



eServices 8.1

Deployment Guide

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

Copyright © 2011–2013 Genesys Telecommunications Laboratories, Inc. All rights reserved.

About Genesys

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

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

Notice

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

Your Responsibility for Your System's Security

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

Trademarks

Genesys and the Genesys logo are registered trademarks of Genesys Telecommunications Laboratories, Inc. All other company names and logos may be trademarks or registered trademarks of their respective holders. © 2012 Genesys Telecommunications Laboratories, Inc. All rights reserved.

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

Technical Support from VARs

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

Customer Care from Genesys

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

Ordering and Licensing Information

Complete information on ordering and licensing Genesys products can be found in the [Genesys Licensing Guide](#).

Released by

Genesys Telecommunications Laboratories, Inc. www.genesyslab.com

Document Version: 81mm_dep_11-2013_v8.1.401.00



Table of Contents

List of Procedures	7
Preface	11
eServices and the CIM Platform	12
CIM Platform	12
eServices	12
Intended Audience	14
Making Comments on This Document	14
Contacting Genesys Technical Support	14
Document Change History	15
New in Document Version 8.1.201.00	15
New in Document Version 8.1.301.00	15
New in Document Version 8.1.401.00	15
Chapter 1	Overview 17
Functions and Components	17
Media Interfaces	18
Data Storage: Universal Contact Server	19
Workflow Control	19
Architecture	22
Processing E-Mail	22
Processing Chat	23
Processing SMS Messages	24
Processing MMS Messages	25
Processing Web Callback	25
Processing Social Media	25
Required Software Components	26
How to Configure eServices	26
New in This 8.1.4 Release	28

Chapter 2	Deployment Task Flow.....	29
	Task Summary: Windows Deployment	29
	Task Summary: UNIX Deployment	32
Chapter 3	Deployment Planning	37
	Overview.....	37
	Component Distribution	38
	Estimating Database Size	39
	Configuration Worksheet	42
	Connections Table	52
Chapter 4	Deploying Third-Party Web Applications	55
	Java Software Development Kit	56
	Installing Apache and Tomcat on Windows	57
	Apache.....	58
	Tomcat	60
	Connecting Tomcat and its Web Server	64
	Setting Up the JK1 Connector with Apache.....	64
	Setting Up the JK1 Connector with IIS 6 or IIS 7	66
	Advanced Configuration of the Tomcat Context	72
	Configuring Apache and Tomcat for Linux or Solaris	76
	Deployments with WebSphere on AIX or Solaris	77
	Minimum Requirements	78
	Deploying KANA Response Live for co-browsing	78
	Prerequisites	78
	Installing and Configuring KANA Response Live for Windows	82
	Installing and Configuring KANA Response Live for Solaris or Linux	88
	Launching a Co-Browsing Session	93
Chapter 5	Model Configuration and Installation on Windows	97
	The Model Environment	98
	Overall Strategy	98
	Prerequisites.....	99
	Web Server and Web Application Container	101
	Configuring eServices Objects	101
	Installing eServices Components	136
	Generalized Installation Procedure.....	137
	Manual Installation of eServices Components	146
	Creating Databases and Running Scripts	152
	Verifying Connections.....	153

	Interaction Server Proxy as Reporting Bus.....	154
	Testing eServices Components	154
	Testing the Setup	157
	Using the Sample Business Processes	157
Chapter 6	Deploying Java Components on 64-bit Windows and UNIX	165
	Old and New Styles of Installation Package	165
	Installing Components on 64-bit Windows Operating Systems	166
	Installing Components on 64-bit UNIX Operating Systems	167
	Supported Versions of JDK/JRE.....	168
Chapter 7	TLS and Client-Side Port	169
	TLS	169
	Client-Side Port	172
Chapter 8	Deploying an E-Mail System in Secured Mode.....	177
	Configuring the Corporate E-Mail Server	177
	Configuring TLS/SSL for E-mail Server.....	178
Chapter 9	Multiple Interaction Servers in a Single Tenant.....	183
	Overview.....	183
	Configuring	184
Chapter 10	Deploying eServices in UNIX Environments.....	185
	Deploying eServices on UNIX Hosts	185
	Note on Java Environment and Libraries for eServices and UCS	186
	Compatibility Package for Red Hat Linux	189
	Deploying Genesys Web Applications for WebSphere on AIX, Solaris, or Windows.....	190
	Environment.....	190
	Deploying Genesys Web Applications for WebLogic on Solaris.....	201
Chapter 11	Manual Deployment—Web API Server and UCS	211
	Introduction.....	211
	Requirements.....	212
	Configuring Web API Server and UCS.....	212
	Installing Web API Server and UCS	215

Chapter 12	Manual Deployment—UCS Proxy, Interaction Server Proxy, and SMS Server.....	223
	Deploying UCS Proxy or Interaction Server Proxy	223
	Requirements.....	224
	Deploying SMS Server	228
Chapter 13	Capture Points	233
Chapter 14	Social Media.....	235
Chapter 15	Uninstalling eServices	237
	Using Wizards to Uninstall eServices	237
	Removing eServices Components	238
	Removing Genesys Desktop	240
Chapter 16	Starting and Stopping Procedures	241
	Starting eServices with Solution Control Interface.....	241
	Starting eServices Manually	242
	Stopping eServices with Solution Control Interface.....	244
Appendix	Deploying an E-Mail Solution Using MIME Customization.....	247
	Overview.....	247
	Configuration Process	248
	Configuring the E-Mail Server Application Object.....	251
	MIME Customization Options Table.....	253
	Message Flow Pattern.....	254
	Incoming E-Mail Message Flow	254
	Outgoing E-Mail Message Flow.....	255
Supplements	Related Documentation Resources	257
	Document Conventions	260
Index	263



List of Procedures

Estimating UCS database size for e-mail interactions	40
Estimating UCS database size for chat interactions.	40
Estimating the size of the Interaction Server database	41
Completing the configuration worksheet	43
Downloading and installing the Java SDK	56
Installing the Apache Web Server on Windows	58
Testing the Apache Web Server on Windows	59
Installing the Tomcat web application container on Windows	60
Testing the Tomcat installation on Windows	62
Installing the JK1 Connector with Apache	64
Testing the JK1 Connector with Apache	65
Installing the JK1 connector with IIS 6	66
Installing the JK1 connector with IIS 7	69
Testing the JK1 Connector with IIS	72
Configuring IIS to serve static content	73
Protecting the WEB-INF Directory	74
Configuring Tomcat to serve multiple contexts.	74
Obtaining Apache documentation from the CD	76
Obtaining Tomcat documentation from the CD and the Web.	77
Installing a new SSL certificate using IIS	80
Installing KANA Response Live for Windows.	82
Configuring KANA Response Live for Windows.	84
Testing the KANA Response Live installation on Windows	87
Installing KANA Response Live for Solaris.	88
Installing KANA Response Live for Linux.	89
Creating the KANA database schema on Solaris or Linux.	90
Configuring KANA Response Live for Solaris or Linux.	91
Testing the KANA installation on Solaris or Linux	92
Launching a co-browsing session	93

Installing the eServices Configuration Wizards	102
Installing and Starting Interaction Workflow Samples	103
Configuring eServices objects using the wizard	104
Specifying a webapi-port for Chat Server	134
Installing and Starting the Web Callback Application	135
Performing an integrated installation of all eServices servers	138
Specifying properties in a special file for passing initialization parameters to the Load Balancer Servlet	142
Specifying system variables for passing initialization parameters to the Load Balancer Servlet	145
Creating the Application object	147
Installing eServices components on Windows	148
Installing eServices components on AIX, Solaris, or Linux	151
Running scripts for your databases	153
Verifying connections	153
Testing the eServices servers	154
Testing the components using the Web-based TestTool810	155
Testing that Genesys Desktop can handle interactions	156
Configuring ABC Simple BP for routing e-mail interactions	158
Configuring ABC Simple Chat BP for routing chat interactions	159
Sending a test e-mail to a Desktop agent	159
Starting a test chat session with a Desktop agent	160
Sending a test web form e-mail to a Desktop agent	161
Configuring WebCallback BP for routing web callback interactions	162
Updating the Interaction Server database for processing web callback interactions	163
Sending a test web callback to a Desktop agent	164
Generating the .truststore file	179
Modifying the E-mail Server startup command line on Windows	179
Modifying the E-mail Server startup command line on Unix	180
Configuring E-mail Server's POP, IMAP, and SMTP ports	181
Deploying eServices on Solaris, AIX, or Linux from a Windows host	187
Installing Genesys web applications for WebSphere on AIX or Solaris	191
Deploying Genesys web applications for WebSphere on AIX, Solaris, or Windows	192
Testing the WebSphere installation on AIX, Solaris, or Windows	199
Installing Genesys web applications for WebLogic	202

Checking your WebLogic configuration file	203
Deploying your web applications	204
Testing the WebLogic installation and deployment	208
Creating or editing the Web API Server application object	212
Creating or editing the UCS application object	213
Creating or editing the UCS Manager application object	214
Installing Web API Server on Windows	215
Installing Web API Server on Solaris or Linux	217
Installing Universal Contact Server on Windows	218
Installing Universal Contact Server on AIX, Solaris, or Linux	219
Installing Universal Contact Server Manager	220
Creating the Proxy application object	225
Installing Universal Contact Server Proxy or Interaction Server Proxy on Windows	226
Installing Universal Contact Server Proxy or Interaction Server Proxy on AIX, Solaris, or Linux	227
Configuring the desktop application to use UCS Proxy or Interaction Server Proxy	228
Creating the SMS Server application object	229
Installing SMS Server on Windows	230
Installing SMS Server on AIX, Solaris, or Linux	231
Configuring eServices Components to work with SMS Server	232
Uninstalling eServices using the wizards	237
Removing eServices components	239
Starting eServices using SCI	241
Starting eServices components from the command line	243
Starting eServices components as services	243
Starting eServices from the Windows Start menu	244
Stopping eServices using SCI	244
Deploying a MIME-compliant e-mail solution	248
Viewing a sample transformer	248
Creating a custom transformer	249
Configuring the E-mail Server application object	251



Preface

Welcome to the *eServices 8.1 Deployment Guide*. This guide introduces you to the architecture, required components, and procedures relevant to the deployment of a Genesys eServices solution in your contact center.

eServices was known as Multimedia in releases 8.0.0, 7.6, and 7.5, and was known as Multi-Channel Routing in releases 7.0 and 7.1.

This document is valid only for the 8.1 release of this product.

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

This preface contains the following sections:

- [eServices and the CIM Platform, page 12](#)
- [Intended Audience, page 14](#)
- [Making Comments on This Document, page 14](#)
- [Contacting Genesys Technical Support, page 14](#)
- [Document Change History, page 15](#)

Genesys eServices is a series of components working together to handle interactions from disparate media-based devices. It allows you to centralize your handling of the various communication channels that customers use to reach your interaction contact center. Bringing your eServices solution into production requires that you deploy a number of Genesys components, as well as several separate, third-party software applications, without which eServices cannot work.

For information about related resources and about the conventions that are used in this document, see the supplementary material starting on [page 257](#).

eServices and the CIM Platform

Genesys eServices (formerly Multimedia) is a cover term for Genesys components that work together to manage interactions whose media is something other than traditional telephonic voice (for example, e-mail or chat).

eServices includes some parts of the Genesys Customer Interaction Management (CIM) Platform, plus certain of the media channels that run on top of the Platform. See Figure 1 on [page 13](#).

CIM Platform

The CIM Platform consists of the following components:

- Management Framework
- Reporting (CC Analyzer, CCPulse+)
- Interaction Management, which in turn consists of:
 - Universal Routing
 - Interaction Workflow
 - Knowledge Management
 - Content Analysis
 - Universal Contact History

On top of the CIM Platform are various media channels. Some, such as Genesys Network Voice, handle traditional telephony. Others, such as Genesys E-mail, handle other media.

eServices

eServices, then, consists of the following components:

- From the CIM Platform, all of Interaction Management components except for Universal Routing. This includes:
 - Interaction Workflow—Centralized handling of interactions irrespective of media type.
 - Knowledge Management—Creation and maintenance of standard responses and screening rules.
 - Content Analysis—Optional enhancement to Knowledge Management, applying natural language processing technology to categorize interactions.
 - Universal Contact History—Storage of data on contacts and on interactions (linked as threads).

Universal Routing is not considered part of eServices because it deals with both traditional telephonic interactions and the nontraditional interactions that are handled in eServices.

- From the media channels, at least one of the following:
 - Genesys E-mail
 - Genesys Chat (formerly Genesys Web Media)
 - Genesys SMS (Short Message Service)
 - Genesys MMS (Multimedia Messaging Service)
 - Genesys Web Callback
 - Genesys 3rd Party Media—A cover term for Genesys 3rd Party Chat, Genesys 3rd Party SMS, and other products that provide the ability to add customized support for various media.
- Optionally, Web Collaboration—the ability for agents and customers to co-browse (simultaneously navigate) shared web pages. This is an option that you can add to either Genesys Chat or Inbound Voice.

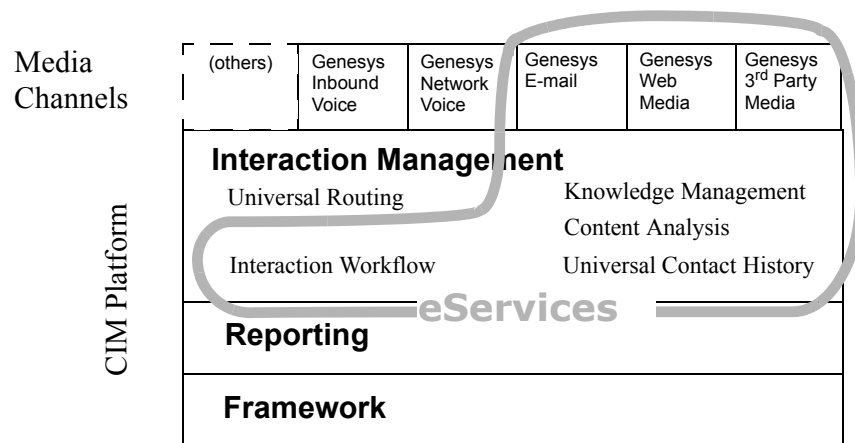


Figure 1: eServices in Relation to the CIM Platform and Media Channels

Note: Although Universal Routing is not considered part of eServices, any functioning solution (platform plus channels) that includes any part of the Interaction Management sector requires Universal Routing.

Licensing

Licensing requirements are:

- For each agent, one eServices Agent seat.
- For each media option, one media channel (E-mail and/or Web Media and/or SMS and/or custom media).
- For Genesys Content Analyzer, NLP Content Analysis license.

See the *Genesys Licensing Guide* for additional details.

Reporting

Reporting templates are available for eServices. For details see the documents in the Reporting Technical Reference Series.

Intended Audience

This document is primarily intended for system administrators and anyone else who wants to configure and install eServices 8.1. It has been written with the assumption that you have a basic understanding of:

- Computer-telephony integration (CTI) concepts, processes, terminology, and applications
- Network design and operation
- Your own network configurations
- Internet, e-mail, chat, and World Wide Web basics

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

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 Customer Care if you have suggestions about the product itself.

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

Contacting Genesys Technical Support

If you have purchased support directly from Genesys, please contact [Genesys Customer Care](#).

Before contacting Customer Care, please refer to the [Genesys Care Program Guide](#) for complete contact information and procedures.

Document Change History

This section lists topics that are new or that have changed significantly since the first release of this document, which was version v8.1.001.00.

New in Document Version 8.1.201.00

- The procedure “Specifying a webapi-port for Chat Server” on [page 134](#) has been modified.
- The section “Web Service Capture Point” has been added to [Chapter 13](#) “Capture Points”.

New in Document Version 8.1.301.00

- Information on E-mail Server and UCS in the tables in Chapter 7, “TLS and Client-Side Port,” on [page 169](#) has been updated.
- Additional description has been added to the section “Configuring the Corporate E-Mail Server” on [page 177](#) in [Chapter 8](#) “Deploying an E-Mail System in Secured Mode.”
- Chapter 9, “Multiple Interaction Servers in a Single Tenant,” on [page 183](#) has been added.

New in Document Version 8.1.401.00

“Interaction Server Proxy as Reporting Bus” on [page 154](#) describes this new feature of Interaction Server Proxy.

1

Overview

Genesys eServices (called Multimedia in 8.0.0 and earlier) is a platform on which you can assemble a coordinated suite of components that processes, manages, and archives customer/agent interactions in the media of your choice. This chapter introduces the components, functionality, and concepts behind them.

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

- [Functions and Components, page 17](#)
- [Architecture, page 22](#)
- [Required Software Components, page 26](#)
- [How to Configure eServices, page 26](#)
- [New in This 8.1.4 Release, page 28](#)

Functions and Components

Figure 2 shows the overall functionality of eServices (called Multimedia in 8.0.0 and before), with any media type. This figure shows functions only, not components.

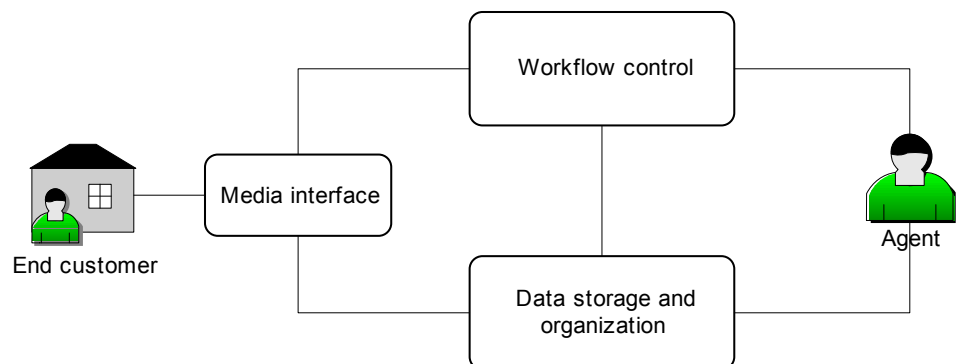


Figure 2: General Functioning of eServices

The three major functions shown in [Figure 2](#) are:

- The **media interface**, which brings interactions into the system. It may interface with e-mail, chat, or other media.
- A **database**, which stores the history of the interaction and associates it with related interactions to form a thread. It also stores contact information and other types of data used at different points in the processing of interactions.
- **Workflow control**, which determines where the interaction goes and what happens to it.

For some media types (such as chat), the media interface also communicates directly with the agent desktop.

The following sections explain these functions further and describe the components involved.

Media Interfaces

The media interfaces available with eServices 8.1 are **E-mail Server**, **Chat Server**, **Social Messaging Server**, and **SMS Server**.

Note: eServices 8.1 also supports the processing of 3rd Party Media interactions with the help of Genesys's 3rd Party Media SDK and Interaction SDK products. See the documentation for those products for more information.

- E-mail Server interfaces with the enterprise mail server and the Genesys Web API Server, bringing in new e-mail interactions and sending out replies or other outbound messages.
- Chat Server works with **Web API Server** to open, conduct, and close chat interactions between agents and customers.
- SMS Server receives and handles SMS and MMS messages sent from a mobile client. SMS Server uses SMPP v3.4 protocol for SMS support, and MM1, MM7 protocols for MMS support. For more details see the "SMS Server" part of the "Administration" section of the "Ongoing Administration and Other Topics" chapter of the *eServices 8.1 User's Guide*.
- Web API Server works with Interaction Server to create, schedule and close callback requests via the web.
- Social Messaging Server provides Social Media functionality, such as support for Facebook, Twitter, and RSS. For more information, see Chapter 14, "Social Media," on [page 235](#).

To the workflow control components, these interfaces transmit operational data about each interaction, consisting of an identifying code plus some data about the interaction (date received, originating party, and so on).

To the data storage components, they transmit the body of the interaction—that is, a transcript of the e-mail or chat session.

Data Storage: Universal Contact Server

Universal Contact Server (UCS) interfaces with a database that stores the following:

- Contact information, such as names, addresses, phone numbers
- Contact history: previous interactions with this contact
- Other data used in processing interactions, such as standard responses and screening rules.

Your eServices installation (as part of the Universal Contact Server installation package) includes scripts for setting up the database (Microsoft SQL Server, Oracle, and DB2 are supported). **Universal Contact Server Manager** provides a user interface for setting data-archiving and pruning functions.

Workflow Control

The components illustrated in [Figure 3](#) handle workflow control.

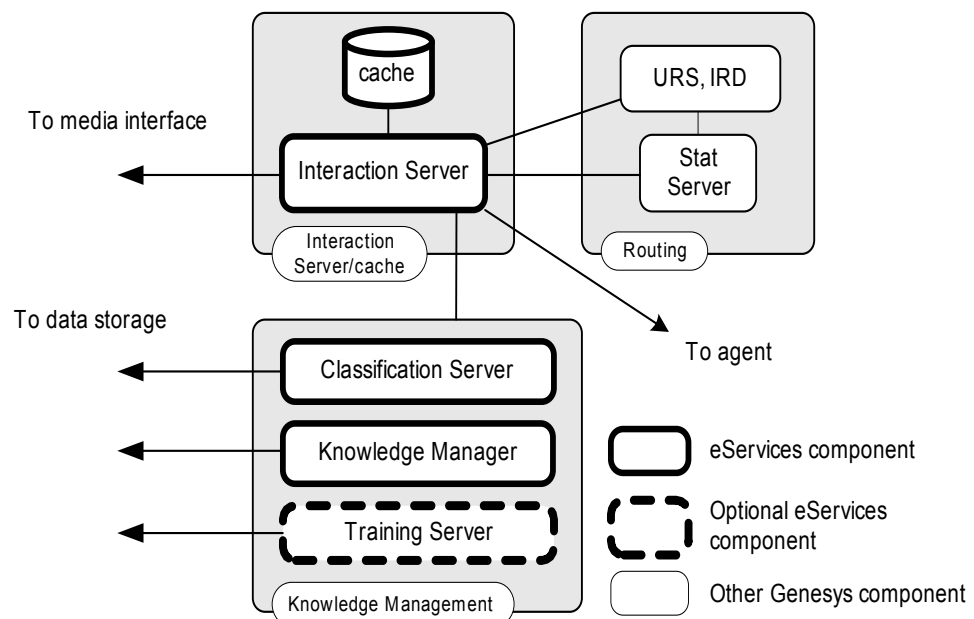


Figure 3: Workflow Control Components

Not shown in [Figure 3](#) are other required Genesys Management Framework components, such as Configuration Server and the Management Layer.

The workflow control components fall into three groups, described in the following sections.

Interaction Server

Interaction Server is the central interchange for interaction flow.

- It receives interaction operational data from the media interface.
- It stores the operational data in a **cache** (a database) while receiving and transmitting information about the interaction. This cache also contains **queues** through which the interaction passes as part of its processing.
- It works in concert with the Routing components to route interactions according to interaction workflows and routing strategies (see the [“Routing”](#) section).
- It provides the means for agents to log in and indicate readiness.

Routing

Routing components include the following:

- **Interaction Routing Designer (IRD)** and **Universal Routing Server (URS)** design and execute **routing strategies**, which trigger functions such as automatic responses and screening; apply logic (segmentation and conditional branching) to the flow; and ultimately deliver the interaction to an agent or other target. Routing strategies are one of the two main types of objects used in interaction workflows (see the [“Interaction Server”](#) section).
- **Interaction Design**, a sub component of IRD, creates and displays **Business Processes**, which plot an overall path for interactions. Interaction workflows map a route for the interaction between contact center objects, principally queues and routing strategies (see the [“Knowledge Management”](#) section). Interaction workflows are executed by Interaction Server.
- **Stat Server** accumulates data about places, agents, and place/agent groups; converts the data into statistically useful information; and passes these calculations to other software applications. In particular, Stat Server provides information to URS about agents’ capacities in terms of the number of interactions, the media type of an interaction, and so on.

Knowledge Management

Genesys Knowledge Management is made up of the following:

- **Classification Server**, which applies **screening rules** when triggered to do so by a routing strategy. Screening rules are basic pattern-matching queries performed on interaction contents. The results of these queries can then be referred to by further routing strategy logic. In the Genesys Content Analyzer option (see below), Classification Server also applies **models** to categorize incoming interactions. Both screening rules and models are stored in the Universal Contact Server database.

- **Training Server**, which trains the system to recognize categories. It is active only in the Content Analyzer option (see below).
- **Knowledge Manager**, which is the user interface component for Knowledge Management. You use Knowledge Manager to:
 - Manage the Standard Response Library, which is a collection of ready-made responses to common inquiries and topics.
 - Manage screening rules.
 - Manage **categories**, which are used to organize standard responses.

Genesys Content Analyzer is an optional enhancement to Knowledge Management, requiring a separate license. It uses natural language processing technology to analyze incoming interactions for assignment to the categories of the standard response category system. The statistical tools that enable this analysis, called **models**, are built up and refined by Training Server as it processes collections of preclassified interactions. Setting up and scheduling these training sessions is another function of Knowledge Manager.

FAQ works with Genesys Content Analyzer to convert your category structure and standard responses into an Frequently Asked Questions (FAQ) list. You can either post the resulting list as text on your web site or use it as the source for an automatic question-answering facility.

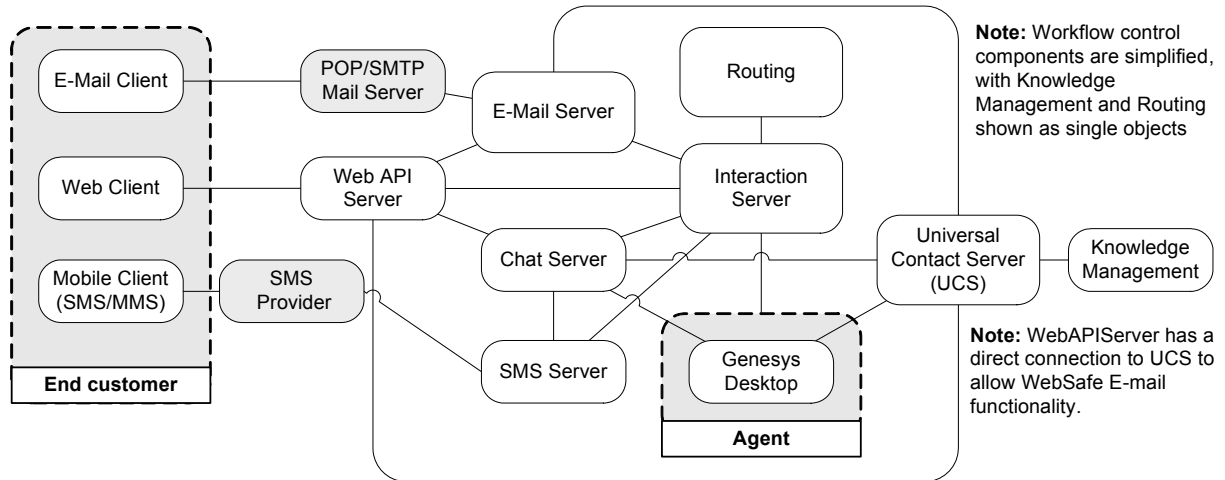
Summary

To summarize interaction flow:

- At the highest level the flow is controlled by interaction workflows that Interaction Server executes.
- Each interaction workflow contains queues and routing strategies.
- Routing strategies may bring in other applications/components to apply processing to the interaction—for example, sending a transcription of the chat session to the customer.:
 - Send an acknowledgment or an automatic reply.
 - Apply a screening rule.
 - Apply content analysis (with Content Analyzer option only).
 - Forward or redirect the interaction.

Architecture

This section presents the architecture for eServices 8.1. [Figure 4](#) shows the main eServices components.



Note: Each Genesys component also connects to the Configuration Server for configuration data.

Figure 4: eServices Architecture

Note: For simplicity, not all connections and details are included in [Figure 4](#). For example, many components (including Web API Server) connect to Stat Server.

Interaction processing generally proceeds according to the type of interaction. The following sections describe how five of the most common interaction types (e-mail, chat, SMS, MMS, and web callback) are processed.

Processing E-Mail

This section describes how e-mails are processed.

1. E-mail interactions arrive in one of two ways:
 - a. If the customer sends ordinary e-mail, the interaction arrives via the enterprise mail server.
 - b. If the customer sends e-mail from a web site (by filling out a web form), the interaction arrives via the Web API Server.
2. E-mail Server stores the body of the interaction in the Universal Contact Server database, and then sends operational data on the interaction to Interaction Server.

3. Interaction Server parks the interaction's operational data in its cache and starts processing the data according to an interaction workflow.
4. What happens next depends on the interaction workflow and the routing strategies that it contains. The system may:
 - Apply a screening rule.
 - Assign the interaction to one or more categories (if Content Analyzer is present).
 - Generate an automatic response.
 - Route the interaction to an agent's desktop, possibly also sending an automatic acknowledgment to the customer.

A supervisor may intervene at various points as long as the interaction's operational data remains in the Interaction Server's cache and the interaction is not being actively worked on by the Routing components.

5. The agent receives the interaction.
6. The agent may then:
 - Simply reply to the interaction.
 - Reply making use of a standard response. With the Content Analyzer option, the interaction may have arrived already equipped with a category assignment and associated suggested response. Otherwise, the agent may search manually for a category with suggested response.
 - Transfer the interaction to another agent.
 - Produce a collaborative response by consulting with other agents.
 - Return the interaction to the system for further processing.
7. When the agent or agents finally release the reply (typically to an Outbound queue in the Interaction Server cache), the interaction workflow may route it to a senior agent or supervisor for QA review. The reviewer decides whether to let the reply continue through the outbound part of the interaction workflow, return it to the agent for revision, or take other action.

Processing Chat

This section describes how chat interactions are processed.

1. Chat interactions begin processing when the Web Client submits a customer's chat request to Chat Server.
2. Chat Server creates a chat session and asks Universal Contact Server to create an interaction record.
3. Chat Server submits the interaction to Interaction Server.
4. Interaction Server places the interaction in its initial queue and begins processing it according to an interaction workflow.

5. The interaction workflow and its component routing strategies may do various things, including sending a message to a customer prior to an agent actually handling the interaction, but eventually they select an agent who is available for chat sessions and send an invitation to that agent to participate in a chat session.
6. The agent connects to the chat session and accepts the invitation.
7. Agent and customer conduct a chat session (exchange with messages and notifications).
8. The chat session ends (by agent request).
9. Chat Server writes the content of the chat session to the Universal Contact Server database and updates the interaction in Interaction Server.
10. Any postprocessing occurs; for example, a transcript of the chat session is e-mailed to the customer.

Processing SMS Messages

This section describes how SMS interactions are processed.

1. SMS messages arrive when a mobile client sends an SMS message to a phone number of a Contact Center. Genesys SMS Server is a recipient and handler of SMS messages.
SMS Server supports two operational modes: **paging mode** and **session (chat) mode**.
Paging mode refers to receiving an individual SMS message from a mobile client and sending back an agent's response (paging inbound), or sending an individual SMS message to a mobile client on a Contact Center initiative (paging outbound).
Session (chat) mode refers to creating and keeping an interactive conversation between a mobile client and an agent in the form of a conventional chat session. All messages received and sent during this session are associated with one interaction, which corresponds to this SMS session.
2. The paging mode incorporates capabilities to send and receive individual SMS messages. SMS server:
 - Submits the incoming SMS messages to Interaction Server as a new interaction.
 - Sends SMS messages to mobile clients on requests from an agent, routing strategy, or application.
3. The session mode supports an interactive conversation between a client and an agent:
 - For an incoming SMS message, SMS Server checks if the mobile client is participating in an active SMS session (Chat Server session).
 - If a session is found, the SMS Server forwards the message to the session.

- If a session is not found, SMS Server creates a new SMS session. SMS Server:
 - Requests Chat Server initiate a chat session.
 - Stores a record about this session.
 - Starts forwarding incoming and outgoing messages between the mobile client and the Chat Server session.

Processing MMS Messages

This section describes how MMS interactions are processed.

MMS messages arrive when a mobile client sends an MMS message to a phone number of a Contact Center. Genesys SMS Server is a recipient and handler of MMS messages. MMS messages can include a subject, text parts, multimedia parts, and a presentation scenario specified in an SMIL (Synchronized Multimedia Integration Language) part of an MMS message. SMS Server supports inbound mode for MMS messages.

Inbound mode refers to receiving an individual MMS message from a movable client and submitting it to Interaction Server as a new interaction.

Processing Web Callback

This section describes how web callback interactions are processed.

1. A customer schedules web callback in the Web Callback sample application.
2. An interaction is created in Interaction Server and placed in the queue.
3. The interaction is processed by the web callback application business process.
4. Optional: An acknowledgement e-mail is sent to the customer.
5. Optional: An interaction is created in Universal Contact Server database by the business process.
6. The customer can reschedule or cancel the web callback.
7. The web callback interaction is delivered to an agent at the scheduled time.
8. The agent calls the customer.
9. If the agent is unable to reach the customer, the agent can reschedule the web callback and the process returns to [Step 6](#).
10. The agent can “mark done” the web callback interaction.

Processing Social Media

For information about processing Social Media, refer to the [eServices Social Media Solution Guide](#), available on the Genesys Documentation Wiki.

Required Software Components

The components that you must have running to operate an eServices 8.1 environment can be grouped into the following categories. In this list, when the item is a generic description it is followed in parentheses by a typical instance:

- **Genesys components**
 - **Framework**
 - Any third-party items that Genesys Framework requires, such as a database engine and databases to run with DB Server. (See the [Genesys Supported Operating Environment Reference Guide](#) for a list of supported databases.)
 - Configuration Layer
 - Management Layer
 - Stat Server (from the Real-Time Metrics Engine)
 - Genesys Universal Routing (either an Enterprise Routing solution or a Network Routing solution)
 - **eServices**
 - Servers
 - Clients
- **Web components**
 - Web Server (such as Microsoft IIS)
 - Browser (such as Microsoft Internet Explorer)
 - Website content (Genesys supplies Web Samples, a starter application that you can customize).
- **Java components**
 - Java virtual machine (such as Java Runtime Environment or JRE)
 - Java web application container (such as Tomcat)

Note: The 8.1 release of Genesys eServices supports only specific releases of Tomcat. Check the [Genesys Supported Operating Environment Reference Guide](#) for more information.

- **Other items**
 - Database engine for Interaction Server and Universal Contact Server databases

How to Configure eServices

You can configure eServices 8.1 in two ways:

- Using the wizards supplied with the software.

- Manually, using the following steps:
 - Creating objects in Configuration Manager, one for each eServices component.
 - Giving the objects the proper settings for options and other attributes.
 - Assigning the various connections that link components together.

Note: Whether you are deploying eServices exclusively to a Windows environment or to a Solaris, Linux, or AIX environment, Genesys strongly recommends that you use wizards to install and configure your eServices solution.

Chapter 4, beginning on [page 55](#), describes the configuration for various third-party software components that eServices uses, such as Apache or Microsoft IIS for the web server. In your final deployment, you will of course substitute your own architecture (more or fewer hosts), settings (Configuration Layer Persons corresponding to actual agents in your environment), and third-party applications (such as WebSphere, rather than Apache or Microsoft IIS).

Chapter 5, beginning on [page 97](#), describes a model wizard deployment on a single Windows host, such as you might use in a lab configuration. (The model does not suggest a production environment.) The sample includes specific values for all settings.

Genesys eServices 8.1 supports Windows, Solaris, Linux, and AIX operating systems. The model installation uses a Windows environment. Deployment for Solaris, Linux, and AIX environments is similar; the differences are described in Chapter 10, beginning on [page 185](#).

If you want to configure and install eServices 8.1 manually in a Windows environment, consult:

- *Configuration Manager 8.1 Help* for instructions on using Configuration Manager.
- *Framework 8.1 Genesys Administrator Help* for instructions on using Genesys Administrator.
- This chapter, Chapter 1, beginning on [page 17](#), and Table 5 on [page 99](#) for a list of required components.
- Chapter 5, beginning on [page 97](#), and each component's configuration steps for the required connections between eServices components.
- *eServices 8.1 Reference Manual* for descriptions of all eServices options.

New in This 8.1.4 Release

This release includes the following new features:

- Interaction Management:
 - Interaction Server Proxy:
 - Ability to function as a dedicated distribution point for reporting events.
 - Support of TLS and IPv6.
 - Interaction Server:
 - Interdependent login: new value `force-logout` for option `allow-multiple-agent-connection` extends ability to configure how agents log in on multiple connections.
 - New option `delayed-logout-panic-threshold` provides ability to notify reporting clients before and after distribution of events reporting agent state change due to a proxy disconnecting.
 - Delayed logout for agent clients that disconnect due to intermediate proxy disconnect, configured using new option `delayed-logout-timeout`. No reporting events are distributed if agents log in and restore their state within a configurable time interval. This helps to reduce the load on reporting clients.
 - Ability to broadcast requests to multiple third-party servers of the same type and combine the responses.
 - Ability for clients to request Agent state information with the new pair of messages `RequestAgentState` and `EventAgentState`.
 - Support of IPv6.
 - Java 7 64-bit support by Integrated Capture Points
 - Support of IBM DB2 10
- Genesys E-mail:
 - Support of Java 7 64-bit
 - Support of IPv6
 - Support for the retrieval of e-mail using Exchange Web Services protocol
- Universal Contact Server (UCS):
 - Support of DB2 10
 - Support of Java 7 64-bit
 - Support of IPv6
- UCS Proxy—support of IPv6



Chapter

2

Deployment Task Flow

This chapter guides you through the various actions you must take to plan, deploy, and test your eServices solution.

You can find task flow information for the following operating systems:

- [Task Summary: Windows Deployment, page 29](#)
- [Task Summary: UNIX Deployment, page 32](#)

Task Summary: Windows Deployment

This task summary describes a model setup of a blended (e-mail and chat), single-tenant eServices solution deployed on a single Windows host machine, such as might be deployed in a lab setting. Complete these actions in the following order to deploy your solution.

Task Summary: Windows Deployment

Objective	Related Procedures and Actions
1. Plan your deployment.	<ol style="list-style-type: none">1. Choose which machines will host eServices and various third party components. For a sample architecture layout, see “Component Distribution” on page 38.2. Choose which machines will host your databases. To help calculate the rough size of your databases, use the formulas in “Estimating Database Size” on page 39.3. Print out and complete the Configuration Worksheet (Table 1 on page 43) with values that fit your deployment.
2. Complete all prerequisite actions.	<ol style="list-style-type: none">1. Verify that all prerequisite components are in place. See Table 5 on page 99.2. “Create Host” on page 100.

Task Summary: Windows Deployment (Continued)

Objective	Related Procedures and Actions
3. Complete all prerequisite actions. (continued)	<ol style="list-style-type: none"> For deployments across multiple machines, see “Create a Shared Directory” on page 101. If you have already installed KANA Response Live, you must stop it from running as a service. For more information, see “Remove the KANA Response Live Start Service” on page 101. “Create Contact, Interaction (and Co-Browsing) Databases” on page 101.
4. Install and configure third-party software for processing Web-based interactions (chat and web e-mail).	<p>Install, connect, and/or configure components that enable web services. Use one of the following web server/application container combinations.</p> <p>Apache and Tomcat</p> <p>For an Apache/Tomcat combination, complete these steps:</p> <ol style="list-style-type: none"> Procedure: Downloading and installing the Java SDK, on page 56. Procedure: Installing the Apache Web Server on Windows, on page 58 Procedure: Installing the Tomcat web application container on Windows, on page 60 Procedure: Installing the JK1 Connector with Apache, on page 64 <p>IIS and Tomcat</p> <p>For an IIS/Tomcat combination, complete these steps:</p> <ol style="list-style-type: none"> Procedure: Installing the Tomcat web application container on Windows, on page 60 Procedure: Installing the JK1 connector with IIS 6, on page 66 or Procedure: Installing the JK1 connector with IIS 7, on page 69
5. Install and configure third-party software for co-browsing * *Skip this step if you are not using co-browsing functionality.	<p>To use Genesys co-browsing, install and configure the KANA Response Live Server.</p> <p>See Table 4 on page 79 for the deployment task flow.</p> <ol style="list-style-type: none"> Procedure: Installing KANA Response Live for Windows, on page 82 Procedure: Configuring KANA Response Live for Windows, on page 84 Procedure: Testing the KANA Response Live installation on Windows, on page 87 Procedure: Launching a co-browsing session, on page 93

Task Summary: Windows Deployment (Continued)

Objective	Related Procedures and Actions
<p>6. Configure eServices objects.</p>	<ol style="list-style-type: none"> 1. Install the configuration wizards. See Procedure: Installing the eServices Configuration Wizards, on page 102. 2. Create sample workflows for routing simple test interactions through the contact center to an agent. See Procedure: Installing and Starting Interaction Workflow Samples, on page 103. 3. Configure eServices objects for installation. See Procedure: Configuring eServices objects using the wizard, on page 104. 4. Specify the webapi-port for Chat Server, for dedicated communication with Web API Server. See Procedure: Specifying a webapi-port for Chat Server, on page 134. 5. Create configuration objects and sample workflows for routing web callback interactions through the contact center to an agent. See Procedure: Installing and Starting the Web Callback Application, on page 135.
<p>7. Install eServices components</p>	<p>Procedure: Performing an integrated installation of all eServices servers, on page 138</p> <p>Note: For deployments across multiple hosts, you need to launch the Integrated Installation one time for each host.</p>
<p>8. Configure and install eServices components not included in the eServices Configuration Wizards</p> <p>* This step is optional. Your deployment might not require any of these components.</p>	<p>Manually deploy the eServices components that are not included in the eServices Configuration Wizard installation.</p> <p>Universal Contact Server Proxy or Interaction Server Proxy</p> <ol style="list-style-type: none"> 1. Procedure: Creating the Proxy application object, on page 225 2. Procedure: Installing Universal Contact Server Proxy or Interaction Server Proxy on Windows, on page 226 3. Procedure: Configuring the desktop application to use UCS Proxy or Interaction Server Proxy, on page 228 <p>SMS Server</p> <ol style="list-style-type: none"> 1. Procedure: Creating the SMS Server application object, on page 229 2. Procedure: Installing SMS Server on Windows, on page 230 3. Procedure: Configuring eServices Components to work with SMS Server, on page 232

Task Summary: Windows Deployment (Continued)

Objective	Related Procedures and Actions
9. Verify connections	<ol style="list-style-type: none"> 1. Double-check that the connections you made with the configuration wizard are in place. For a list of necessary connections, see Table 2 on page 52. 2. From Configuration Manager, add a connection from Stat Server to Interaction Server (or to Interaction Server Proxy; see page 154). Stat Server uses this connection to find and route interactions to available eServices agents. <p>For more information, see Procedure: Verifying connections, on page 153.</p>
10. Modify your databases to process interactions	<p>Procedure: Running scripts for your databases, on page 153</p>
11. Test the components	<ol style="list-style-type: none"> 1. Procedure: Testing the eServices servers, on page 154 2. Procedure: Testing the components using the Web-based TestTool810, on page 155 3. Procedure: Testing that Genesys Desktop can handle interactions, on page 156
12. Test the setup	<ol style="list-style-type: none"> 1. Procedure: Configuring ABC Simple BP for routing e-mail interactions, on page 158 2. Procedure: Configuring ABC Simple Chat BP for routing chat interactions, on page 159 3. Procedure: Sending a test e-mail to a Desktop agent, on page 159 4. Procedure: Starting a test chat session with a Desktop agent, on page 160 5. Procedure: Sending a test web form e-mail to a Desktop agent, on page 161 6. Procedure: Configuring WebCallback BP for routing web callback interactions, on page 162 7. Procedure: Updating the Interaction Server database for processing web callback interactions, on page 163 8. Procedure: Sending a test web callback to a Desktop agent, on page 164

Task Summary: UNIX Deployment

For Solaris, AIX, or Linux deployments, Genesys recommends that you use the eServices Configuration Wizards on a Windows host to configure and install components to other hosts. Complete the actions in the following order to deploy your solution.

Task Summary: Solaris/AIX/Linux Deployment

Objective	Related Procedures and Actions
1. Prepare your environment.	<ol style="list-style-type: none"> 1. Define an environment variable for your eServices directory (for “old-style” installation only; see explanation on page 165). GES_HOME_810 2. Create a shared directory on your UNIX host that your Windows host can access. OR Create a shared directory on the Windows host that your UNIX hosts can access. 3. If you have already installed KANA Response Live, you must stop it from running as a service. Use the following command: <code>rm -f /etc/rc2.d/S88hipctl</code> 4. Create your databases.
2. Configure eServices objects for installation	<p>Launch the eServices Configuration Wizard from a Windows host to configure the components that will run in the Solaris, Linux, or AIX environments.</p> <ol style="list-style-type: none"> 1. Procedure: Installing the eServices Configuration Wizards, on page 102 2. Procedure: Installing and Starting Interaction Workflow Samples, on page 103 3. Procedure: Configuring eServices objects using the wizard, on page 104
3. Install Third Party Components (omit this step for “new-style” installation packages; see explanation page 165).	<p>On each host where you plan to install components, run the Java Environment and Libraries for eServices and UCS installation package from the following directory: <platform>\ThirdPartyComponents</p> <p>Note: For an explanation of the difference between Third Party Components and ThirdPartyApplications, see the note on page 56.</p>
4. Install the eServices components	<p>Run the installation package for each remaining component: <component>\<platform></p>

Task Summary: Solaris/AIX/Linux Deployment (Continued)

Objective	Related Procedures and Actions
<p>5. Configure and install eServices components not included in the eServices Configuration Wizards</p>	<p>Manually deploy the eServices components that are not included in the eServices Configuration Wizard installation.</p> <p>Universal Contact Server Proxy or Interaction Server Proxy</p> <ol style="list-style-type: none"> 1. Procedure: Creating the Proxy application object, on page 225 2. Procedure: Installing Universal Contact Server Proxy or Interaction Server Proxy on AIX, Solaris, or Linux, on page 227 3. Procedure: Configuring the desktop application to use UCS Proxy or Interaction Server Proxy, on page 228 <p>SMS Server</p> <ol style="list-style-type: none"> 1. Procedure: Creating the SMS Server application object, on page 229 2. Procedure: Installing SMS Server on AIX, Solaris, or Linux, on page 231 3. Procedure: Configuring eServices Components to work with SMS Server, on page 232

Task Summary: Solaris/AIX/Linux Deployment (Continued)

Objective	Related Procedures and Actions
<p>6. Install and configure third-party software for processing Web-based interactions</p>	<p>Install, connect, and/or configure components that enable web services. Use one of the following web server/application container combinations.</p> <p>Solaris—WebLogic Web Application</p> <ol style="list-style-type: none"> 1. Procedure: Installing Genesys web applications for WebLogic, on page 202 2. Procedure: Checking your WebLogic configuration file, on page 203 3. Procedure: Deploying your web applications, on page 204 4. Procedure: Testing the WebLogic installation and deployment, on page 208 <p>Solaris—Apache and Tomcat</p> <ul style="list-style-type: none"> • Procedure: Obtaining Apache documentation from the CD, on page 76 • Procedure: Obtaining Tomcat documentation from the CD and the Web, on page 77 <p>AIX or Solaris—WebSphere Web Application</p> <ol style="list-style-type: none"> 1. Procedure: Deploying Genesys web applications for WebSphere on AIX, Solaris, or Windows, on page 192 2. Procedure: Testing the WebSphere installation on AIX, Solaris, or Windows, on page 199 <p>Linux—Apache and Tomcat</p> <ol style="list-style-type: none"> 1. Procedure: Obtaining Apache documentation from the CD, on page 76 2. Procedure: Obtaining Tomcat documentation from the CD and the Web, on page 77

Task Summary: Solaris/AIX/Linux Deployment (Continued)

Objective	Related Procedures and Actions
<p>7. Install and configure third-party software for co-browsing. (Solaris and Linux only)</p>	<p>To use Genesys co-browsing, install and configure the KANA Response Live Server. See Table 4 on page 79 for the deployment task flow.</p> <p>Solaris or Linux</p> <ol style="list-style-type: none"> 1. Procedure: Installing KANA Response Live for Solaris, on page 88 2. Procedure: Installing KANA Response Live for Linux, on page 89 3. Procedure: Creating the KANA database schema on Solaris or Linux, on page 90 4. Procedure: Configuring KANA Response Live for Solaris or Linux, on page 91 5. Procedure: Testing the KANA installation on Solaris or Linux, on page 92 6. Procedure: Launching a co-browsing session, on page 93



Chapter

3

Deployment Planning

This chapter describes, in general terms, how to plan an effective deployment process for eServices 8.1. It covers these topics:

- [Overview, page 37](#)
- [Component Distribution, page 38](#)
- [Estimating Database Size, page 39](#)
- [Configuration Worksheet, page 42](#)
- [Connections Table, page 52](#)

For further information about specific issues, see also the “Ongoing Administration and Other Topics” chapter of the *eServices 8.1 User's Guide*.

Note: Genesys recommends that you configure eServices using the wizards provided with it and that, for Windows deployments, you install it using the integrated installation package. These utilities configure and install all eServices components

Overview

Chapter 5, beginning on [page 97](#), and Chapter 10, beginning on [page 185](#), describe deployment in detail. Before beginning the deployment process there are several things you can think about and do to make the process easier.

Genesys recommends that you configure eServices using the wizards provided with it and that, for Windows deployments, you install it using the integrated installation package. These utilities configure and install all eServices components.

This chapter provides general information on deployment planning. For further information about specific issues, see also the “Ongoing Administration and Other Topics” chapter of the *eServices 8.1 User's Guide*.

Component Distribution

Genesys recommends that you divide eServices and associated components among several host machines. See [Figure 5](#) for a sample architecture.

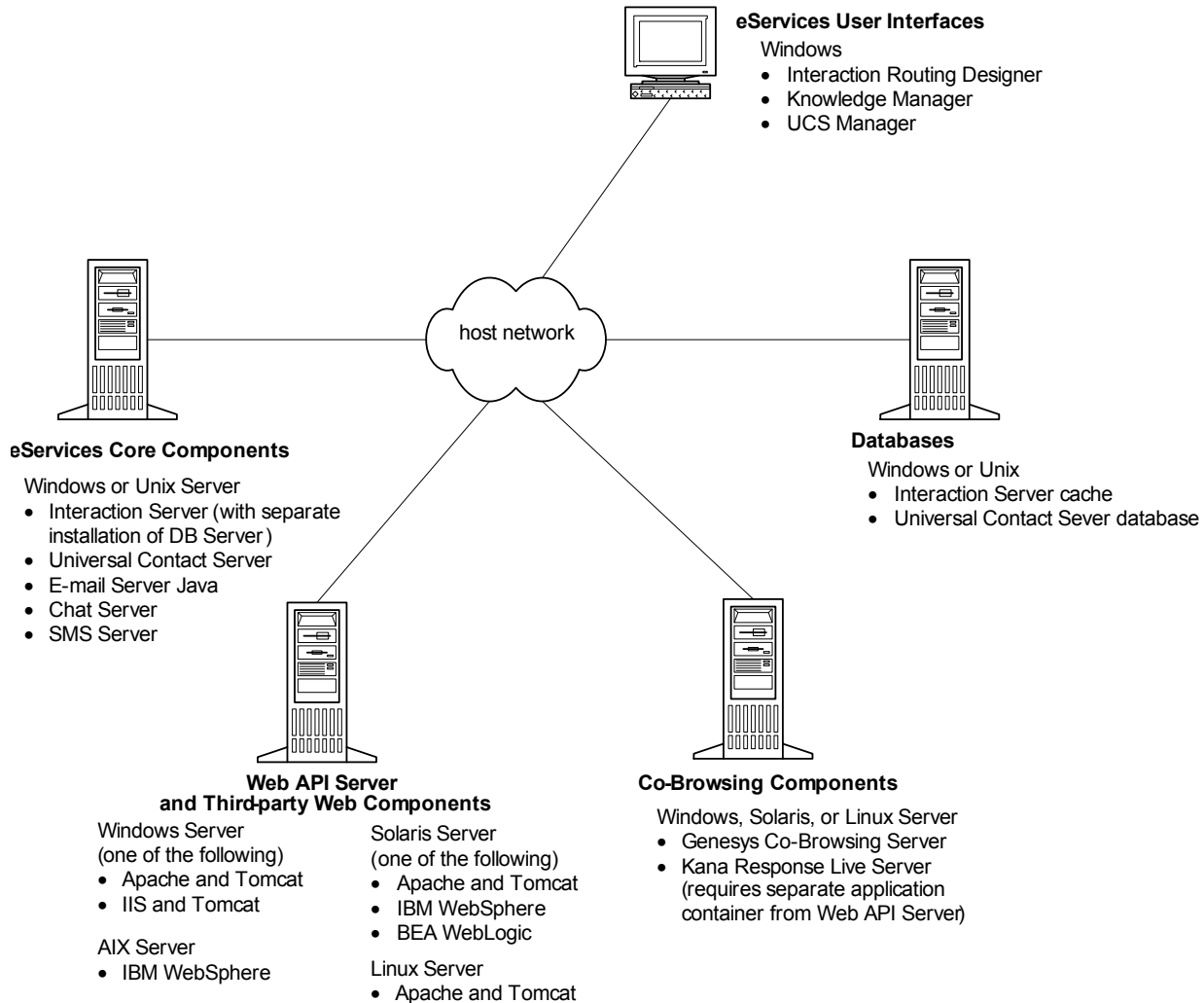


Figure 5: Sample Architecture for a Single-Tenant eServices Deployment

When choosing host machines for your eServices deployment, keep the following recommendations in mind:

- Keep web servers for different applications on separate machines. For example, Kana Response Live server installs its own instance of the Apache Tomcat application container, which you should keep separate from the Apache Tomcat used for Web API Server (according to the procedures available in this guide).
- Install eServices graphical user interface (GUI) applications on Windows hosts only. These GUI applications include:
 - Knowledge Manager

- Universal Contact Server Manager
- Interaction Workflow Samples
- Based on the load and nature of contact center media (e-mail, chat, or blended), you might need to deploy the following components on separate machines:
 - Universal Contact Server
 - E-mail Server
 - Chat Server
 - SMS Server

For information about distributing eServices components across multiple hosts for load-balancing purposes, see the “Load Balancing and Backup Configuration” chapter of the *eServices 8.1 User’s Guide*.

- You should also deploy the corporate mail server on a separate computer.

Note: For low-level load configurations such as a functional lab environment, all components can be installed on a single host.

Estimating Database Size

This section suggests some ways of estimating the size of the Universal Contact Server (UCS) and Interaction Server databases. Please keep in mind that these are rough estimates only. The size of a database file on a hard drive depends on a number of factors other than the number of interactions it contains. For example:

- The size of a database file can vary according to the DBMS platform you are using.
- Interactions are not uniform in size. An e-mail may be as small as 4 KB or it may be 200 KB or more. Also, interactions can include attachments, which can increase size greatly.

Note: You can configure Universal Contact Server Manager to automatically archive and/or prune old messages to keep this database to a manageable size. See *eServices 8.1 Universal Contact Server Manager Help* for more information.

Event Logger functionality was introduced in Multimedia 7.6.1. This feature requires its own database. For more information on Event Logger, including creating the database, refer to the “Event Logger” section in the “Interaction Server: Advanced Topics” section of the “Ongoing Administration and Other Topics” chapter of the *eServices 8.1 User’s Guide*.

Procedure: Estimating UCS database size for e-mail interactions

Purpose: To calculate the estimated amount of disk space required for the UCS database to store e-mail interactions.

Start of procedure

1. Multiply the number of interactions expected each day by the number of days you want to keep the interactions.
2. Multiply the result by 25 KB (estimating 20 KB for the message body and 5 KB for headers, comments, and indexes).

For example

If you anticipate 100 incoming interactions per day and you want to keep 90 days' worth of history, reserve $100 \times 90 \times 25$ KB or 225 MB of disk space.

Note: Though the average interaction will probably be less than 25 KB in size, this is a reliable estimate of the amount of disk space required for contact and history records

3. Add 1 KB for each related contact in the database. So for 100,000 contacts, allow 100 MB for the database.

End of procedure

Next Steps

- [Procedure: Estimating UCS database size for chat interactions](#)

Procedure: Estimating UCS database size for chat interactions

Purpose: To calculate the estimated amount of disk space required for the UCS database to store chat interactions.

Start of procedure

1. Multiply the number of chat interactions expected each day by the number of days you want to keep the interactions.
2. Multiply the result by 5 KB.

For example

If you anticipate 1000 incoming chat interactions per day, and you want to keep 90 days' worth of history, reserve $1000 \times 90 \times 5$ KB, or 450 MB of disk space.

3. Add 1 KB for each related contact in the database. So for 100,000 contacts, allow 100 MB for the database.

End of procedure**Next Steps**

- [Procedure: Estimating the size of the Interaction Server database](#)

Procedure:
Estimating the size of the Interaction Server database

Purpose: To calculate the estimated amount of disk space required by the Interaction Server database.

Start of procedure

1. Use the following formula:

$$(3.5 + d)m = x \text{ KB}$$

Where:

- 3.5 KB is the size of system data and business properties (also called operational data). See the initialization script for the particular limit for each property.
- d is the size, in KB, of attached user data. Attached data is normally 4 KB or less, but it may be more. eServices imposes an upper limit on the size of attached user data written in the logs. However, there is no limit set for the total amount of attached data that Interaction Server has to process.

Tip: For more information about limits, see the *eServices 8.1 User's Guide*.

- m is the expected maximum number of active interactions. In choosing this number for Genesys E-mail, take into consideration the large number of e-mails that may build up during off hours or system malfunctions.

Tip: For example, if you limit attached user data to 64 KB and you expect a maximum of 10,000 active interactions, allow about 700 MB for the database.

End of procedure

Next Steps

- [Procedure: Completing the configuration worksheet](#)

Configuration Worksheet

Before you begin deployment, it is helpful to assemble information that the configuration wizards will require you to supply.

Use the following worksheet (see [Table 1](#)) to plan your eServices deployment. After filling out the worksheet you are ready to start the configuration wizards.

This worksheet lists:

- Values (mostly Application object names and port numbers) that the wizards ask you to invent. You can invent them ahead of time in case you want to incorporate some systematically (for example, you may want all Application object names to start with a certain prefix, or you may want port numbers to be consecutive or to all start with the same two digits).
- Model Setup Values used during our model deployment. These provide a reference to help you during the deployment procedures.
- Names (or other attributes) of non-eServices objects that the wizards ask you to enter. Some of these are likely to exist already. Of those which do not, you can create them before, after, or (in some cases) while running the eServices Wizard. Here are two examples:
 - eServices requires several databases (tablespaces): one for Interaction Server, two for UCS, and one for Co-Browsing Server (if you plan on deploying it). See “Databases and Associated Configuration Objects” on [page 44](#) for more information.
 - eServices requires a Person or Access Group object that UCS can use to log in to Configuration Server (see “Login account” on [page 47](#)).

The wizards also ask you to select some items from a list. This worksheet does not list most items of this type, but they are described in Chapter 5, beginning on [page 97](#).

Procedure: Completing the configuration worksheet

Purpose: To provide a printable reference sheet that lists all the parameters required to deploy an eServices solution.

Start of procedure

1. Print out [Table 1](#).
2. Add deployment information for your environment in the Your Value column.
3. Keep a copy of this worksheet with you during the deployment process.

End of procedure

Next Steps

- If you already have a web server and servlet for Web API Server installed, continue with Chapter 5 on [page 97](#).
- If you still need to install a web server and servlet, before you start the configuration of eServices objects, continue with:

Windows Chapter 4 on [page 55](#)

UNIX Chapter 10 on [page 185](#)

Table 1: Configuration Worksheet

Field	Value	Model Setup Value	Description
Log In to Wizard Manager			
User name			User name for logging in to Configuration Server
User password			Password for logging in to Configuration Server
Application			Configuration Server Manager Application object name
Host			Host of Configuration Server
Port			Port on Configuration Server host where Configuration Server listens for queries

Table 1: Configuration Worksheet (Continued)

Field	Value	Model Setup Value	Description
Solution			
Name		MM81	Invent a name for your eServices solution.
Tenant (in multi-tenant environment)			
Name			Name of the Tenant object holding the eServices solution that you are configuring.
Databases and Associated Configuration Objects Note: Before installing eServices, you must configure and install a DBMS and create four databases/tablespaces, one for Interaction Server, two for Universal Contact Server (one main, one archive), and one for Co-Browsing Server (if you plan on deploying it). You can do this before or after running the eServices Wizard.			
Interaction Server Database Objects			
Database Access Point (DAP)			
DAP Name		MM81_Ixn_DAP	Invent a name for the DAP to use with Interaction Server. If you intend to use the Event Logger or message queue functionality you will need additional DAPs. Configuration and deployment is explained in the “Interaction Server: Advanced Topics” section of the “Ongoing Administration and Other Topics” chapter of the <i>eServices 8.1 User’s Guide</i> .

Table 1: Configuration Worksheet (Continued)

Field	Value	Model Setup Value	Description
DB Server			
Name		MM81_IxnDBServer	Name of the DB Server application object.
DBMS type		Microsoft SQL Server	Type of database management system you are using (Microsoft SQL, Oracle, DB2).
Host		MMHost	Host where you want DB Server installed.
Port		6110	Port where DB Server listens for queries.
SNMP Port		Not included in the model setup.	Port where DB Server listens for management commands (optional).
DBMS name		MMHost	Name of your database management system.
DBMS type		Microsoft SQL Server	Type of database management system you are using (Microsoft SQL, Oracle, DB2).
Database name		IxnDB	Name of the database (tablespace) for Interaction Server.
User name		sa	User name for logging in to the DBMS.
Password		<password for user sa>	Password for logging in to the DBMS.

Table 1: Configuration Worksheet (Continued)

Field	Value	Model Setup Value	Description
Universal Contact Server (UCS) Database Objects			
DB Server			
Not needed because UCS uses JDBC to connect with the database.			
Database Access Point (DAP)			
DAP name		MM81_UCS_DAP	Invent a name for the DAP to use with UCS.
Host		MMHost	Host name where database is running.
Port		1433	Port where the database listens for queries (for example, 1521 is the default for Oracle, 1433 for Microsoft SQL, 50000 for DB2).
DBMS type		Microsoft SQL Server	Type of database management system you are using (Microsoft SQL, Oracle, DB2).
Database name		Customer	Name of the database (tablespace) for UCS.
User name		sa	User name for logging in to the DBMS.
Password		<password for user sa>	Password for logging in to the DBMS.
Universal Contact Server			
Name		MM81_UCS	Invent a name for your UCS Application object.
Host		MMHost	Host where you want UCS installed.
Port		6120	Listening port for requests.

Table 1: Configuration Worksheet (Continued)

Field	Value	Model Setup Value	Description
Universal Contact Server API port		Default port value	Port for Remote Method Invocation (RMI) connection to the UCS API. Must be different from standard server port.
Login account		Not specified for the model setup. See the description, and choose a Person or Access Group accordingly.	The User Account (Person) or Access Group that UCS will use to log in to Configuration Server. Note: This Person or Access Group must have Administrator or higher access rights so it can update information in the Configuration Database. Use an existing account with these rights or create one for the purpose.
Multimedia Switch			
Name		In the model setup, we select New-Style Interaction Server, which creates a Multimedia switch and switching office automatically in the background.	Select or create a multimedia switch. Note: Include only one multimedia switch per tenant to preserve Stat Server and URS support for Interaction Server.
Switching Office			
Name		In the model setup, we select New-Style Interaction Server, which creates a Multimedia switch and switching office automatically in the background.	Select or create a switching office.

Table 1: Configuration Worksheet (Continued)

Field	Value	Model Setup Value	Description
Interaction Server			
Name		MM81_IxnSrv	Invent a name for your Interaction Server Application Object.
Host		MMHost	Host where you want Interaction Server installed.
Port		6130	Port where Interaction Server listens for queries.
Stat Server			
Name		In the model setup, we select the existing, properly configured prerequisite Stat Server. See Table 5 for details.	Select or create a Stat Server.
Host			Host of Stat Server.
Port			Port where Stat Server listens for queries.
Application Cluster (optional)			
Name		Model setup does not use the Application Cluster.	Invent a name for your Application Cluster.
Web API Server			
Name		MM81_WebAPIServer	Invent a name for your Web API Server Application Object.
Host		MMHost	Host where you want Web API Server installed.
Port		80	Port for web server (defaults are 80 for HTTP, 443 for HTTPS); used for load balancing between web clients.
UCS Manager			
Name		MM81_UCSMgr	Invent a name for your UCS Manager Application Object.
E-Mail Server (for Genesys E-mail)			
Name		MM81_E-mailServer	Invent a name.

Table 1: Configuration Worksheet (Continued)

Field	Value	Model Setup Value	Description
Host		MMHost	Host where you want E-mail Server installed.
Port		6150	Listening port for requests
POP server		MMHost	The name of your corporate POP server.
SMTP server		MMHost	The name of your corporate SMTP server if different from the POP server.
POP3 user name and password		Not specified in the model setup. Use valid user information for your corporate mail server.	Address, user name and password for logging on to your POP server.
Default e-mail address domain		MMHost	Domain to be appended to e-mail (both incoming and outgoing) that does not already have a domain specified in its To address field.
External agent e-mail address		external@MMHost	Address to be used as the From address in messages to external agents and as the To address in messages from external agents. Typically, this is different from your general corporate e-mail addresses.
Integrated Web Form processing port (optional)		7777	Listening port number for Web Form-processing entry point.
Automated Reply Agent		Agent1	Notional agent name used in automatic responses. Typically this is something generic like Genesys Technical Support.

Table 1: Configuration Worksheet (Continued)

Field	Value	Model Setup Value	Description
E-mail account(s) for customer access		<mailboxName1>@MM Host <mailboxName2>@MM Host	Account name and e-mail address. These are the accounts on your corporate POP server that E-mail Server will pull e-mails from and bring them into the eServices system. Be sure to also create them on your corporate mail server if they do not exist already.
Classification Server			
Name		MM81_ClassificationServer	Invent a name.
Host		MMHost	Host where you want Classification Server installed.
Port		6160	Listening port for requests.
Training Server			
Name		MM81_TrainingServer	Invent a name.
Host		MMHost	Host where you want Training Server installed.
Port		6170	Listening port for requests.
Knowledge Manager			
Name		MM81_KnowledgeManager	Invent a name.
Chat Server (for Genesys Chat)			
Name		MM81_ChatServer	Invent a name.
Host		MMHost	Host where you want Chat Server installed.
Port (ID=default)		6180	Listening port for requests.
Port (ID=webapi) (recommended)		6181	A specific port for web client connections only.

Table 1: Configuration Worksheet (Continued)

Field	Value	Model Setup Value	Description
Port (ID=ESP)		6182	Listening port for ESP requests.
ESP default nickname		system	Name to be used in place of agent's name for use in automated chatting.
Co-Browsing Server			
Name		MM81_Co-BrowsingServer	Invent a name.
Host		MMHost2	Host on which you want Co-Browsing Server installed.
Port		6195	Listening port for requests.
Co-Browsing Server alias			Invent a four-character name.
Co-Browsing Server Database Access Point (DAP)			
Note: If you use the UCS DAP for Co-Browsing Server as well, this is not needed. Genesys does not recommend using the same DAP.			
DAP name		In the model setup, we use the UCS DAP and Database for Co-Browsing Server.	Invent a name for the DAP to use with Co-Browsing Server.
Host			Host name where database is running.
Port			Port where the database listens for queries (for example, 1521 is the default for Oracle, 1433 for Microsoft SQL, 50000 for DB2).
DBMS type			Type of database management system you are using (Microsoft SQL, Oracle, DB2).
Database name			Name of the database (tablespace) for Co-Browsing Server.

Table 1: Configuration Worksheet (Continued)

Field	Value	Model Setup Value	Description
User name		In the model setup, we use the UCS DAP and Database for Co-Browsing Server.	User name for logging on to the DBMS.
Password			Password for logging on to the DBMS.

Connections Table

Table 2 on [page 52](#) lists connections that you set in the Configuration Layer for eServices components. eServices 8.1 supports Transport Layer Security (TLS) and client-side port functionality for some connections. See Chapter 7 on [page 169](#) for more information.

In addition to the information in [Table 2](#), keep in mind that:

- Connections to Message Server are required only if you are planning to use it for application log output.
- A Local Control Agent (LCA) runs on each host. All Genesys components on a host connect to the local LCA. Refer to the *Framework 8.1 Deployment Guide* for details on configuring LCAs.
- For Reporting purposes, you must also add a connection from your Reporting Stat Server to Interaction Server (or to Interaction Server Proxy; see “Interaction Server Proxy as Reporting Bus” on [page 154](#)).

Note: It is possible to have multiple connections to some components. These components are marked with a *. For information about how and why you use multiple connections, please see the “Load Balancing and Configuration” chapter of the *eServices 8.1 User’s Guide*.

Table 2: eServices Application Connections

Application	Connects to
Chat Server	Interaction Server Message Server Universal Contact Server
Classification Server	Message Server Universal Contact Server

Table 2: eServices Application Connections (Continued)

Application	Connects to
E-mail Server	Interaction Server Message Server Universal Contact Server
Interaction Server	Chat Server (via the ESP port)* Classification Server* E-mail Server* Interaction Server DAP* Message Server SMS Server Stat Server (for reporting)* Universal Contact Server* Application Cluster*
Knowledge Manager	Universal Contact Server
Training Server	Message Server Universal Contact Server
Universal Contact Server	Message Server Stat Server* UCS DAPs
Universal Contact Server Manager	Universal Contact Server
Co-Browsing Server	JDBC-based DAP Message Server
SMS Server	Message Server Chat Server* Interaction Server Solution Control Server

Table 2: eServices Application Connections (Continued)

Application	Connects to
Web API Server	Application Cluster* Chat Server (via the webapi port)* Co-Browsing Server E-mail Server* Interaction Server Message Server Solution Control Server Stat Server* Universal Contact Server* Web API Server ^a

a. If this Web API Server is acting as a load balancer for other Web API Servers.

4

Deploying Third-Party Web Applications

A web server and a web application container are prerequisites for deploying eServices. Your environment may already have its own established configuration of these components. This chapter describes the steps you must take to install and configure common combinations of these applications, including the Apache Web Server and the Apache Tomcat web application container in Windows, Linux, and Solaris environments, as well as the initial setup for WebSphere in AIX and Solaris environments. Be sure to check the [Genesys Supported Operating Environment Reference Guide](#) for the supported versions of the third-party software you are using. This chapter includes these sections:

- [Java Software Development Kit, page 56](#)
- [Installing Apache and Tomcat on Windows, page 57](#)
- [Connecting Tomcat and its Web Server, page 64](#)
- [Configuring Apache and Tomcat for Linux or Solaris, page 76](#)
- [Deployments with WebSphere on AIX or Solaris, page 77](#)
- [Deploying KANA Response Live for co-browsing, page 78](#)

Note: It is important to distinguish Third Party *Components* from third-party *applications*.

eServices Third Party Components—refers to a collection of files, mostly Java libraries, that eServices uses for its normal operation. Third Party Components are included in the integrated install, as described in [Procedure: Performing an integrated installation of all eServices servers](#), on [page 138](#). In eServices 8.0.1, Third Party Components were renamed to Java Environment and Libraries for eServices and UCS. The Java Environment and Libraries for eServices and UCS installation package places these files in various locations on the host machine.

ThirdPartyApplications—refers to a directory on the eServices product CD that contains the web server Apache, the application container Tomcat, and various supporting files. eServices requires a web server and an application container, but Apache and Tomcat are not the only supported types, and you are not required to install the particular copies in this directory; they are provided as a convenience.

Java Software Development Kit

Procedure:

Downloading and installing the Java SDK

Purpose: Download and install the Java SDK to support Tomcat functionality (web server-side Java content compilation and execution).

Start of procedure

1. Determine the correct Java software development kit (SDK) for your system. For supported releases of Java SDK on Windows, Solaris, Linux, and AIX systems, see the [Genesys Supported Operating Environment Reference Guide](#).

Tip: Web API Server 8.1 supports JDK 5.0 and 6.0, which use the extended version number format of 1.5.0.

2. Select one of the URLs from the list below, according to your system. Paste the URL into your browser, replacing <release_number> with 1.5.0, or the supported release number that you found in [Step 1](#).

Solaris http://java.sun.com/j2se/<release_number>/install-solaris.html

Linux http://java.sun.com/j2se/<release_number>/install-linux.html

Windows http://java.sun.com/j2se/<release_number>/install-windows.html

AIX <http://www-1.ibm.com/support/docview.wss?rs=180&context=SSEQTP&uid=swg24006713>

3. Make sure the JAVA_HOME environment variable points to the directory where Java SDK is located. For example, on Solaris it might be this path:

```
JAVA_HOME=/usr/local/jdk1.5.0_22
```

```
export JAVA_HOME
```

Or for Windows, set the JAVA_HOME environment variable to something like
c:\jdk1.5.0_22

End of procedure

Next Steps

- If you are deploying the Apache HTTP server and Tomcat Application Container on a Windows platform, see [Procedure: Installing Apache and Tomcat on Windows](#), on [page 57](#).
- If you are deploying the Apache HTTP server and Tomcat Application Container on a Solaris or Linux platform, see [Procedure: Configuring Apache and Tomcat for Linux or Solaris](#), on [page 76](#).

Installing Apache and Tomcat on Windows

This section describes procedures for installing the Apache Web Server and the Tomcat web application container in a Windows environment. The installation packages for both Apache and Tomcat are included on your eServices 8.1 CD.

If you are using Microsoft's Internet Information Server (IIS) as your web server, skip the instructions for installing Apache, proceed to the Tomcat

instructions, and then use the steps in “Setting Up the JK1 Connector with IIS 6 or IIS 7” on [page 66](#) for further configuration details.

Apache

Genesys provides the Apache Web Server with your eServices software. Look for it in the third-party applications portion of the installation CDs.

Procedure: Installing the Apache Web Server on Windows

Purpose: To install the Apache Web Server on Windows.

Start of procedure

1. Shut down any other web servers (such as IIS) on the machine where Apache is to run.
2. On the Interaction Management CD, go to the directory `ThirdPartyApplications\Apache<release_number>\Windows\binary`. Find and double-click `apache_<release_number>-win32-x86-no_ssl.msi`.
3. Click Next at the Welcome window.
4. Accept the license agreement, and click Next.
5. Click Next at the Read This First page.
6. At the Server Information prompt, enter the following information:
 - a. Network domain
 - b. Server name

Note: Servers may have an external name that differs from the internal name. Be sure to use the right name for your situation.

- c. Administrator's e-mail address.
 - d. Choose whether to run as a service. Keep the default if you want Apache running as a service; otherwise select the radio button to manually start and stop Apache. Click Next to continue.
7. Choose a typical or custom install.
 8. Choose a destination folder. Genesys recommends that you use a short path name, such as `C:\ApacheGroup`.

Warning! Do *not* include any spaces in the folder name (for example, use `ApacheGroup` instead of `Apache Group`).

9. Click **Install**. The Installation Wizard uses the settings you specified to install and configure Apache.
10. Click **Finish**. The installation is complete.

Note: Throughout this document, <APACHE_HOME> represents the folder you define here for your Apache installation. During installation, a sub-folder named Apache2 may be created. If this is the case, <APACHE_HOME> represents the folder you define here as well as the sub-folder Apache2 (C:\ApacheGroup\Apache2).

End of procedure

Next Steps

- If you plan on using Genesys Web Collaboration (Co-Browsing Server), you need to implement a Secure Socket Layer (SSL) for your web server. See [Procedure: Installing a new SSL certificate using IIS](#), on page 80.
- Otherwise, proceed with [Procedure: Testing the Apache Web Server on Windows](#).

Procedure: Testing the Apache Web Server on Windows

Purpose: To verify the Apache Web Server installation was successful.

Prerequisites

- Apache Web Server has been installed. See [Procedure: Installing the Apache Web Server on Windows](#), on page 58.

Start of procedure

1. From the Windows taskbar, select **Start > Programs > Apache HTTP Server <release_number> > Control Apache Server > Start**.

Note: If Apache is already running, then the Start command displays an error message. Continue with [Step 2](#) to confirm that Apache is running correctly.

2. Open a web browser and go to Apache's host (http://<servername_or_IP>:80). If Apache is running correctly, you will see a white page with the message **It works!**

End of procedure

Next Steps

- Install the Tomcat web application. See [Procedure: Installing the Tomcat web application container on Windows](#), on page 60

Tomcat

Genesys provides the Tomcat web application container with your eServices software. Look for it in the ThirdPartyApplications folder of the installation CDs.

Note: The 8.1 release of Genesys eServices supports only specific releases of Tomcat. Check the [Genesys Supported Operating Environment Reference Guide](#) for more information.

Procedure: Installing the Tomcat web application container on Windows

Purpose: Tomcat is needed to provide standards-compliant support for servlets and JSP pages.

Start of procedure

1. On the eServices Interaction Management CD, go to the directory ThirdPartyApplications\Tomcat_Servlet<release_number>\Windows\binary. Find and double-click apache-tomcat-<release_number>.exe.
2. At the Welcome to the Apache Tomcat Setup Wizard window, click Next.
3. Click **I Agree** to accept the license agreement.
4. At the Installation Options window, select **Full** from the Select type of install drop-down list (see Figure 6 on [page 61](#)).

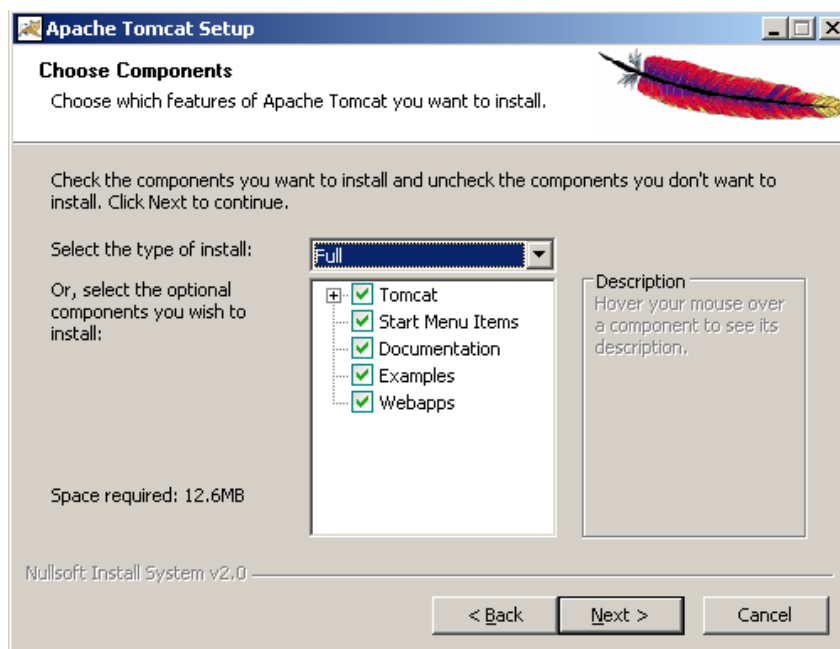


Figure 6: Tomcat Installation Options

5. Click Next to continue.
6. At the Choose Install Location window, enter a path to your Tomcat installation location, and click Next.

Warning! Do not include any spaces in the folder name (for example, use Tomcat6 instead of Tomcat 6).

7. Specify the HTTP/1.1 Connector Port and Administrator Login details for your Tomcat installation, and click Next.

Tip: For the HTTP/1.1 Connector Port, you can use the default value of 8080.

8. Confirm the location of your Java SDK, and click Install. The Installation Wizard uses the settings you specified to install and configure Tomcat.
9. Click Finish. The installation is complete.
10. Verify the values of the following environment variables (Start/My Computer/Properties/Advanced/Environment Variables), or create them if they do not already exist:
 - CATALINA_HOME—Your Tomcat installation directory.
 - JAVA_HOME—Your JDK base directory, as described under “Java Software Development Kit” on [page 56](#).

Note: Throughout this document, <CATALINA_HOME> represents the base directory of your Tomcat installation.

End of procedure

Next Steps

- [Procedure: Testing the Tomcat installation on Windows.](#)

Procedure: Testing the Tomcat installation on Windows

Purpose: To verify the Tomcat installation was successful.

If they exist, you should always start and stop Tomcat using the batch files <CATALINA_HOME>\bin\startup.bat and <CATALINA_HOME>\bin\shutdown.bat. Using shutdown.bat to stop Tomcat stops the executable and also closes various files that would otherwise remain open.

If there are no batch files, use the Tomcat Monitor (visible as an icon in the system taskbar) to start and stop Tomcat.

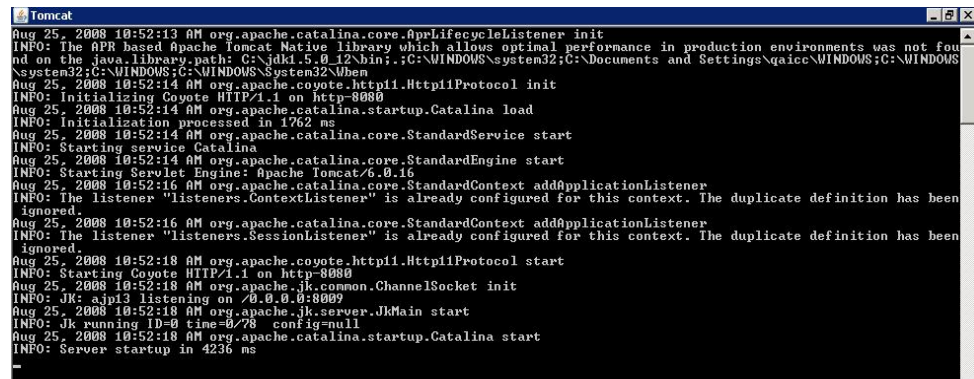
Tip: You can find an icon for the Tomcat Monitor in the system taskbar:



To start or stop Tomcat, right-click the Tomcat Monitor icon and select your action: Start service, Stop service, and so on.

Start of procedure

1. Verify that Apache is already running.
2. Start Tomcat using <CATALINA_HOME>\bin\startup.bat. If Tomcat is running correctly, you will see a console window (see [Figure 7](#)).



```

Aug 25, 2008 10:52:13 AM org.apache.catalina.core.AprLifecycleListener init
INFO: The APR based Apache Tomcat Native library which allows optimal performance in production environments was not found on the java.library.path: C:\jdk1.5.0_12\bin;.;C:\WINDOWS\system32;C:\Documents and Settings\qaicc\WINDOWS\C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem
Aug 25, 2008 10:52:14 AM org.apache.coyote.http11.Http11Protocol init
INFO: Initializing Coyote HTTP/1.1 on http-8080
Aug 25, 2008 10:52:14 AM org.apache.catalina.startup.Catalina load
INFO: Initialization processed in 1762 ms
Aug 25, 2008 10:52:14 AM org.apache.catalina.core.StandardService start
INFO: Starting service Catalina
Aug 25, 2008 10:52:14 AM org.apache.catalina.core.StandardEngine start
INFO: Starting Servlet Engine: Apache Tomcat/6.0.16
Aug 25, 2008 10:52:16 AM org.apache.catalina.core.StandardContext addApplicationListener
INFO: The listener "listeners.ContextListener" is already configured for this context. The duplicate definition has been ignored.
Aug 25, 2008 10:52:16 AM org.apache.catalina.core.StandardContext addApplicationListener
INFO: The listener "listeners.SessionListener" is already configured for this context. The duplicate definition has been ignored.
Aug 25, 2008 10:52:18 AM org.apache.coyote.http11.Http11Protocol start
INFO: Starting Coyote HTTP/1.1 on http-8080
Aug 25, 2008 10:52:18 AM org.apache.jk.common.ChannelSocket init
INFO: JK: ajp13 listening on /0.0.0.0:8009
Aug 25, 2008 10:52:18 AM org.apache.jk.server.JkMain start
INFO: JK running ID=0 time=0/78 config=null
Aug 25, 2008 10:52:18 AM org.apache.catalina.startup.Catalina start
INFO: Server startup in 4236 ms

```

Figure 7: Tomcat Console Window

- Further test the Tomcat installation by opening a browser to <http://localhost:8080>. You should see the web page displayed in Figure 8.

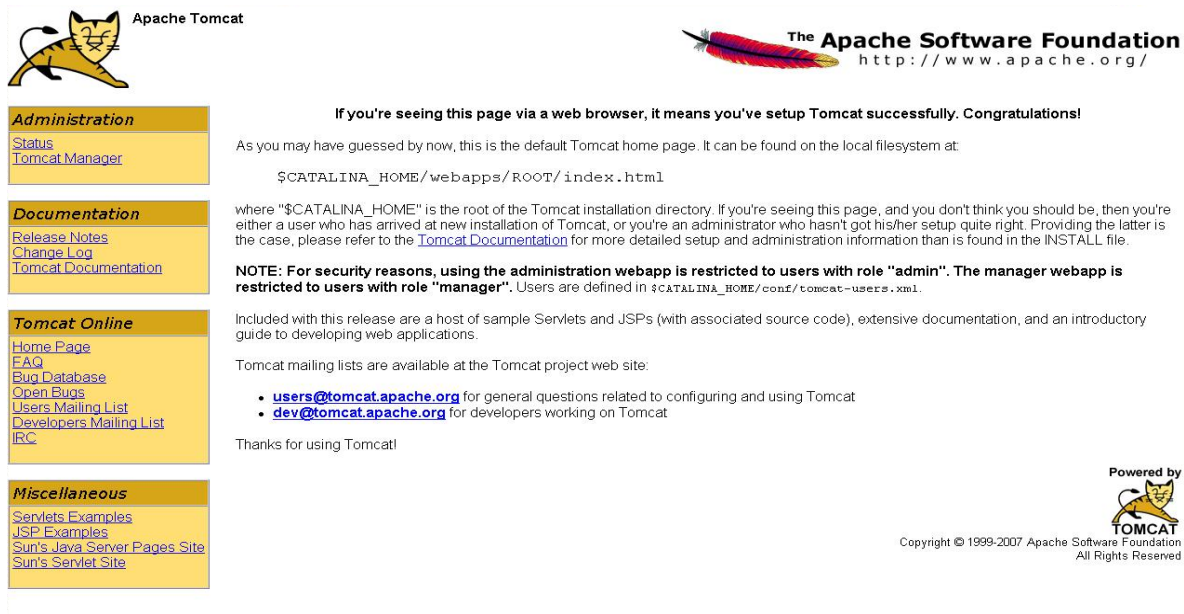


Figure 8: Successful Tomcat Installation Web Page

- Shut down Tomcat. Use `shutdown.bat` if it exists, or press `Ctrl+C` from the console window if the batch file does not exist.
- Shut down Apache. Select `Start > Programs > Apache HTTP Server <release_number> > Control Apache Server > Stop`.

End of procedure

Next Steps

- Connect Tomcat and its web server. See [Procedure: Installing the JK1 Connector with Apache](#), on page 64 or [Procedure: Installing the JK1 connector with IIS 6](#), on page 66.

Connecting Tomcat and its Web Server

This section describes connecting Tomcat and its web server on the Windows platform.

Setting Up the JK1 Connector with Apache

Procedure: Installing the JK1 Connector with Apache

Purpose: To install the JK1 connector to invisibly integrate Tomcat with a web server such as Apache or IIS.

Prerequisites

- Apache web server is installed. See [Procedure: Installing the Apache Web Server on Windows](#), on page 58.
- Tomcat is installed. See [Procedure: Installing the Tomcat web application container on Windows](#), on page 60.

Start of procedure

1. On the Interaction Management CD, navigate to `ThirdPartyApplications\Tomcat\jk_apache_for_windows\binary` and locate the file that matches your version of Apache:
 - `mod_jk-1.2.28-httpd-2.2.3.so` (for Apache 2.2 or later)
 - `mod_jk-1.2.28-httpd-2.0.52.so` (for Apache 2.0.52 or later)
2. Copy this file to your `<APACHE_HOME>/modules` directory, and rename it to `mod_jk.so`.
3. In the `<CATALINA_HOME>/conf/jk` directory, create a file called `workers.properties`, with the following contents:

```
worker.list=ajp13
worker.ajp13.port=8009
worker.ajp13.host=localhost
worker.ajp13.type=ajp13
```

Note: If there is no `jk` directory, create one.

4. In the `<APACHE_HOME>/conf` directory, create a file called `mod_jk.conf`, with the following contents:

```
<IfModule !mod_jk.c>
  LoadModule jk_module "<APACHE_HOME>/modules/mod_jk.so"
</IfModule>
JkWorkersFile "<CATALINA_HOME>/conf/jk/workers.properties"
JkLogFile "<CATALINA_HOME>/logs/mod_jk.log"
JkLogLevel emerg
JkMount /examples/* ajp13
```
5. Add the following statement to the end of the `<APACHE_HOME>/conf/httpd.conf` file:

```
Include "<APACHE_HOME>/conf/mod_jk.conf"
```

End of procedure

Next Steps

- After completing the installation, you should test the JK1 connector to ensure that it functions properly. See [Procedure: Testing the JK1 Connector with Apache](#).

Procedure: Testing the JK1 Connector with Apache

Purpose: To verify the JK1 connector is working correctly.

Prerequisites

- JK1 connector is installed. See [Procedure: Installing the JK1 Connector with Apache](#), on page 64.

Start of procedure

1. Start Tomcat and Apache.
Open a web browser and enter the following address:
`http://localhost/examples/jsp/jsp2/el/basic-arithmetic.jsp`
You should see the web page shown in [Figure 9](#).

JSP 2.0 Expression Language - Basic Arithmetic

This example illustrates basic Expression Language arithmetic. Addition (+), subtraction (-), multiplication (*), division (/ or div), and modulus (% or mod) are all supported. Error conditions, like division by zero, are handled gracefully.

EL Expression	Result
$\$(1)$	1
$\$(1 + 2)$	3
$\$(1.2 + 2.3)$	3.5
$\$(1.2E4 + 1.4)$	12001.4
$\$(-4 - 2)$	-6
$\$(21 * 2)$	42
$\$(3/4)$	0.75
$\$(3 \text{ div } 4)$	0.75
$\$(3/0)$	Infinity
$\$(10\%4)$	2
$\$(10 \text{ mod } 4)$	2
$\$((1==2) ? 3 : 4)$	4

Figure 9: JK1 Test Page

End of procedure

Next Steps

- If you want your eServices solution to include co-browsing functionality, see [Procedure: Deploying KANA Response Live for co-browsing](#), on [page 78](#).
- Otherwise, you can begin to deploy the Model Setup of the eServices components. See Chapter 5 on [page 97](#).

Setting Up the JK1 Connector with IIS 6 or IIS 7

eServices 8.1 supports IIS 6 and IIS 7.

Note: Make sure that IIS 6 Management Compatibility is installed on the host with IIS 7 (go to Server Manager > Roles > Web Server (IIS) > Management Tools and install “IIS 6 Management Compatibility” if it is not already installed).

The procedures for setting up the JK1 connector with each version is very similar, but there are a few minor differences, so the instructions are broken into two separate procedures. Refer to the procedure for the version of IIS that you are using.

Procedure: Installing the JK1 connector with IIS 6

Purpose: To install the JK1 connector with IIS 6.

Prerequisites

- Tomcat is installed. See [Procedure: Installing the Tomcat web application container on Windows](#), on page 60.

Start of procedure

1. On the Interaction Management CD, go to ThirdPartyApplications\Tomcat\jk_apache_for_windows\binary, locate isapi_redirect<release number>.dll, and place it in c:\jakarta-tomcat\bin.
2. Create the properties files workers.properties and uriworkermap.properties in c:\jakarta-tomcat\conf.
Here is a minimum workers.properties, using just ajp13 to connect your IIS web server to the Tomcat engine.

```
# workers.properties
#
# This file provides minimal jk configuration properties needed to
# connect to Tomcat.
# Define 1 real worker using ajp13
#
worker.list=ajp13w
#
# Set properties for worker ajp13w
# Note that the name and the type do not have to match.
#
worker.ajp13w.type=ajp13
worker.ajp13w.host=localhost
worker.ajp13w.port=8009
```

Here is a minimum example of uriworkermap.properties:

```
# This file provides sample mappings for example
# ajp13w worker defined in workermap.properties
/examples/*=ajp13w
```

Note: The workers.properties and uriworkermap.properties files must be saved in Microsoft Windows/MS-DOS format. Otherwise, an error will occur when installing the Web API Server and setup will not proceed.

3. In the registry, create a new registry key named:
HKEY_LOCAL_MACHINE\SOFTWARE\Apache Software Foundation\Jakarta Isapi Redirector\1.0
4. Add a string value with the name extension_uri and a value of
/jakarta/isapi_redirect.dll.

5. Add a string value with the name `log_file` and a value pointing to where you want your log file to be (for example `c:\jakarta-tomcat\logs\isapi.log`).
6. Add a string value with the name `log_level` and a value for your log level (can be debug, info, error or emerg).
7. Add a string value with the name `worker_file` and a value which is the full path to your `workers.properties` file (for example, `c:\jakarta-tomcat\conf\workers.properties`).
8. Add a string value with the name `worker_mount_file` and a value which is the full path to your `uriworkermap.properties` file (for example `c:\jakarta-tomcat\conf\uriworkermap.properties`).
9. Using the IIS management console, add a new virtual directory to your IIS web site. The name of the virtual directory must be `jakarta`. Its physical path should be the directory where you placed `isapi_redirect.dll` (in our example it is `c:\jakarta-tomcat\bin`). While creating this new virtual directory assign it with execute access.
10. Using the IIS management console, add `isapi_redirect.dll` as a filter in your IIS web site. The name of the filter should reflect its task (for example, `tomcat`), its executable must be our `c:\jakarta-tomcat\bin\isapi_redirect.dll`.
11. Open IIS Manager, as shown in Figure 10.

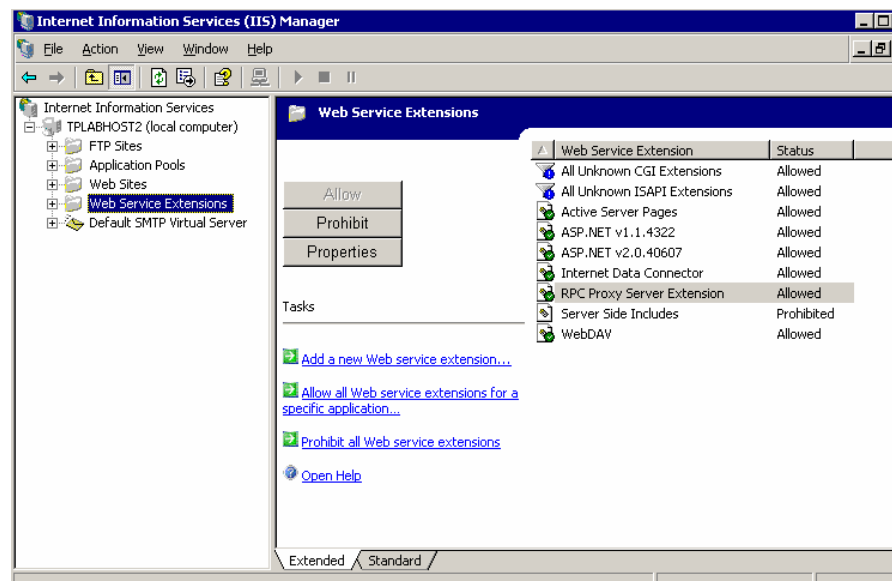


Figure 10: IIS Manager

12. Click Add New Web Service Extension. The resulting dialog box is shown in Figure 11 on page 69.

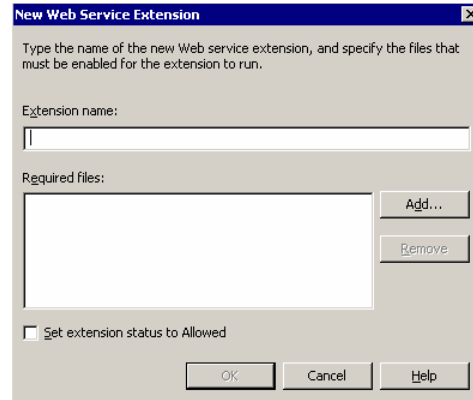


Figure 11: New Web Service Extension

13. In the New Web Service Extension dialog box:
 - a. In the Extension name box, type `jakarta`.
 - b. In the Required files box, click Add and browse to the location of the `isapi_redirect.dll` file.
 - c. Select the Set extension status to Allowed check box, then click OK.
14. Restart the IIS service. Ensure that the filter you added in [Step 10](#) is marked with a green up-pointing arrow.

End of procedure

Next Steps

- After completing the installation, you should test the JK1 connector to ensure that it functions properly. See [Procedure: Testing the JK1 Connector with IIS](#).

Procedure: Installing the JK1 connector with IIS 7

Purpose: To install the JK1 connector with IIS 7.

Prerequisites

- Tomcat is installed. See [Procedure: Installing the Tomcat web application container on Windows](#), on page 60.

Start of procedure

1. On the Interaction Management CD, go to ThirdPartyApplications\Tomcat\jk_apache_for_windows\binary, locate isapi_redirect<release number>.dll, and place it in c:\jakarta-tomcat\bin.
2. Create the properties files workers.properties and uriworkermap.properties in c:\jakarta-tomcat\conf.
Here is a minimum workers.properties, using just ajp13 to connect your IIS web server to the Tomcat engine.

```
# workers.properties
#
# This file provides minimal jk configuration properties needed to
# connect to Tomcat.
# Define 1 real worker using ajp13
#
worker.list=ajp13w
#
# Set properties for worker ajp13w
# Note that the name and the type do not have to match.
#
worker.ajp13w.type=ajp13
worker.ajp13w.host=localhost
worker.ajp13w.port=8009
```

Here is a minimum example of uriworkermap.properties:

```
# This file provides sample mappings for example
# ajp13w worker defined in workermap.properties
/examples/*=ajp13w
```

Note: The workers.properties and uriworkermap.properties files must be saved in Microsoft Windows/MS-DOS format. Otherwise, an error will occur when installing the Web API Server and setup will not proceed.

3. In the registry, create a new registry key named:
HKEY_LOCAL_MACHINE\SOFTWARE\Apache Software Foundation\Jakarta Isapi Redirector\1.0
4. Add a string value with the name extension_uri and a value of
/jakarta/isapi_redirect.dll.
5. Add a string value with the name log_file and a value pointing to where you want your log file to be (for example
c:\jakarta-tomcat\logs\isapi.log).

6. Add a string value with the name `log_level` and a value for your log level (can be `debug`, `info`, `error` or `emerg`).
7. Add a string value with the name `worker_file` and a value which is the full path to your `workers.properties` file (for example, `c:\jakarta-tomcat\conf\workers.properties`).
8. Add a string value with the name `worker_mount_file` and a value which is the full path to your `uriworkermap.properties` file (for example `c:\jakarta-tomcat\conf\uriworkermap.properties`).
9. Open IIS 7 Manager.
10. Navigate to your host.
11. Double click on the ISAPI and CGI Restrictions icon.
12. On the right hand panel, click `Add...`
13. Point the path to your `isapi_redirect.dll` file and give it a description, for example, `jakarta`.
14. Check `Allow extension path to execute`.
15. Click on the Default Website and navigate to `ISAPI Filters`.
16. On the right hand panel, click on `Add...` give the filter a name (for example, `jakarta`), and point to your `isapi_redirect.dll` file.
17. Navigate to your Default Website again.
18. Add a virtual directory and name it `jakarta`. Point this to the directory containing `isapi_redirect.dll`.
19. Click on the newly created virtual directory `jakarta`.
20. Navigate into `Handler Mappings`, on the right hand side there is a panel called `Actions`. Click on `Edit Feature Permissions` and click the `execute` permission.
21. Navigate to your host context again and do a restart.

Note: If you are running a 32 bit version of Tomcat with a 32 bit JK connector on a x64 OS, you have to configure the `Application Pool` -> `Advanced Settings` -> `Enable 32 bit Applications` = `TRUE`.

End of procedure

Next Steps

- After completing the installation, you should test the JK1 connector to ensure that it functions properly. See [Procedure: Testing the JK1 Connector with IIS](#).

Procedure: Testing the JK1 Connector with IIS

Purpose: To verify the JK1 connector is working correctly.

Prerequisites

- The JK1 connector is installed. See [Procedure: Installing the JK1 connector with IIS 6](#), on page 66 or [Procedure: Installing the JK1 connector with IIS 7](#), on page 69.

Start of procedure

1. Start Tomcat. Start or restart IIS.
2. Open a web browser and go to:
`http://localhost/examples/jsp/jsp2/el/basic-arithmetic.jsp`
You should see the web page shown in [Figure 12](#).

JSP 2.0 Expression Language - Basic Arithmetic

This example illustrates basic Expression Language arithmetic. Addition (+), subtraction (-), multiplication (*), division (/ or div), and modulus (% or mod) are all supported. Error conditions, like division by zero, are handled gracefully.

EL Expression	Result
<code>\$(1)</code>	1
<code>\$(1 + 2)</code>	3
<code>\$(1.2 + 2.3)</code>	3.5
<code>\$(1.2E4 + 1.4)</code>	12001.4
<code>\$(-4 - 2)</code>	-6
<code>\$(21 * 2)</code>	42
<code>\$(3/4)</code>	0.75
<code>\$(3 div 4)</code>	0.75
<code>\$(3/0)</code>	Infinity
<code>\$(10%4)</code>	2
<code>\$(10 mod 4)</code>	2
<code>\$((1==2) ? 3 : 4)</code>	4

Figure 12: JK1 Test Page

End of procedure

Next Steps

- Optional: [Advanced Configuration of the Tomcat Context](#).

Advanced Configuration of the Tomcat Context

To perform additional advanced configuration of your IIS-Tomcat implementation, see the following:

- [Procedure: Configuring IIS to serve static content](#), on page 73

- [Procedure: Protecting the WEB-INF Directory](#), on page 74
- [Procedure: Configuring Tomcat to serve multiple contexts](#), on page 74

Procedure: Configuring IIS to serve static content

Purpose: To configure IIS-Tomcat so that IIS handles all static content (.html, .jpg, and so on), leaving Tomcat to process the Java Server Pages (JSP) and servlets.

Genesys eServices web applications (Web API Server) use a combination of static web pages and active java content. You can reduce processing time by letting IIS serve static pages, even if these files are part of a context already served by Tomcat. For example, the .html and .gif files in the examples context do not need to be served from the Tomcat process—it is enough for IIS to serve them.

Start of procedure

1. Configure IIS to include the Tomcat contexts.

Adding a Tomcat context to IIS requires the addition of a new IIS virtual directory that covers the Tomcat context. For example, you could add a /example IIS virtual directory that covers the D:\tomcat\webapps\examples directory.

2. Configure the redirector to leave the static files for IIS.

To configure the redirector, you must edit the `uriworkermap.properties` file to specify the URL-path pattern(s) that you want Tomcat to handle (usually only JSPs and servlets). For the examples context, you must replace the following line:

```
/examples/*=ajp13
```

with the following two lines:

```
/examples/*.jsp=ajp13
```

```
/examples/servlet/*=ajp13
```

The second line is more explicit; it actually instructs the redirector to redirect only the following:

- Requests to resources under /examples/servlet/.
- Resources under /examples/ whose name ends with .jsp.

You can be more explicit and provide lines such as the following:

```
/examples/servlet/<servletname>=ajp13
```

This instructs the redirector to redirect requests whose URL path equals /examples/servlet/<servletname> to the worker named ajp13.

End of procedure

Next Steps

- If you want to do any more advanced configuration of the Tomcat context, return to the list of configuration options—see [Advanced Configuration of the Tomcat Context, page 72](#).
- If you want your eServices solution to include co-browsing functionality, see [Deploying KANA Response Live for co-browsing, page 78](#).
- Otherwise, you can begin to deploy the Model Setup of the eServices components—see [Chapter 5 on page 97](#).

Procedure:
Protecting the WEB-INF Directory

Each servlet application (context) has a special directory named WEB-INF. This directory contains sensitive configuration data and Java classes, and you must prevent web users from accessing or even seeing these directories. Although you can use the IIS management console to do this, the administrator must remember that it has been done. To avoid this complication, the redirector plugin automatically protects your WEB-INF directories by rejecting any request that contains WEB-INF in its URL-path.

Start of procedure

- You do not need to perform any actions to obtain this WEB-INF protection. The redirector plugin does this automatically.

End of procedure**Next Steps**

- If you want to do any more advanced configuration of the Tomcat context, return to the list of configuration options—see [Advanced Configuration of the Tomcat Context, page 72](#).
- If you want your eServices solution to include co-browsing functionality, see [Deploying KANA Response Live for co-browsing, page 78](#).
- Otherwise, you can begin to deploy the Model Setup of the eServices components—see [Chapter 5 on page 97](#).

Procedure:
Configuring Tomcat to serve multiple contexts

Purpose: To configure the `workers.properties` file so that different Tomcat processes can serve different contexts—for example, to spread the load among different machines.

Summary

For Tomcat to serve multiple contexts, you must:

- Define the *workers* (the Tomcat processes or instances) that you want to use.
- Assign each context its own worker.

Start of procedure

1. Open the `workers.properties` file, located in the `<CATALINA_HOME>/conf` directory.
2. In the `worker.list` entry, add workers using the following format:
`worker.list=<a comma separated list of worker names>`

Example

```
worker.list=ajp13, ajp13second
```

3. Add entries for each individual worker, one for each associated host and port, using the following format:
`worker.<worker name>.<property>=<property value>`

Example

```
worker.ajp13.host=localhost
worker.ajp13.port=8007
worker.ajp13second.host=otherhost
worker.ajp13second.port=8007
```

4. You can now assign the workers to serve the various contexts (one worker for each context).

For example, look at the following `uriworkermap.properties` fragment:

```
/examples/*=ajp13
/webpages/*=ajp13second
```

The `examples` context is served by `ajp13` while the `webpages` context is served by `ajp13second`.

End of procedure

Next Steps

- If you want to do any more advanced configuration of the Tomcat context, return to the list of configuration options—see [Advanced Configuration of the Tomcat Context, page 72](#).
- If you want your eServices solution to include co-browsing functionality, see [Deploying KANA Response Live for co-browsing, page 78](#).
- Otherwise, you can begin to deploy the Model Setup of the eServices components—see [Chapter 5 on page 97](#).

Configuring Apache and Tomcat for Linux or Solaris

This section points you to third-party documentation on how to perform further Apache and Tomcat configuration for Solaris or Linux. You can obtain third-party documentation using either of these methods:

- [Procedure: Obtaining Apache documentation from the CD](#)
- [Procedure: Obtaining Tomcat documentation from the CD and the Web](#)

Note: For information about deploying Solaris or AIX with IBM WebSphere, or Solaris with BEA WebLogic, see the following sections:

- [Deploying Genesys Web Applications for WebSphere on AIX, Solaris, or Windows, page 190](#)
 - [Deploying Genesys Web Applications for WebLogic on Solaris, page 201](#)
-

Procedure: Obtaining Apache documentation from the CD

Start of procedure

1. From the Interaction Management CD, locate the .html file applicable to your operating system from one of the following folders:

Solaris

ThirdPartyApplications\Apache2.2.14\solaris\apache_documentation.html

Linux

ThirdPartyApplications\Apache2.2.14\linux\apache_documentation.html

2. Open the apache_documentation.html file. This document can assist you in installing and building Apache for a Solaris or Linux environment.

End of procedure

Next Steps

- If you have not done so already, configure and install eServices as outlined in:
 - [Chapter 5 on page 97](#)
 - [Deploying eServices on UNIX Hosts, page 185](#)

Procedure: Obtaining Tomcat documentation from the CD and the Web

Purpose: To obtain Tomcat documentation.

Start of procedure

1. From the Interaction Management CD, locate the .html file:
`ThirdPartyApplications\TomCat_Servlet6.0.20\unix\tomcat_documentation.html`
2. Open the `tomcat_documentation.html` file. This document includes information about installing, configuring, and operating an Apache Tomcat server.
3. For documentation about installation, building, and configuring `mod_jk` for UNIX, see:
`http://tomcat.apache.org/connectors-doc/index.html`
4. The site referenced above also describes how to connect Apache to Tomcat on UNIX.

Note: You can access links to these same locations from the port 8080 page of your local host when your Tomcat server is running:
`http://localhost:8080/docs`

End of procedure

Next Steps

- If you have not done so already, configure and install eServices as outlined in:
 - [Chapter 5 on page 97](#)
 - [Deploying eServices on UNIX Hosts, page 185](#)

Deployments with WebSphere on AIX or Solaris

If you intend to use WebSphere on AIX or Solaris for eServices and its Web-based interaction capabilities, necessary components, such as an Enterprise Archive project, are dynamically generated during installation of the Genesys Web API Server for AIX or Solaris. (This feature was introduced in the 7.2 release.) Details on how to deploy web projects are set out in

Chapter 10, beginning on [page 185](#), after configuration and installation of eServices itself.

Minimum Requirements

For hardware and operating system minimum requirements, see the [Genesys Supported Operating Environment Reference Guide](#). Ensure that you have at least the following amount of free disk space on your installation host.

WebSphere Application Server 6.1

For AIX ® (32-bit WebSphere Application Server) or Sun Solaris:

- Total amount of disk space required is 1030 MB
- Minimum 512 MB physical memory; 1 GB recommended

Software

Use IBM HTTP Server 6.1 and IBM WebSphere 6.1.

Deploying KANA Response Live for co-browsing

KANA Response Live is used to enable the eServices Web Collaboration feature, which allows an agent to direct a client's web browser so that they can view websites at the same time.

This section describes how to install and configure KANA Response Live on the following platforms.

Table 3: KANA Response Live Supported Platforms

OS	Web Server	Application Container
Windows	Microsoft Internet Information Server (IIS)	Apache Tomcat
Solaris	Apache HTTP server	Apache Tomcat
Linux	Apache HTTP server	Apache Tomcat

Prerequisites

- Create a new Microsoft SQL or Oracle database to use with KANA Response Live.

- Request and install an SSL certificate for your server. See [Procedure: Installing a new SSL certificate using IIS](#), on page 80.

Table 4: Task Flow to Deploy KANA Response Live

Objective	Related Procedures and Actions
1. Install KANA Response Live	<p>Follow the procedure for your operating system:</p> <ul style="list-style-type: none"> • Procedure: Installing KANA Response Live for Windows, on page 82 • Procedure: Installing KANA Response Live for Solaris, on page 88 • Procedure: Installing KANA Response Live for Linux, on page 89
2. Create the database schema.	<p>Windows</p> <p>On Windows deployments, the database schema is created automatically during installation.</p> <p>Solaris and Linux</p> <p>For Solaris and Linux deployments, you must manually create the schema:</p> <ul style="list-style-type: none"> • Procedure: Creating the KANA database schema on Solaris or Linux, on page 90
3. Configure additional settings.	<p>Windows</p> <p>For Windows deployments:</p> <ol style="list-style-type: none"> 1. Change the <code>router.enabled</code> option in the <code>config.properties</code> file to <code>false</code>. 2. Point the Home Directory in IIS to your KANA installation folder. <p>For the full procedure, see Procedure: Configuring KANA Response Live for Windows, on page 84.</p> <p>Solaris or Linux</p> <p>For Solaris or Linux deployments:</p> <ol style="list-style-type: none"> 1. Change the <code>router.enabled</code> option in the <code>config.properties</code> file to <code>false</code>. 2. Install a valid SSL certificate in the Apache Web Server directory. <p>For the full procedure, see Procedure: Configuring KANA Response Live for Solaris or Linux, on page 91.</p>

Table 4: Task Flow to Deploy KANA Response Live (Continued)

Objective	Related Procedures and Actions
4. Test the installation.	<p>Windows</p> <p>For Windows, you can start running the KANA Response Live server by launching the .bat files in kana\hbrook\bin.</p> <p>For the full procedure, see Procedure: Testing the KANA Response Live installation on Windows, on page 87.</p> <p>Solaris or Linux</p> <p>For Solaris or Linux, you can start the KANA server using the following command:</p> <pre>/usr/local/kana/hipctl start</pre> <p>For the full procedure, see Procedure: Testing the KANA installation on Solaris or Linux, on page 92.</p>
5. Test co-browsing functionality.	<p>Use the following URLs to start a test co-browsing session:</p> <ul style="list-style-type: none"> • Windows: http://<hostname>/index.html • Solaris or Linux: http://<hostname>/responselive/index.html <p>For the full procedure, see Procedure: Launching a co-browsing session, on page 93.</p>

Procedure: Installing a new SSL certificate using IIS

Purpose: To configure an SSL certificate for your Kana Response Live server, using Microsoft IIS Manager.

Start of procedure

1. To open IIS Manager, select Start > Administrative Tools > Internet Information Services (IIS) Manager.
2. In the left frame, expand Server_Name(local computer), find and right-click Default Web Site, and then select Properties.
3. On the Directory Security tab, in the Secure communications section, click Server Certificate.
The Web Server Certificate Wizard opens.
4. On the Welcome page, click Next.

5. Select **Create a new certificate**, and then click **Next**.
6. Select **Prepare the request now, but send it later**, and then click **Next**.
7. Enter a name for your certificate, select a bit length (the higher the bit length, the stronger the encryption), and then click **Next**.
8. On the **Organization Information** page, enter your company name in the **Organization** box and a descriptive name for your organizational unit in the **Organizational unit** box.
9. Enter either the fully qualified domain name (FQDN) or the server name as the **Common name** for your website—the address you want users to enter when browsing to your site.

Tip: For the **Common name**, make sure you use a valid DNS name. It must match the external FQDN of your website. For intranet-only purposes, you can use the internal FQDN for your sever, or the NetBIOS name.

10. Enter your location information, and then click **Next**. Do not use abbreviations for **State/province** or **City/locality**.
The **Certificate Request File Name** page opens.
11. Enter the path and file name where you want to save your certificate request, and then click **Next**.
12. Review the summary of your certificate request information, and then click **Next**.
IIS creates the certification file.
13. Click **Finish**.
14. At this point, you need to send the certification file to the online Certificate Authority of your choice. Contact the Certificate Authority for directions on how to submit your request.
15. When you receive the response file from the Certificate Authority, you need to reopen the certificate installation wizard. Again on the **Directory Security** tab of your **Default Web Site Properties** dialog box, click **Server Certificate**. Click **Next** to continue.
16. Select **Process the pending request** and install the certificate, and then click **Next**.
17. Enter the location of the certificate file you received from the Certificate Authority, and then click **Next**.
18. Review the summary, and then click **Next**.
19. Click **Finish**.

End of procedure

Next Steps

- Install KANA Response Live. See [Procedure: Installing KANA Response Live for Windows](#).

Installing and Configuring KANA Response Live for Windows

Use these procedures to install and configure KANA Response Live in a Windows environment.

Procedure: Installing KANA Response Live for Windows

Purpose: To install KANA Response Live for Windows.

Prerequisites

- An SSL certificate is installed. See [Procedure: Installing a new SSL certificate using IIS](#), on [page 80](#).

Start of procedure

1. Locate and double-click the KANA Response Live setup file. For Windows, this file is called V<version>_Win_Tomcat_IIS_DB_CHAT.exe.

Note: For eServices 8.1, the current supported version for Windows is KANA Response Live Version 10 R3. The file on the 8.0 CD is V10.3.0.14_Win_Tomcat_IIS_DB_CHAT.exe.

2. Enter your Web Server and Application Server host names, and click Next.

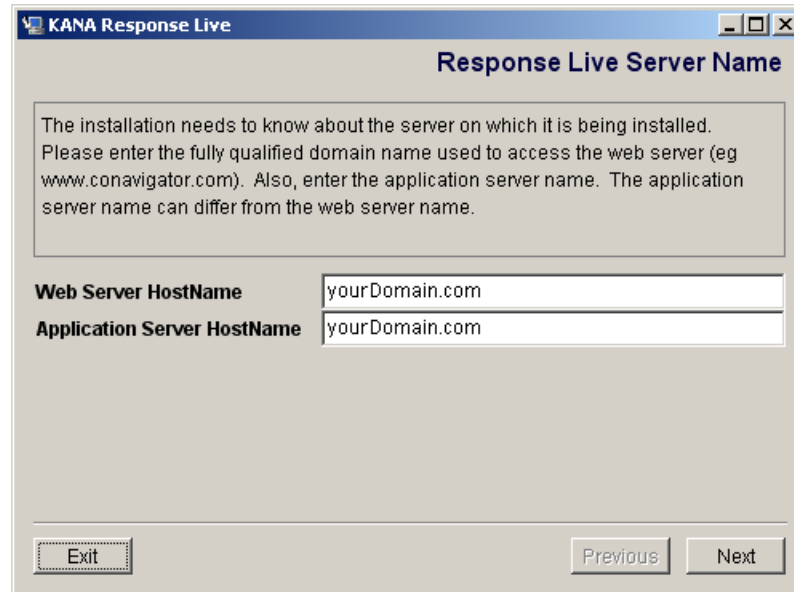


Figure 13: KANA Response Live Installation

3. Select the database type you created, and click **Next**.
4. Enter login parameters for the database instance that you created, and click **Next**.
5. Enter a Server ID, and click **Next**. This Server ID is a unique name of four characters or less.
6. If you did not enter valid database login information, the Database Check Result page displays an error message. To correct the information, click **Previous** to return to the appropriate pages. To continue, click **Next**.
7. Click **Next** to start the preinstallation checks.

If you did not install a valid SSL certificate before installing KANA Response Live, a warning appears (see [Figure 14](#)). Click **Next** to continue, and install your SSL certificate after completing the installation.

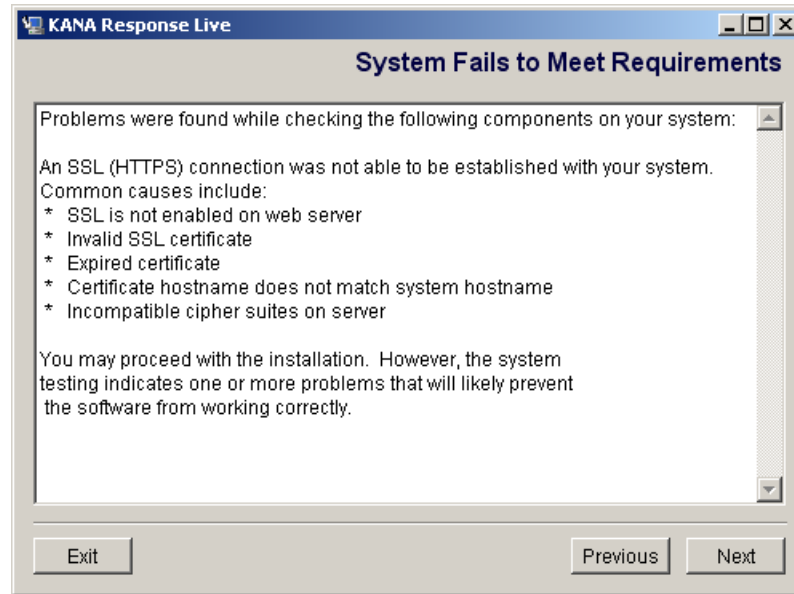


Figure 14: System Requirements Warning

8. Click **Next** at the **Introduction** page.
9. Choose the installation folder for KANA Response Live, and click **Install**.

Warning! Do *not* include any spaces in the folder name (for example, use `ResponseLive` instead of `Response Live`).

10. Click **Next**.
11. Leave the **Yes** checkbox unselected, and click **Next**.
12. At the **Launch Page** page, click **Next**.
13. Click **Done**. The installation is complete.

End of procedure

Next Steps

- Configure KANA Response Live. See [Procedure: Configuring KANA Response Live for Windows](#).

Procedure: Configuring KANA Response Live for Windows

Purpose: After installing KANA Response Live on your Windows machine, you need to make the following additional configurations:

- Update your `config.properties` file. Change the `router.enabled` option in the `config.properties` file to `false`.
- Change your IIS Default Web Site so that the Home Directory points to your KANA installation folder.
- Apply Patch5 for Kana Response Live Version 10 R3, by using `ResponseLive10R3_Patch5.zip` in the `ThirdPartyApplications/KANAResponseLivePatch5for10R3` folder on the Genesys Web Collaboration CD.

Note: KANA Response Live has many other functions and configurable options that are not described in this *Deployment Guide*. To ensure compatibility, Genesys does not recommend making use of additional functionality or changing any options other than those described below.

Prerequisites

- KANA Respond Live is installed. See [Procedure: Installing KANA Response Live for Windows](#), on page 82.

Start of procedure

1. To update your `config.properties` file, complete the following actions:
 - a. Open the `config.properties` file located in `<KANA_HOME>\hbroot\conf`.
 - b. Change the `router.enabled` option to `false`.
 - c. Save and close the properties file.
2. To change your IIS Default Web Site home directory, complete the following actions:
 - a. Open Internet Information Services (IIS) Manager.
 - b. Right-click on Default Web Site, and select the Properties option ([Figure 15](#)).

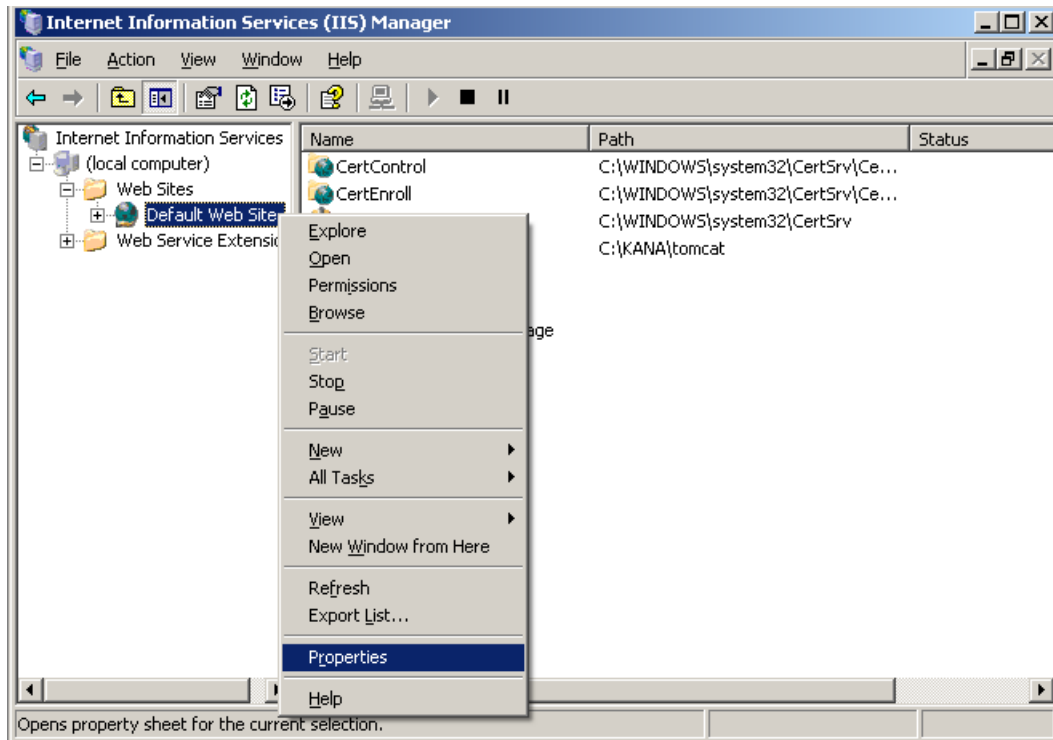


Figure 15: Internet Information Services (IIS) Manager

- c. On the Default Web Site Properties page, select the Home Directory tab (Figure 16).

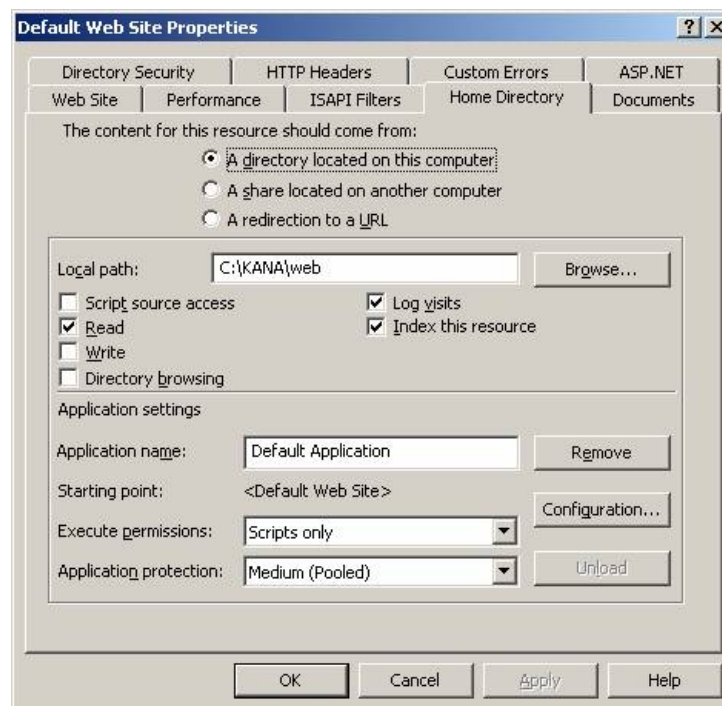


Figure 16: Default Web Site Properties

- d. Change the Local path value to <KANA_HOME>\web.
- e. Click OK to save the change.

End of procedure

Next Steps

- Test the installation. See [Procedure: Testing the KANA Response Live installation on Windows](#), on page 87.

Note: To configure Kana Response Live application server to support a single remote web server, make the following change on the Response Live application server:
In the System Administration Tool, edit the ISystems properties for the Response Live Server. In the Host Servers, change Login Cookie Domain to the IIS Server domain name.

Procedure: Testing the KANA Response Live installation on Windows

Purpose: This procedure describes how to test the KANA Response Live installation on Windows.

Prerequisites

- KANA Response Live is configured. See [Procedure: Configuring KANA Response Live for Windows](#), on page 84.

Start of procedure

1. Locate and double-click the following files (in the order listed) to start your KANA Response Live installation:
 - a. hipregistry.bat
 - b. router.bat
 - c. tomcat.bat
2. Start (or restart) your Web Server.
3. Follow the steps listed in [Procedure: Launching a co-browsing session](#), on page 93 to test co-browse functionality.

End of procedure

Installing and Configuring KANA Response Live for Solaris or Linux

To install KANA Response Live on a Solaris or Linux system, you must use a user that has superuser permissions and has write permissions in the `/usr/local` directory, such as `root`.

Procedure: Installing KANA Response Live for Solaris

Purpose: To install KANA Response Live for Solaris.

Start of procedure

1. Transfer the KANA Response Live software package to the system, using the following command:
`pkgadd -d Cona_sparc-V<version>_sol7-sparc.`

Note: For eServices 8.1, the current supported version for Solaris is KANA Response Live Version 10R2. Use the `Cona_sparc-V10.2.0.9-sol7-sparc` file on the CD.

2. Select 1 and press Enter.
3. Type the fully qualified domain name of your system and press Enter.
4. A message will appear asking where you want to install. Press Enter to select the default location `/usr/local/kana`. You cannot install in any other directory, as this path name is hard-coded.
5. A message will appear asking for the packages to install. Press Enter to select the default all option and start the installation process. The installer creates the KANA Response Live files and copies them to the KANA Response Live installation directory `/usr/local/kana`.
6. Select the database type that you created. For details, see “Prerequisites” on [page 78](#).
7. At the prompt, type the database host name and press Enter.
8. Enter the following information about your database, as prompted:
 - a. Database server name.
 - b. Database instance name.

Note: The length of the database instance name should not be greater than 12.

- c. Administrator's user name.
 - d. Administrator's password.
9. Enter a server ID. This ID is a unique name of four characters or less.
 10. Press Y to confirm the information you entered, and to transfer the package.

End of procedure

Next Steps

- Create the database schema. See [Procedure: Creating the KANA database schema on Solaris or Linux](#), on page 90.

Procedure: Installing KANA Response Live for Linux

Purpose: To install KANA Response Live for Linux.

Start of procedure

1. Transfer the KANA Response Live software package to the system, using the following command:
`./pkginst add`

Note: For eServices 8.1, the current supported version for Linux is KANA Response Live Version 10R3. Use the `cona_linux-V10.3.0.14-linux.tar` file on the CD. You will have to unzip this file. To do so, enter `tar -xvf <filename>` at the command prompt. Then you can enter the command in [Step 1](#).

2. Press Enter.
3. Type the fully qualified domain name of your system and press Enter.
4. Select the database type that you created. For details, see "Prerequisites" on [page 78](#).
5. Enter the following information about your database, as prompted:
 - a. Database server name.
 - b. Database instance name.

Note: The length of the database instance name should not be greater than 12.

- c. Administrator's user name.
- d. Administrator's password.

6. Enter a server ID. This ID is a unique name of four characters or less.

Note: The following message can be ignored if it appears during installation:
./postinstall_k: line 37: ndd: command not found
It will not have any affect on the installation.

7. The installation generates a dummy certificate.

Note: Installation on Linux ends *without* a successful completion message (as it does after installation on Solaris).

End of procedure

Next Steps

- Create the database schema. See [Procedure: Creating the KANA database schema on Solaris or Linux](#), on page 90.

Procedure: Creating the KANA database schema on Solaris or Linux

Purpose: To create the KANA database schema on Solaris or Linux.

Prerequisites

- KANA Response Live is installed. See [Procedure: Installing KANA Response Live for Solaris](#), on page 88, or [Procedure: Installing KANA Response Live for Linux](#), on page 89.

Start of procedure

1. Enter the following commands to start the installation script:

```
cd /usr/local/kana/hbroot/bin
./clinstaller.sh
```
2. Select the database type that you created. For details, see “Prerequisites” on [page 78](#).
3. Enter the following information about your database, as prompted:
 - a. Database instance name.

Note: The length of the database instance name should not be greater than 12.

- b. Administrator's user name.
 - c. Administrator's password.
 - d. Database server name.
4. Type the KANA Response Live server name and press Enter.
5. Enter the server ID that you specified in [Step 9](#) of the “Installing KANA Response Live for Solaris” section.
6. Enter the database tablespace name.
7. Press Y to confirm the information that you have entered, and to install the database schema.

End of procedure

Next Steps

- Configure KANA Response Live. See [Procedure: Configuring KANA Response Live for Solaris or Linux](#), on [page 91](#).

Procedure: Configuring KANA Response Live for Solaris or Linux

Purpose: To configure KANA Response Live for Solaris or Linux.

Prerequisites

- KANA Response Live is installed. See [Procedure: Installing KANA Response Live for Solaris](#), on [page 88](#), or [Procedure: Installing KANA Response Live for Linux](#), on [page 89](#).

Note: KANA Response Live has many other functions and configurable options that are not described in this *Deployment Guide*. To ensure compatibility, Genesys does not recommend making use of additional functionality or changing any options other than those described below.

Start of procedure

1. Update your `config.properties` file:
 - a. Open the `config.properties` file located in `/usr/local/kana/hbroot/conf`.
 - b. Change the `router.enabled` option to `false`.
 - c. Save and close the `config.properties` file.

2. In order for the KANA Response Live Server to function, you must install a valid SSL certificate in the packaged Apache Web Server located at:
`/usr/local/kana/apache`
3. For Linux only: apply Patch5 for KANA Response Live Version 10 R3, by using `ResponseLive10R3_Patch5.zip` in the `ThirdPartyApplications/KANAResponseLivePatch5for10R3` folder on the Genesys Web Collaboration CD.

End of procedure

Next Steps

- Test the installation. See [Procedure: Testing the KANA installation on Solaris or Linux](#), on page 92.

Procedure: Testing the KANA installation on Solaris or Linux

Purpose: This procedure describes how to test the KANA installation.

Prerequisites

- The KANA database schema exists. See [Procedure: Creating the KANA database schema on Solaris or Linux](#), on page 90.
- KANA Response Live is configured. See [Procedure: Configuring KANA Response Live for Solaris or Linux](#), on page 91.

Start of procedure

1. Start your KANA Response Live installation using the following command:
`/usr/local/kana/hipctl start`

Note: After the Genesys Web Collaboration Server is installed, you do not use this command to start or stop KANA Response Live. Instead, use the following commands:

Start Genesys Web Collaboration:
`/usr/local/kana/hipctl_start.sh`

Stop Genesys Web Collaboration:
`/usr/local/kana/hipctl_stop.sh`

2. Follow the steps listed in [Step 1](#) on [page 93](#) to test co-browsing functionality.

End of procedure

Next Steps

- Launch a co-browsing session. See [Procedure: Launching a co-browsing session](#), on [page 93](#).

Launching a Co-Browsing Session

Complete the following procedure to launch a test co-browsing session using either the Windows, Solaris, or Linux deployment of KANA Response Live.

Procedure: Launching a co-browsing session

Purpose: To test co-browsing functionality. Once KANA Response Live is installed, you can use two hosts to test the co-browsing sessions.

Prerequisites

- KANA Response Live is installed.

Start of procedure

1. On the first host, open the KANA Response Live launch page in a web browser (see Figure 17 on [page 94](#)).

Windows: `http://<hostname>/index.html`

Solaris or Linux: `http://<hostname>/responseLive/index.html`

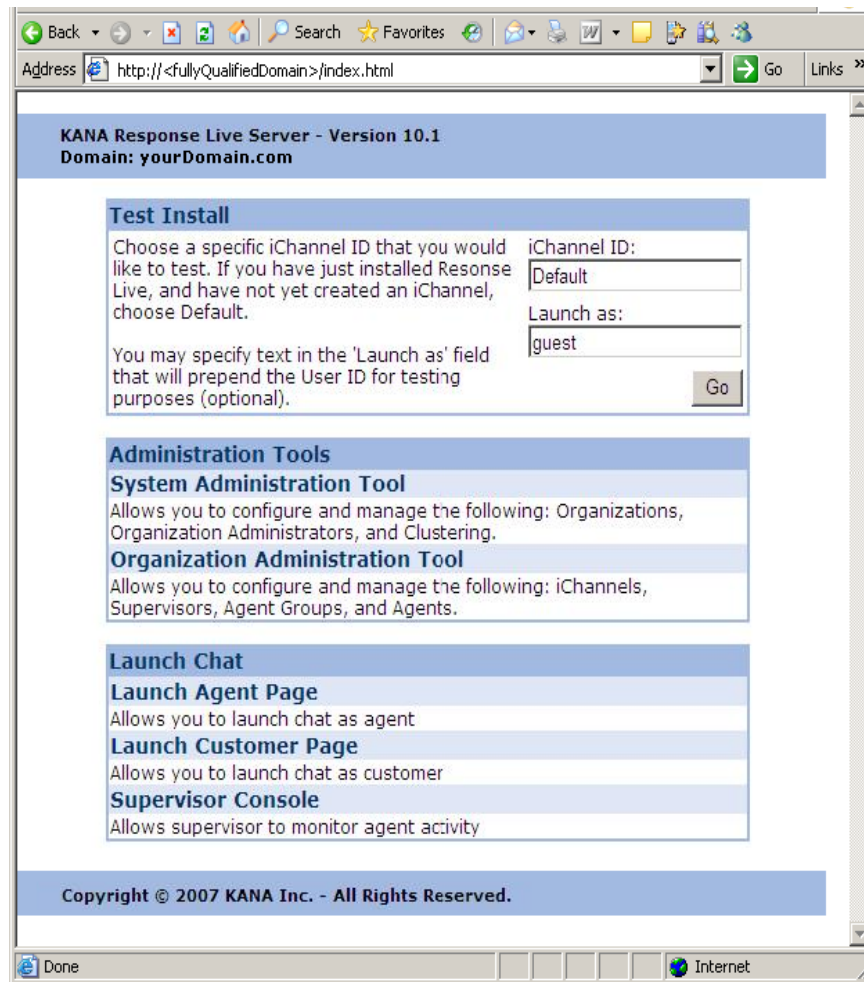


Figure 17: KANA Response Live Launch Page

2. Enter the channel and user ID, and click Go.
iChannel ID: Default
Launch as: guest
3. On the second host, repeat [Steps 1 and 2](#).
4. On the first host, enter the user ID for the second host into the Connect With field (see [Figure 18 on page 95](#)), and click Go.

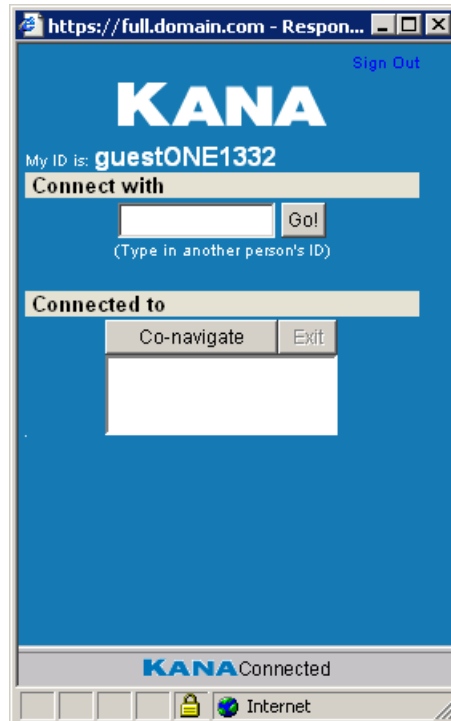


Figure 18: Web Collaboration Connection Window

5. On the second host, click Yes to accept the Web Collaboration session (see [Figure 19](#)).

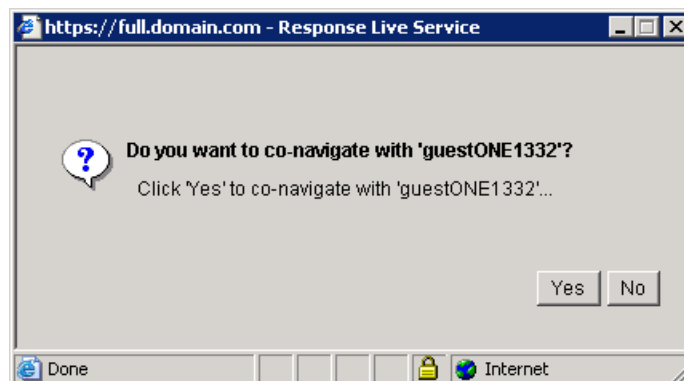


Figure 19: Request for Web Collaboration

A new web browser window opens on both hosts, showing the shared web session.

Note: Before installing the Genesys Co-Browsing components, you should remove the KANA Response Live start service. See “Remove the KANA Response Live Start Service” on [page 101](#) for details.

End of procedure

Next Steps

- Install and configure eServices. See [Chapter 5 on page 97](#).



Chapter

5

Model Configuration and Installation on Windows

This chapter describes a model single-tenant setup, using a Windows environment, such as might be deployed in a lab setting. This model setup, which includes all Genesys and third-party components, is assumed to be a first-time installation.

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

- [The Model Environment, page 98](#)
- [Overall Strategy, page 98](#)
- [Prerequisites, page 99](#)
- [Web Server and Web Application Container, page 101](#)
- [Configuring eServices Objects, page 101](#)
- [Installing eServices Components, page 136](#)
- [Manual Installation of eServices Components, page 146](#)
- [Creating Databases and Running Scripts, page 152](#)
- [Verifying Connections, page 153](#)
- [Testing eServices Components, page 154](#)
- [Testing the Setup, page 157](#)

This chapter assumes that you are familiar with the use of the Genesys Configuration Manager. See *Framework 8.0 Configuration Manager Help* for detailed directions on using Configuration Manager.

For your own deployment, be sure to use the “Configuration Worksheet” on [page 42](#) to help you plan the specifics of your solution.

The Model Environment

The model environment is deployed on one computer with the network name `MMHost` and with the following software components:

- Genesys Framework
- Universal Routing
- Genesys Desktop (a Genesys-designed agent desktop, not part of eServices)
- Stat Server (part of Real-Time Metrics Engine)
- POP3 e-mail server
- All eServices components
- Web components, including web server and Java components

Overall Strategy

The strategy for configuring and installing the model environment is as follows:

1. Prerequisites

You must have some items installed and operating before starting, including Framework, Routing, and various third-party applications. This document does not describe installation procedures for these components beyond what is available in Chapter 4, beginning on [page 55](#). (For detailed information on deploying a given non-eServices component, see the related *Deployment Guide* for that component.)

You should also create a shared directory to receive the installation packages for the eServices components.

2. Web Server and Web Application Container

Install, connect, and/or configure components that enable web services.

- Install and connect Apache, then Tomcat. (See [Chapter 4](#).)

3. Configuration

Install and use the eServices Configuration Wizards that Genesys supplies to guide you through the process of configuring this solution. These wizards help you configure a number of different components, and each component wizard has its own name. Throughout this section, however,

these wizards will generally be referred to as the eServices Configuration Wizards, unless identifying a specific component wizard by name is more helpful.

Note: You can configure most of the components and contact center objects you need from within the eServices Configuration Wizards interface. Interaction Server Proxy, Universal Contact Server Proxy, and SMS Server cannot be configured within the eServices Configuration Wizards interface. See Chapter 12, beginning on [page 223](#) for information about the manual deployment of these components.

4. Install the eServices Configuration Wizards.
 - a. Install Interaction Workflow Samples.
 - b. Install the Web Callback Application.
 - c. Configure contact center objects—Switching Office, Switch, Places.
 - d. Configure eServices objects—Media servers, clients, other components.
5. **Installation**

Install the eServices 8.1 components.

 - a. Perform integrated installation of eServices servers.
 - b. Install Genesys Desktop.
6. **Testing the setup**
 - a. Test individual components.
 - b. Test your model setup as a whole and troubleshoot any problems.

The following sections of this chapter describe these procedures in detail.

Prerequisites

The model setup uses the components listed in [Table 5](#).

Table 5: Prerequisite Components

Type of Component	Component Used
Operating System	A supported Windows environment.
Web Server	Apache Web Server.
Web Application Container	Tomcat.
E-mail (POP3) Server	Third-party e-mail server.

Table 5: Prerequisite Components (Continued)

Type of Component	Component Used
Java Environment	Java Runtime Environment (JRE). For supported versions, see the Genesys Supported Operating Environment Reference Guide .
Database Server	Microsoft SQL 2008. For other supported databases, see the Genesys Supported Operating Environment Reference Guide .
License Server	License Server 9.5; License File: 7260@MMHost . eServices 8.1 also supports FLEXNet Publisher v11.9 License Manager.
Genesys Framework, including <ul style="list-style-type: none"> Management Layer Stat Server (part of Real-Time Metrics Engine) 	Genesys Framework, single tenant.
	Management Layer.
	Stat Server Application object name Stat_Server . If installing a new Stat Server: <ul style="list-style-type: none"> Do so prior to running the eServices 8.1 Configuration Wizards. Make sure you select Stat Server Java Extensions (SSJE). SSJE requires a Java Runtime Environment (JRE) on the computer hosting this instance of Stat Server. If using an existing Stat Server: <ul style="list-style-type: none"> You will likely need to enable SSJE. This requires creating new Java sections in the Stat Server application object and installing new java extension files. For details, consult the <i>Framework 8.1 Stat Server Deployment Guide</i>.
Genesys Universal Routing	Genesys Universal Routing, with a multimedia routing strategy enabling e-mail and chat.

Be sure you have these components installed and operating properly before installing eServices 8.1. For instructions, see the documentation for Genesys Framework and Universal Routing.

Create Host If you plan on installing eServices on the same computer as Genesys Framework, you should already have a Host object created and available for your eServices deployment. However, if you have not already done so, or if you want to install eServices on a different computer than Framework, create

the Host object in the Configuration Layer for the machine that will host your model environment.

Create a Shared Directory

If you plan to install eServices components across multiple hosts, create a shared directory on your local drive or network to receive the installation packages. Genesys recommends that you do this before starting the configuration and installation process. (This is unnecessary if you plan to install eServices on a single host.)

Remove the KANA Response Live Start Service

If you plan to install Genesys Co-Browsing components, you should remove the KANA Response Live start service. For Windows systems, check the Windows Services application to ensure that KANA Response Live Service is configured to start manually and is not running. For Solaris systems, use the `rm -f /etc/rc2.d/S88hipctl` command to remove the start service.

Create Contact, Interaction (and Co-Browsing) Databases

Create three databases in your SQL Server RDBMS. Two of these will accommodate Universal Contact Server contact information, and the other will handle Interaction Server's one table for tracking transient data. If you are also deploying Co-Browsing Server, you may want to create an additional database for it. (Alternatively, you can use the UCS database to store Co-Browsing Server information.) You will use the names of these databases during your creation of eServices Database Access Points (DAPs). After configuration and installation, you will run scripts against these databases to make them available to Genesys servers.

Note: For DB2 RDMS users, before running your database scripts, you need to create an additional table space with a corresponding buffer pool page size of at least 8 KB.

Web Server and Web Application Container

The installation of the web server and the web application container are described in Chapter 4, beginning on [page 55](#). Any further instructions in this chapter coincide with steps you may need to take in using wizards to deploy eServices.

Configuring eServices Objects

The first step in configuring eServices 8.1 is to install the eServices 8.1 Configuration Wizards.

The second step is to install the Interaction Workflow Samples. Doing so provides you with a number of Configuration Layer objects that are useful later, when implementing the model environment..

Note: Because eServices requires specific configuration objects, including certain Queue and View objects, if you do not install the Interaction Workflow Samples first, you must create these objects manually to allow the wizard to install the solution. If you create these objects prior to running the wizard, the wizard uses them. If you do not install the Samples first, and if you have not created these objects previously, you must create and configure the default Queue objects for E-mail Server and Chat Server prior to completing the eServices deployment.

The eServices 8.1 Configuration Wizards take you through a step-by-step configuration process, allowing you to create Configuration Layer objects and to associate them with each other as needed. After you configure each component, the wizard copies the software to the directory you specify. Once you have configured all the eServices components, you will install them with the eServices Installation Starter. See “Installing eServices Components” on [page 136](#) for more details.

Note: At certain points, the configuration or installation wizards might prompt you to restart your computer. If you receive such a prompt, allow the restart to occur, then continue with the configuration or installation process.

Procedure:

Installing the eServices Configuration Wizards

Purpose: To install the configuration wizards used to deploy an eServices solution.

Prerequisites

- Interaction Management CD.
- Start the Configuration Layer.

Start of procedure

1. Install the eServices Configuration Wizards and Genesys Wizard Manager.
In the root directory of the Interaction Management CD, find Setup.exe. Run Setup.exe and follow the directions. You may want to read the Wizard Advisory supplied with the wizard.

2. If you intend to configure your Stat Server as you configure eServices, you should also install the configuration wizard for this product.

End of procedure

Next Steps

- [Procedure: Installing and Starting Interaction Workflow Samples](#)

Procedure: Installing and Starting Interaction Workflow Samples

Purpose: To instantiate objects in the Configuration Layer, which you will find helpful when completing the full eServices configuration and installation.

Interaction Workflow Samples use a number of preconfigured Script objects of various subtypes, including Simple Routing, Interaction Queue, Workbin, and Interaction View.

Prerequisites

- Interaction Management CD.

Start of procedure

1. Install the Interaction Workflow Samples.
 - a. On your Interaction Management CD, find and double-click Setup.exe in the solution_specific\InteractionWorkflowSamples\windows directory.
 - b. At the welcome page for the installation wizard, click Next.
 - c. Specify the destination for Interaction Workflow Samples, and click Next.
 - d. Click Install.
 - e. Click Finish.
2. Start the Interaction Workflow Samples.
 - a. From the Windows taskbar, select Start > Programs > Genesys Solutions > eServices 8.1.0 > Interaction Workflow Samples > Start Interaction Workflow Samples. This launches the Interaction Workflow Samples Wizard.
 - b. Login to Configuration Manager using your user name and password.
 - c. At the Welcome to Interaction Workflow Samples Wizard window, click Next.

- d. If using a multi-tenant environment, select the **Tenant** and click **Next**.

Tip: During the installation, the setup procedure does not look for possible name conflicts between existing objects and new components from the Interaction Workflow Samples. It overwrites any existing objects.

In order to prevent the loss of existing objects, Genesys recommends you install the new samples into a separate tenant. Alternatively, you should use IRD's Business processes export capability to create backups of workflows and strategies related to an existing configuration.

- e. Specify a destination directory for the strategy files, and click **Next**.

You may want to select the directory used for your Interaction Routing Designer strategies, unless it already contains strategies with same file names, which would then be overwritten. If you do decide to use that particular directory, make a backup of its content before proceeding.

Tip: Do not select the **StrategyFiles** directory created during installation of your Interaction Workflow Samples as the target here. This causes file "collision," since the source and target files are then identical.

- f. At the **Import Completed** window, click **Next**.

- g. Click **Finish**.

Once you exit the wizard, you can view the new objects (over 180 of them) installed with these Samples, by opening the <tenant>\Scripts folder in Configuration Manager.

End of procedure

Next Steps

- [Procedure: Configuring eServices objects using the wizard](#)

Procedure: Configuring eServices objects using the wizard

Purpose: To configure all eServices components required for an eServices solution. Use the eServices Configuration Wizards to launch the main wizard and any supplementary wizards necessary for your deployment.

This procedure provides values for a model setup on a single Windows host, with two databases. Substitute the model values for the values that you entered on your [Configuration Worksheet](#) on [page 43](#).

Note: If you are deploying Co-browsing Server, you must install that component and the supporting third-party software, KANA Response Live Server, on a separate host from the Web API Server. Other than Co-Browsing Server, all other components can reside on the same host machine.

Prerequisites

- [Procedure: Installing the eServices Configuration Wizards](#), on [page 102](#).
- [Procedure: Installing and Starting Interaction Workflow Samples](#), on [page 103](#).
- Interaction Management CD (for eServices Objects).
- Management Framework CD (for adding DB Servers).
- Depending on the type of solution that you purchased, you also require the following CDs:
 - Genesys E-mail CD
 - Knowledge Management CD
 - Genesys Chat CD
 - Genesys Web Collaboration CD
- A license server running on the host computer.
- A completed Configuration Worksheet (see Table 1 on [page 43](#)).

Summary of the Wizards

The eServices Configuration Wizards consist of a main wizard and multiple secondary wizards, which are launched automatically when you indicate that you want to configure particular components. The wizard presents only those pages that are applicable for your deployment, or for the stage of deployment that you have reached.

[Table 6](#) summarizes the flow of the main and secondary wizards to deploy the model setup. The information in the Key Actions and Inputs (Model Setup)

column gives you the model-specific data inputs and actions for that section of the wizard.

Table 6: Configuration Wizard Work Flow

Wizard Step	Key Actions and Inputs (Model Setup)
1. Launch the eServices Configuration Wizard, page 112	Start > Programs > Genesys Solutions > eServices 8.1.0 > eServices Configuration Wizards > Start Wizard Manager
2. Give your solution a name, page 113	Solution name: MM81
3. Select Simple or Custom Configuration, page 113	Select Custom Configuration.
4. Copy the IP to your shared directory, page 113	Copy the IP from the Interaction Management CD, and paste it to your shared directory. For more info, see Procedure: Create a Shared Directory , on page 101 .
5. Select or add a Message Server, page 114	Add a Message Server from the existing Framework components.

Table 6: Configuration Wizard Work Flow (Continued)

Wizard Step	Key Actions and Inputs (Model Setup)
Add the Database Access Points and associated DB Servers:	<p>At the Solution Components: Data Access Point page, click Add to launch the Database Access Points Wizard, which you run several times to create the following:</p> <ul style="list-style-type: none"> • Interaction Server DAP • DB Server for Interaction Server database • A shared DAP for Universal Contact Server and Co-Browsing Server <p>Note: You must create all required DAPs and DB Servers here, before moving on to other components. The wizard does not permit you to create DAPs later in the configuration.</p> <p>If you intend to use the Event Logger functionality that was introduced in version 7.6.1, or the message queue functionality that was introduced in 8.0.1, you will need additional DAPs. Configuration and deployment is explained in the “Interaction Server: Advanced Topics” section of the “Ongoing Administration and Other Topics” chapter of the <i>eServices 8.0 User’s Guide</i>.</p>
6. Create the Interaction Server DAP, page 114	Name = MM81_Ixn_DAP
7. Add the Interaction DB Server, page 114	<p>Name = MM81_IxnDBServer Host = MMHost Default Port = 6110 DBMS Type = mssql</p> <p>For the model setup, you can skip TLS Settings and Backup Server Information. Select MM81_IxnDBServer to finish configuring the DAP.</p>

Table 6: Configuration Wizard Work Flow (Continued)

Wizard Step	Key Actions and Inputs (Model Setup)
8. Configure the Interaction Server DAP, page 115	<p>DBMS Type = mssql DBMS Name = MMHost Database Name/SID = IxnDB User Name = sa Password = <password for user sa> Case Conversion = any</p> <p>Note: The Interaction Server DAP must have its Case Conversion attribute (on the DB Info tab) set to any or upper. Setting it to lower causes an error when Interaction Server initializes.</p>
9. Create the Universal Contact Server DAP (Main), page 116	<p>This DAP uses a JDBC connection. You do not need to create a DB Server, but make sure you enable JDBC access and clear the DBMS name.</p> <p>DAP Name = MM81_UCS_DAP Database Name/SID = Customer JDBC Port = 1433 JDBC Role = Main</p>
10. Add the Universal Contact Server, page 117	<p>At the Solutions Components: Universal Contact Server Page, click Add to launch the Universal Contact Server Wizard.</p> <p>Name = MM81_UCS Server Host = MMHost Server Default Port = 6120</p>
11. Select the type of Interaction Server, page 118	<ul style="list-style-type: none"> • If you select New-Style Interaction Server, the wizard creates a multimedia Switch object for you. Continue at Step 13. • If you select Switch-based Interaction Server and the wizard detects a properly configured Multimedia type Switch object, continue at Step 14. • If you select Switch-based Interaction Server and the wizard does not detect a properly configured Multimedia-type Switch object, continue at Step 12. <p>For the model setup, select New-Style Interaction Server and continue at Step 13.</p>

Table 6: Configuration Wizard Work Flow (Continued)

Wizard Step	Key Actions and Inputs (Model Setup)
12. Create a Multimedia Switch object (Switch-based Interaction Server only) , page 119	For Switch-based Interaction Server only. Our model setup uses the following names: Switch = MM81_Switch Switching Office = MM81_Switching_Office Switch Type = Multimedia Switch
13. Configure Framework Resources , page 120	This wizard lists all current Framework resources (Agents, Places, and so on), and prompts you to create any missing resources that the solution requires. Note that: <ul style="list-style-type: none"> • If you selected New-Style Interaction Server, this wizard launches before you configure the Interaction Server itself. • If you selected Switch-based Interaction Server, this wizard launches as the last step in configuring the Interaction Server.
14. Configure the Interaction Server , page 122	Name = MM81_IxnSrv Host = MMHost Default Port = 6130 License Server Host = MMHost License Server Port = 7260
15. Select a Stat Server , page 124	This wizard connects your existing Stat Server to all necessary components with one action—click ADD.
16. Add a Universal Routing Server , page 124	Add your existing Universal Routing Server to the solution. For the full procedure, see page 124 .
17. Add an Application Cluster , page 124	Use this wizard to group components for load-balancing. Otherwise, you can skip this step.

Table 6: Configuration Wizard Work Flow (Continued)

Wizard Step	Key Actions and Inputs (Model Setup)
18. Add the Web API Server, page 125	<p>For our model setup, enter the following values:</p> <p>Name = MM81_WebAPIServer Host = MMHost Default Port = 80 (or 443 if using https)</p> <p>Add connections to: Message Server, UCS, Stat Server, Interaction Server, SCS</p> <p>You can also use this wizard to add .NET Web API Server & Samples. Follow the same steps, but copy the following installation package:</p> <p>DotNetWebAPIServer_CompoundSamples</p>
19. Add the Universal Contact Server Manager, page 126	<p>For our model setup, name the UCS Manager:</p> <p>MM81_UCSMgr</p> <p>Add a connection to UCS.</p>
20. Select Additional CDs for Installation, page 127	<p>Launch the wizards for the rest of your purchased components—the wizards launch one after another, depending on how many you select.</p> <p>For the full procedure, see page 127</p>
21. Add E-mail Server, page 127	<p>Skip this step if your solution does not include e-mail interactions.</p> <p>Name = MM81_E-mailServer Port = 6150</p> <p>Add connections to: Message Server, Interaction Server, and UCS</p> <p>POP Server Info</p> <p>For the model setup, use MMHost as the POP3 and SMTP server host. Adjust accordingly if your enterprise is using a POP3 server.</p> <p>For details about creating addresses and accounts, and enabling web form e-mail, see the full procedure on page 127.</p>

Table 6: Configuration Wizard Work Flow (Continued)

Wizard Step	Key Actions and Inputs (Model Setup)
<p>Complete the Knowledge Management Wizard:</p> <p>22. Add the Classification Server, page 129</p> <p>23. Add the Training Server, page 130</p> <p>24. Add the Knowledge Manager, page 130</p>	<p>Skip these steps if you did not purchase Knowledge Manager.</p> <p>Name = MM81_ClassificationServer Port = 6160</p> <p>Add connections to: Message Server and UCS.</p> <p>Name = MM81_TrainingServer Port = 6170</p> <p>Add connections to: Message Server and UCS.</p> <p>Name = MM81_KnowledgeManager</p>
25. Add the Chat Server, page 131	<p>Skip this step if your solution does not include chat interactions.</p> <p>Name = MM81_ChatServer Port (default) = 6180 Port (webapi) = 6181 Port (ESP) = 6182</p> <p>Add connections to: Message Server, Interaction Server, and UCS</p>
26. Add the Co-Browsing Server, page 132	<p>Skip this step if your solution does not include co-browsing functionality.</p> <p>Name = MM800_Co-BrowsingServer Default Port = 6195</p> <p>Add connections to: Message Server and a JDBC-based DAP (MM81_UCS_DAP in the model setup).</p>
27. Create Capacity Rules, page 132	<p>Select a folder in the Configuration Server where the rules will be stored, and then select the rules that eServices will use as its default.</p>
28. Complete the wizard, page 133	<p>If you quit the wizard before you reach this point, the solution details do not enter the Configuration Database. However, any components that you create remain available for further configuration.</p>

Start of procedure

Note: The details of your solution configuration are entered into the Configuration Database when you have completed the wizard. Exiting the wizard prior to completion requires that you begin the solution configuration from the beginning. Any components you create along the way, however, are available during subsequent solution configuration using the wizard.

1. Launch the eServices Configuration Wizard

- a. To launch the Genesys Wizard Manager, from the Windows taskbar, select Start > Programs > Genesys Solutions > eServices 8.1.0 > eServices Configuration Wizards > Start Wizard Manager.
- b. Click log into the Configuration Layer. This opens the main Wizard Manager window (see Figure 20).

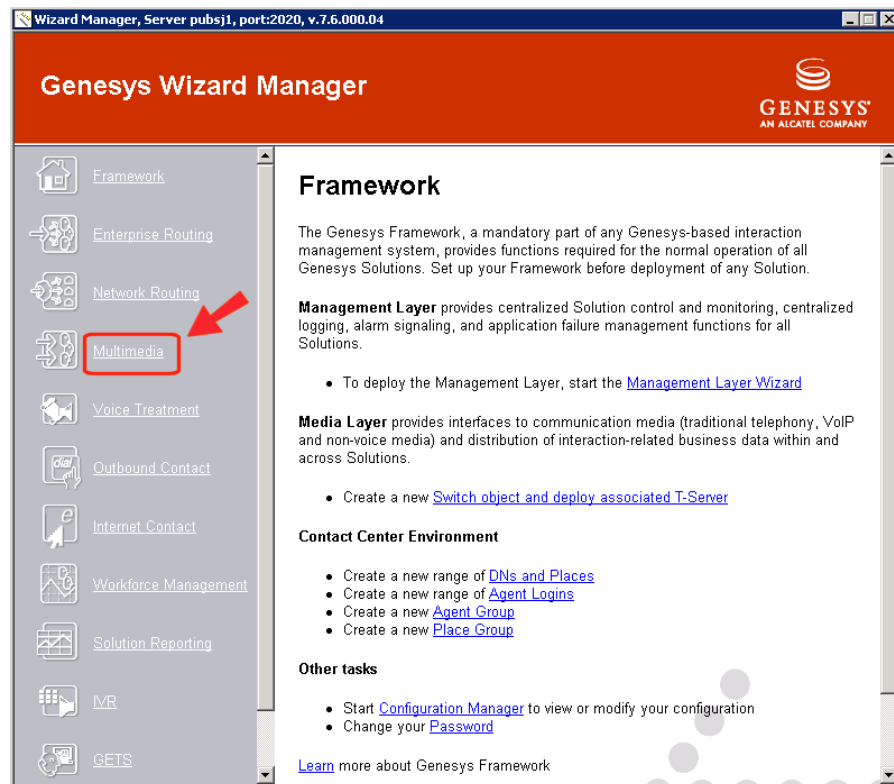


Figure 20: Wizard Manager

- c. Select Multimedia from the menu on the left side of the window (Figure 20), and then select Deploy Multimedia Solution in your contact center. The eServices Configuration Wizard Welcome page opens.
- d. Click Next to begin configuring eServices objects.

2. Give your solution a name

At the **Solution Name** page, enter a name for your solution and click **Next**.

Name = MM81

3. Select Simple or Custom Configuration

- a. At the **Configuration Process Selection** page, select one of the following:

- Simple single-host configuration
- Custom Configuration

For our model setup, select **Custom Configuration**, as shown in [Figure 21](#).

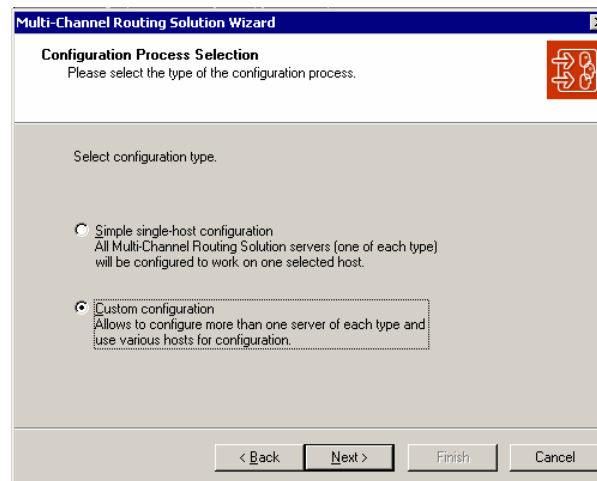


Figure 21: Configuration Process Selection

- b. Click **Next**.

Note: The Simple single-host configuration option is available for a predefined host in Windows deployments only. This chapter describes the custom configuration process. If you choose the simple configuration with Windows, use this chapter by skipping those steps that the wizard does not present to you during deployment.

4. Copy the IP to your shared directory

- a. At the **Installation Package** page, click **Have Disk** and navigate to the **Interaction Management CD**.
- b. To select a destination for the package, click **Browse** and navigate to your shared directory (see “[Create a Shared Directory](#)” on [page 101](#)). Click **Next**.

Note: eServices uses this group of components for installation. In later steps you will copy to this directory installation packages that are specific to each eServices component.

- c. At the **Installation Ready** page, confirm that your software is ready for installation and click **Next**.
5. **Select or add a Message Server**
 - a. At the **Solution Components: Message Server** page, select or add a Message Server.
 - b. Click **Next**.
6. **Create the Interaction Server DAP**

At this point in the wizard, you begin creating your DAPs and associated DB Servers. You must create all DAPs and DB Servers here, before moving on to other components. The eServices Configuration Wizard does not permit you to create DAPs later in the configuration. Relaunch the Database Access Point Wizard as necessary.

 - a. At the **Solution Components: Data Access Point** page, click **Add**.
 - b. At the **Browse for Application** page, click the **New Application** icon as shown in [Figure 22](#).



Figure 22: New Application Icon

- c. Create a new DAP.
- d. Enter your Database Access Point name:
Name = MM81_Ixn_DAP
7. **Add the Interaction DB Server**
 - a. Click the folder icon beside **DB Server**, and then click the **New Application** icon to run the DB Server Wizard.
 - b. At the **DB Server Name** page, enter a name for your DB Server:
Name = MM81_IxnDBServer
 - c. At the **Server Information** page, enter your host and port information:
Host = MMHost
Default Port = 6110
 - d. At the **Installation Package** page, copy the DB Server installation package. Select a source (the Management Framework CD) and a destination (the shared directory). Click **Next**.
 - e. At the **Installation Ready** page, confirm that your DB Server software is ready for installation, and then click **Next**.

- f. At the Listening Ports and Transport Layer Security (TLS) Settings page, click Next.

Note: For our model setup, we do not require any additional ports. If you want to install additional ports, you can do so in Configuration Manager later.

- g. DBMS Type = mssql
- h. Log Configuration:
Click Next.

Note: For our model setup, we use only the default settings for all log options.

- i. At the Backup Server Information page, because no backup servers are configured, clear the checkbox and click Next.
- j. Click Finish to exit the DB Server Wizard. The Database Access Point Wizard will resume.

8. Configure the Interaction Server DAP

- a. Browse for Application (DB Server):
Select the DB Server you just configured (MM81_IxnDBServer) and click OK.
- b. Database Access Point Name and Type:
The Name and DB Server are already filled in. Clear the Enable JDBC access box and click Next.

Note: It is important to clear the Enable JDBC access box. Failing to do so can lead to a configuration problem.

- c. Database Information:
DBMS Type = mssql
DBMS Name = MMHost
Database Name/SID = IxnDB
User Name = sa
Password = <password for user sa>
- d. Case Conversion = any

Note: The Interaction Server DAP must have its Case Conversion attribute set to any or upper. Setting it to lower causes an error when Interaction Server initializes.

- e. Click Finish to complete the Database Access Point Wizard and return to the eServices Configuration Wizard.

- f. Browse for Application:
Select the Database Access Point that was just configured (MM81_Ixn_DAP) and click OK.

9. Create the Universal Contact Server DAP (Main)

For our model deployment, create a single DAP and database for both UCS and Co-Browsing Server. In your deployment, you can create both a Main database and an Archive database. You can also create a fourth database for Co-Browsing Server.

Note: UCS and Co-Browsing Server connect to their databases directly through JDBC. You do not need to create DB Server Application objects for these servers

- a. Solution Components: Database Access Point:
Click Add to open the Database Access Point Wizard. Click the New Application icon to configure a new DAP.
- b. Database Access Point Name and Type:
Name = MM81_UCS_DAP
DB Server = [NONE]
Select Enable JDBC access.
- c. Database Information:
DBMS Type = mssql
DBMS Name = ""
Database Name / SID = Customer
User Name = sa
Password = <password for user sa>

Warning! If you do not clear the DBMS Name box, your UCS and your Co-Browsing Server will not work properly.

- d. JDBC Connection:
Host = MMHost
Port (for JDBC) = 1433 (Microsoft SQL default)
Role = Main

Note: If your RDBMS is Microsoft SQL Server on a different host, select the host where you have Microsoft SQL Server installed, and enter the connection information for that instance of the database.

- e. Case Conversion = any
- f. Click Finish to complete the Database Access Point configuration and return to the eServices Configuration Wizard.

- g. Browse for Application:
Select the Database Access Point that was just configured (MM81_UCS_DAP) and click OK.
- h. Click Next.

Warning! If you are using a Microsoft SQL 2005 database, an additional configuration step is recommended for the UCS DAP.

In Configuration Manager, create a settings section on the options tab. Create a new option, setting the option name as prepare and the option value to false.

For additional details, see the “Ongoing Administration and Other Topics” chapter of the *eServices 8.1 User’s Guide*.

10. Add the Universal Contact Server

- a. Solution Components: Universal Contact Server:
Click Add.
- b. Browse for Application (Contact Server):
Click the New Application icon.
- c. Universal Contact Server Name:
Name = MM81_UCS
- d. Log Configuration:
Click Next.
- e. Server Information:
Host = MMHost
Default Port = 6120
- f. Universal Contact Server API Port:
Accept the default port value, or enter a port number where UCS should listen for third-party protocol connections. Click Next.
- g. Application Connections: Message Server:
Click Add, select the Message Server, and then click Finish and Next.
- h. Application Connections: Database Access Point:
Click Add, select MM81_UCS_DAP, and then click Finish and Next.
- i. Universal Contact Server Login Account:
UCS uses the Configuration Layer to pass some of its information back and forth to certain components. Select the user account (Configuration Layer Person object) or Access Group that UCS is to use to log in to the Configuration Layer. The selected account or access group must have write access to the tenant in use.

Note: For more information on access requirements for this user account, see “Limitations: UCS” in the “Ongoing Administration and Other Topics” chapter of the *eServices 8.1 User’s Guide*.

- j. **Installation Package:**
 - i. Click **Have Disk**, navigate to the **Interaction Management CD**, and then click **OK**.
 - ii. Click **Browse**, navigate to your shared directory, and then click **Next**.
 - k. **Installation Ready:**
Click **Next**.
 - l. Click **Finish** to exit the **Universal Contact Server Wizard** and return to the **eServices Configuration Wizard**.
 - m. **Browse for Application (Contact Server):**
Select the **Universal Contact Server** and click **OK**.
 - n. Click **Next**.
11. **Select the type of Interaction Server**
- At the **Interaction Server Selection** page, select the type of **Interaction Server** that you plan to use. For the model setup, select **New-style Interaction Server** (see Figure 23 on [page 118](#)).

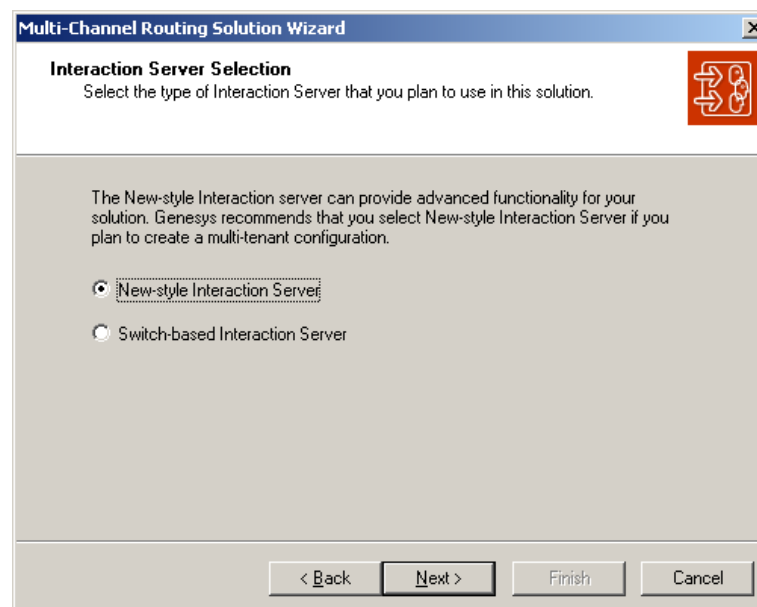


Figure 23: Interaction Server Application Object Choice

- If you select **New-Style Interaction Server**, the **eServices Configuration Wizard** automatically creates a **multimedia Switch** object for you in the background. Continue at [Step 13 on page 120](#).
- If you select **Switch-based Interaction Server**, the **eServices Configuration Wizard** checks for the required, telephony-related **Configuration Layer** resources.

- If your configuration includes a properly configured Multimedia type Switch object, continue at “Configure the Interaction Server” on [page 122](#).
- If your configuration does not include a properly configured Multimedia type Switch object, continue at “Create a Multimedia Switch object (Switch-based Interaction Server only)” on [page 119](#).

Note: Before attempting to run your eServices solution, check Configuration Manager for the existence of a Multimedia Switch object. If for some reason the wizard has failed to create one, use the Framework Wizard to create it. No other configuration is required; the components that require this switch are able to locate it automatically.

12. Create a Multimedia Switch object (Switch-based Interaction Server only)
 - a. If you select Switch-based Interaction Server, the wizard checks for the following:
 - A Multimedia-type switch
 - A connection from the switch to a T-Server type Interaction Server
 If the wizard fails to detect either the switch or its connection to T-Server, then it issues a warning (see [Figure 24](#)).

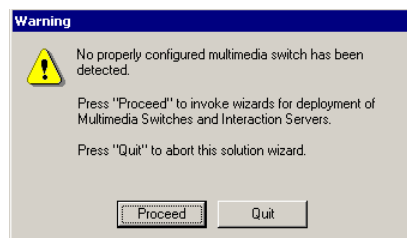


Figure 24: Switch Warning

- b. Click Proceed to launch the Switch Wizard.

Note: You can use a preexisting Switch object (that is connected to a properly configured Interaction Server) from your Configuration Layer, if available.

Multimedia Switch and Switching Office

- c. To create a new Multimedia type Switch object, provide a unique name for it and click Next. (The wizard allows you to create the new Switch only in folders designated to contain objects of type Switch.)

Name = MM81_Switch

Note: Even though the Configuration Layer permits assignment of multiple Switches of type Multimedia Switch to a single Interaction Server Application object, Genesys does not currently support such a configuration. This type of configuration can lead to unpredictable behavior on the part of the application that is connected to the switches.

- d. **Switching Office:**
Click Add to create a new Switching Office object. Provide the name and type, and click Finish.
Name = MM81_Switching_Office
Switch Type = Multimedia Switch
- e. At the Switch Summary page, click Next. See [Figure 25](#) for a sample summary of a Switch object's properties.

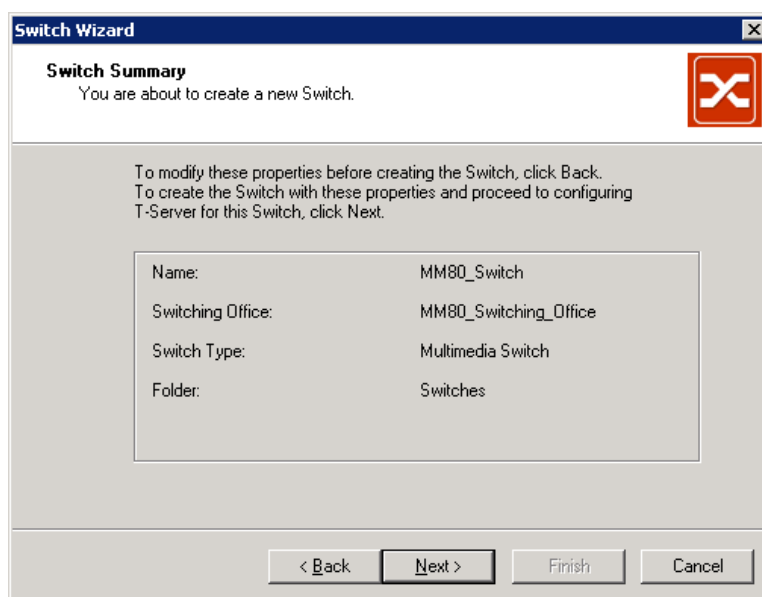


Figure 25: Multimedia Switch Summary Information

13. Configure Framework Resources

- a. If you select New-Style Interaction Server, the wizard opens the Framework Resources page (see [Figure 26](#) on [page 121](#)). This page lists the following:
 - Current Framework resources
 - Any missing resources required to support the media types served by your switch
- b. To create the missing resources now, select Create missing objects and review the Framework resources again, and then click Next. For our model setup, select this option.

In your own deployment, you can choose to wait until after you finish the wizard to create the required objects. Select **Continue** the solution creation process with the Framework resources as they are now.

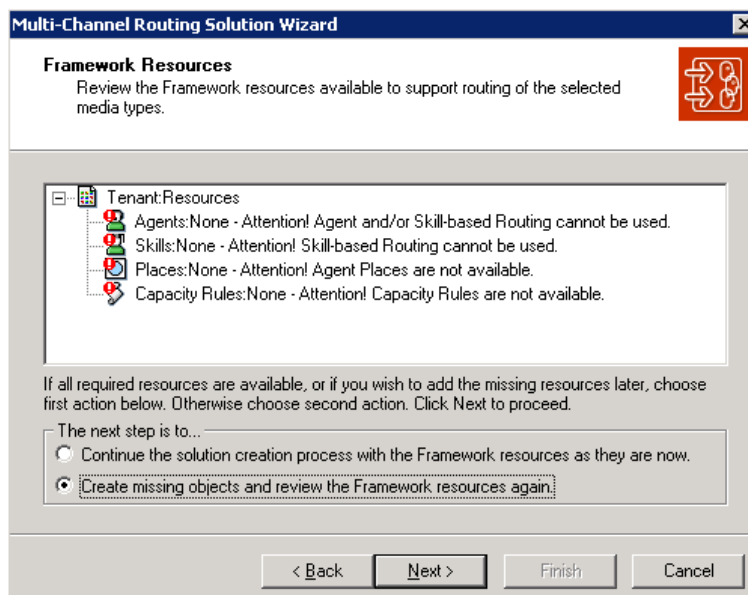


Figure 26: Framework Resources

- c. At the Framework Objects page, select the object type you want to create from the drop-down menu. For our model setup, create and configure the various objects as follows:

Table 7: Framework Resource Object Types

Object Type	Model Setup
Place	<p>Create Place objects to facilitate the routing of multimedia interactions:</p> <p>Place Name = Agent1Place</p> <p>Note: Do not create any DNs associated with this Place object, though you may want to create a Place Group object for this place, as suggested in the wizard.</p>
Capacity Rule	<p>By default, as the final step of the solution's configuration, the wizard creates and installs three capacity rules for your tenant. Do not create any capacity rules here (although you can, if you have previously installed the Resource Capacity Wizard, which is available on the Real-Time Metrics Engine CD).</p>

Table 7: Framework Resource Object Types (Continued)

Object Type	Model Setup
Agent	<p>Create Person objects to represent agents in your model setup and assign agent IDs to them. (Also use this step to create any Person objects you intend to use as the autoreply agents for supplying agent-related information in standard responses). Refer to the <i>Framework 7 Deployment Guide</i> for the full list of steps required to create Person objects.</p> <p>Agent Name = Agent One Employee ID = Agent1 User Name = Agent1 Password = Agent1</p> <p>Note: The agent values supplied here are for general use in this chapter's model setup, and do not correspond to the actual steps suggested by the wizard for the creation of Person objects. A Person object's user name must be unique within the Configuration Database. This is not a concern for this model installation, which is in a single-tenant environment. If you were installing more than one set of eServices components in a multi-tenant environment, you would use distinct names across tenants.</p>
Agent Login:	Create Agent Login objects to enable agents to log in to the switch. The agent login code must correspond to the agent login numbering of your switch.
Skill:	You do not need to create skills for this model installation.

14. Configure the Interaction Server

- a. **Solution Components: Interaction Server:**
Click Add and then click the New Application icon to run the Interaction Server Wizard.

Note: If you use a switch-based Interaction Server in a multi-tenant environment, you need to install one Interaction Server per tenant that requires one. That is, for reporting purposes, a given Interaction Server cannot serve the needs of multiple tenants.

The new style Interaction Server supports multi-tenancy. One Interaction Server can serve multiple tenants. This functionality was added in release 7.2. Releases prior to 7.2 do not support multi-tenancy.

- b. **Interaction Server Name:**
Name = MM81_IxnSrv
- c. **Log Configuration.** Accept the default log settings and click Next.

- d. **Server Information:**
Host = MMHost
Default Port = 6130
- e. Use the Connection Wizard to connect to Message Server.
- f. **Application Connections: Database Access Point:**
Add the DAP you created for Interaction Server, MM81_Ixn_DAP.
A message box will appear asking you to confirm the DAP settings.
Click **Yes** to review the settings, and then click **OK** to confirm the settings. Click **Next**.
- g. **Application Connections: Servers for Third-Party Protocol**
If you already installed related eServices components (Universal Contact Server, E-mail Server, and Classification Server, for instance), you can connect to them now using this page.
For our model setup, click **Next**
- h. **License Server Connection:**
Specify the license options to use (either a license host and port or a local filename) and click **Next**.
License Server Host = MMHost
License Server Port = 7260
- i. **Feature license counters:**
Specify the number of licenses for Interaction Server features. Click **Next**.
- j. **Installation Package:**
Copy the Interaction Server installation package. Select a source and destination. Click **Next**.
- k. **Installation Ready:**
Confirm that your Interaction Server software is ready for installation, and then click **Next**.
- l. Click **Finish** to exit the Interaction Server Wizard and return to the eServices Configuration Wizard.
- m. Do one of the following:
 - If you selected **New-style Interaction Server**, the wizard returns you to the **Browse for Application (Interaction Server)** page. Select the Interaction Server and click **OK**.
 - If you selected **Switch-based Interaction Server**, the wizard asks if you want to configure Framework resources. Refer to [Procedure: Configure Framework Resources](#), on [page 120](#) for details on how to create resources you might need in your environment.
- n. Click **Next**.

15. Select a Stat Server

Solution Components: Stat Server:

Click Add and select a Stat Server. This connects all the appropriate components in your eServices solution to this Stat Server.

Note: Even though the wizard allows you to create a Stat Server here, do not do so. Prior to using the wizard, you should have one Stat Server per eServices tenant already configured and installed in your environment. (Table 5 on [page 99](#) lists prerequisite details.)

If you receive the warning message in [Figure 27](#), this indicates that Stat Server does not yet have a connection to Interaction Server (or to Interaction Server Proxy; see [page 154](#)). This connection is required for both the routing and reporting of eServices interactions.

- If you click Yes, you may receive an error message if the Stat Server wizard is not installed on your machine. The connection cannot be created at this time.
- If you click No (the recommended approach), make sure that you add the connection from Stat Server to Interaction Server before you attempt to use routing or reporting with the eServices solution.

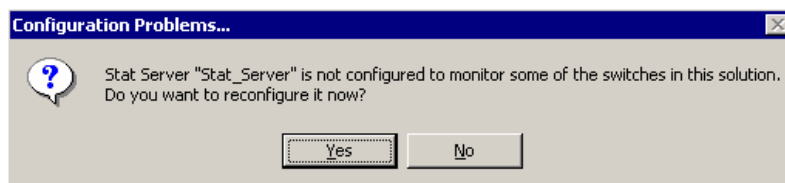


Figure 27: Offer to Reconfigure Stat Server

16. Add a Universal Routing Server

At the Universal Routing Server (URS) page, select the URS that you created and installed as one of the prerequisites for your eServices installation. If you receive a warning message similar to the one shown in [Figure 27](#), but stating that URS is not configured to monitor Interaction Server, click Yes to reconfigure now, or click No to reconfigure later.

Tip: If you click No, make sure you add the required connections after configuration and installation are complete. For more information, see [Procedure: Verifying Connections](#), on [page 153](#)

17. Add an Application Cluster

At the Application Cluster page, add an application cluster if you intend to group your eServices components into groups for load-balancing purposes. After giving a name to your new application cluster, the Application Cluster Wizard invites you to group, for instance, Chat Server, E-mail Server, and Web API Server, as well as other application clusters. See [Figure 28](#).

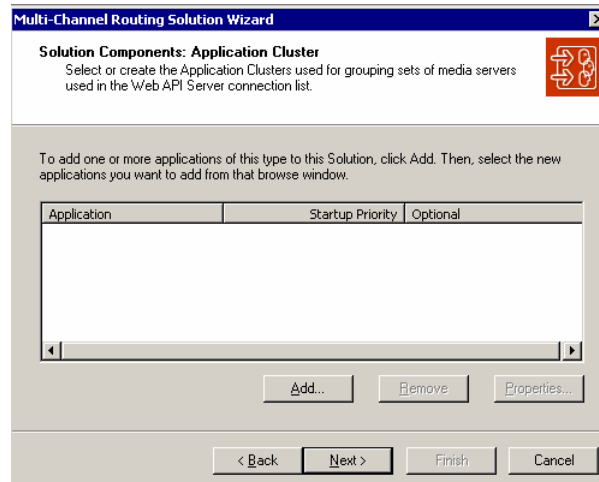


Figure 28: Application Connections in Application Cluster Wizard

18. Add the Web API Server

- a. **Solution Components: Web API Server:**
Click **Add** and then click the **New Application** icon to run the Web API Server Wizard.
- b. **Web API Server Name:**
Name = MM81_WebAPIServer
- c. **Log Configuration:**
Accept the default log settings and click **Next**.

Warning! For Web API Server .NET, you must use Configuration Manager to specify the *full path* of your log file in the `log/all` section of the options tab in your Web API Server Application object. If only a file name is specified, Web API Server does not create a log file.

Full permissions should be granted to the Everyone user for the folder containing your log file.

- d. **Server Information:**
Enter the server information, and click **Next**.
Host = MMHost
Default Port = 80 (Use your web server's http port here. The value might be 443 if you are using the https port.)
- e. **Application Connections: Message Server:**
Use the Connection Wizard to connect to Message Server. Click **Next**.
- f. **Application Connections: Servers for Load Balancing:**
Connect to the applications that Web API Server will use for load balancing and click **Next**. Such applications might include Chat Server (via the `webapi` port) and E-mail Server, if you have them. If you do not have them, click **Next**.

- g. Application Connections: Universal Contact Server:
Use the Connection Wizard to connect to Universal Contact Server. Click Next.
- h. Application Connections: Stat Server:
Use the Connection Wizard to connect to Stat Server. Click Next.
- i. Application Connections: Interaction Server:
Use the Connection Wizard to connect to Interaction Server. Click Next.
- j. Application Connections: Solution Control Server:
Use the Connection Wizard to connect to Solution Control Server. Click Next.

Note: Web API Server will not work without a connection to Solution Control Server.

- k. Installation Package:
Copy the Web API Server installation package. Select a source and destination. Click Next.
- l. Installation Ready:
Confirm that your Web API Server software is ready for installation, and then click Next.
- m. Click Finish to exit the Web API Server Wizard and return to the eServices Configuration Wizard.
- n. Browse for Application (Web API Server):
Select the Web API Server and click OK.
- o. Click Next.

19. Add the Universal Contact Server Manager

- a. Solution Components: Universal Contact Server Manager:
Click Add and then click the New Application icon to run the Universal Contact Server Manager Wizard.
- b. Universal Contact Server Manager Name:
Name = MM81_UCSMgr
- c. Application Connections: Universal Contact Server:
Use the Connection Wizard to connect to Universal Contact Server. Click Next.
- d. Installation Package:
Copy the UCS Manager installation package. Select a source and destination. Click Next.
- e. Installation Ready:
Confirm that your UCS Manager software is ready for installation, and then click Next.
- f. Click Finish to exit the Universal Contact Server Manager Wizard and return to the eServices Configuration Wizard.

- g. Browse for Application (Contact Server Manager):
Select the Universal Contact Server Manager and click OK.
- h. Click Next.

20. Select Additional CDs for Installation

Based on how you plan to implement eServices, identify any additional CDs you need to complete this installation.

The list of CDs that the wizard presents (shown in [Figure 29](#)) may include some you have not purchased. Select only those CDs that apply to your environment. For the purposes of the model environment, this section describes an installation using all possible CDs. If you select more than one CD, the wizard presents the appropriate component-related wizards in order.

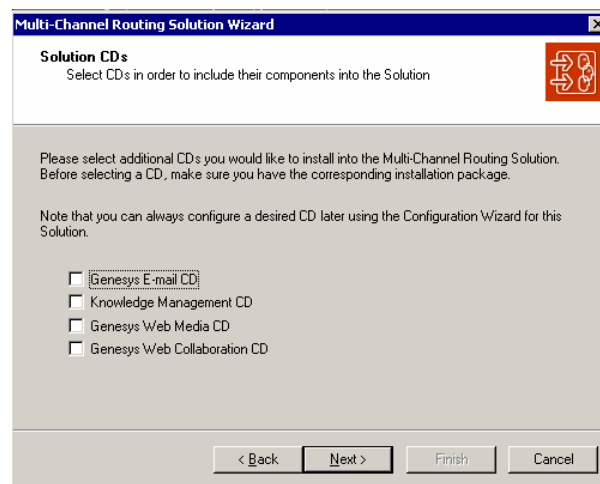


Figure 29: Other CDs for eServices Installation

21. Add E-mail Server

- a. Solution Components: E-mail Server:
Click Add and then click the New Application icon to run the E-mail Server Wizard.
- b. E-mail Server Name:
Name = MM81_E-mailServer
- c. Log Configuration:
Accept the default log settings and click Next.
- d. Server Information:
Host = MMHost
Default Port = 6150
- e. Use the wizard to connect to Message Server, Interaction Server, and Universal Contact Server.

f. The Names of the POP3 and SMTP Servers:

POP3 server name = MMHost
 POP server type = POP3
 SMTP server name = MMHost

Notes: This model installation uses MMHost as its POP3 and SMTP servers.

If you are using your enterprise POP3 server, enter its name as the POP3 server name value. You do not need to enter a value for the SMTP server name if it is the same server that you used for POP3 server name.

g. POP3 User Name:

Add information about the e-mail account identified as the administrator on the POP3 server.

- Address = <mailboxName>@MMHost
- User name = <username>
- Password = <password>

h. Default Domain for E-mail Addresses:

E-mail address default domain name = MMHost

i. External Agent E-mail Address:

External Agent e-mail address = external@MMHost

j. Web Form Processing:

Enable Web Form Processing by selecting the check box, and provide a port where the Web API Server should listen for requests—for example IWF Processing Port = 7777. (See [Figure 30.](#))

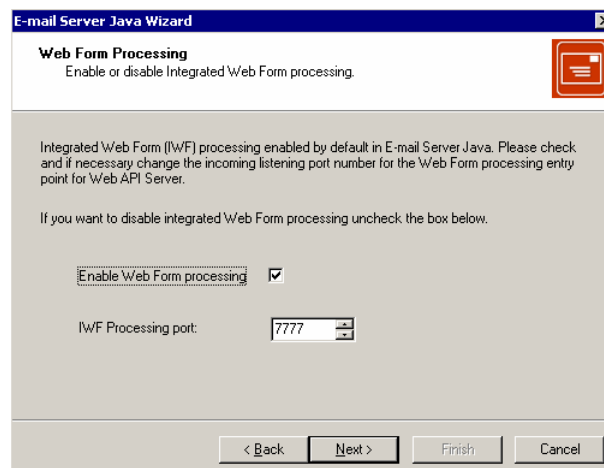


Figure 30: Information for Web Form Processing

k. Automated Reply Agent:

This e-mail address is used to supply field values in automated responses generated in the contact center. Select an existing Person object from the Configuration Layer.

- l. Auto-reply Agent = Agent1
- m. Installation Package:
Copy the E-mail Server installation package. Select a source and destination. Click Next.
- n. Installation Ready:
Confirm that your E-mail Server software is ready for installation, and then click Next and Finish.
- o. Browse for Application (E-mail Server):
Select the E-mail Server and click OK.
- p. E-mail Accounts:
Configure e-mail accounts that exist on your corporate e-mail server for customers to use when contacting your company—for example:
Customer access account 1 = <mailboxName1>@MMHost
Customer access account 2 = <mailboxName2>@MMHost

Note: If you installed the Interaction Workflow Samples, then your configuration will automatically include the following three e-mail accounts (which are used in the samples):
Customer support, Tech support, and Warranty support.

If you did not install the Interaction Workflow Samples before running the wizard, then no e-mail accounts are shown. You must add and configure at least one e-mail account before continuing.

- q. Click Finish to exit the E-Mail Wizard.
22. Add the Classification Server
- If you selected the Knowledge Management CD in [Step 20](#) on [page 127](#), the wizard now guides you through the configuration of the Classification Server, Training Server, and Knowledge Manager.
- a. Solution Components: Classification Server:
Click Add, and then click the New Application icon to run the Classification Server Wizard.
 - b. Classification Server Name:
Name = MM81_ClassificationServer
 - c. Log Configuration:
Accept the default log settings and click Next.
 - d. Server Information:
Host = MMHost
Default Port = 6160
 - e. Application Connections:
Use the wizard to connect to Message Server and Universal Contact Server.

- f. **Access to License:**
Specify the license options to use (either a license host and port or a local filename) and click Next.
License Server Host = MMHost
License Server Port = 7260
- g. **Installation Package:**
Copy the Classification Server installation package. Select a source and destination. Click Next.
- h. **Installation Ready:**
Confirm that your Classification Server software is ready for installation, and then click Next.
- i. Click **Finish** to exit the Classification Server Wizard and return to the Knowledge Management Wizard.
- j. **Browse for Application (Classification Server):**
Select the Classification Server and click OK.

23. Add the Training Server

- a. **Solution Components: Training Server:**
Click **Add**, and then click the **New Application** icon to run the Training Server Wizard.
- b. **Training Server Name:**
Name = MM81_TrainingServer
- c. **Log Configuration:**
Accept the default log settings and click Next.
- d. **Server Information:**
Host = MMHost
Default Port = 6170
- e. Use the **Connection Wizard** to connect to Message Server and Universal Contact Server.
- f. **Installation Package:**
Copy the Training Server installation package. Select a source and destination. Click Next.
- g. **Installation Ready:**
Confirm that your Training Server software is ready for installation, and then click Next.
- h. Click **Finish** to exit the Training Server Wizard and return to the Knowledge Management Wizard.
- i. **Browse for Application (Training Server):**
Select the Training Server and click OK.

24. Add the Knowledge Manager

- a. **Solution Components: Knowledge Manager:**
Click **Add**, and then click the **New Application** icon to run the Knowledge Manager Wizard.
- b. **Knowledge Manager Name:**
Name = MM81_KnowledgeManager

- c. Use the Connection Wizard to connect to Universal Contact Server.
- d. Knowledge Manager Options:
Indicate that you have a Classification Server Content Analysis license installed by selecting the check box of the same name. If you do not have this license, do not select this check box.
- e. Installation Package:
Copy the Knowledge Manager installation package. Select a source and destination. Click Next.
- f. Installation Ready:
Confirm that your Knowledge Manager software is ready for installation, and then click Next.
- g. Click Finish to exit the Knowledge Manager Wizard and return to the Knowledge Management Wizard.
- h. Browse for Application (Knowledge Manager):
Select the Knowledge Manager and click OK. Click Finish to exit the Knowledge Management Wizard.

25. Add the Chat Server

- a. Solution Components: Chat Server.
Click Add, and then click the New Application icon to run the Chat Server Wizard.
- b. Chat Server Name:
Name = MM81_ChatServer
- c. Log Configuration. Accept the default log settings and click Next.
- d. Server Information:
Host = MMHost
Port = 6180
- e. Use the Connection Wizard to connect to Message Server, Interaction Server, and Universal Contact Server.
- f. Installation Package:
Copy the Chat Server installation package. Select a source and destination. Click Next.
- g. Installation Ready:
Confirm that your Chat Server software is ready for installation, and then click Next.
- h. Change ports:
At this time you can change the ESP and webapi ports if necessary. ESP refers to the ability of URS to submit messages directly from a strategy into a chat session—for example, to greet a customer who is waiting for an agent. The webapi port is to be used by Web API Server.
- i. Click Finish to exit the Chat Server Wizard and return to the Genesys Web Media Wizard.
- j. Browse for Application (Chat Server):
Select the Chat Server and click OK. Click Next.
- k. Click Finish to exit the Genesys Web Media Wizard.

26. Add the Co-Browsing Server

- a. **Solution Components: Co-Browsing Server:**
Click **Add**, and then click the **New Application** icon to run the Co-Browsing Server Wizard.
- b. **Co-Browsing Server Name:**
Name = MM800_Co-BrowsingServer
- c. **Log Configuration:**
Accept the default log settings and click **Next**.
- d. **Server Information**
Host = MMHost2
Default Port = 6195

Tip: You must install Co-Browsing Server and the supporting third-party software, KANA Response Live Server, on a separate host from the Web API Server—MMHost2 for the model setup. To install and configure Kana Response Live, see [“Deploying KANA Response Live for co-browsing” on page 78](#).

- e. Use the wizard to connect to Message Server and a JDBC-based DAP. (You can use the MM81_UCS_DAP that you created.)
- f. **Installation Package:**
Copy the Web Collaboration installation package. Select a source and destination. Click **Next**.
- g. **Installation Ready:**
Confirm that your Co-Browsing Server software is ready for installation, and then click **Next**.
- h. Click **Finish** to exit the Co-Browsing Server Wizard and return to the Genesys Web Collaboration Wizard.
- i. **Browse for Application (Co-Browsing Server):**
Select the Co-Browsing Server and click **OK**. Click **Next**.
- j. Click **Finish** to exit the Genesys Web Collaboration Wizard and return to the eServices Configuration Wizard.
- k. **Solution Components: Web API Server:**
Follow the on-screen instructions to configure servers for load balancing. Click **Next**.

27. Create Capacity Rules

- a. **Resource Capacity Rules Deployment:**
Select a folder in the Configuration Server where eServices should store its resource capacity rules. Click **Next**.
- b. **Resource Capacity Rules Selection:**
From the drop-down menu, select the capacity rule you want to use as the default for eServices. The wizard assigns that rule to the Tenant object associated with your deployment in the previous step. Click **Next**.

28. Complete the wizard

a. Solution Summary:

At this point the wizard lists the solution components you have collected for use with eServices. Click **Next** to finish the configuration or **Back** to change the configuration of some solution component. See Figure 31 on [page 133](#) for a sample of the Solution Summary page.

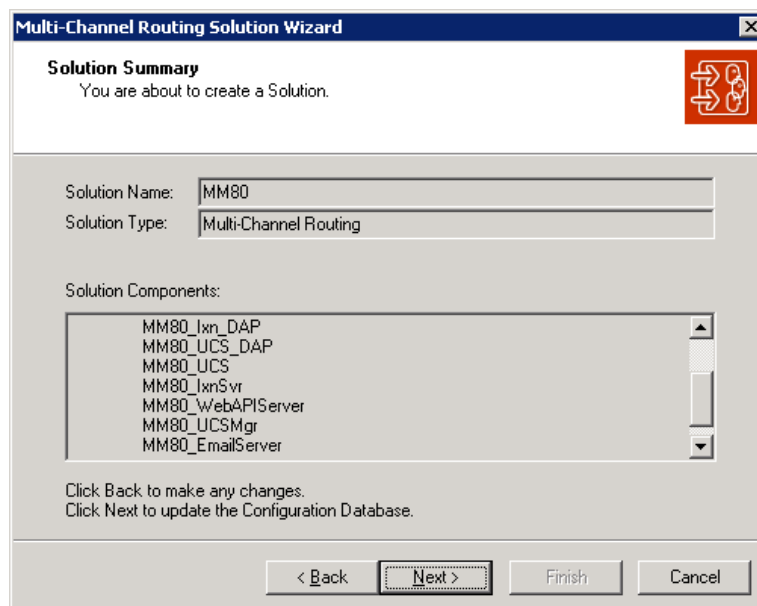


Figure 31: Solution Summary

b. Completing the eServices Solution Wizard:

Click **Finish** to confirm your solution configuration. Remember that you can change configuration details at a later time as well.

Note: Only at this point, when you exit the wizard, are the details of your solution configuration entered into the Configuration Database. Exiting the wizard prior to this point requires that you begin the solution configuration from the beginning. Any components you create along the way, however, are available during subsequent solution configuration using the wizard.

End of procedure

Next Steps

- For Chat Server, Genesys recommends that you specify an additional port for dedicated communication with Web API Server: see [Procedure: Specifying a webapi-port for Chat Server](#), on [page 134](#).

- If you want Chat Server to use the same port for communicating with both Web API Server and your call center agents (not recommended), then continue at [Procedure: Installing and Starting the Web Callback Application](#), on page 135 and [Procedure: Performing an integrated installation of all eServices servers](#), on page 138.

Procedure: Specifying a webapi-port for Chat Server

Purpose: To establish a dedicated port for Chat Server communication with Web API Server.

Note: Chat Server can function using a single port for communicating with both web clients and agents. However, for system security Genesys recommends that you specify an additional `webapi-port`, which allows only web clients to communicate through it. Only this port should be opened/exposed to outside connections, thus prohibiting agent connections over a firewall.

Start of procedure

1. In Configuration Manager, on the `Server Info` tab of the Chat Server Application object, click `Add Port`.
2. In the resulting `Port ID` dialog, enter the following:
 - `Port ID` = `webapi`
 - `Communication Port` = 6181 (refer to the [Procedure: Configuration Worksheet](#), on page 43 if using different values than the model).
3. Click `OK`, then `OK` again.
Your change takes effect immediately.

End of procedure

Next Steps

- [Procedure: Installing and Starting the Web Callback Application](#), on page 135.
- [Procedure: Performing an integrated installation of all eServices servers](#), on page 138

Procedure: Installing and Starting the Web Callback Application

Purpose: To instantiate objects in the Configuration Layer and provide database scripts, which you will find helpful when completing the web callback workflow configuration.

The Web Callback Application use a number of preconfigured Script objects of various subtypes, including Simple Routing, Interaction Queue, Interaction View, Subroutine.

Prerequisites

- Interaction Management CD.

Start of procedure

1. Install the Web Callback Application.
 - a. On your Interaction Management CD, find and double-click Setup.exe in the solution_specific\ Web Callback Application \Windows directory.
 - b. At the welcome page for the installation wizard, click Next.
 - c. Specify the destination for the Web Callback Application, and click Next.
 - d. Click Install.
 - e. Click Finish.
2. Start the Web Callback Application.
 - a. From the Windows taskbar, select Start > Programs > Genesys Solutions > eServices 8.1.0 > eServices Web Callback Application > Start Web Callback Application. This launches the Web Callback Application Wizard.
 - b. Login to Configuration Manager using your user name and password.
 - c. At the Welcome to the Web Callback Application Wizard window, click Next.
 - d. If using a multi-tenant environment, select the Tenant and click Next.

Tip: During the installation, the setup procedure does not look for possible name conflicts between existing objects and new components from the Web Callback Application. It overwrites any existing objects. In order to prevent the loss of existing objects, Genesys recommends you install the new samples into a separate tenant. Alternatively, you should use IRD's Business processes export capability to create backups of workflows and strategies related to an existing configuration.

- e. Specify a destination directory for the strategy files, and click Next. You may want to select the directory used for your Interaction Routing Designer strategies, unless it already contains strategies with same file names, which would then be overwritten. If you do decide to use that particular directory, make a backup of its content before proceeding.

Tip: Do not select the StrategyFiles directory created during installation of your Web Callback Application as the target here. This causes file “collision,” since the source and target files are identical.

- f. At the Import Completed window, click Next.
 - g. Optionally, select a resource capacity rule which takes into account webcallback media type. Click Next.
 - h. Click Finish.
3. Once you exit the wizard, you can view the following:
- New objects installed with these Samples, by opening the <tenant>\Scripts folder in Configuration Manager
 - Database scripts in the <Web Callback Application installation>\Scripts folder for updating the Interaction Server database.
 - UCS_impex.kme file in the <Web Callback Application installation> folder for importing Web Callback standard responses via Knowledge Manager.
 - WebCallback distribution for processing Agent Group in the <tenant>\Agent Groups folder in Configuration Manager
 - WCB_CUSTOMER_NUMBER Attribute Value in the <tenant>\Business Attributes\Interaction Custom Properties\Attribute Values folder in Configuration Manager
 - webcallback Media Type in the <tenant>\ Business Attributes\ Media Type\ Attribute Values folder in Configuration Manager.
 - WebCallback Transaction in the <tenant>\Transactions folder in Configuration Manager.

End of procedure

Next Steps

- [Procedure: Performing an integrated installation of all eServices servers, on page 138.](#)

Installing eServices Components

This section describes the installation process for eServices 8.1 components after you have finished configuring them with the wizards.

For Windows-only installations and the Windows portions of Solaris, Linux, and AIX installations, Genesys provides an installation package that integrates the installation of most or all eServices 8.1 components. The installation package is copied to one host as part of the process of running the eServices Configuration Wizards. Start `IPStarter.exe`, located in the `Windows/` subdirectory of your installation directory, one time for each Windows host that is part of your eServices deployment. (Each time you run `IPStarter.exe`, you are installing the appropriate components for the given Windows host.)

For Solaris, Linux, or AIX installations, after going through the configuration process, install eServices components by first manually copying them from their deployed locations (where the wizards placed them) to each Solaris, Linux, or AIX host. Then manually install each component separately by launching its own setup. Detailed information on Solaris, Linux, and AIX installations is available in “Deploying eServices on UNIX Hosts” on [page 185](#).

Note: Prior to the 8.1.3 release, for all platforms you need to install the Java Environment and Libraries for eServices and UCS, before installing individual components. (This does not affect Chat Server, Interaction Server, Web API Server, and Co-Browsing Server.)

In releases prior to 8.0.1, the Java Environment and Libraries for eServices and UCS were named Third Party Components.

On Windows, if you use `IPStarter.exe`, that executable takes care of this for you automatically. On each Solaris, Linux, or AIX host, however, you must manually run the Java Environment and Libraries for eServices and UCS setup before you run the setup for the other eServices components on that host.

For an explanation of the difference between Third Party Components (executable) and ThirdPartyApplications (directory), see the note on [page 56](#).

For the purposes of this section, the host where you have the eServices installation directory is called the *local host*. Any other hosts that you configured to receive eServices components are called *remote hosts*.

Generalized Installation Procedure

The general outline of how the integrated installation proceeds is as follows:

1. Local host
 - a. The installer presents a list of eServices components that you configured to run on the local host, and asks you to verify it.
 - b. The installer presents a list of components for which you did not specify a host, and asks if you want to install them on the local host.
 - c. The installer installs the appropriate components on the local host.

2. Remote hosts

- a. From each remote host in turn, you must access the installation package from its shared directory on the local host (see “Create a Shared Directory” on [page 101](#)).
- b. The installation package then begins to install on the remote host.

Individual Component Installation

If for some reason you want to install individual components rather than using the integrated installation package, be aware that the directory structure of the individual installation packages looks like this:

```
<componentA>
AIX
Linux
Solaris
Windows
<componentB>
AIX
Linux
Solaris
Windows
...
aix
ThirdPartyComponents
Linux
ThirdPartyComponents
solaris
ThirdPartyComponents
windows
Configuration Wizards
ThirdPartyComponents
```

That is, for each eServices component other than Common and Third Party Components, there is a directory named for the component. That directory contains subdirectories holding the installation packages for the supported operating systems.

To perform an integrated installation of all eServices Servers, see the next section.

Procedure:

Performing an integrated installation of all eServices servers

Purpose: To install the eServices components that were configured with the eServices Configuration Wizards.

Prerequisites

- An eServices solution has been configured with the wizards. See [Procedure: Configuring eServices objects using the wizard](#), on [page 104](#).

Start of procedure

1. Starting the Integrated Installation
 - a. Go to your shared directory, on which you had the Genesys Wizard Manager place the eServices components (for example, \\MMHost\GenesysSetup). Find and run windows\IPStarter.exe.
 - b. At the Welcome window, click Next to proceed.
 - c. If any components were not assigned to a specific computer while running the Configuration Wizards, the Host independent components dialog page lists those components. Select the check box of any of these components that you want to install on the current machine (the host computer of the installation package). Click Next.
 - d. At the Ready to start dialog page, click Next to proceed.
 - e. The Integrated Installation takes you through each component's necessary installation steps. Prior to each subsequent component's installation, the installer displays a list of component directories (see [Figure 32](#)) containing:
 - Components that have been installed, each with a green check mark next to it.
 - Components not yet installed, with no markings next to them.
 - The component to be installed next, with a blue check mark next to it.

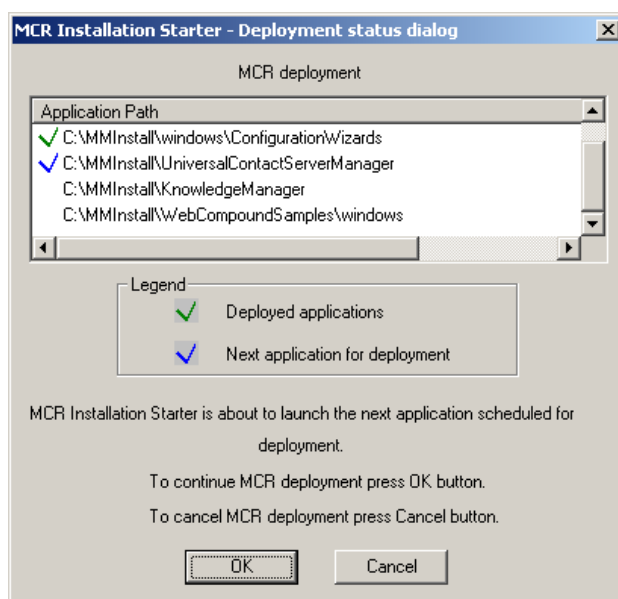


Figure 32: Deployment Status Window

2. Installing Java Environment and Libraries for eServices and UCS (Third Party Components)

Note: For an explanation of the difference between Third Party Components (executable) and ThirdPartyApplications (directory), see the note on [page 56](#).

- a. On the Choose Solution Destination Location page, enter the location of your solution destination folder and click Next.
- b. Confirm the location of your Java Environment and Libraries for eServices and UCS and click Install.
- c. Click Finish.

3. Installing Configuration Wizards

If you do not already have the Genesys eServices Configuration Wizards installed on this host, the Installation Starter will install them for you in the existing Wizards directory. You may need to restart your host after eServices Configuration Wizard installation.

If you already have the Genesys eServices Configuration Wizards installed on this host, click Cancel on the first page of the eServices Configuration Wizard setup.

4. Installing Universal Contact Server

- a. Click Next, and then click Install.
- b. Click Finish.

Note: Once you install any given eServices component on your computer, the wizard uses that directory for all other eServices components. You are not allowed to install these components across different directories.

5. Installing Interaction Server

- a. Click Next, then Install, if ready for installation.
- b. Click Finish.

6. Installing Web API Server

Note: If Co-Browsing Server is going to be installed with Web API Server on the same host, then you must install KANA Response Live Server first. See “Deploying KANA Response Live for co-browsing” on [page 78](#) for details.

In addition to following the steps listed here, installing the Web API Server requires that you refer back to the installation of your web server and your web services container. Installation of those components is described in Chapter 4, “Deploying Third-Party Web Applications,” on [page 55](#).

To install the Web API Server:

- a. Select the web server. Click **Next**.
 - b. Browse to the web server location. Click **Next**.
 - c. Browse to the web application container location. Click **Next**.
 - d. Select the tenant ID. Click **Next**.
 - e. Specify the Co-Browse Server name and location of the `worker.properties` file you created during installation of the web application container (`<CATALINA_HOME>\conf\jk` directory). Click **Next**.
 - f. Select the `ajp13` worker. Click **Next**.
 - g. Select a server-side code page—that is, the preferred language for processing all requests to Web API Server. Click **Next**.
 - h. Click **Install** if ready for installation.
 - i. Click **Finish**.
 - j. Next steps:
 - i. If necessary, create a special text file and update the `web.xml` configuration file to read parameters from this special file. See [Procedure: Specifying properties in a special file for passing initialization parameters to the Load Balancer Servlet](#), on [page 142](#) for details.
 - ii. If you want Web API Server to use system variables instead of, or in addition to, the parameters that are specified in the `web.xml` configuration file, specify system variables for passing initialization parameters to the Load Balancer Servlet. See [Procedure: Specifying system variables for passing initialization parameters to the Load Balancer Servlet](#), on [page 145](#) for details.
-
7. Installing E-Mail Server
 - a. Click **Next**, then **Install** if ready for installation.
 - b. Click **Finish**.
 8. Installing Classification Server
 - a. Click **Next**, then **Install** if ready for installation.
 - b. Click **Finish**.

9. Installing Training Server
 - a. Click **Next**, then **Install** if ready for installation.
 - b. Click **Finish**.
10. Installing Chat Server
 - a. Click **Next**, then **Install** if ready for installation.
 - b. Click **Finish**.
11. Installing Universal Contact Server Manager
 - a. Click **Next**, and then **Install**.
 - b. Click **Finish**.
12. Installing Knowledge Manager
 - a. Click **Next**, then **Install**.
 - b. Click **Finish**.

Note: The error log is written to the following location: Documents and Settings\<local account>\My Documents\KM_ErLog. The log is generated when Knowledge Manager is started. This change was made in release 7.6.1 to comply with Vista security systems.

13. Installing Co-Browsing Server
 - a. Enter the fully qualified domain name of the web server and the unique server identifier. Click **Next**.
 - b. Browse to the root folder containing your KANA Response Live server. (See “Deploying KANA Response Live for co-browsing” on [page 78](#).) Click **Next**.
 - c. Browse to the destination folder where the application will be installed, and click **Next**.
 - d. Click **Install**, then **Finish**.

The integrated installation is now complete.

End of procedure

Next Steps

- [Procedure: Creating Databases and Running Scripts](#)

Procedure: Specifying properties in a special file for passing initialization parameters to the Load Balancer Servlet

Purpose: Web API Server has been modified in eServices 8.1. The way in which parameters are passed to initialize Web API Server has been changed. In

8.1, the parameters may be stored in a special properties file. This addresses the restriction of previous versions where properties might be specified only in the web.xml file. In earlier versions, you had to stop Web API Server and manually update the web.xml file in different locations to use the new properties. While Web API Server will still need to be restarted for changes to take effect, only the properties file needs to be updated. Note the following limitations:

- The properties file has to be placed in the same location for each server on which Web API Server is deployed.
- The parameters in this file should represent connection parameters related to specific host.
- The account under which the Web Application Server is running must have read access permissions to the location of the properties file.
- This solution will work on all platforms, not only WebSphere, as long as the path to the properties file is properly configured according to the operating system requirements.
- Standard UNIX, Windows, and Linux installations do not offer this functionality by default. You must configure the web.xml file manually later.

Start of procedure

1. Create applications for each instance/clone/cell of the Web Application Server. After that he/she should follow the next steps to configure the same application over the cluster of Web Application Servers.
2. Create Web or Enterprise Application Archive and deploy it using the Web Application cluster. During this step, you must update the web.xml with information about the location of the properties file. The following is an example web.xml file using the current configuration style. The current configuration of the Load Balancing servlet requires storing host_config_serv, port_config_serv and app_config_serv in the web.xml file:

```
<servlet>
  <servlet-name>SvcDispatcherServlet</servlet-name>
  <display-name>SvcDispatcherServlet</display-name>

  <servlet-class>com.genesyslab.webapi.core.LoadBalancerServlet</servlet-
class>
  <init-param>
    <param-name>host_config_serv</param-name>
    <param-value>[ToBeChanged: <CONFIG_SERVER_HOST
value>]</param-value>
  </init-param>
  <init-param>
```

```

    <param-name>port_config_serv</param-name>
    <param-value>[ToBeChanged: <CONFIG_SERVER_PORT
value>]</param-value>
  </init-param>
  <init-param>
    <param-name>app_config_serv</param-name>
    <param-value>[ToBeChanged: <CONFIG_SERVER_APP value>]</param-value>
  </init-param>
<---- Added since 8.1.0 release ---->
  <init-param>
    <param-name>transport_address</param-name>
    <param-value></param-value>
  </init-param>
  <init-param>
    <param-name>transport_port</param-name>
    <param-value></param-value>
  </init-param>
  <load-on-startup>100</load-on-startup>
</servlet>

```

The alternate configuration points the Load Balancing servlet to a file with the required properties:

```

<servlet>
  <servlet-name>SvcDispatcherServlet</servlet-name>
  <display-name>SvcDispatcherServlet</display-name>

  <servlet-class>com.genesyslab.webapi.core.LoadBalancerServlet</servlet-
class>
    <init-param>
      <param-name>external_properties_file</param-name>
      <param-value>[ToBeChanged: <PATH_TO_PROPERTIES_FILE
value>]</param-value>
    </init-param>
    <load-on-startup>100</load-on-startup>
</servlet>

```

3. The Web Application cluster deploys this information on all cells/clones that are configured in it.

On each host, create and update the properties file with information specific to the application configured on the specific host (Host1, Host2 ... HostN). All of these files have to be stored at the same location according to the information in the web.xml file. The format of the properties file uses the standard approach for such files—each parameter is stored on a new line, with the name of the parameter and its value separated by “=”.

For example:


```

host_config_serv=config_server_host,backup_config_server_host
port_config_serv=config_server_port,backup_config_server_port
app_config_serv=config_server_app_name
transport_address=client_side_host_mapping
transport_port=client_side_port_mapping

```

In this configuration, the value of the `host_config_serv` parameter contains the primary and backup Configuration Server hosts separated by “,”. The value of the `port_config_serv` parameter contains primary and backup Configuration Server ports separated by “,”. The `app_config_serv` parameter contains information about the name of the application in Configuration Server. The `transport_address` and `transport_port` parameters contain information about client-side host/port definition for the connection to Configuration Server.

4. During initialization, Load Balancer will check the presence of the path to the configuration file. If everything is fine, it will initialize using parameters from it, otherwise it will use the old way of initialization.
5. Restart the Web Application Server.

End of procedure

Procedure: Specifying system variables for passing initialization parameters to the Load Balancer Servlet

Purpose: Another approach is to use environmental variables from the operating system to get access to the information required for Genesys Web API Server to initialize the Load Balancing Servlet.

Start of procedure

1. Create the following system variables:
 - `host_config_serv`—hosts of the primary/backup Configuration Servers, separated by “,”
 - `port_config_serv`—ports of the primary/backup Configuration Servers, separated by “,”
 - `app_config_serv`—name of the application in Configuration Server
 - `transport_address`—client-side host definition for the connection to Configuration Server
 - `transport_port`—client-side port definition for the connection to Configuration Server

Example 1:

```
host_config_serv=cshost.mycompany.com
port_config_serv=2020
app_config_serv=WebApiServer81
transport_address=127.0.0.1
transport_port=47774
```

Example 2:

```
host_config_serv=primarycshost.mycompany.com, backupcshost.mycompany.com
port_config_serv=2020, 4020
app_config_serv=WebApiServer81
transport_port=47774
```

End of procedure

Note: During startup of Web API Server, parameters from the properties file have top priority. After that, Web API Server will check for parameters in the environment variables. If parameters are present in the system and in the web.xml file, the parameters in the properties file will be used first, then data from the system variables, and parameters in the web.xml file will be used last.

Manual Installation of eServices Components

eServices components can also be installed manually. This section provides instructions on manually deploying eServices. The eServices Configuration Wizard can be used to configure the components, or they can be configured separately.

If wizards were not used to configure your eServices components, begin at [Procedure: Creating the Application object](#), on page 147.

If the wizards were used to configure your eServices solution, begin at [Procedure: Installing eServices components on Windows](#), on page 148 or [Procedure: Installing eServices components on AIX, Solaris, or Linux](#), on page 151.

Procedure: Creating the Application object

Purpose: Before installing the eServices component, a corresponding Application object must exist in the Configuration Database.

Prerequisites

- The product CD for the component you are installing.

Start of procedure

1. Create an Application object for the application if it does not already exist.
 - a. Import the application template from the product CD.
 - b. Create a new Application object based on the template.
2. Open the Properties dialog box of the Application object.
3. On the Server Info tab:
 - In the Host box, enter the name of the desired host.
 - In the Communication Port box, enter the port the component will use for listening.
4. On the Start Info tab enter some characters in the Working Directory, Command Line, and Command Line Arguments fields. These characters will be over-written with the correct values during the installation, but they cannot be left blank at this point.
5. On the Connections tab, add the appropriate connections. Refer to “Verifying Connections” on [page 153](#).
6. Click Apply.

End of procedure

Next Steps

- Install the application. See [Procedure: Installing eServices components on Windows](#) or [Procedure: Installing eServices components on AIX, Solaris, or Linux](#), on [page 151](#).

Procedure: Installing eServices components on Windows

Prerequisites

- The Application object has been configured. See [Procedure: Creating the Application object](#), on [page 147](#). Or, the application was configured using the eServices Configuration Wizards. See [Procedure: Configuring eServices objects using the wizard](#), on [page 104](#).
- (Prior to the 8.1.3 release only:) Java Environment and Libraries for eServices and UCS are installed on the host machine. Java Environment and Libraries for eServices and UCS are included in the integrated install, as described in [Procedure: Performing an integrated installation of all eServices servers](#), on [page 138](#). These components are not required for every eServices component.
- The product CD for the component you are installing.

Start of procedure

1. Locate the Setup.exe for the component you are installing (available on the Product CD).
2. Double-click Setup.exe.
3. Enter the login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password

4. For components that support client-side port functionality, a dialog box like the one shown in [Figure 33](#) will appear. See “Client-Side Port” on [page 172](#).

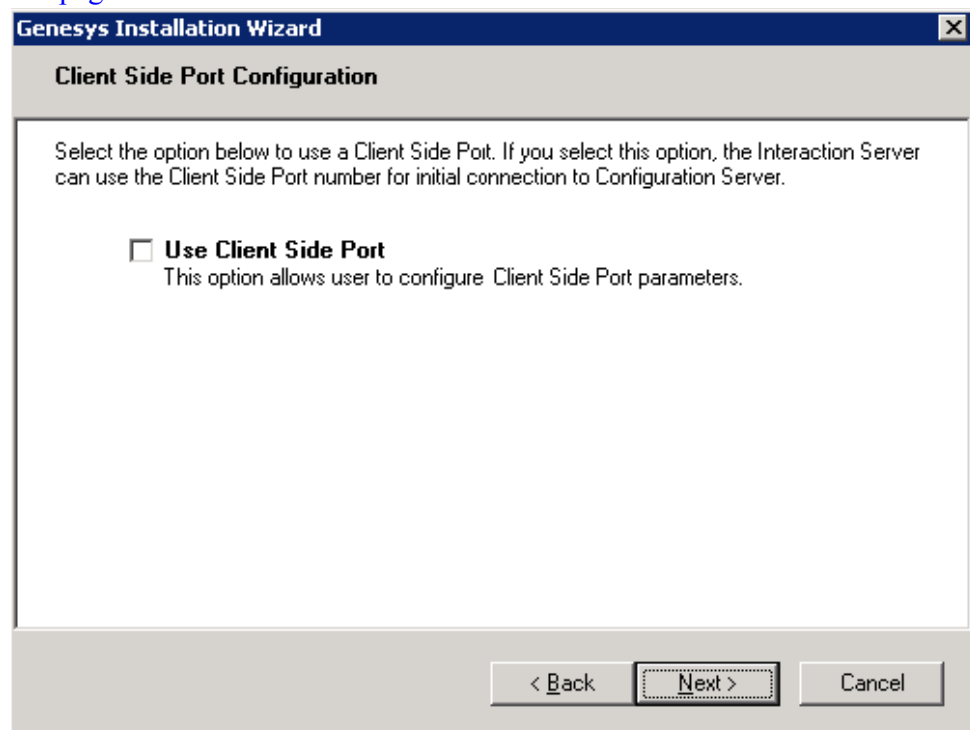


Figure 33: Use Client Side Port

5. If you are not setting up client-side port configuration, click Next and continue at [Step 8](#) on [page 150](#).

Note: Client-side port configuration is not supported for every eServices component. This dialog will only appear for Chat Server and Interaction Server installations. This step is optional. Refer to the *Genesys 8.1 Security Deployment Guide* for more information.

6. If you are setting up client-side port configuration for the initial connection to Configuration Server as described in the *Genesys 8.1 Security Deployment Guide*, select the Use Client Side Port check box to reveal additional fields (see Figure 34).

The screenshot shows a window titled "Genesys Installation Wizard" with a sub-header "Client Side Port Configuration". The main text says: "Select the option below to use a Client Side Port. If you select this option, the Interaction Server can use the Client Side Port number for initial connection to Configuration Server." Below this is a checked checkbox labeled "Use Client Side Port" with a tooltip that says "This option allows user to configure Client Side Port parameters." Underneath is a section titled "Configuration Options" containing two rows of labels and text boxes: "Specify Client Side Port Number." with a "Port:" label and a text box, and "Specify Client Side IP Address." with an "IP Address:" label and a text box. At the bottom are three buttons: "< Back", "Next >", and "Cancel".

Figure 34: Client Side Port Number and IP Address

7. Specify the following parameters and click Next:
Port—Enter any free port number (this is not the Listening port in the Server Info tab of the Application object).
IP Address—Enter the IP Address of the computer on which you are installing and running the application.

Note: After you have entered this information, the installation process will add the necessary command-line arguments (-transport-address and -transport-port) for connecting to Configuration Server during Application startup.

8. Select the appropriate Application object from the list.
9. Click Install.

End of procedure

Procedure: Installing eServices components on AIX, Solaris, or Linux

Prerequisites

- The Application object has been configured. See [Procedure: Creating the Application object](#), on [page 147](#).
- The product CD for the component you are installing.

Start of procedure

1. Locate the `install.sh` file for the application you are installing.
2. Start the installation script with the command `>install.sh`
3. Press Enter to confirm the host name for the installation
4. Enter the login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password
5. If the component you are installing supports client-side configuration, you will see the following prompt:
`Do you want to use Client Side Port option (y/n)?`
If you are not setting up client-side port configuration, enter `n` and continue at [Step 8](#) on [page 151](#).

Note: Client-side port configuration is not supported for every eServices component. This dialog will only appear for Chat Server and Interaction Server installations. This step is optional. Refer to the *Genesys 8.1 Security Deployment Guide* for more information.

6. If you are setting up client-side port configuration for the initial connection to Configuration Server as described in the *Genesys 8.1 Security Deployment Guide*, enter `y`.
7. Specify the following parameters:
 - Port—Enter any free port number (this is not the Listening port in the Server Info tab of the Application object).
 - IP Address—Enter the IP Address of the computer on which you are installing and running the application
8. From the list of applications, select one and enter its number in the list.
9. Press Enter to confirm the suggested destination directory, or choose another one.

10. Answer other questions, if required.

Note: On some Red Hat Linux platforms, eServices 8.1 components might fail to start and produce the error `./cfgutility: error while loading shared libraries: libstdc++.so.5: cannot open shared object file: No such file or directory`. Refer to “Compatibility Package for Red Hat Linux” on [page 189](#) for information about correcting this issue.

End of procedure

Creating Databases and Running Scripts

If you have not already done so, create the databases for Interaction Server, Universal Contact Server, and Co-Browsing Server. For UCS, you may also want to create an Archive database to store older UCS data and ease storage in the Main UCS database.

Note: If you intend to use the Event Logger functionality that was introduced in release 7.6.1, you will need to create another database and run the corresponding script. Configuration and deployment of Event Logger is explained in the “Event Logger” section in the “Interaction Server: Advanced Topic” section of the “Ongoing Administration and Other Topics” chapter of the *eServices 8.1 User's Guide*.

Procedure: Running scripts for your databases

Start of procedure

1. At the starting directory for each component, locate the SQL script folder:

Component	Script Folder
Interaction Server	Script
UCS	sql-scripts
Co-Browsing Server	dbscripts

2. Run the database scripts for each component.

Note: Only one Universal Contact Server database pair (Main and Archive) is allowed per Tenant.

End of procedure

Next Steps

- Confirm the components' connections are correct. See [“Verifying Connections”](#).

Verifying Connections

At this point in the eServices installation process, you must update the configuration settings for some components before continuing. Refer to Table 2 on [page 52](#) for a list of connections that must be set in the Configuration Layer. Refer to the *eServices 8.1 Reference Manual* for information on option settings.

Procedure: Verifying connections

Start of procedure

1. In Configuration Manager, verify that the required connections have been made in the eServices Application objects. See Table 2 on [page 52](#) for a detailed list of the connections you need to add to the eServices objects.
2. Add a connection from Stat Server to Interaction Server, or to Interaction Server Proxy (see “Interaction Server Proxy as Reporting Bus” on [page 154](#)).

3. Add a connection from Universal Routing Server to Interaction Server.

End of procedure**Next Steps**

- Test the components. See [“Testing eServices Components”](#).

Interaction Server Proxy as Reporting Bus

You can connect Stat Server to Interaction Server Proxy rather than directly to Interaction Server. This can be advantageous if you are running multiple Stat Servers.

To do this,

- Deploy Interaction Server Proxy, as described in “Deploying UCS Proxy or Interaction Server Proxy” on [page 223](#).
- Wherever this Guide speaks of connecting Stat Server to Interaction Server, instead make the connection to Interaction Server Proxy.

Testing eServices Components

The minimum required components for testing include:

- Local Control Agent
- DB Server
- Configuration Server
- Message Server
- Solution Control Server
- Solution Control Interface
- Stat Server
- Universal Routing Server

Procedure:**Testing the eServices servers**

Purpose: To test that the eServices servers will successfully launch and run free of errors.

Tip: General instructions for starting and stopping your eServices solution are in Chapter 16 on [page 241](#). This section provides startup instructions for the sole purpose of testing your solution.

Prerequisites

- Both Universal Contact Server and Interaction Server have connections to Stat Server (if you have not already made these connections).
- Both Stat Server and Universal Routing Server have connections to Interaction Server.
- Genesys Framework is running.
- Before testing chat and web form e-mail, restart the host(s) running the Chat Server, and ensure that your POP3 e-mail server is up and running.

Start of procedure

1. Start the servers in this order:
 - a. DB Server for Interaction Server
 - b. Interaction Server
 - c. Universal Contact Server
 - d. E-mail Server
 - e. Chat Server
 - f. Classification Server
 - g. Training Server
 - h. Co-Browsing Server
 - i. Web Server and Web Application Container for Web API Server launch
2. After each server starts, check its console window for errors.

End of procedure**Next Steps**

- [Procedure: Testing the components using the Web-based TestTool810](#)

Procedure:
Testing the components using the Web-based TestTool810

An application for testing the components, called TestTool810, is included as part of the installation of Web API Server. Use this tool to test the various Java classes and servlets that run the eServices solution.

Note: TestTool810 supports Internet Explorer. It does not support Mozilla Firefox.

Start of procedure

1. Open a web browser window and enter the following URL:
<http://<WebAPIServerhost>/WebAPI810/TestTool810>

Note: The address used to access the testing application is case sensitive.

You should see a window that looks like the one in [Figure 35](#).

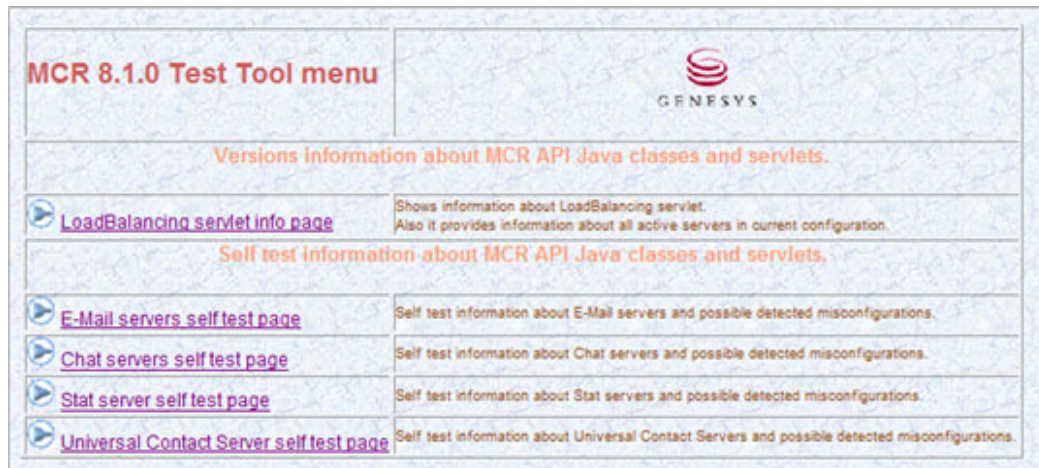


Figure 35: eServices Test Tools Page

2. Click any of the links to launch the various testing tools. The results of the test display in the bottom frame of the web page.

End of procedure**Next Steps**

- Verify interactions can be handled by Genesys Desktop. See [Procedure: Testing that Genesys Desktop can handle interactions](#).

Procedure: Testing that Genesys Desktop can handle interactions

Purpose: To log in the Genesys Desktop agent that will receive test eServices interactions (e-mail, chat, web e-mail, and web callback).

Test agent handling of interactions by configuring and installing Genesys Desktop. For instructions on how to do this, see the *Genesys Desktop 7.6 Deployment Guide*.

Note: After installing Genesys Desktop, use Configuration Manager to ensure that the Desktop Application object has a connection to your Interaction Server or if applicable, Interaction Server Proxy.

Start of procedure

1. Enter the following URL into your web browser:
`http://<Host_IP_Address:port>/gdesktop`
2. To log in an agent, enter the following information:
 - Agent ID
 - Password
 - The name of the agent's associated Place

End of procedure

Next Steps

- Test the installation's functionality. See [“Testing the Setup”](#).

Testing the Setup

Next you should test your installation's functionality. This requires that you implement an interaction workflow. The workflow and its associated routing strategies can be very simple. The simplest way to test the setup is to use the ABC Simple BP and the ABC Simple Chat BP business processes that are installed as part of your Interaction Workflow Samples installation.

Using the Sample Business Processes

The ABC Simple BP receives e-mail interactions that enter the system and delivers them to an agent group. This business process also allows the agent who works on a given interaction to reply to it and send a resulting outbound e-mail. The ABC Simple Chat BP does essentially the same thing, but for chat interactions; it receives chat interactions that enter the system and delivers them to an agent group. See the *Universal Routing 8.0 Business Process User's Guide* for more details on these and the other sample business processes. That guide also describes how to work with strategies and queues in the Interaction Design portion of IRD.

Procedure: Configuring ABC Simple BP for routing e-mail interactions

Start of procedure

1. Set E-mail Server's default-inbound-queue option (in the email-processing section) to the value Inbound queue, which is the name of the inbound queue in ABC Simple BP.

Note: If you installed the Interaction Workflow Samples before using the wizard installation, you will find that two Interaction Queue objects named Inbound queue and Outbound queue already exist in your <Tenant>\Scripts folder in Configuration Manager. Use the Inbound queue name as the value for this E-mail Server default queue option.

2. The Person object you plan to use for handling e-mail interactions must be a member of the Agent Group named E-mail distribution for processing, which serves as a target for interactions in the Process ABC strategy of this sample business process.
3. From the Interaction Design portion of IRD, if they are not already activated, activate the two strategies that this business process uses, Process ABC and Send ABC.

See the *Universal Routing 8.0 Business Process User's Guide* and *Universal Routing 8.0 Interaction Routing Designer Help* for additional instructions on creating and activating/loading routing strategies.

End of procedure

Next Steps

- To test chat functionality, configure the business process to route a chat. See [Procedure: Configuring ABC Simple Chat BP for routing chat interactions](#), on page 159.
- If chat is not being used, send a test e-mail. See [Procedure: Sending a test e-mail to a Desktop agent](#), on page 159 and [Procedure: Sending a test web form e-mail to a Desktop agent](#), on page 161.

Procedure: Configuring ABC Simple Chat BP for routing chat interactions

Start of procedure

Be sure you have set the following prior to testing this workflow:

1. If you installed the Interaction Workflow Samples before using the wizard installation, you will find that an Interaction Queue object named Chat inbound queue already exists in your <Tenant>\Scripts folder in Configuration Manager. That queue is also the value for the Chat Server default option (in the endpoints:<tenant_id> section). Make sure this is the case.
2. The Person object you plan to use for handling chat interactions must be a member of the Agent Group named Chat distribution for processing, which serves as a target for interactions in the Chat inbound strategy strategy of this sample business process.
3. From the Interaction Design portion of IRD, if they are not already activated, activate the three strategies this business process uses: Chat send transcript email strategy, Chat request transcript send strategy, and Chat inbound strategy.

See the *Universal Routing 8.0 Business Process User's Guide* and *Universal Routing 8.0 Interaction Routing Designer Help* for additional instructions on creating and activating/loading routing strategies.

End of procedure

Next Steps

- To test chat, launch a chat session. See [Procedure: Starting a test chat session with a Desktop agent](#), on page 160.
- To test e-mail, send a test e-mail. See [Procedure: Sending a test e-mail to a Desktop agent](#) and [Procedure: Sending a test web form e-mail to a Desktop agent](#), on page 161.

Procedure: Sending a test e-mail to a Desktop agent

Purpose: To send a test e-mail from your installed eServices solution to a Genesys Desktop agent using the ABC Simple BP routing strategy.

Start of procedure

1. If you have not already started the necessary components (as noted in “Testing eServices Components” on [page 154](#)), do so now.
2. Open your e-mail client and send an e-mail to the POP box that E-mail Server checks (the value of that component’s address option).
3. Ready an agent at the Genesys Desktop and look for the arrival of the e-mail interaction.
4. If the inbound e-mail fails (for example, E-mail Server does not see the message or E-mail Server sends the message to a bad directory), check each of the E-mail Server options, logs, and the routing strategies for possible errors.

End of procedure**Next Steps**

- To test chat, see [Procedure: Starting a test chat session with a Desktop agent](#).
- To send a test web form e-mail from the Genesys MCR 8.1 Platform SDK Java Samples Pages, see [Procedure: Sending a test web form e-mail to a Desktop agent](#), on [page 161](#)

Procedure:
Starting a test chat session with a Desktop agent

Purpose: To start a test chat session between a sample customer from the Genesys MCR 8.1 Platform SDK Java Samples Pages and a Genesys Desktop agent, using the ABC Simple Chat routing strategy.

Start of procedure

1. Start the web server and the web application container, if you have not already.
2. If you have not already started the necessary components (as noted in “Testing eServices Components” on [page 154](#)), do so now.
3. Ready an agent at the Genesys Desktop.
4. Open the Genesys MCR 8.1 Platform SDK Java Samples Pages by opening a browser navigating to `http://<WebAPIServerhost>/WebAPI810`.
5. Select the New samples based on PSDK Java API link to open the samples page.
6. Click Chat Sample with "user" typing notification.
7. Fill in customer data and click Start Chat.

8. Look for the arrival of the chat interaction at the Genesys Desktop.
9. If chat does not work (for example, a disconnected from chat server message appears or the customer joins but the agent does not), check each of the manual install steps. Specifically, check the:
 - Chat Server options.
 - Web API Server options.
 - WebAPI810 parameters, contained in the constants.jsp file.

End of procedure

Next Steps

- To test e-mail, see [Procedure: Sending a test e-mail to a Desktop agent](#), on [page 159](#) and [Procedure: Sending a test web form e-mail to a Desktop agent](#), on [page 161](#), otherwise you have completed all steps for testing the setup of eServices 8.1.

Procedure: Sending a test web form e-mail to a Desktop agent

Purpose: To send a test web form e-mail from the Genesys MCR 8.1 Platform SDK Java Samples Pages to a Genesys Desktop agent.

Start of procedure

1. If you have not already started the necessary components (as noted in “Testing eServices Components” on [page 154](#)), do so now.
2. Open the Genesys MCR 8.1 Platform SDK Java Samples Pages by opening a browser navigating to `http://<WebAPIServerhost>/WebAPI810`.
3. Select the New samples based on PSDK Java API link to open the samples page.
4. Select the E-mail over the Web link.
5. Fill in customer data and click Submit.
6. Ready an agent at the Genesys Desktop and look for the arrival of the web form e-mail interaction.
7. If web form e-mail does not work (for example, no thank you page displays, a thank you page displays with an error, or a thank you page displays but the e-mail never pops to the desktop), check each of the manual install steps.

End of procedure

Next Steps

- To prepare testing configuration for web callback processing, see [Procedure: Configuring WebCallback BP for routing web callback interactions](#) and [Procedure: Updating the Interaction Server database for processing web callback interactions](#), on page 163.

Procedure: Configuring WebCallback BP for routing web callback interactions

Purpose: Set the following prior to testing this workflow.

Start of procedure

1. If you installed the Web Callback Application before using the wizard installation, you will find that an Interaction Queue object named `New` already exists in your `<Tenant>\Scripts` folder in Configuration Manager. That queue is also the value for the Web API Server `wcbNewQueue` (in the `\...\SimpleSamples810\constants.jsp`). Make sure this is the case.
2. The Person object you plan to use for handling web callback interactions must be configured to work with voice media and be a member of the Agent Group named `WebCallback` distribution for processing, which serves as a target for interactions in the `Delivering` strategy of this sample business process.
3. Start Knowledge Manager and import the `UCS_impex.kme` file located in the folder `<Web Callback Application installation>` to receive standard responses related to WebCallback BP.
4. Open the `Send Email Notification` subroutine. Select the corresponding standard responses in the two `Acknowledgement receipt` objects. Save changes in the subroutine and then in IWD.
5. From the Interaction Design portion of IRD, if they are not already activated, activate the following strategies this business process uses:
 - `Preprocessing`
 - `Rescheduled by Agent`
 - `Rescheduled by Customer`
 - `Delivering`
 - `Stop By Customer`
 - `Stop By Agent`
 - `Expired Transfer Callbacks`
 - `Expired Conference Callbacks`

- Outbound notification email sending

Note: See the *Universal Routing 8.0 Business Process User's Guide* and *Universal Routing 8.0 Interaction Routing Designer Help* for additional instructions on creating and activating/loading routing strategies.

End of procedure

Next Steps

- To complete preparing testing configuration for web callback processing, see [Procedure: Updating the Interaction Server database for processing web callback interactions](#).

Procedure: Updating the Interaction Server database for processing web callback interactions

Purpose: To run the scripts required to update the Interaction Server database for processing web callback interactions.

Start of procedure

Run the appropriate script for your Interaction Server database. The scripts are located in the following directories:

- for DB2, in the <Web Callback Application installation>\Scripts\Db2 folder
- for MS SQL, in the <Web Callback Application installation>\Scripts\MsSQL folder
- for Oracle, in the <Web Callback Application installation>\Scripts\Oracle folder

End of procedure

Next Steps

- To test web callback, see [Procedure: Sending a test web callback to a Desktop agent](#), otherwise you have completed all steps for testing the setup of eServices 8.1.

Procedure:

Sending a test web callback to a Desktop agent

Purpose: To send a test web callback from the Genesys MCR 8.1 Platform SDK Java Samples Pages to a Genesys Desktop agent.

Start of procedure

1. If you have not already started the necessary components (as noted in “Testing eServices Components” on [page 154](#)), do so now.
2. Open the Genesys MCR 8.1 Platform SDK Java Samples Pages by opening a browser and navigating to `http://<WebAPIServerhost>/WebAPI810`.
3. Select the New samples based on PSDK Java API link to open the samples page.
4. Select the WebCallback link.
5. Fill in customer data and click Request callback.
6. Ready an agent at the Genesys Desktop and look for the arrival of the web callback interaction.

End of procedure

Next Steps

- You have completed all steps for testing the setup of eServices 8.1.

6

Deploying Java Components on 64-bit Windows and UNIX

This chapter describes how to deploy the eServices 8.1 Java components on 64-bit Windows and 64-bit UNIX operating systems. It covers these topics:

- [Old and New Styles of Installation Package, page 165.](#)
- [Installing Components on 64-bit Windows Operating Systems, page 166](#)
- [Installing Components on 64-bit UNIX Operating Systems, page 167](#)
- [Supported Versions of JDK/JRE, page 168](#)

Old and New Styles of Installation Package

In the 8.1.0 release, a new type of installation package (IP) was introduced for SMS Server. This “new-style” IP does not differentiate between 32- and 64-bit Java platforms, unlike the “old-style” IP which required different procedures for 32-bit and 64-bit Java. Between the 8.1.0 and 8.1.3 releases, some other Java components switched to the “new-style” IP and some retained the “old-style.” Starting with the 8.1.3 release, all Java-based components use the new style of IP. [Table 8](#) lists the components in question and shows when (and if) they changed to the new-style IP.

Note: Knowledge Manager and UCS Manager do not support 64-bit Java, but they did shift to the new-style IP in release 8.1.3.

In the following procedures, any differences between the deployment of the two IP types is clearly noted. [Table 8](#) lists the components in question and shows when (and if) they changed to the new-style IP.

Table 8: Old- and New-Style IPs

Component	Change to New-Style IP
Classification Server	8.1.3 and later
E-mail Server	8.1.2 and later
Knowledge Manager	8.1.3 and later
SMS Server	all 8.1.x releases
Social Messaging Server	8.1.1 and later
Training Server	8.1.3 and later
Universal Contact Server	8.1.1 and later
UCS Manager	8.1.1 and later

Installing Components on 64-bit Windows Operating Systems

The shared steps for all IPs are the following:

1. Verify the Oracle JDK for Windows x64, version 1.6.0_26 or higher (as long as it is within the 1.6 family) is installed.
2. Run the component's IP (located in the `Windows_x64` folder on the CD).
3. The next steps are:
 - For the “old-style” IPs (see [Table 8](#)),
 - i. Install the MCR Java Environment and Libraries for eServices and UCS.
 - ii. Manually modify the file `<installation folder>\<filename>.ini`. Replace the placeholder `<<JVM 64-BIT LOCATION>>` with the actual location of the `jvm.dll` file of the 64-bit JDK that is installed.

Note: In the preceding step, `<filename>` could be either `JavaServerStarter.ini`, `JavaEmailServerDriver.ini`, or `ContactServerDriver.ini`.

- For the “new-style” IP, when you are prompted with a list of JDK choices, select the required version (1.6.0_26 or higher) of Oracle JDK.

Installing Components on 64-bit UNIX Operating Systems

The shared steps for all IPs are the following:

1. Verify that JRE x64 for UNIX is installed.
 - For Oracle JRE, the version must be 1.6.0_26 or higher.
 - For IBM JRE (for AIX), the version must be 1.6.0 build pap6460sr9fp2-20110627_03(SR9 FP2) or higher.
2. Run the component’s UNIX IP. When prompted, enter the path to the Java executable (from JRE x64).
3. The next steps are
 - For the “old-style” IPs:
 - i. For Classification Server only: install the MCR Java Environment and Libraries for eServices and UCS
 - ii. Manually modify the file `<installation folder>\<filename>.sh` as outlined in [Table 9](#). For each component, update the specified file. Replace the text in the “Old” column with the text in the “New” column. See [Table 8](#) on [page 166](#) for a list of old versus new IPs.

Table 9: Updates for 64-bit UNIX Installations

Component	Filename	Old	New
Social Messaging Server	smsserver.sh	\$GES_HOME_810/jre/bin/java	<path to Java executable, including executable name>
Classification Server	Clfserver.sh		
Universal Contact Server	contactServer.sh	JAVA_HOME=\${GES_HOME}/jre	JAVA_HOME=<JRE location> ^a
E-mail Server Java	emailServer.sh		

a. For Solaris installations, the line “JAVACMD=\${JAVA_HOME}/bin/java” must be adjusted because x64 java executable is located in the folder “/bin/sparcv9”

- For the “new-style” IP, enter the path to the Java executable (from JRE x64) when you are prompted to do so during the installation.

Supported Versions of JDK/JRE

Regarding supported versions of JDK/JRE, keep the following in mind:

- JDK/JRE versions that are *lower* than specified in the foregoing procedures are not supported.
- For the 8.1.3 release,
 - JDK/JRE for Oracle 1.8 or higher is not supported, except that Knowledge Manager additionally does not support JDK/JRE 1.7.
 - JDK/JRE 1.8 for AIX or higher is not supported.
- For releases prior to 8.1.3,
 - JDK/JRE for Oracle 1.7 or higher is not supported.
 - JDK/JRE 1.6.1 for AIX or higher is not supported.

7

TLS and Client-Side Port

This chapter describes TLS and client-side port functionality in eServices 8.1. It includes these sections:

- [TLS, page 169](#)
- [Client-Side Port, page 172](#)

TLS

eServices 8.1 supports Transport Layer Security (TLS) protocol to secure data exchange between components. [Table 10](#), [Table 11](#), and [Table 12](#) in this chapter list various component connections and indicates whether the specific connection supports TLS. In these tables, the components listed in the first column are connecting to the components that are listed in the top row.

Further information on using TLS is available as follows:

- Refer to the *Genesys 8.1 Security Deployment Guide* for information about how to configure TLS for eServices components.
- For detailed information on using TLS with UCS and UCS clients, see the “TLS” section of the “Security and Authentication” page of the [Context Services User’s Guide](#).
- UCS, E-mail Server, and UCS Manager use Genesys Platform SDK to implement TLS connectivity. Refer to the “Using and Configuring Security Providers” page of the [Platform SDK Developer’s Guide](#) for detailed configuration of TLS providers, as well as recommended Java versions for TLS on Windows.

Table 10: TLS Support 1

Connection	Configura- tion Server	Configura- tion Server Proxy	Solution Control Server	Message Server	Interaction Server	Universal Contact Server
Interaction Server	Yes	Yes	N/A	Yes	Yes	Yes
Interaction Server Proxy	Yes	Yes	N/A	Yes	Yes	N/A
Universal Contact Server	Yes	Yes	N/A	Yes	N/A	
Universal Contact Server Proxy	Yes	Yes	N/A	Yes	N/A	Yes
E-mail Server	Yes	Yes	N/A	Yes	Yes	Yes
Chat Server	Yes	Yes	N/A	No	Yes	Yes
SMS Server	Yes	Yes	Yes	No	Yes	N/A
Social Messaging Server	Yes	Yes	N/A	No	Yes	N/A
Classifica- tion Server	Yes	Yes	N/A	No	N/A	No
Web API Java	Yes	Yes	Yes	No	Yes	Yes
Web API .NET	Yes	Yes	Yes	No	Yes	Yes

Table 11: TLS Support 2

Connection	E-mail Server	Chat Server	SMS Server	Social Messaging Server	Classification Server
Interaction Server	Yes	Yes	No	No	No
Universal Contact Server	N/A	N/A	N/A	N/A	N/A
Universal Contact Server Proxy	N/A	N/A	N/A	N/A	N/A
E-mail Server		N/A	N/A	N/A	N/A
Chat Server	N/A		N/A	N/A	N/A
SMS Server	N/A	No		N/A	N/A
Social Messaging Server	N/A	N/A	N/A		N/A
Classification Server	N/A	N/A	N/A	N/A	
Web API Java	Yes	Yes	N/A	N/A	N/A
Web API .NET	Yes	Yes	N/A	N/A	N/A

Table 12: TLS Support 3

Connection	Web API Java	Web API .NET	DB Server	Universal Routing Server	Stat Server
Interaction Server	N/A	N/A	Yes	N/A	Yes
Universal Contact Server	N/A	N/A	N/A	N/A	No

Table 12: TLS Support 3 (Continued)

Connection	Web API Java	Web API .NET	DB Server	Universal Routing Server	Stat Server
Universal Contact Server Proxy	N/A	N/A	N/A	N/A	N/A
E-mail Server	N/A	N/A	N/A	N/A	N/A
Chat Server	N/A	N/A	N/A	N/A	N/A
SMS Server	N/A	N/A	N/A	N/A	N/A
Social Messaging Server	N/A	N/A	N/A	N/A	N/A
Classification Server	N/A	N/A	N/A	N/A	N/A
Web API Java			N/A	N/A	Yes
Web API .NET			N/A	N/A	Yes

Note: Training Server does not support the TLS connection type. Stat Server and Universal Routing Server support TLS with their connections to Interaction Server. Universal Contact Server Proxy supports TLS with its connection to Universal Contact Server. No other connections to eServices components are supported by UCS Proxy.

Client-Side Port

eServices 8.1 supports client-side port definition between components as outlined in [Table 13](#), [Table 14](#), and [Table 15](#). In these tables, the components listed in the first column are connecting to the components that are listed in the

top row. Refer to the *Genesys 8.1 Security Deployment Guide* for information about how to configure client-side port parameters.

Table 13: Client-side Port Support 1

Connection	Configura- tion Server	Configura- tion Server Proxy	Solution Control Server	Message Server	Interaction Server	Universal Contact Server
Interaction Server	Yes	Yes	N/A	Yes	Yes	Yes
Interaction Server Proxy	Yes	Yes	N/A	Yes	Yes	N/A
Universal Contact Server	Yes	Yes	N/A	Yes	N/A	
Universal Contact Server Proxy	Yes	Yes	N/A	Yes	N/A	Yes
E-mail Server	Yes	Yes	N/A	Yes	Yes	Yes
Chat Server	Yes	Yes	N/A	Yes	Yes	Yes
SMS Server	Yes	Yes	Yes	Yes	Yes	N/A
Social Messaging Server	Yes	Yes	N/A	Yes	Yes	N/A
Classifica- tion Server	Yes	Yes	N/A	Yes	N/A	No
Web API Java	Yes	Yes	Yes	Yes	No	No
Web API .NET	Yes	Yes	Yes	Yes	No	No

Table 14: Client-side Port Support 2

Connection	Universal Contact Server Proxy	E-mail Server	Chat Server	SMS Server	Social Messaging Server	Classification Server
Interaction Server	N/A	Yes	Yes	Yes	Yes	Yes
Universal Contact Server	N/A	N/A	N/A	N/A	N/A	N/A
Universal Contact Server Proxy		N/A	N/A	N/A	N/A	N/A
E-mail Server	N/A		N/A	N/A	N/A	N/A
Chat Server	N/A	N/A		N/A	N/A	N/A
SMS Server	N/A	N/A	No		N/A	N/A
Social Messaging Server	N/A	N/A	N/A	N/A		N/A
Classification Server	N/A	N/A	N/A	N/A	N/A	
Web API Java	N/A	No	No	N/A	N/A	N/A
Web API .NET	N/A	No	No	N/A	N/A	N/A

Table 15: Client-side Port Support 3

Connection	Web API Java	Web API .NET	DB Server	Universal Routing Server	Stat Server
Interaction Server	N/A	N/A	No	N/A	Yes
Universal Contact Server	N/A	N/A	N/A	N/A	No

Table 15: Client-side Port Support 3 (Continued)

Connection	Web API Java	Web API .NET	DB Server	Universal Routing Server	Stat Server
Universal Contact Server Proxy	N/A	N/A	N/A	N/A	N/A
E-mail Server	N/A	N/A	N/A	N/A	N/A
Chat Server	N/A	N/A	N/A	N/A	N/A
SMS Server	N/A	N/A	N/A	N/A	N/A
Social Messaging Server	N/A	N/A	N/A	N/A	N/A
Classification Server	N/A	N/A	N/A	N/A	N/A
Web API Java			N/A	N/A	No
Web API .NET			N/A	N/A	No

Note: Training Server does not support the client-side port connection type.

8

Deploying an E-Mail System in Secured Mode

This chapter describes how to configure an e-mail system to work in secured mode using TLS/SSL. This applies to POP3, IMAP4, and SMTP. The purpose is to generate and install a public/private key pair. The information in this chapter is divided among the following sections:

- [Configuring the Corporate E-Mail Server, page 177](#)
- [Configuring TLS/SSL for E-mail Server, page 178](#)

Configuring the Corporate E-Mail Server

Configure TLS/SSL in the Corporate E-mail Server. Follow the constructor recommendations to generate a certificate and configure TLS/SSL on ports POP3, IMAP and SMTP.

The following is an example of generation of a certificate with keytool (keytool is a Java utility that is available with the JRE. The utility can be found in <eServices_Install_Dir>/jre/bin for Unix operating systems, and in <eServices_Install_Dir>\jre\bin for Windows operating systems):

```
keytool -genkey -v -alias hostname.example.com
-dname "CN=hostname.example.com, OU=IT, O=ourcompany, C=FR" -keypass
<certificate_password>
-keystore <certificate_name>.keystore -storepass <certificate_password>
-keyalg "RSA" -sigalg "SHA1withRSA" -keysize 2048 -validity 3650
```

The arguments used in this command are the following:

- alias—Defines an alias in keystore, to store the key.
- dname—Distinguished Name, a comma-separated list made up of the following, in the following order:

CN—Common Name. This must be the name of the host where the corporate e-mail server is running. It must be the host name used in E-mail Server's settings; for example, if connecting to a POP 3 server, the option `server` in the `pop-client` section must have this value.

OU—Organizational Unit Name

O—Organization Name

L—Locality Name (city)

S—State

C—Country Name

Notes:

- The abbreviations are not case-sensitive.
- Only CN is required.

-keypass—Password of the key of the certificate.

-keystore—Specifies the keystore used.

-storepass—Password of the keystore.

-keyalg—Algorithm used to generate the key. Possible values are DSA and RSA. More information is available at <http://docs.oracle.com/javase>.

-sigalg—Specifies the algorithm used to sign the key.

-keysize—Specifies the size of the key.

-validity—Defines the validity of the certificate, in days. The value in the example is 3,650 days, or 10 years.

Configuring TLS/SSL for E-mail Server

This section describes procedures for configuring your E-mail Server application to work with TLS/SSL. This configuration is supported for E-mail Server version 7.0 and above. The following procedures are included in this section:

- [Procedure: Generating the .truststore file, on page 179](#)
- [Procedure: Modifying the E-mail Server startup command line on Windows, on page 179](#)
- [Procedure: Modifying the E-mail Server startup command line on Unix, on page 180](#)
- [Procedure: Configuring E-mail Server's POP, IMAP, and SMTP ports, on page 181](#)

Procedure: Generating the .truststore file

Prerequisites:

- The corporate e-mail server is configured to work in secured mode. See “Configuring the Corporate E-Mail Server” on [page 177](#).

Start of procedure

1. From the certificate on the Corporate E-mail Server, extract the public key.

The following is an example of extracting a public key using keytool:

```
keytool -export -v -alias hostname.example.com -file  
<certificate_name>.cer -keystore <certificate_name>.truststore  
-storepass <certificate_password>
```

```
keytool -import -alias hostname.example.com -file  
<certificate_name>.cer -keystore client.truststore -storepass  
<certificate_password>
```

At this point, the `client.truststore` file contains the public key.

2. Copy it to the host on which E-mail Server is running.

End of procedure

Next Steps

- Provide E-mail Server with the location of the `.truststore` file. See [Procedure: Modifying the E-mail Server startup command line on Windows](#) or [Procedure: Modifying the E-mail Server startup command line on Unix](#).

Procedure: Modifying the E-mail Server startup command line on Windows

Purpose: To provide E-mail Server with the location of the `.truststore` file on startup.

Prerequisites

- The `.truststore` file has been created. See [Procedure: Generating the .truststore file](#), on [page 179](#).

Start of procedure

1. Open `JavaEmailServerDriver.ini` in a text editor.
2. In the `[JavaArgs]` section, add the following:
`-Djavax.net.ssl.trustStore=<path to certificate>`
3. Save and close the file.

End of procedure**Next Steps**

- Configure ports required for E-mail Server to work in secured mode. See [Procedure: Configuring E-mail Server's POP, IMAP, and SMTP ports](#).

Procedure:
Modifying the E-mail Server startup command line on Unix

Purpose: To provide E-mail Server with the location of the `.truststore` file on startup.

Prerequisites

- The `.truststore` file has been created. See [Procedure: Generating the .truststore file](#), on page 179.

Start of procedure

1. Locate the E-mail Server startup file (`emailServer.sh`).
2. Open the file in a text editor and modify the startup command line so E-mail Server can locate the `.truststore` file. For example:

```
java -Djavax.net.ssl.trustStore="<path to certificate>" --Xmx512M
....
```
3. Save and close the file.

End of procedure**Next Steps**

- Configure ports required for E-mail Server to work in secured mode. See [Procedure: Configuring E-mail Server's POP, IMAP, and SMTP ports](#).

Procedure: Configuring E-mail Server's POP, IMAP, and SMTP ports

Purpose: The ports must be configured in order for the e-mail system to work in secured mode.

Prerequisites

- The `.truststore` file has been generated and E-mail Server's startup command line has been modified. See [Procedure: Generating the .truststore file](#), on page 179 and [Procedure: Modifying the E-mail Server startup command line on Windows](#), on page 179 or [Procedure: Modifying the E-mail Server startup command line on Unix](#), on page 180.

Start of procedure

1. In Configuration Manager, open the properties for your E-mail Server application.
2. In the Options tab, locate the `[pop-client]` section for IMAP and configure the type, port, and `enable-ssl` options. For example:
`[pop-client1]`
`type = IMAP`
`port = 993` (the default SSL port for IMAP)
`pop-connection-security = ssl-tls`
3. Locate the `[pop-client]` section for POP3 and configure the type, port, and `enable-ssl` options. For example:
`[pop-client2]`
`type = POP3`
`port = 995` (the default SSL port for POP3)
`pop-connection-security = ssl-tls`
4. Locate the `[smtp-client]` section and configure the port and `enable-ssl` options. For example:
`port = 465` (the default SSL port for SMTP)
`smtp-connection-security = ssl-tls`
5. Save your changes.

End of procedure

9

Multiple Interaction Servers in a Single Tenant

This chapter discusses configuring multiple Interaction Server instances in a single tenant. It covers the following topics:

- [Overview, page 183](#)
- [Configuring, page 184](#)

Overview

If you deploy separate solutions within a single tenant, the usual architecture has a single Interaction Server processing all the interactions according to business processes that are defined for the entire tenant (which may include separate business processes for the separate solutions). The server also uses a single database to store all the interactions.

However, assigning a separate instance of Interaction Server to each solution can provide the following benefits.

Functional Separation

- Stopping or terminating one solution does not affect the others.
- New solutions can be deployed with their own Interaction Servers without disturbing existing solutions.
- The Interaction Servers assigned to different solutions can be configured most appropriately to handle the specific solution
- The overall system performance can be higher:
 - The servers do not affect each other and can utilize different hardware.
 - The servers use separate databases.
- Each solution can be managed and maintained separately by different teams.

Data Separation

Physically separating the data of different solutions within one tenant provides the following benefits:

- Databases can be managed separately. Backup or restoration of one solution does not affect other solutions.
- Data corruption or hardware failures in one database only affects one solution.
- Performance is improved.

Note: If you have separate objects in a single tenant for use with multiple Interaction Servers, you must similarly use multiple instances of Stat Server, desktop applications, reporting applications, and any other application that supports a connection to only one application of Interaction Server or T-Server type.

Configuring

You can use Configuration Server security to allow a specific server instance to work only with a subset of Business Processes within a tenant. At a high level, the procedure is as follows:

1. Use the Interaction Design window of Interaction Routing Designer to create an account and associate a Business Process with it, as described in the “Last Step in Creating a Business Process” topic of *Universal Routing 8.1 Interaction Routing Designer Help* (control of access to other configuration s is not supported at this time).
2. Associate any other desired Business Processes with this account.
This creates an Access Group object that has access to the associated Business Process.
3. In Genesys Administrator or Configuration Manager, create a Person object and add it to the Access Group.
4. Configure the selected Interaction Server instance to run under this new Person account:
 - Configuration Manager: In the Log On As area of the Security tab of the Interaction Server Application object, select This Account, then select the desired Person in the resulting Add User dialog box.
 - Genesys Administrator: On the Configuration tab of the Interaction Server Application object, go to the Server Info area, clear the Log On As System checkbox, and click Browse to select an account.



Chapter

10

Deploying eServices in UNIX Environments

This chapter describes procedures for configuring and installing eServices components in UNIX environments—Solaris, AIX, and Linux—as well as for configuring the Web API components in a WebSphere or WebLogic environment.

This chapter has these sections:

- [Deploying eServices on UNIX Hosts, page 185](#)
- [Compatibility Package for Red Hat Linux, page 189](#)
- [Deploying Genesys Web Applications for WebSphere on AIX, Solaris, or Windows, page 190](#)
- [Deploying Genesys Web Applications for WebLogic on Solaris, page 201](#)

Note: You can deploy all eServices 8.1 components on UNIX platforms, except for the following user interfaces:

- Knowledge Manager
 - Universal Contact Server Manager
 - Interaction Workflow Samples
-

Deploying eServices on UNIX Hosts

Configuring and installing eServices 8.1 on Solaris, AIX, or Linux follows the general procedure introduced in Chapter 5, beginning on [page 97](#). You can use the eServices 8.1 Configuration Wizards on a Windows host to configure components that will run in UNIX environments. For the installation phase, however, you should proceed in a slightly different fashion than you would for a Windows-only environment. The main difference is that after configuration, you will need to manually copy the individual component installation packages

to each UNIX host, as initially suggested in “Installing eServices Components” on [page 136](#). Also consider the guidelines under “Copying Installations to Remote Computers” in the “Genesys Wizards” section of Chapter 4, “Deployment Overview,” of the *Framework 8.1 Deployment Guide*.

After you copy the software to the UNIX hosts, do one of the following:

- Run the setup for individual components manually.
- Use the eServices CD on each host to run the setup for the components directly from that CD.

Note on Java Environment and Libraries for eServices and UCS

Some components require Java Environment and Libraries for eServices and UCS (JELEU), supplied as part of eServices. For all platforms, before you install these components, you must first install the Java Environment and Libraries for eServices and UCS. On Windows, if you use `IPStarter.exe`, it takes care of this automatically. On Solaris, AIX, or Linux, however, you must manually run the Java Environment and Libraries for eServices and UCS setup on each host before you run the setup for the other eServices components for that host. [Table 16](#) lists eServices components according to their requirement for JELEU.

Table 16: Components and Releases not Requiring JELEU

Component	JELEU Required?
Interaction Server	No
Interaction Server Proxy	No
UCS Proxy	No
SMS Server	No
Web API Server	No
E-mail Server	Prior to release 8.1.2 only
Social Messaging Server	Prior to release 8.1.1 only
UCS	Prior to release 8.1.1 only
UCS Manager	Prior to release 8.1.1 only
Classification Server	Prior to release 8.1.3 only
Knowledge Manager	Prior to release 8.1.3 only
Training Server	Prior to release 8.1.3 only

Interaction Server, Chat Server, Interaction Server Proxy, UCS Proxy, and Web API Server are exceptions to this guideline.

Interaction Server, Chat Server, Interaction Server Proxy, UCS Proxy, and Web API Server (on all platforms) do not require Java Environment and Libraries for eServices and UCS.

For an explanation of the difference between Third Party Components (executable) and ThirdPartyApplications (directory), see the note on [page 56](#).

Procedure: Deploying eServices on Solaris, AIX, or Linux from a Windows host

Purpose: To suggest how you might use the eServices Configuration Wizards on a Windows host to configure and install components to other hosts for Solaris, AIX, or Linux deployments.

Start of procedure

1. (This step is not required for release 8.1.3 and higher): On each Solaris, AIX, or Linux host, set the environment variable `GES_HOME_810` to target the directory containing eServices components. Refer to the online UNIX manual for the syntax for the `set` command in the shell you use. If for any reason you cannot complete the deployment of your eServices Solution during one login session, you may need to set the variable again after a new login.

Warning! Do not include any spaces in the value of the `GES_HOME_810` environment variable (for example, use `/home/lab/GCTI/eServices81` instead of `/home/lab/GCTI/eServices 8.1`).

2. When you run the eServices Configuration Wizard, it will ask you to specify the destination for copying installation packages. Do one of the following:
 - Create a shared directory on your Solaris, AIX, or Linux computer for the installation package deployment, making sure that it is accessible from the Windows host.
 - Create a shared directory on a Windows host (as recommended in “Create a Shared Directory” on [page 101](#) for the Windows installation), making sure that it is accessible from the Solaris, Linux, or AIX host(s) that will run your eServices components.

In either case you must copy all installation packages to the Solaris, Linux, or AIX computer(s) that will host your eServices solution.

3. On a Windows machine, run the eServices Configuration Wizards as described in Chapter 5, beginning on [page 97](#).

The Wizards copy the installation packages using a directory structure such as the following:

```
<componentA>
    IBM_AIX
    Linux
    Solaris
    Windows_2008
    Windows_Server_2003

<componentB>
    IBM_AIX
    Solaris
    Windows_2008
    Windows_Server_2003

...
aix
    ThirdPartyComponents

linux
    ThirdPartyComponents

solaris
    ThirdPartyComponents

windows
    CommonWizardComponentSet
    ThirdPartyComponents
```

4. After completing configuration using the Wizards, begin installation by running the installation package in `<platform>\ThirdPartyComponents` on each host where you plan to install components.
5. Continue installation by running the installation package for each remaining eServices component. You will find that package in the `<platform>` subdirectory contained in the directory named for each component.

End of procedure

Next Steps

- For Solaris deployments, you need to install and configure one of the following web application servers:
 - [Deploying Genesys Web Applications for WebLogic on Solaris, page 201](#)
 - [Deploying Genesys Web Applications for WebSphere on AIX, Solaris, or Windows, page 190](#)
 - [Configuring Apache and Tomcat for Linux or Solaris, page 76](#)
- For AIX deployments, you need to deploy the following web application server:

[Deploying Genesys Web Applications for WebSphere on AIX, Solaris, or Windows, page 190](#)

- For Linux deployments, you need to install and configure the following:
 - [Configuring Apache and Tomcat for Linux or Solaris, page 76](#)

Compatibility Package for Red Hat Linux

On some Red Hat Linux platforms, eServices 8.1 components might fail to start and produce the following error:

```
./cfgutility: error while loading shared libraries: libstdc++.so.5:
cannot open shared object file: No such file or directory
```

To correct this issue, a compatibility package must be added to the Operating System. The exact name of the package depends on the OS version of your Linux host. Here are several examples for different Linux versions:

Example for Red Hat 4:

```
alexey@rh40pd32$ rpm -qf /usr/lib/libstdc++.so.5
compat-libstdc++-33-3.2.3-47.3
```

The name of package indicates that it is not a part of the OS, but the part of a compatibility package.

Example for Red Hat 5:

```
alexey@rh50-vm$ rpm -qf /usr/lib/libstdc++.so.5
compat-libstdc++-33-3.2.3-61
```

The name of package indicates that it is not a part of the OS, but the part of a compatibility package.

Note: A package newer than `compat-libstdc++-33-3.2.3-61` might already exist for Red Hat 5 Linux.

If you have yum installed then you can use it to get the package from a predefined Red Hat repository. For example:

```
yum -y install compat-libstdc++-33
```

If you do not have yum installed then you need to find package manually and use rpm for installation. For example:

```
rpm -i
ftp://ftp.pbone.net/mirror/archive.fedoraproject.org/fedora/linux/core/
6/x86_64/os/Fedora/RPMS/compat-libstdc++-33-3.2.3-61.i386.rpm
```

This package could possibly be found here:

<http://rpm.pbone.net/index.php3/stat/4/idpl/3416184/com/compat-libstdc++-33-3.2.3-61.i386.rpm.html>

Note: Refer to the *Genesys Supported Operating Environment Reference Guide* for more information.

Deploying Genesys Web Applications for WebSphere on AIX, Solaris, or Windows

In eServices 8.1, you can use the IBM WebSphere Application Server in AIX, Solaris, and Windows deployments. After configuring and installing eServices as outlined in Chapter 5 on [page 97](#), and in “Deploying eServices on UNIX Hosts” on [page 185](#), you need to enable your WebSphere deployment to use the Web API Server and the Multimedia 8.1 Samples.

For AIX deployments, WebSphere is the only supported web server. For Windows deployments, Web API Server 8.1.0 can work with WebSphere, Apache/Tomcat or IIS/Tomcat platforms. For Solaris, however, you can use WebSphere, Apache/Tomcat, or WebLogic in your deployment. For WebSphere deployment procedures, see:

- [Installing Genesys web applications for WebSphere on AIX or Solaris, page 191](#)
- [Deploying Genesys web applications for WebSphere on AIX, Solaris, or Windows, page 192](#)
- [Testing the WebSphere installation on AIX, Solaris, or Windows, page 199](#)

Note: For information about deploying on Solaris with Tomcat or WebLogic, see the following sections:

- “Configuring Apache and Tomcat for Linux or Solaris” on [page 76](#)
 - “Deploying Genesys Web Applications for WebLogic on Solaris” on [page 201](#)
-

Prerequisites

This section assumes that you have your WebSphere environment ready to go, and that you have completed the model setup. The steps outlined in this section use values you entered for that model installation.

Environment

This document refers to the root directory for IBM WebSphere as `<WEBSPHERE_HOME>`. The root directory for the IBM HTTP Server, an Apache-based Web Server, is `<IHS_HOME>`. `<MCR_API_EAR>` represents the base

directory of your MCR_API_EAR.ear file location. A sample directory structure in this case might be:

```
<WEBSPPHERE_HOME>/AppServer/profiles/AppSrv01/installedApps/DefaultNodeCell/MCR_API_EAR.ear.
```

Note: For instructions on how to configure and install Genesys Web API Server see the Web API Server sections of the following procedures:

- [Procedure: Configuring eServices objects using the wizard](#), on page 104
 - [Procedure: Performing an integrated installation of all eServices servers](#), on page 138
-

Procedure:

Installing Genesys web applications for WebSphere on AIX or Solaris

Prerequisites

- The environment variable GES_HOME_810 is set. See [page 187](#).

Start of procedure

1. Run the install.sh script for Web API Server for WebSphere.
2. Enter the name of the host on which you are installing Web API Server.
3. If you have not specified configuration information using a Wizard, you are prompted to enter the following login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password
4. Enter the backup Configuration Server host and port.
5. Specify which application to install.
6. Select the code page (the character set pair for processing requests from computers with different locale settings).
7. Enter the fully qualified host name where Co-Browsing Server is installed.
8. Enter the Configuration Server tenant name.
9. Press Enter to confirm the destination directory, and to begin the installation.

End of procedure

Next Steps

- Deploy Genesys web applications for WebSphere. See [Procedure: Deploying Genesys web applications for WebSphere on AIX, Solaris, or Windows](#).

Procedure: Deploying Genesys web applications for WebSphere on AIX, Solaris, or Windows

The following procedures and sample screens apply to WebSphere version 6.1.0.21. For other versions of WebSphere, variations may occur.

Start of procedure

1. Launch your WebSphere Application Server.
2. Login to the Administrative Console (typically using the address `http://<yourhost>:9060/ibm/console`) and proceed to the main page (see [Figure 36](#)).

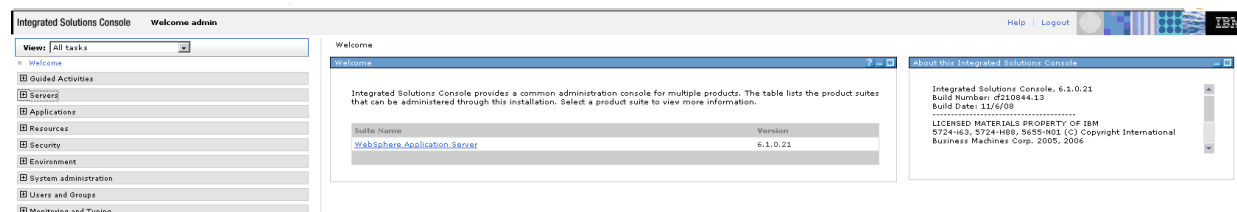


Figure 36: Administration Console—Main Page

3. In the left frame of the main page, click **Applications > Enterprise Applications** (see [Figure 37](#)).

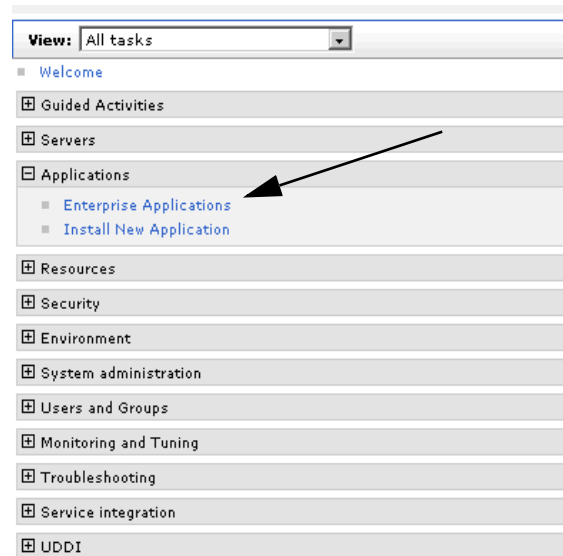


Figure 37: Click Enterprise Applications

4. After the Enterprise Application page opens, in the right frame, select Default Application, and then click Stop (see [Figure 38](#)).

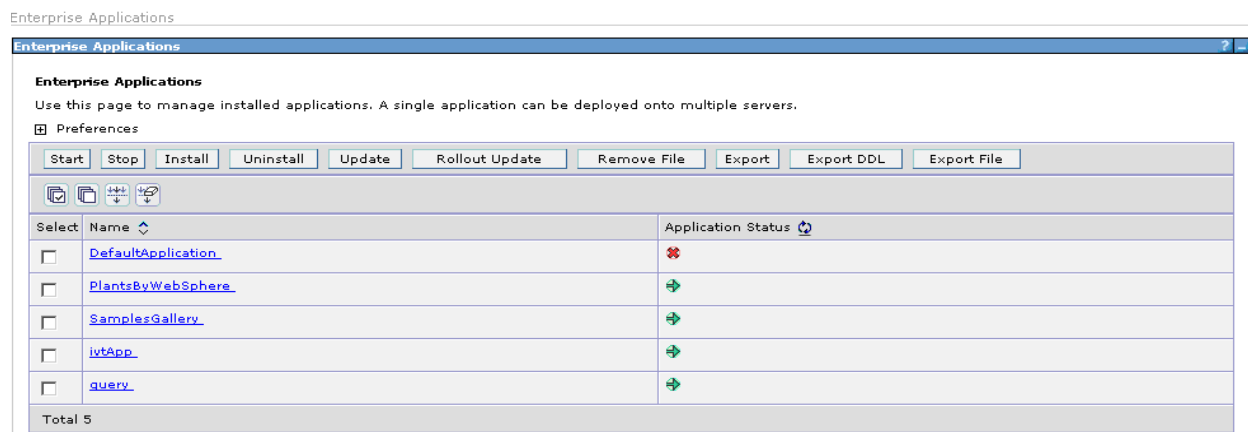


Figure 38: Stop the Default Application

Tip: You must click Stop, because the WebSphere default application shares the same context root as the one you created in your EAR project.

5. After the page refreshes, in the left frame, click Applications > Install New Application (see [Figure 39](#)).

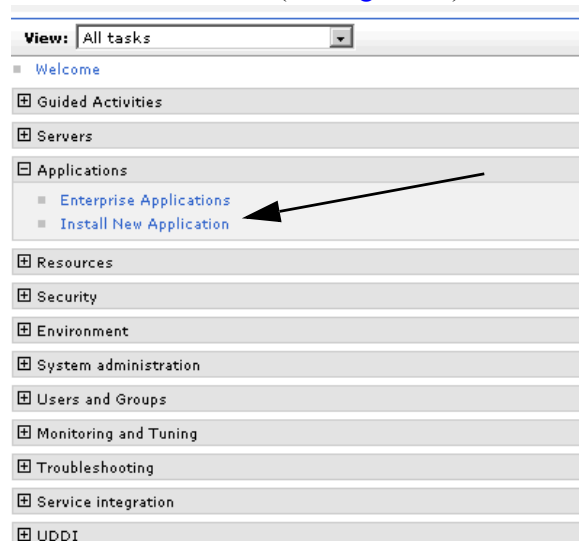


Figure 39: Click Install New Application

6. After the Install New Application page opens, in the right frame, enter the full path to the MCR_API_EAR.ear file that was prepared when you ran the installation package script (see [Figure 40](#)). Click Next.

Figure 40: Enter the Path to the MCR API EAR File

Tip: If you experience difficulties, check access permissions for your UNIX user account as it applies to the .ear file.

- At the Step 1 page (see [Figure 41](#)), retain all defaults, or change the Application Name value (retaining defaults for the remaining fields), and click Next.

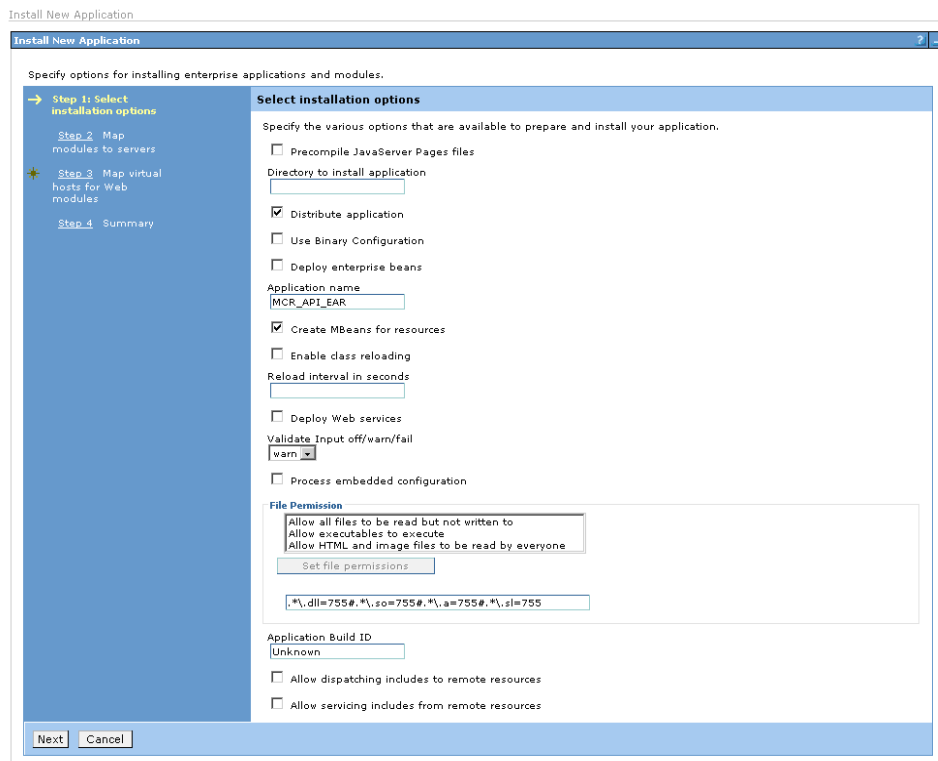


Figure 41: Change Application Name

- At the Step 2 page (see [Figure 42](#)), select your web and application servers from the Clusters and Servers drop-down list, and then click Apply. Click Next.

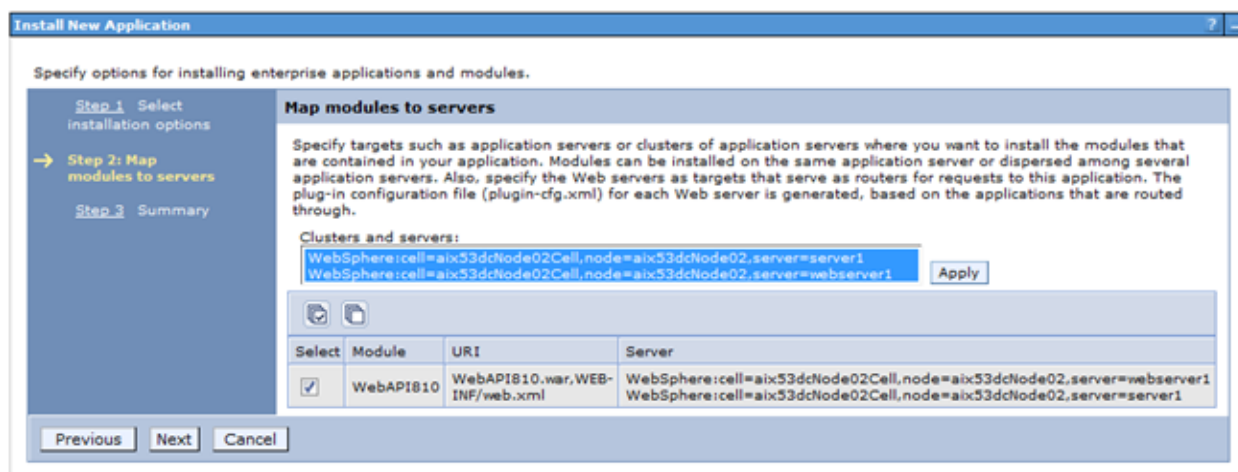


Figure 42: Select Your Web and Application Servers

9. At the Step 3 page (see [Figure 43](#)), review the summary of the your install options. If it is correct, click **Finish**. Otherwise, click **Previous** to return to an earlier page and make any necessary changes.

Install New Application

Specify options for installing enterprise applications and modules.

[Step 1: Select installation options](#)
[Step 2: Map modules to servers](#)
[Step 3: Map virtual hosts for Web modules](#)
→ Step 4: Summary

Summary	
Summary of installation options	
Options	Values
Precompile JavaServer Pages files	No
Directory to install application	
Distribute application	Yes
Use Binary Configuration	No
Deploy enterprise beans	No
Application name	MCR_API_EAR
Create MBeans for resources	Yes
Enable class reloading	No
Reload interval in seconds	
Deploy Web services	No
Validate Input off/warn/fail	warn
Process embedded configuration	No
File Permission	.*\,dl=755#.*\,so=755#.*\,a=755#.*\,sl=755
Application Build ID	Unknown
Allow dispatching includes to remote resources	No
Allow servicing includes from remote resources	No
Cell/Node/Server	Click here

[Previous](#)
[Finish](#)
[Cancel](#)

Figure 43: Review the Summary

After a few moments, you will receive a message notifying you that the installation was successful (see [Figure 44](#)).

Installing...

If there are enterprise beans in the application, the EJB deployment process can take several minutes. Please do not save the configuration until the process completes.

Check the SystemOut.log on the Deployment Manager or server where the application is deployed for specific information about the EJB deployment process as it occurs.

ADMA5016I: Installation of MCR_API_EAR started.

ADMA5067I: Resource validation for application MCR_API_EAR completed successfully.

ADMA5058I: Application and module versions are validated with versions of deployment targets.

ADMA5005I: The application MCR_API_EAR is configured in the WebSphere Application Server repository.

ADMA5053I: The library references for the installed optional package are created.

ADMA5005I: The application MCR_API_EAR is configured in the WebSphere Application Server repository.

ADMA5001I: The application binaries are saved in /data/VS61/AppServer/profiles/AppSrv01/vstemp/62668751/workspace/cells/sol01Qq64Node01/Cell/applications/MCR_API_EAR/MCR_API_EAR.ear

ADMA5005I: The application MCR_API_EAR is configured in the WebSphere Application Server repository.

SECJ0400I: Successfully updated the application MCR_API_EAR with the appContextIDForSecurity information.

ADMA5011I: The cleanup of the temp directory for application MCR_API_EAR is complete.

ADMA5013I: Application MCR_API_EAR installed successfully.

Application MCR_API_EAR installed successfully.

To start the application, first save changes to the master configuration.

Changes have been made to your local configuration. You can:

- [Save](#) directly to the master configuration.
- [Review](#) changes before saving or discarding.

To work with installed applications, click the "Manage Applications" button.

[Manage Applications](#)

Figure 44: Notice of Successful Installation

10. Click the **Save to Master Configuration** link in the middle of the right frame.

11. If you are using the IBM HTTP Server to provide frontline access to WebSphere, you must also do the following:
 - a. In the left frame, click Servers > Web Servers (see Figure 45).

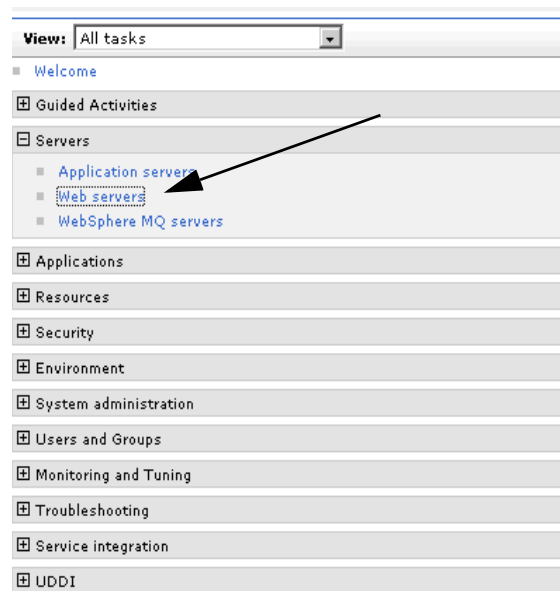


Figure 45: Click Web Servers

- b. In the right frame select your web server. Click Generate Plug-in. You will see the Update web server plug-in configuration confirmation page (as shown in Figure 46).

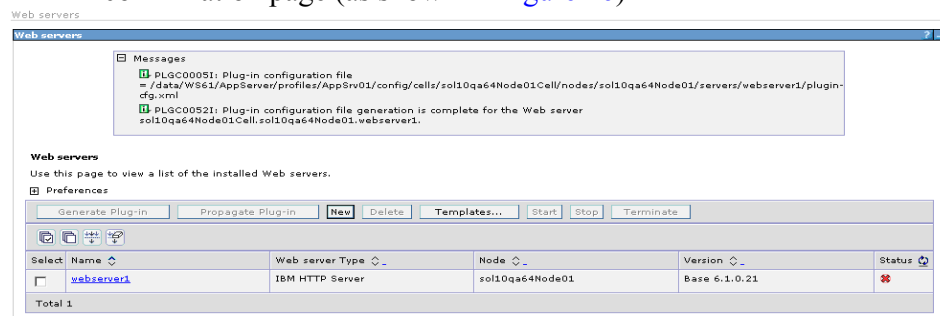
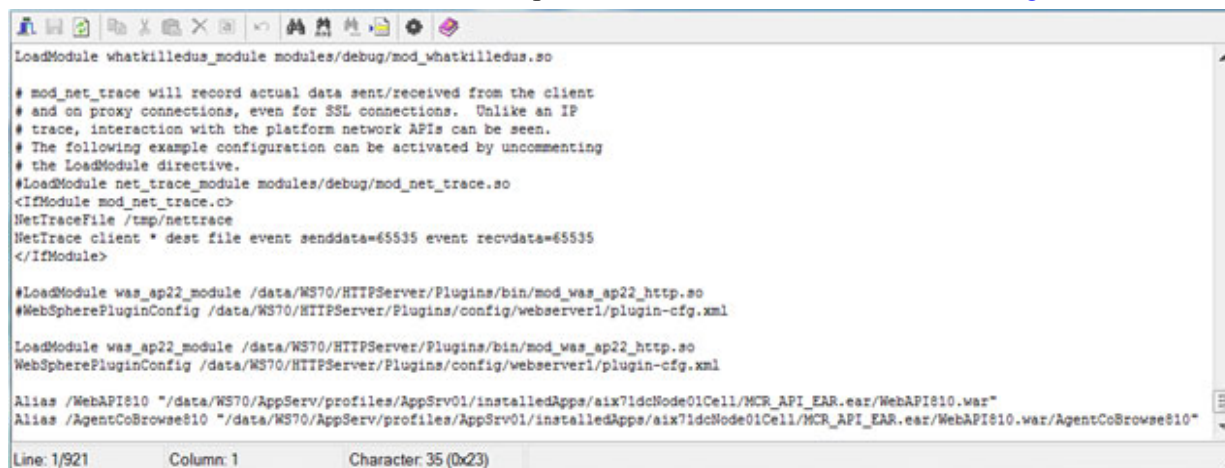


Figure 46: Update Web Server Plug-in Confirmation Page

- c. Update the web server configuration file to accommodate the specifics of your deployment. To do this, locate the <IHS_HOME>/conf/httpd.conf file and add following text (one for each web project, with your environment's specifics), indicating a virtual root, to the end of it:

```
Alias /AgentCoBrowse810 "<MCR_API_EAR>/WebAPI810.war/AgentCoBrowse810"
Alias /WebAPI810 "<MCR_API_EAR>/WebAPI810.war"
```

For an example of an amended http.conf file, see [Figure 47](#):



```
LoadModule whatkilledus_module modules/debug/mod_whatkilledus.so

# mod_net_trace will record actual data sent/received from the client
# and on proxy connections, even for SSL connections. Unlike an IP
# trace, interaction with the platform network APIs can be seen.
# The following example configuration can be activated by uncommenting
# the LoadModule directive.
#LoadModule net_trace_module modules/debug/mod_net_trace.so
<IfModule mod_net_trace.c>
NetTraceFile /tmp/nettrace
NetTrace client * dest file event senddata=65535 event recvddata=65535
</IfModule>

#LoadModule was_ap22_module /data/WS70/HTTPServer/Plugins/bin/mod_was_ap22_http.so
#WebSpherePluginConfig /data/WS70/HTTPServer/Plugins/config/webserver1/plugin-cfg.xml

LoadModule was_ap22_module /data/WS70/HTTPServer/Plugins/bin/mod_was_ap22_http.so
WebSpherePluginConfig /data/WS70/HTTPServer/Plugins/config/webserver1/plugin-cfg.xml

Alias /WebAPI810 "/data/WS70/AppSrv/profiles/AppSrv01/installedApps/aix71dcNode01Cell/MCR_API_EAR.ear/WebAPI810.war"
Alias /AgentCoBrowse810 "/data/WS70/AppSrv/profiles/AppSrv01/installedApps/aix71dcNode01Cell/MCR_API_EAR.ear/WebAPI810.war/AgentCoBrowse810"
```

Figure 47: Sample http.conf File

- d. Save the http.conf file.
 - e. Restart the IHS Web Server.
12. If needed, create a special text file and update the web.xml configuration file to read parameters from this special file. See [Procedure: Specifying properties in a special file for passing initialization parameters to the Load Balancer Servlet](#), on page 142 for details.
 13. Specify system variables for passing initialization parameters to the Load Balancer Servlet. Do this if you want Web API Server to use system variables instead of, or in addition to, the parameters that are specified in the web.xml configuration file. See [Procedure: Specifying system variables for passing initialization parameters to the Load Balancer Servlet](#), on page 145 for details.
 14. Back in WebSphere, in the left frame, go to Applications > Enterprise Applications.

15. At the Enterprise Applications page, in the right frame, select your Web API Server EAR application and click Start.

The status arrow next to your EAR application should change to green as shown in [Figure 48](#).

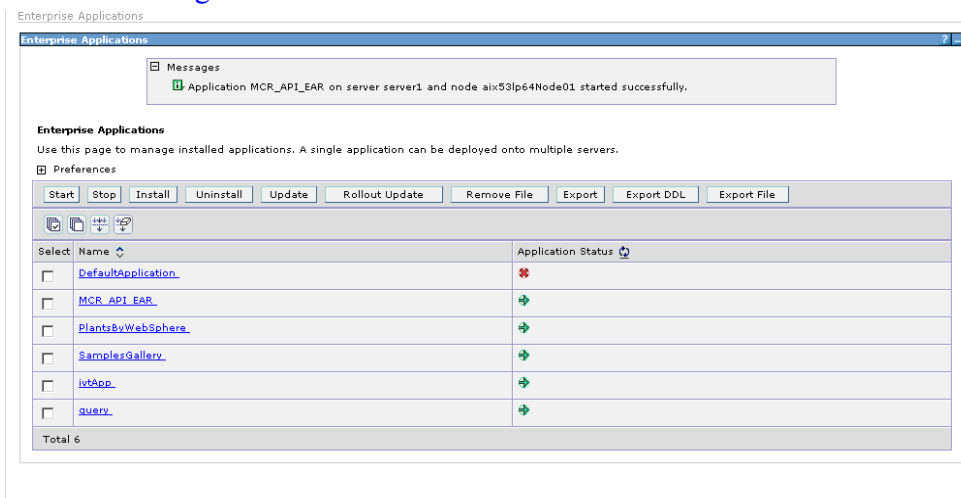


Figure 48: Start Your EAR Application

End of procedure

Next Steps

- Test the WebSphere installation. See [Procedure: Testing the WebSphere installation on AIX, Solaris, or Windows](#).

Procedure: Testing the WebSphere installation on AIX, Solaris, or Windows

Purpose: Use the Genesys MCR 8.1 Platform SDK Java Samples Pages to test your IBM WebSphere deployment.

Start of procedure

1. Launch the IBM HTTP Server.

2. Enter the following address in your browser:

`http://<yourhost>/WebAPI810/`

The Multimedia 8.1 Platform SDK Java Start page opens (see [Figure 49](#)).



Figure 49: Multimedia Platform SDK Java Start Page

3. Select the Samples based on Java API link. The samples page opens (see Figure 50). If you can access and work with those sites, you successfully performed the installation.

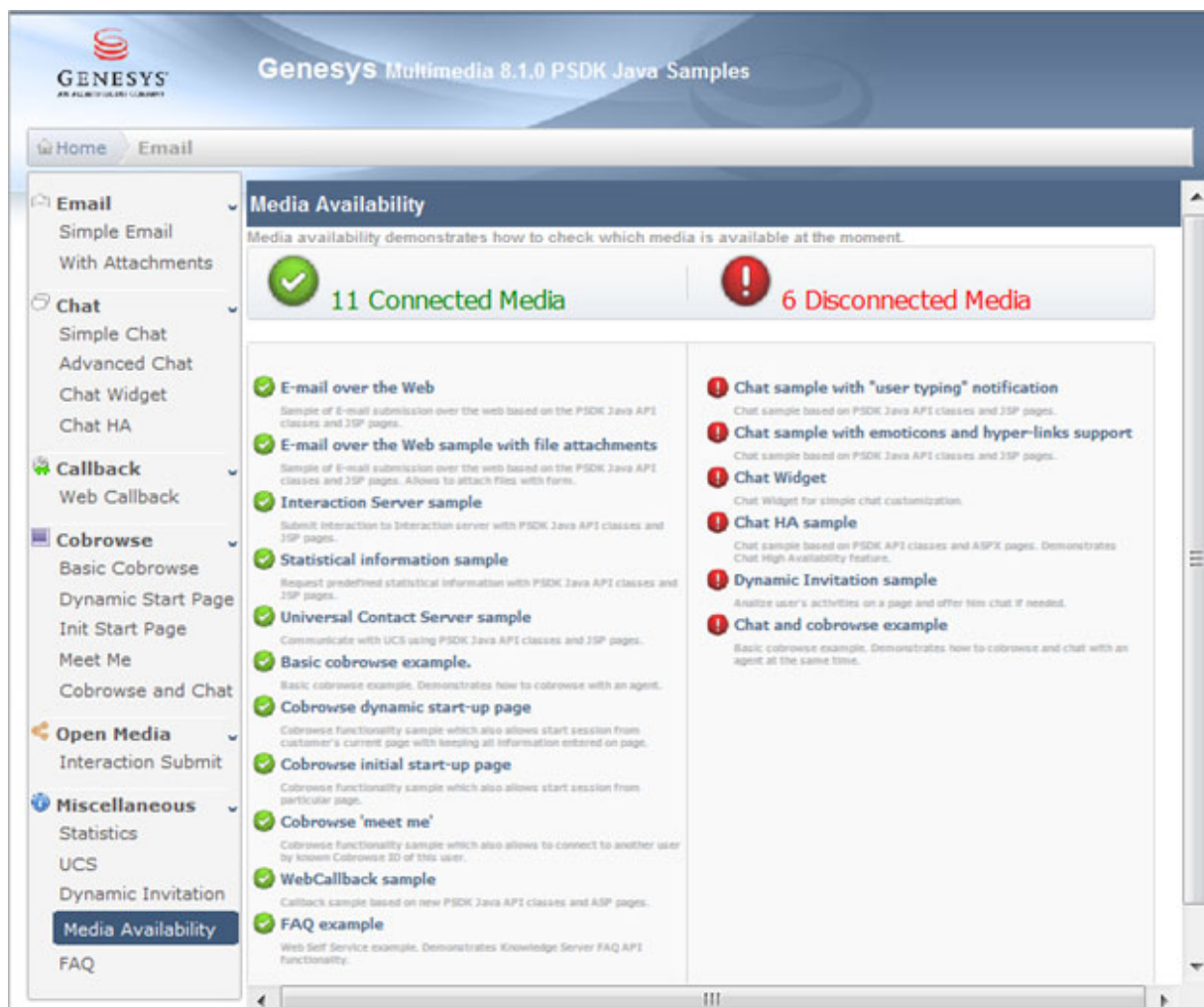


Figure 50: eServices Platform SDK Java Samples

End of procedure

Next Steps

- You have completed all the actions required to deploy Genesys web applications for WebSphere for AIX, Solaris, or Windows.

Deploying Genesys Web Applications for WebLogic on Solaris

For deployment on Solaris, after configuring and installing eServices as outlined in Chapter 5, beginning on [page 97](#), and “Deploying eServices on

UNIX Hosts” on [page 185](#), you need to enable a web application to use Web API Server and the Multimedia 8.1 Samples. You can deploy any of the following web applications on Solaris hosts:

- BEA WebLogic Server—See this section for deployment details.
- IBM WebSphere Application Server—See “Deploying Genesys Web Applications for WebSphere on AIX, Solaris, or Windows” on [page 190](#) for details.
- Apache HTTP Server and Apache Tomcat Application Container—See “Configuring Apache and Tomcat for Linux or Solaris” on [page 76](#) for details.

Prerequisites

This section assumes that you have your WebLogic environment installed and ready, and that you have completed the model setup. The steps outlined in this section use values you entered for that model installation.

Procedure: Installing Genesys web applications for WebLogic

Prerequisites

- The environment variable `GES_HOME_810` is set. See [page 187](#).

Start of procedure

1. Run the `install.sh` script.
2. Enter the host name where you will install the Web API Server.
3. If you have not specified configuration information using a Wizard, you are prompted to enter the login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password
4. Enter the backup Configuration Server host and port.
5. Specify which application to install.
6. Choose the web server type, and then enter the web server root directory.
7. Enter the WebLogic domain directory that you are using.
8. Select the code page·character set pair for processing requests from computers with different locale settings.
9. Enter the fully qualified host name where Co-Browsing Server is installed.
10. Enter the Configuration Server tenant name.

11. Press Enter to confirm the destination directory, and to begin the installation.

The installer will extract the necessary files to the locations you specified, and will update the following files:

- <Apache root>/conf/httpd.conf.
- All related *.jsp files.
- startWebLogic.sh.

End of procedure

Next Steps

- Verify the WebLogic configuration file is correct. See [Procedure: Checking your WebLogic configuration file](#).

Procedure: Checking your WebLogic configuration file

Purpose: To confirm that your eServices web application will run correctly. WebLogic automatically generates a config.xml configuration file during the first run of the domain created in DEVELOPMENT mode. This configuration file specifies which Web API applications WebLogic starts.

Start of procedure

1. Locate and open your WebLogic config.xml file.
2. Search for the following lines inside your configuration file:


```
<app-deployment>
  <name>_appsdir_MCR_API_EAR_ear</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>
  <source-path>autodeploy\MCR_API_EAR.ear</source-path>
  <security-dd-model>DDOnly</security-dd-model>
  <staging-mode>nostage</staging-mode>
</app-deployment>
```

End of procedure

Next Steps

- [Procedure: Deploying your web applications](#), on page 204

Procedure: Deploying your web applications

Purpose: Genesys web applications for WebLogic are deployed automatically during the installation process. However, if you choose to create a custom package based on the Web API Server, then you will have to use WebLogic to manually deploy the related web applications.

Start of procedure

1. Start your Admin and Managed servers, if they are not already running.
2. Open your web browser and start the Administrative console.
3. Log in using your user name and password.
4. Open the Deployments section for your domain, and click the Lock & Edit button as shown in Figure 51 on page 204.

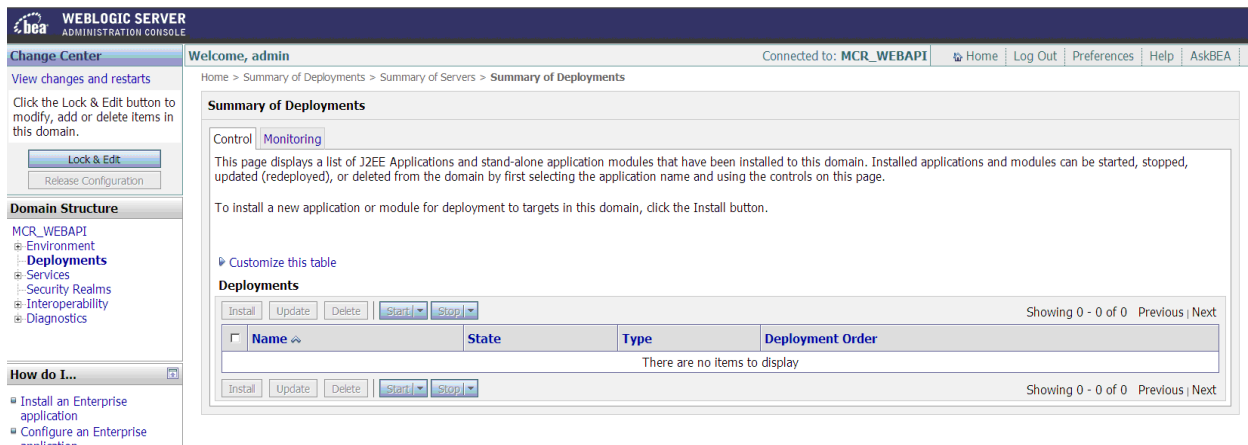


Figure 51: Deploy a Web Application in WebLogic

5. Click the Install button. See Figure 52.

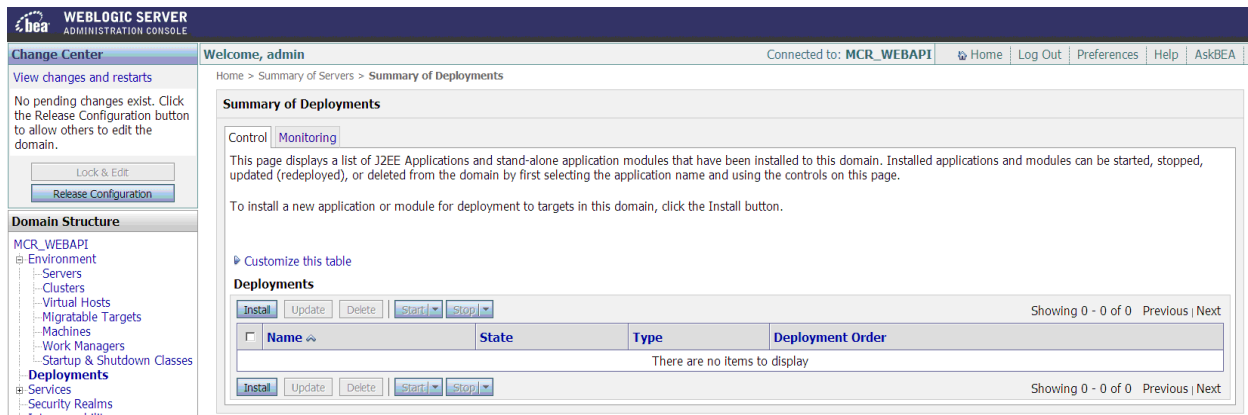


Figure 52: Click Install

- Specify the full path to your MCR_API_EAR.ear application archive and click Next. See [Figure 53](#).

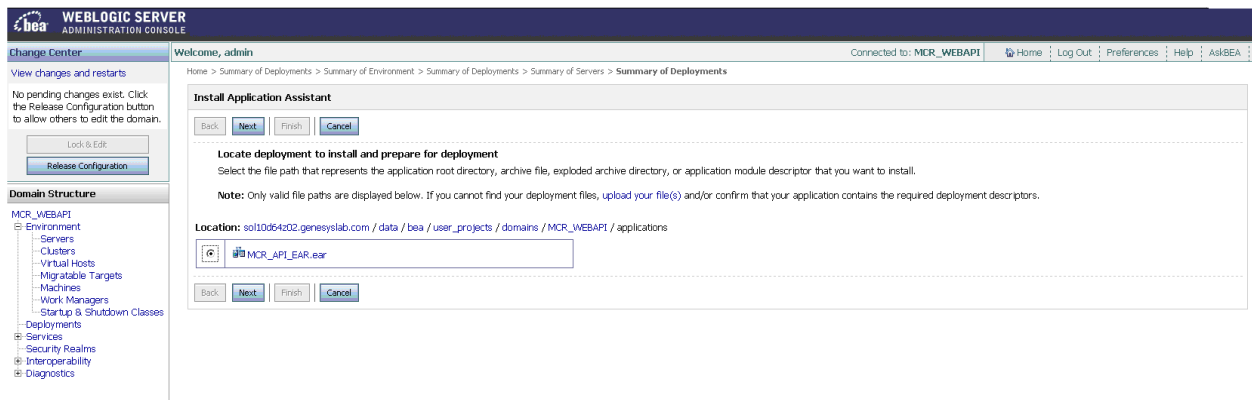


Figure 53: Specify the Path

- Select **Install** this deployment as an application and click Next. See [Figure 54](#).

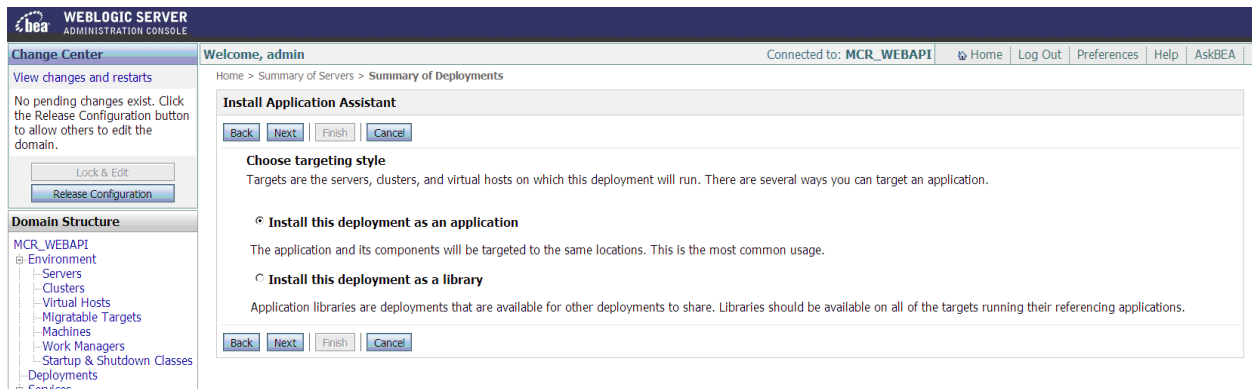


Figure 54: Install this Deployment as an Application

- Select your Managed Servers and click Next. See [Figure 55](#).

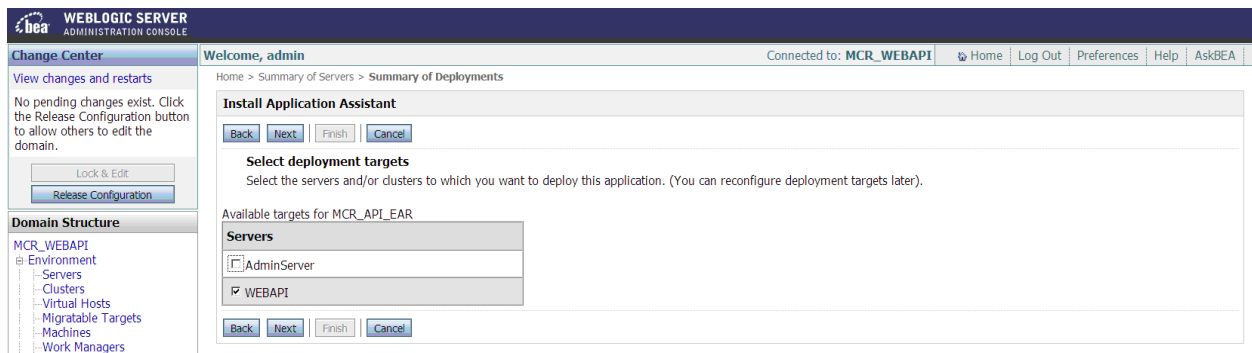


Figure 55: Select your Managed Servers

9. Modify optional settings or accept the defaults. Click Next. See [Figure 56](#).

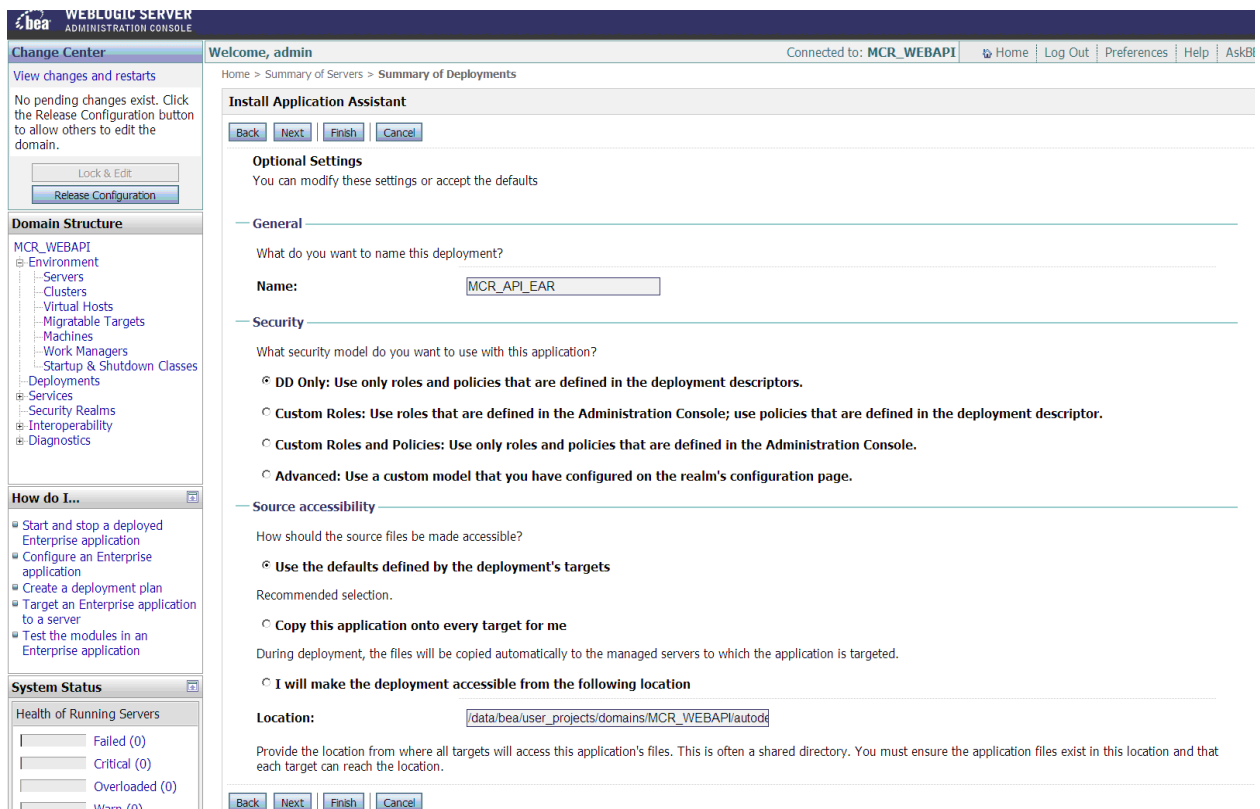


Figure 56: Modify Settings

10. Review your choices and click Finish to complete the deployment. See [Figure 57](#) on [page 207](#).

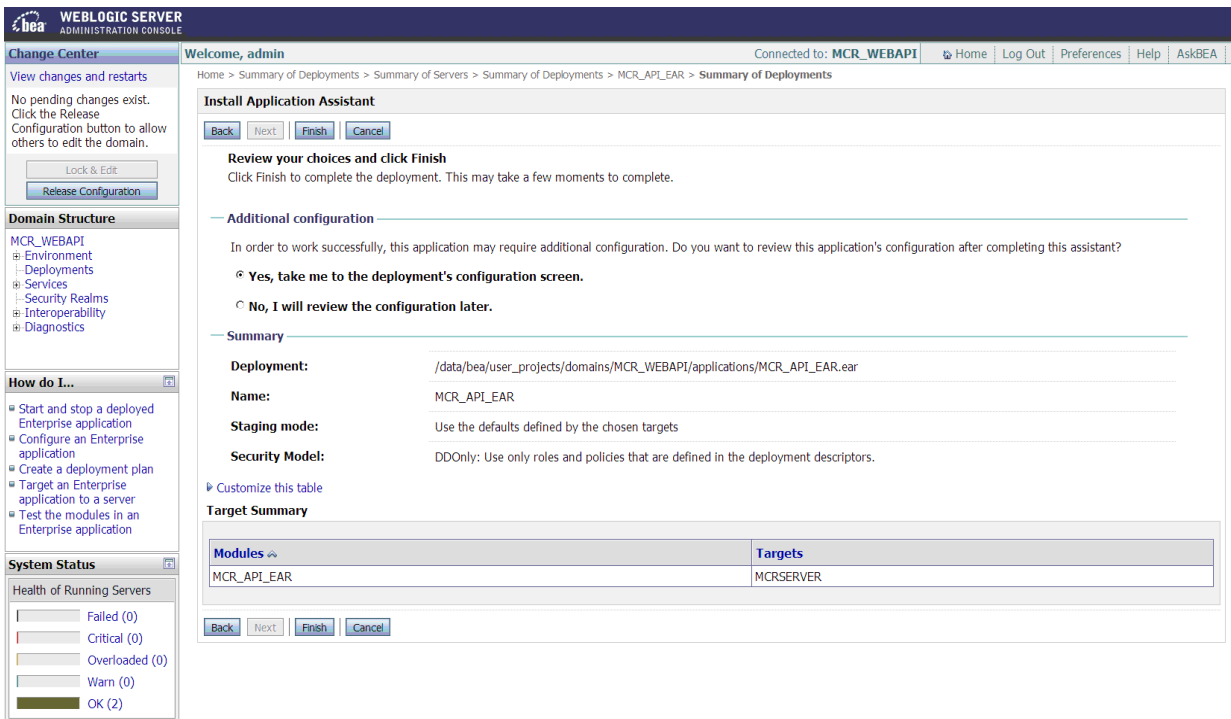


Figure 57: Review Choices

11. Click the **Save** and then click the **Activate Changes** button. See Figure 58.

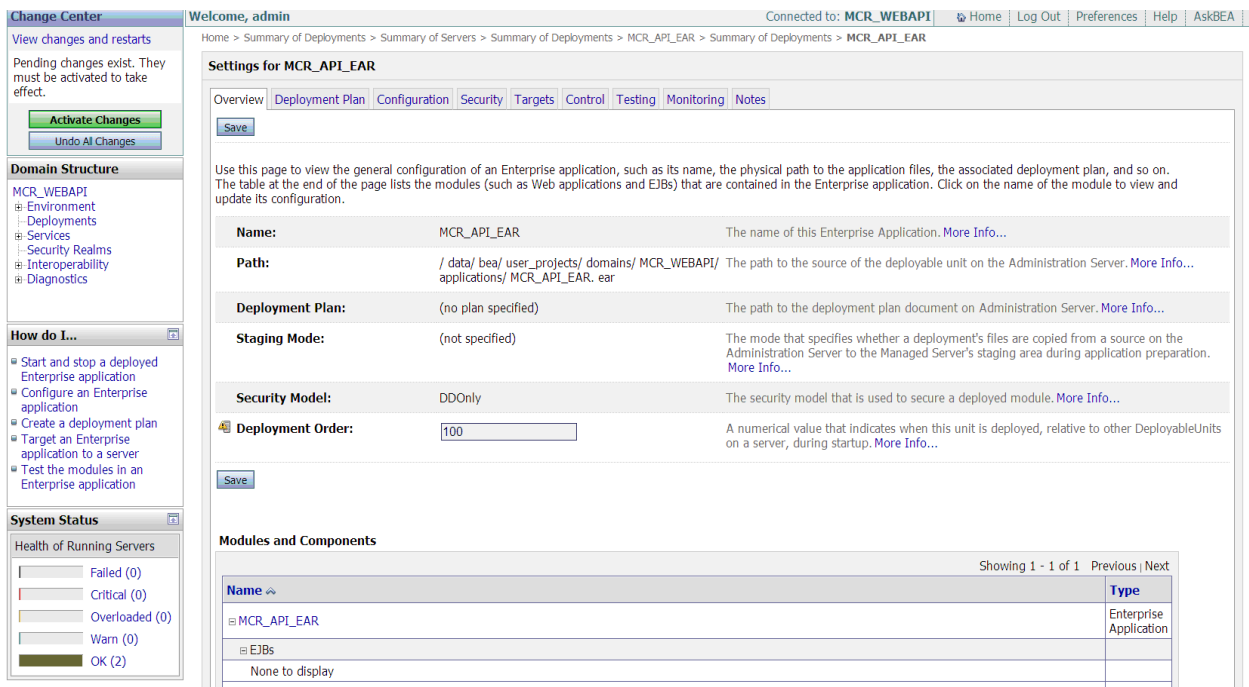


Figure 58: Activate Changes

12. The deployment is now complete. Check your summary page to ensure that the WebAPI810 module is contained in the Enterprise application, as shown in [Figure 59](#).

The screenshot shows the WebLogic Administration Console interface. The left sidebar contains navigation links for Change Center, Domain Structure, How do I..., and System Status. The main content area is titled 'Welcome, admin' and shows the 'Settings for MCR_API_EAR' page. The 'Messages' section indicates that all changes have been activated. The 'Settings' section includes tabs for Overview, Deployment Plan, Configuration, Security, Targets, Control, Testing, Monitoring, and Notes. The 'Overview' tab is selected, showing a table of settings for the MCR_API_EAR application. Below the settings, the 'Modules and Components' section displays a list of modules, including the MCR_API_EAR Enterprise Application and the /WebAPI810 Web Application.

Name	Value	Description
Name:	MCR_API_EAR	The name of this Enterprise Application. More Info...
Path:	/ data/ bea/ user_projects/ domains/ MCR_WEBAPI/ applications/ MCR_API_EAR. ear	The path to the source of the deployable unit on the Administration Server. More Info...
Deployment Plan:	(no plan specified)	The path to the deployment plan document on Administration Server. More Info...
Staging Mode:	(not specified)	The mode that specifies whether a deployment's files are copied from a source on the Administration Server to the Managed Server's staging area during application preparation. More Info...
Security Model:	DDOnly	The security model that is used to secure a deployed module. More Info...
Deployment Order:	100	A numerical value that indicates when this unit is deployed, relative to other DeployableUnits on a server, during startup. More Info...

Name	Type
MCR_API_EAR	Enterprise Application
EJBs	
None to display	
Modules	
/WebAPI810	Web Application

Figure 59: Summary Page

End of procedure

Next Steps

- Test the installation. See [Procedure: Testing the WebLogic installation and deployment](#), on page 208.

Procedure: Testing the WebLogic installation and deployment

Purpose: To verify the WebLogic installation and deployment was successful.

Start of procedure

1. Ensure Apache is running.
2. Start WebLogic.
3. In a browser, enter the following address:
`http://<Apache host>:<Apache Listen port>/WebAPI810`

If you can access and work with this site, you have successfully completed the installation and deployment.

Note: The address listed above is case sensitive.

End of procedure

Next Steps

- You have completed all the actions required to deploy Genesys web applications for WebLogic on Solaris.



Chapter

11

Manual Deployment—Web API Server and UCS

This chapter describes the manual deployment of Web API Server, Universal Contact Server (UCS) and UCS Manager.

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

- [Introduction, page 211](#)
- [Configuring Web API Server and UCS, page 212](#)
- [Installing Web API Server and UCS, page 215](#)

Introduction

It is possible that you may want to use Web API Server or UCS and UCS Manager without deploying the other eServices components. For example, Genesys Voice Callback requires these components. In such a situation you may want to install these components on their own without using the eServices Configuration Wizards or the eServices integrated installation. This chapter describes procedures for such manual configuration and installation.

Note: The installation procedures outlined in this chapter are specific to Apache and Tomcat on Windows and Apache and Tomcat on Solaris or Linux. Other combinations of web servers, application containers, and operating systems are supported, but their installation procedures may vary. Refer to Chapter 10 on [page 185](#) for information about installation on WebLogic and WebSphere.

Requirements

These components require certain other components to operate:

- The Web API Server requires a web server and an application container. See [Chapter 4](#) for a description of installing these objects.
- Prior to the 8.1.3 release, UCS and UCS Manager require Java Environment and Libraries for eServices and UCS installed on the host. Java Environment and Libraries for eServices and UCS are included in the integrated install, as described in [Procedure: Performing an integrated installation of all eServices servers](#), on [page 138](#). Java Environment and Libraries for eServices and UCS can also be installed manually, as described in [Procedure: Installing Universal Contact Server Manager](#) (see [Step 1](#) on [page 220](#)).

Note: For an explanation of the difference between Third Party Components (executable) and ThirdPartyApplications (directory), see the note on [page 56](#).

Configuring Web API Server and UCS

In the Configuration Layer, create or edit the required Application objects. Refer to the following procedures:

1. [Procedure: Creating or editing the Web API Server application object](#), on [page 212](#)
2. [Procedure: Creating or editing the UCS application object](#), on [page 213](#)
3. [Procedure: Creating or editing the UCS Manager application object](#), on [page 214](#)

Procedure: Creating or editing the Web API Server application object

Purpose: To create or update the Application object for Web API Server.

Start of procedure

1. Create an Application object if it does not already exist.
 - a. Import the Web API Server application template from the Interaction Management CD.
 - b. Create a new Application object based on the template.
2. Open the Properties dialog box of the Application object.

3. On the **Server Info** tab:
 - In the **Host** box, enter the name of the desired host.
 - In the **Communication Port** box, enter the number of the web server HTTP port.
4. On the **Start Info** tab, enter the working directory.
5. On the **Connections** tab, add connections to:
 - Solution Control Server (mandatory)
 - Universal Callback Server
 - Chat Server
 - E-mail Server
 - Interaction Server
 - Universal Contact Server
 - Stat Server
 - Configuration Server, if you want support for Genesys Advanced Disconnect Detection Protocol (ADDP)
 - Co-Browsing Server

Only the connection to Solution Control Server is mandatory. Make additional connections based on your planned configuration.

End of procedure**Next Steps**

- Configure the Universal Contact Server Application object. See [Procedure: Creating or editing the UCS application object](#).

Procedure:
Creating or editing the UCS application object

Purpose: To configure the Universal Contact Server Application object.

Prerequisites

- UCS requires one database (the main database). It can optionally use a second (archive) database to store older UCS data and ease storage in the main database.

Start of procedure

1. Create the main and (optional) archive databases if you have not already done so. See also “Creating Databases and Running Scripts” on [page 152](#).
2. Run the SQL script located in the `sql_scripts` folder of the UCS starting directory.

3. Create a Database Access Point (DAP) for the main database. Create a second DAP for the archive database if you have one. Use the JDBC connection type.
4. Create a UCS Application object if it does not already exist.
 - a. Import the UCS application template from the Interaction Management CD.
 - b. Create a new Application object based on the template.
5. Open the Properties dialog box of the Application object.
6. On the Server Info tab, enter the host name and communication port.
7. On the Start Info tab, enter the working directory.
8. On the Connections tab, add connections to:
 - Message Server
 - Stat Server
 - The two DAPs that you created in [Step 3](#).Click Apply.
9. On the Security tab, in the Log On As area, select the This Account check box, and then select a Person or Access Group with privileges that are high enough to include write permission.
10. On the Options tab, ports section, specify for the option ucsapi a valid network port number that UCS will use for connections from Java clients.

End of procedure

Next Steps

- Configure the UCS Manager Application object. See [Procedure: Creating or editing the UCS Manager application object](#).

Procedure: Creating or editing the UCS Manager application object

Purpose: To create and configure the UCS Manager Application object.

Start of procedure

1. Create a UCS Manager Application object if it does not exist.
 - a. Import the UCS Manager application template from the Interaction Management CD.
 - b. Create a new Application object based on the template.

2. On the **Connections** tab, add a connection to UCS and click **Apply**.

End of procedure

Next Steps

- [Installing Web API Server and UCS, page 215](#)

Installing Web API Server and UCS

Complete the following procedures to install Web API Server and UCS:

1. [Procedure: Installing Web API Server on Windows, on page 215](#) or [Procedure: Installing Web API Server on Solaris or Linux, on page 217](#).
2. [Procedure: Installing Universal Contact Server on Windows, on page 218](#) or [Procedure: Installing Universal Contact Server on AIX, Solaris, or Linux, on page 219](#).
3. [Procedure: Installing Universal Contact Server Manager, on page 220](#).

Procedure: Installing Web API Server on Windows

Purpose: To manually install Web API Server on a Windows host.

Prerequisites

- The Web API Server Application object exists in Configuration Manager. See [Procedure: Creating or editing the Web API Server application object, on page 212](#).
- A web server and an application container are installed on the host. See [Procedure: Installing the Apache Web Server on Windows, on page 58](#) and [Procedure: Installing the Tomcat web application container on Windows, on page 60](#).
- The Interaction Management CD.

Start of procedure

1. Locate the **Setup.exe** for Web API Server on the Interaction Management CD.
2. Double-click **Setup.exe**.
3. Select the web server. Click **Next**.
4. Browse to the web server location. Click **Next**.
5. Browse to the web application container location. Click **Next**.

6. Enter the login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password
7. If necessary, specify the solution destination directory. Click **Next**.
8. Enter the backup Configuration Server Host and Port. Click **Next**.
9. Select the tenant ID. Click **Next**.
10. Specify the Co-Browsing Server name and location of the `worker.properties` file you created during installation of the web application container (<CATALINA_HOME>\conf\jk directory). Click **Next**.
11. Select the `ajp13` worker. Click **Next**.
12. Select a server-side code page (the preferred language for processing all requests to Web API Server). Click **Next**.
13. Specify the Agent Place ID that will be used in the Web Callback sample.
14. Click **Install** if you are ready for installation.
15. Click **Finish**.

End of procedure

Next Steps

- If needed, create a special text file and update the `web.xml` configuration file to read parameters from this special file. See [Procedure: Specifying properties in a special file for passing initialization parameters to the Load Balancer Servlet](#), on page 142 for details.
- Specify system variables for passing initialization parameters to the Load Balancer Servlet. Do this if you want Web API Server to use system variables instead of, or in addition to, the parameters that are specified in the `web.xml` configuration file. See [Procedure: Specifying system variables for passing initialization parameters to the Load Balancer Servlet](#), on page 145 for details.
- Install Universal Contact Server. See [Procedure: Installing Universal Contact Server on Windows](#), on page 218 or [Procedure: Installing Universal Contact Server on AIX, Solaris, or Linux](#), on page 219.
- Install Universal Contact Server Manager. See [Procedure: Installing Universal Contact Server Manager](#), on page 220.

Procedure: Installing Web API Server on Solaris or Linux

Purpose: To manually install Web API Server on a Solaris or Linux host.

Note: This installation procedure is specific for Apache and Tomcat on Solaris or Linux. Other combinations of web servers, application containers, and operating systems are supported, but their installation procedures may vary. Refer to Chapter 10 on [page 185](#) for information about installation on WebLogic and WebSphere.

Prerequisites

- The Web API Server Application object exists in Configuration Manager. See [Procedure: Creating or editing the Web API Server application object](#), on [page 212](#).
- A web server and an application container are installed on the host. See “Configuring Apache and Tomcat for Linux or Solaris” on [page 76](#).
- The Interaction Management CD.
- The environment variable `GES_HOME_810` is set. See [page 187](#).

Start of procedure

1. Run the `install.sh` script for Web API Server.
2. Enter the name of the host on which you are installing Web API Server.
3. If you have not specified configuration information using a Wizard, you are prompted to enter the following login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password
4. Enter the backup Configuration Server Host and Port.
5. Select the web server.
6. If necessary, browse to the web server location.
7. Select the web application container.
8. If necessary, browse to the web application container location.
9. Select the code page (the character set pair for processing requests from computers with different locale settings).
10. Specify the location of the `worker.properties` file you created during installation of the web application container (`<CATALINA_HOME>\conf\jk` directory).

11. Select the `ajp13` worker.
12. Enter the fully qualified host name where Co-Browsing Server is installed.
13. Enter the Configuration Server tenant name.
14. Specify the Agent Place ID that will be used in the Web Callback sample.
15. Press Enter to confirm the destination directory, and to begin the installation.

End of procedure

Next Steps

- If needed, create a special text file and update the `web.xml` configuration file to read parameters from this special file. See [Procedure: Specifying properties in a special file for passing initialization parameters to the Load Balancer Servlet](#), on page 142 for details.
- Specify system variables for passing initialization parameters to the Load Balancer Servlet. Do this if you want Web API Server to use system variables instead of, or in addition to, the parameters that are specified in the `web.xml` configuration file. See [Procedure: Specifying system variables for passing initialization parameters to the Load Balancer Servlet](#), on page 145 for details.
- Install Universal Contact Server. See [Procedure: Installing Universal Contact Server on Windows](#), on page 218 or [Procedure: Installing Universal Contact Server on AIX, Solaris, or Linux](#), on page 219.
- Install Universal Contact Server Manager. See [Procedure: Installing Universal Contact Server Manager](#), on page 220.

Procedure: Installing Universal Contact Server on Windows

Prerequisites

- The UCS Application object has been configured. See [Procedure: Creating or editing the UCS application object](#), on page 213.
- (For releases prior to 8.1.1 only:) Java Environment and Libraries for eServices and UCS are installed on the host machine. Java Environment and Libraries for eServices and UCS are included in the integrated install, as described in [Procedure: Performing an integrated installation of all eServices servers](#), on page 138. They can also be installed manually from the Interaction Management CD, as described in [Procedure: Installing Universal Contact Server Manager](#) (see [Step 1](#) on page 220).
- The Interaction Management CD.

Start of procedure

1. Locate the `Setup.exe` for Universal Contact Server (available on the Interaction Management CD).
2. Double-click `Setup.exe`.
3. Enter the login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password
4. Select the appropriate UCS Application object from the list.
5. Click `Install`.

End of procedure**Next Steps**

- Install Universal Contact Server Manager. See [Procedure: Installing Universal Contact Server Manager](#), on [page 220](#).

Procedure:
Installing Universal Contact Server on AIX, Solaris, or Linux**Prerequisites**

- The Universal Contact Server Application object has been configured. See [Procedure: Creating or editing the UCS application object](#), on [page 213](#).
- (For releases prior to 8.1.1 only:) Java Environment and Libraries for eServices and UCS are installed on the host machine. Java Environment and Libraries for eServices and UCS are included in the integrated install, as described in [Procedure: Performing an integrated installation of all eServices servers](#), on [page 138](#). eServices Third Party Components can also be installed manually from the Interaction CD.
- The Interaction Management CD.
- (For releases prior to 8.1.1 only:) The environment variable `GES_HOME_810` is set. See [page 187](#).

Start of procedure

1. Locate the `install.sh` file for Universal Contact Server.
2. Start the installation script with the command `>install.sh`
3. For UCS 8.1.1 and later, provide the absolute path to the directory containing the Java executable.

4. Press Enter to confirm the host name for the installation.
5. Enter the login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password
6. From the list of applications, select one and enter its number in the list.
7. Press Enter to confirm the suggested destination directory, or choose another one inside the directory referred to by the `GES_HOME_810` variable.
8. Answer other questions, if required.

End of procedure

Next Steps

- Install Universal Contact Server Manager. See [Procedure: Installing Universal Contact Server Manager](#).

Procedure: Installing Universal Contact Server Manager

Purpose: To install an instance of Universal Contact Server Manager. Instances of UCS Manager can be installed on multiple hosts.

Prerequisites

- The Universal Contact Server Manager Application object exists in Configuration Manager. See [Procedure: Creating or editing the UCS Manager application object](#), on page 214.
- The Interaction Management CD.

Start of procedure

1. Install the Java Environment and Libraries for eServices and UCS on the host.
 - a. Locate and double-click `Setup.exe`.
 - b. Enter or browse to the location of the solution destination folder. Click Next.

c. Click **Install**.

Note: This step can be omitted if either of the following is true:

- Java Environment and Libraries for eServices and UCS were already installed on the host as part of an integrated install. See [Procedure: Performing an integrated installation of all eServices servers](#), on [page 138](#) for more information.
 - You are using UCS Manager 8.1.1 or later.
-

2. Locate and double-click `Setup.exe` for UCS Manager on the Interaction Management CD.
3. Click **Next**.
4. For UCS Manager 8.1.1 and later:
 - a. Enter or browse to the location of the destination folder. Click **Next**.
 - b. Select JDK version in the list. Click **Next**.
5. Click **Install**.
6. Click **Finish**.

End of procedure

Next Steps

- You have completed the steps to manually install Web API Server and UCS.



Chapter

12

Manual Deployment—UCS Proxy, Interaction Server Proxy, and SMS Server

This chapter describes the deployment of eServices components that are not included in the eServices Configuration Wizards. These components must be installed manually.

Universal Contact Server Proxy and Interaction Server Proxy were introduced in release 7.6.1 and SMS Server was new in release 8.0.0.

This chapter includes these sections:

- [Deploying UCS Proxy or Interaction Server Proxy, page 223](#)
- [Deploying SMS Server, page 228](#)

Deploying UCS Proxy or Interaction Server Proxy

Large numbers of custom desktop (ESP client) connections to Interaction

Notes: UCS Proxy and Interaction Server Proxy support High Availability in Warm Standby mode.

UCS Proxy and Interaction Server Proxy are **not** supported with Genesys Agent Desktop and desktops that connect using Interaction SDK. A custom desktop that connects using Platform SDK Contact can use UCS Proxy and Interaction Server Proxy.

Server and UCS may give rise to performance issues. To mitigate the issues caused by a high load on the server, Genesys introduced Interaction Server Proxy and UCS Proxy in release 7.6.1. Desktop applications can be configured

to connect to these Proxy servers instead of the main server, significantly reducing the load on the server. For example, it is easier for the server to handle 20,000 clients that operate through ten proxies (only ten connections) than to handle the same 20,000 clients that each connect separately.

Notes: These Proxy components are not part of the eServices Configuration Wizards and must be installed manually.

UCS Proxy only supports connections from desktop applications; Interaction Server Proxy supports connections from desktop applications and from Stat Server.

Because of the many variables in deployment (choice of operating system, number of clients, details of architecture) is so broad, it is not possible to provide exact guidelines as to when deploying Proxy servers would be advantageous. However it may be stated that you can anticipate performance issues when the number of clients exceeds 10,000.

Requirements

These components work with Universal Contact Server and Interaction Server. In order for the respective Proxy servers to work properly, all components that are needed by Universal Contact Server and Interaction Server must be installed. This will depend on your environment and how you are using Universal Contact Server and Interaction Server. For more details, refer to the chapter that is applicable to you:

- Chapter 5, “Model Configuration and Installation on Windows,” on [page 97](#) for a typical eServices Solution running on Windows.
- Chapter 10, “Deploying eServices in UNIX Environments,” on [page 185](#) for a sample deployment of an eServices Solution running on UNIX.
- Chapter 11, “Manual Deployment—Web API Server and UCS,” on [page 211](#), for environments that are not using all eServices components, but that do require Universal Contact Server, such as Voice Callback.

Deploying the Proxy servers can be divided into the following procedures:

1. [Procedure: Creating the Proxy application object](#), on [page 225](#)
2. [Procedure: Installing Universal Contact Server Proxy or Interaction Server Proxy on Windows](#), on [page 226](#) or [Procedure: Installing Universal Contact Server Proxy or Interaction Server Proxy on AIX, Solaris, or Linux](#), on [page 227](#).
3. [Procedure: Configuring the desktop application to use UCS Proxy or Interaction Server Proxy](#), on [page 228](#)

Procedure: Creating the Proxy application object

Purpose: Before installing the Proxy component, a corresponding `Application` object must exist in the Configuration Database.

Prerequisites

- A Universal Contact Server `Application` object (for UCS Proxy) or Interaction Server `Application` object (for Interaction Server Proxy) must already exist.
- The Interaction Management CD.

Start of procedure

1. Create an `Application` object for the Proxy if it does not already exist.
 - a. Import the correct application template from the Interaction Management CD for the type of Proxy you are creating.
 - b. Create a new `Application` object based on the template.
2. Open the `Properties` dialog box of the `Application` object.
3. On the `Server Info` tab:
 - In the `Host` box, enter the name of the desired host.
 - In the `Communication Port` box, enter the port the Proxy will use for listening.
4. On the `Start Info` tab enter some characters in the `Working Directory`, `Command Line`, and `Command Line Arguments` fields. These characters will be over-written with the correct values during the installation, but they cannot be left blank at this point.
5. On the `Connections` tab, add connections to:
 - the primary Universal Contact Server (for UCS Proxy) or Interaction Server (for Interaction Server Proxy). This connection is mandatory.
 - Message Server (optional)
6. Click `Apply`.
7. If you would like to configure your Proxy for HA, repeat this procedure for the second Proxy instance.

End of procedure

Next Steps

- Install UCS Proxy. See [Procedure: Installing Universal Contact Server Proxy or Interaction Server Proxy on Windows](#) or [Procedure: Installing Universal Contact Server Proxy or Interaction Server Proxy on AIX, Solaris, or Linux](#), on page 227.

**Procedure:
Installing Universal Contact Server Proxy or
Interaction Server Proxy on Windows****Prerequisites**

- The Proxy Application object has been configured. See [Procedure: Creating the Proxy application object](#), on page 225.
- The Interaction Management CD.

Start of procedure

1. Locate the Setup.exe for the Proxy (available on the Interaction Management CD).
2. Double-click Setup.exe.
3. Enter the login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password
4. Select the appropriate Proxy Application object from the list.
5. Click Install.

End of procedure**Next Steps**

- Update the desktop application to use the Proxy. See [Procedure: Configuring the desktop application to use UCS Proxy or Interaction Server Proxy](#).

Procedure: Installing Universal Contact Server Proxy or Interaction Server Proxy on AIX, Solaris, or Linux

Prerequisites

- The Proxy Application object has been configured. See [Procedure: Creating the Proxy application object](#), on [page 225](#).
- The Interaction Management CD.
- Review the section “Compatibility Package for Red Hat Linux” on [page 189](#).

Start of procedure

1. Locate the `install.sh` file for the Proxy.
2. Start the installation script with the command `>install.sh`
3. Press Enter to confirm the host name for the installation
4. Enter the login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password
5. From the list of applications, select one and enter its number in the list.
6. Press Enter to confirm the suggested destination directory, or choose another one.
7. Answer other questions, if required.

Note: On some Red Hat Linux platforms, eServices 8.1 components might fail to start and produce the error `./cfgutility: error while loading shared libraries: libstdc++.so.5: cannot open shared object file: No such file or directory`. Refer to “Compatibility Package for Red Hat Linux” on [page 189](#) for information about correcting this issue.

End of procedure

Next Steps

- Update the desktop application to use the Proxy. See [Procedure: Configuring the desktop application to use UCS Proxy or Interaction Server Proxy](#).

Procedure: Configuring the desktop application to use UCS Proxy or Interaction Server Proxy

Purpose: In order to take advantage of the benefits provided by UCS Proxy or Interaction Server Proxy, your desktop Application object must be updated to connect to the Proxy instead of the main Universal Contact Server or Interaction Server

Note: UCS Proxy and Interaction Server Proxy are **not** supported with Genesys Agent Desktop and desktops that connect using Interaction SDK. A custom desktop that connects using Platform SDK Contact can use the Proxy servers.

Prerequisites

- Your desktop application is installed and configured.
- The Proxy is installed and configured. See [Procedure: Creating the Proxy application object](#), on page 225 and [Procedure: Installing Universal Contact Server Proxy or Interaction Server Proxy on Windows](#), on page 226.

Start of procedure

1. Login to Configuration Manager.
2. Locate the desktop Application object and open its Properties dialog box.
3. On the Connections tab, remove the connection to the main Universal Contact Server or Interaction Server.
4. Add a connection to the UCS Proxy or Interaction Server Proxy.
5. Click Apply.

End of procedure

Deploying SMS Server

SMS Server was introduced in 8.0.0. This component is not part of the eServices Configuration Wizards and must be installed manually.

Deploying SMS Server can be divided into the following procedures:

1. [Procedure: Creating the SMS Server application object](#), on page 229
2. [Procedure: Installing SMS Server on Windows](#), on page 230 or [Procedure: Installing SMS Server on AIX, Solaris, or Linux](#), on page 231

3. [Procedure: Configuring eServices Components to work with SMS Server, on page 232](#)

Procedure: Creating the SMS Server application object

Purpose: To create and configure the SMS Server Application object in Configuration Manager. This is needed to install SMS Server.

Prerequisites

- The SMS Server CD.
- The following Java versions:
 - In release 8.1.3 and later,
 - Windows: JDK 1.6 or JDK 1.7 for all eServices Java components.
 - UNIX: JRE 1.6 or JRE 1.7.
 - In releases prior to 8.1.3, JDK 1.6 on Windows and JRE 1.6 on UNIX.

Note: In this release, SMS server is supported with Java 1.6.0_26 or higher (within the 1.6 family); in release 8.1.3 and higher SMS Server also supports Java 1.7. The 64-bit version is required for 64-bit Windows.

Start of procedure

1. Create an Application object for SMS Server if it does not already exist.
 - a. Import the SMS Server application template from the CD.
 - b. Create a new Application object based on the template.
2. Open the Properties dialog box of the Application object.
3. On the Server Info tab:
 - In the Host box, enter the name of the desired host.
 - In the Communication Port box, enter the port SMS Server will use.
4. On the Start Info tab enter some characters in the Working Directory, Command Line, and Command Line Arguments fields. These characters will be over-written with the correct values during the installation, but they cannot be left blank at this point.
5. On the Connections tab, add a connection to Interaction Server.
6. If this is for a multi-tenant environment, add the tenant(s) on the Tenants tab.

7. Click **Apply**.

End of procedure

Next Steps

- Install SMS Server. See [Procedure: Installing SMS Server on Windows](#), on page 230 or [Procedure: Installing SMS Server on AIX, Solaris, or Linux](#), on page 231.

Procedure: Installing SMS Server on Windows

Prerequisites

- The SMS Server Application object has been configured. See [Procedure: Creating the SMS Server application object](#), on page 229.
- Java is installed on the host.

Note: In this release, SMS server is supported with Java 1.6.0_26 or higher (within the 1.6 family); in release 8.1.3 and higher SMS Server also supports Java 1.7. The 64-bit version is required for 64-bit Windows.

- The SMS Server CD.

Start of procedure

1. Locate the `Setup.exe` for SMS Server (available on the SMS Server CD).
2. Double-click `Setup.exe`.
3. Select `Java`.
4. Select `Use Client Side Port` if applicable.
5. Enter the login information for your Configuration Server:
 - `Host`
 - `Port`
 - `User`
 - `Password`
6. Select the appropriate SMS Server Application object from the list.
7. Select destination location.
8. Select `JDK`.
9. Click `Install`.

End of procedure

Next Steps

- Update existing eServices components to work with SMS Server. See [Procedure: Configuring eServices Components to work with SMS Server](#), on [page 232](#)

Procedure: Installing SMS Server on AIX, Solaris, or Linux

Prerequisites

- The SMS Server Application object has been configured. See [Procedure: Creating the SMS Server application object](#), on [page 229](#).
- JRE x64, or x32, (depending on the Operating System version) is installed.

Note: For Oracle, Java version 1.6.0_26 or higher (within the 1.6 family) is supported. In release 8.1.3 and higher, JRE 1.7 is also supported.

For IBM, Java (OS AIX) version 1.6.0 build pap6460sr9fp2-20110627_03(SR9 FP2) for AIX or higher (within the 1.6 family) is supported. In release 8.1.3 and higher, Java 1.7 is also supported.

- The SMS Server CD.
- Review the section “Compatibility Package for Red Hat Linux” on [page 189](#).

Start of procedure

1. Locate the `install.sh` file for SMS Server.
2. Start the installation script with the command `>install.sh`
3. On request, enter the absolute path for the directory of the Java executable.
4. Press Enter to confirm the host name for the installation
5. Enter the login information for your Configuration Server:
 - Host
 - Port
 - User
 - Password
6. Choose whether the application will use Client Side Port.
7. From the list of applications, select one and enter its number in the list.
8. Press Enter to confirm the suggested destination directory, or choose another one.

9. Answer other questions, if required.

Note: On some Red Hat Linux platforms, eServices 8.1 components might fail to start and produce the error `./cfgutility: error while loading shared libraries: libstdc++.so.5: cannot open shared object file: No such file or directory`. Refer to “Compatibility Package for Red Hat Linux” on [page 189](#) for information about correcting this issue.

End of procedure

Next Steps

- Optionally, configure SMS Server to work with Chat Server in session mode:
 - a. Add a connection to Solution Control Server.
 - b. Add a connection to one or more Chat Servers.
- Update existing eServices components to work with SMS Server. See [Procedure: Configuring eServices Components to work with SMS Server](#), on [page 232](#)

Procedure: Configuring eServices Components to work with SMS Server

Purpose: To ensure eServices components are configured correctly to work with SMS Server.

Prerequisites

- The SMS Server Application object exists. See [Procedure: Creating the SMS Server application object](#), on [page 229](#).

Start of procedure

1. In Configuration Manager, locate the Interaction Server Application object and open its properties.
2. On the Connections tab, add a connection to SMS Server.
3. Click Apply.

End of procedure



Chapter

13 Capture Points

Interaction Server 8.1 supports four integrated capture points:

- Java Messaging Service (JMS) Capture Point captures interactions from source systems that use JMS as a message bus.
- File Capture Point captures interactions from XML files.
- Database Capture Point captures interactions from databases.
- Web Service Capture Point (8.1.2 and later) provides a web service interface for interaction-related requests such as submit, stop, update, hold, resume, and get info, as well as for ping requests.

In previous version of this Deployment Guide, this chapter described how to deploy the integrated capture point functionality. That information is now provided in the *eServices Integrated Capture Points* document, which you can access at <http://docs.genesyslab.com/Documentation/ES>.



Chapter

14 Social Media

eServices 8.1 includes Social Media functionality, such as support for Facebook and Twitter. For more information about Social Media in eServices, including how to deploy the new Social Messaging Server and other components, refer to the *eServices Social Media Solution Guide*, available on the Genesys Documentation Wiki at <http://docs.genesyslab.com>.



Chapter

15

Uninstalling eServices

This chapter describes how to uninstall eServices components. It covers the following topics:

- [Using Wizards to Uninstall eServices, page 237](#)
- [Removing eServices Components, page 238](#)
- [Removing Genesys Desktop, page 240](#)

Using Wizards to Uninstall eServices

Complete the following procedure to uninstall eServices.

Procedure: Uninstalling eServices using the wizards

Start of procedure

1. Launch the Genesys Wizards Manager.
2. Click **Multimedia** on the left side of the window. The Wizard presents a list of currently installed eServices Solutions. Selecting one of them enables you to change or uninstall the solution (see [Figure 60](#)).

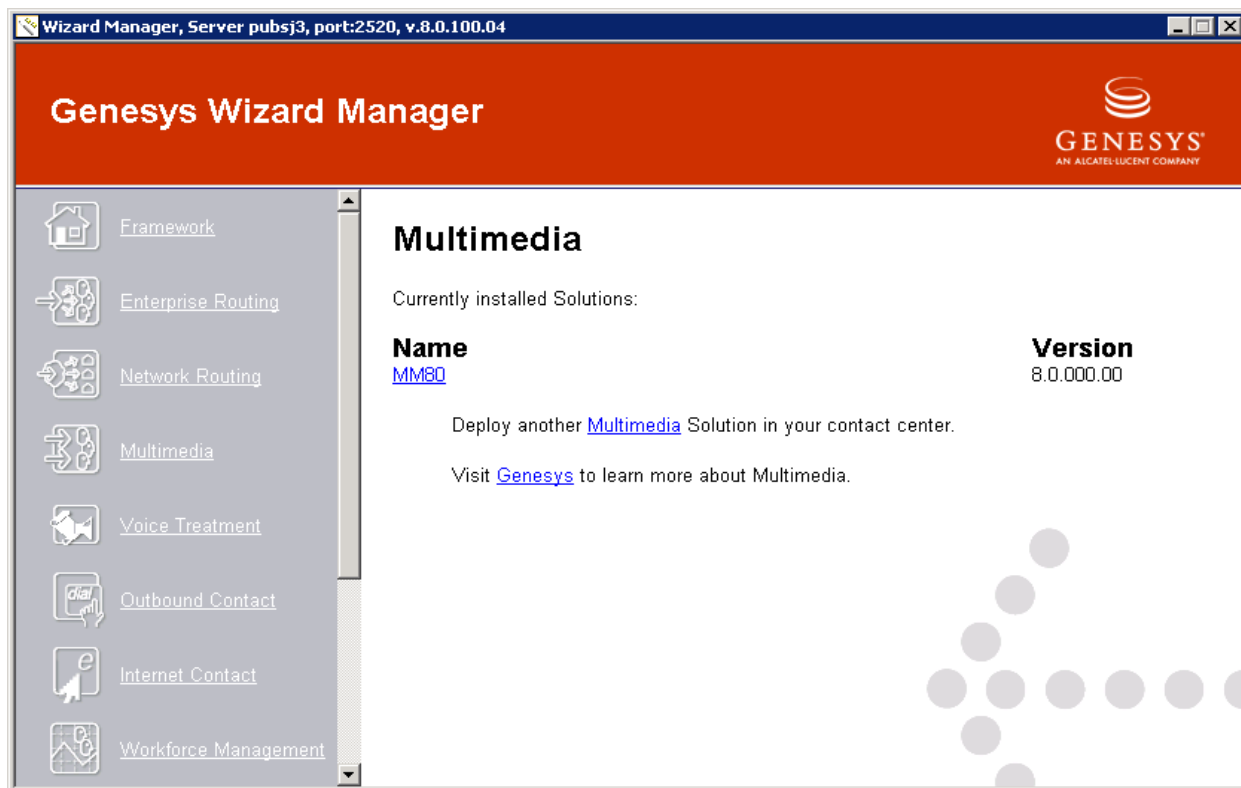


Figure 60: Genesys Wizard Manager

3. Click **Uninstall**. The Wizard asks you to confirm your decision, and then uninstalls the solution.

End of procedure

Next Steps

- Remove the eServices components from each host machine. See [Procedure: Removing eServices components](#).

Removing eServices Components

Use the following procedure to remove eServices components from each host machine.

Procedure: Removing eServices components

Start of procedure

1. From the Windows Start menu, open the Control Panel (Settings > Control Panel) and click Add or Remove Programs.
2. At the Add or Remove Programs dialog box, select these programs:
 - Genesys .Net Web API Server & Samples 8.1
 - Genesys Chat Server 8.1
 - Genesys Classification Server 8.1
 - Genesys Co-Browsing Server 8.1
 - Genesys E-mail Server 8.1
 - Genesys Interaction Server 8.1
 - Genesys Interaction Server Proxy 8.1
 - Genesys Interaction Workflow Samples 8.1
 - Genesys Knowledge Manager 8.1
 - Genesys eServices Configuration Wizards 8.1
 - Genesys Java Environment and Libraries for eServices and UCS 8.1
 - Genesys SMS Server 8.1
 - Genesys Social Messaging Server 8.1
 - Genesys Training Server 8.1
 - Genesys Universal Contact Server 8.1
 - Genesys Universal Contact Server Proxy 8.1
 - Genesys Universal Contact Server Manager 8.1
 - Genesys Web API Server 8.1
 - Genesys eServices Web Callback Application 8.1
3. Click Change/Remove. Repeat this step as needed for each component listed in [Step 2](#).
4. Using Windows Explorer, browse to the GCTI main directory and delete the complete eServices subdirectory, including all subfolders.
5. In your RDBMS, delete the tables and databases that relate to eServices. For the model setup presented in Chapter 5, beginning on [page 97](#), for instance, this would mean deleting the IxnDB, Customer, and, possibly, co-browse databases from the Microsoft SQL 2000 RDBMS on host MMHost. If you have an Archive instance of your Universal Contact Server database, you would delete its Customer database as well. If you have an Event Logger database, you would delete that as well.

6. Consult with your database administrator about deleting the database files themselves.

End of procedure**Next Steps**

- You have completed all the necessary steps to uninstall your eServices 8.1 Solution.

Removing Genesys Desktop

To remove Genesys Desktop, refer to the *Genesys Desktop 7.6 (or later) Deployment Guide*.



Chapter

16

Starting and Stopping Procedures

This chapter describes how to start and stop eServices 8.1. It covers these topics:

- [Starting eServices with Solution Control Interface, page 241](#)
- [Starting eServices Manually, page 242](#)
- [Stopping eServices with Solution Control Interface, page 244](#)

Starting eServices with Solution Control Interface

Complete the following procedure to start eServices using the Solution Control Interface (SCI).

Procedure: Starting eServices using SCI

Note: You cannot use Solution Control Interface to start or stop Web API Server. See [“Starting eServices Manually”](#) for instructions on how to start your Web API Server.

Prerequisites

You must start all of the following Genesys components before starting SCI:

- Local Control Agent
- Primary Database Server
- Configuration Server

- Solution Control Server (SCS)

Start of procedure

1. Start SCI.
2. Go to the **Solutions** view.
3. Select the name of your eServices solution on the **List** pane. (For the model installation, the name is **MM81**.)
4. Click **Start** on the toolbar or select **Start** from the shortcut menu and confirm the action in the dialog box.

The command to start eServices is sent to SCS, which uses Local Control Agents (LCAs) to activate the solution components in the Solution configuration object.

After all of the eServices components have reported a status of **Running** within the configured timeout, SCI reports a successful start of eServices by changing the solution status from **Pending** to **Running**.

Note: Because many components are shared by a number of solutions, some eServices components can have status **Running** before the solution as a whole has started.

End of procedure

Next Steps

- You have completed all of the steps to start eServices using SCI.

For more information, see *Framework 8.0 Solution Control Interface Help*. To view the Help file, open SCI, and then launch the Help file.

For information about the following Management Layer topics, see the documentation for Genesys Framework 8.1:

- Message Server
- Solution Control Server
- Solution Control Interface
- Local Control Agent
- Log Database

Starting eServices Manually

This section describes how to manually start eServices and the eServices servers. For information on manually starting Database Server, Message Server, and Stat Server, see the *Framework 8.1 Deployment Guide*. For Universal Routing, see the *Universal Routing 8.1 Deployment Guide*.

Procedure: **Starting eServices components from the command line**

Start of procedure

1. A *.bat (or *.sh for UNIX) file for Interaction Server is available in its installation directory (as is the case with almost all eServices components). You can use this file to start Interaction Server (or the given component). Or you can start Interaction Server by using a command line with the following command-line options:

- host The name of the host where Configuration Server is running
- port The Configuration Server port
- app The Application object name in the Configuration database
- l The port and host of the license server

You can place the command line with the specified command-line parameters in your own *.bat file. For Solaris, Linux, and AIX environments, you should also place the command line parameters in the startup script.

Example

```
interaction_server.exe -host cs-host -port 2020  
-app "IxnSrv" -l 7260@MMHost
```

End of procedure

Procedure: **Starting eServices components as services**

Services may include LCA and the web server.

Start of procedure

1. From the Windows Start menu, select Settings > Control Panel.
2. In the Control Panel window, select Administrative Tools > Component Services.
3. In the Component Services window, verify that a given service is running. If it is not, start it.

End of procedure

Procedure:
Starting eServices from the Windows Start menu

You can start all eServices applications from the Windows Start menu.

Start of procedure

- From the Windows Start menu, select Start > Programs > Genesys Solutions > eServices 8.1.0 > <ApplicationName> > <Shortcut to Application>.

End of procedure**Note on E-mail Server**

Before starting E-mail Server, make sure the computer where it is installed has TCP/IP loaded and that it can communicate with the corporate mail server. If the server has a dial-up connection to the Internet, connect to the Internet before starting E-mail Server.

As soon as you start the e-mail POP server, E-mail Server retrieves all the messages stored in the specified POP mailbox and stores them in the Interaction Server database. If you are evaluating e-mail functionality, empty the POP mailbox ahead of time.

Stopping eServices with Solution Control Interface

Complete the following procedure to stop eServices using SCI.

Procedure:
Stopping eServices using SCI**Start of procedure**

1. Start SCI.
2. Go to the Solutions view.
3. Select the name of your eServices solution on the List pane (MM81 in the model setup).

4. Click Stop on the toolbar or select Stop from the shortcut menu and confirm the action in the dialog box.

The command to stop eServices is sent to SCS, which uses LCAs to terminate the solution components in the reverse order of the component startup. (The component startup order is defined in the Solution configuration object.)

SCI reports a successful de-activation of eServices after all of the solution components have terminated within the configured timeout. Upon a successful de-activation of eServices, its status changes from Running to Stopped.

Note: Because many components are shared by a number of solutions, some eServices components can continue to have the status Running after the solution has stopped.

End of procedure

Next Steps

- You have completed all the steps to stop eServices using SCI.



Appendix

Deploying an E-Mail Solution Using MIME Customization

Multipurpose Internet Mail Extensions (MIME) is a standard that allows e-mail messages to include graphics, audio or video files, or text in languages other than English.

This appendix describes how to create a custom transformer that enables you to modify the content of e-mails and still ensure that they are compliant with MIME standards.

- [Overview, page 247](#)
- [Configuration Process, page 248](#)
- [Message Flow Pattern, page 254](#)

Overview

MIME customization in a Genesys e-mail handling environment ensures that you have normalized formats for all e-mail messages processed by your system and stored in the UCS database. To accomplish this, your e-mail solution uses a custom transformer to transform any MIME content in incoming or outgoing messages.

You implement this solution by using MIME customization APIs (as described in “Creating a custom transformer” on [page 249](#)) to change the content of incoming and outgoing e-mails. After creating the custom MIME transformer, you configure the E-mail Server Application object to enable the MIME customization option. For details, see “Configuring the E-Mail Server Application Object” on [page 251](#).

Your custom server then does all of the work required to handle incoming and outgoing e-mail messages as needed.

- When an e-mail message is received, E-mail Server uses the custom MIME transformer to transform the message before saving it in the UCS database.
- When an e-mail message is sent, E-mail Server uses the custom MIME transformer to transform the message (if it requires MIME customization) before sending it to the external e-mail server.

Configuration Process

Complete the following procedure to deploy a MIME-compliant e-mail solution.

Procedure:

Deploying a MIME-compliant e-mail solution

Start of procedure

1. [Procedure: Viewing a sample transformer](#)
2. [Procedure: Creating a custom transformer](#)
3. [Procedure: Configuring the E-mail Server application object](#), on [page 251](#)

End of procedure

Procedure:

Viewing a sample transformer

E-mail Server 8.1 includes a sample incoming MIME transformer. This sample is an “MS-TNEF Microsoft specific format to MIME” transformer, and can be found in the `mimeapi` subdirectory.

Start of procedure

1. Open the directory where E-mail Server 8.1 is installed.
2. In the `mimeapi` subdirectory, you can find the following files:
 - `esj-mime-api-doc.jar`—Javadoc documentation for the API.
 - `esj-mime-api.jar`—The actual API archive.
 - `samples`—A subdirectory, containing the following files:
 - `TNEFMimeTransformer.java`—Java source code for this sample.

- `readme.txt`—A readme file describing the sample.

Tip: Use the instructions in the `readme.txt` to download the JTNEF library and then compile the source. After you complete these two actions, you can use the sample to transform MS-TNEF incoming mails into regular MIME messages.

End of procedure

Next Steps

- Create a custom transformer. See [Procedure: Creating a custom transformer](#), on [page 249](#).

Procedure: Creating a custom transformer

Overview

To transform MIME content of e-mail messages, you must create a custom transformer. Your transformer code needs two Java classes: one to implement the API for transforming MIME content of incoming e-mail, and another to provide an API that transforms MIME content of outgoing e-mail. Both of these classes return the transformation result, and contain the following parameters:

- `input`—The MIME message content.
- `config`—Properties contained in the `[mime-custom-outbound-properties]` section of your E-mail Server Application object.
- `debugLogStream`—Log object to be used for debugging purposes.

Each custom class that you create should satisfy the following conditions:

- Implements one of the two appropriate interfaces (see “[Incoming E-Mail Customization API](#)” and “[Outgoing E-Mail Customization API](#)” for details).
- Is thread-safe.
- Does not change the MIME message ID.

Start of procedure

1. Create two Java classes: one for transforming incoming e-mail, another for transforming outgoing e-mail. Use the following interfaces:
 - [Incoming E-Mail Customization API](#), [page 250](#)
 - [Outgoing E-Mail Customization API](#), [page 250](#)
2. Bundle these two classes into a JAR file called `mimecustomization.jar`.

3. Place the new JAR file in the esj\lib\external\ folder of your E-mail Server installation.

End of procedure

Next Steps

- [Procedure: Configuring the E-mail Server application object](#), on [page 251](#)

Incoming E-Mail Customization API

Your custom transformer for incoming e-mail must implement the following interface:

EmailInTransformer Interface

```
public interface EmailInTransformer {
    public TransformerResult transform(byte[] input, java.util.Properties config,
        java.io.PrintStream debugLogStream);
    public class TransformerResult {
        public static TransformerResult noTransformationNeeded();
        public static TransformerResult succesfull(byte[] transformedInput);
        public static TransformerResult failure(String failureReason, FailureAction
            failureAction);
        // Implementation details skipped
    }

    public class FailureAction {
        public static final FailureAction RETRY;
        public static final FailureAction BYPASS_TRANSFORMATION;
        public static final FailureAction DEPEND_ON_BAD_FORMAT_OPTION;
        // Implementation details skipped
    }
}
```

Outgoing E-Mail Customization API

The only difference between this API and the incoming e-mail customization API is that the `failureAction` parameter and class have been removed. If a transformation fails, then the original message is sent to the external e-mail server.

Your custom transformer for outgoing e-mail must implement the following interface:

EmailOutTransformer Interface

```
public interface EmailOutTransformer {
    public TransformerResult transform(byte[] input, java.util.Properties config,
```

```

        java.io.PrintStream debugLogStream);
    public static class TransformerResult {
        public static TransformerResult noTransformationNeeded();
        public static TransformerResult succesfull(byte[] transformedInput);
        public static TransformerResult failure(String failureReason);
        // Implementation details skipped
    }
}
}

```

Configuring the E-Mail Server Application Object

After you create the custom MIME transformer, configure the E-mail Server Application object to specify the class names and configuration settings used with your custom transformer. E-mail Server will use the values you specify to transform content that is sent and received as e-mail messages.

Procedure: Configuring the E-mail Server application object

Start of procedure

In Configuration Manager, configure the following sections on the Options panel of your E-mail Server Application object:

- [mime-custom-inbound-properties]—Content in this section is passed to the `EmailInTransformer.transform()` method using the `config` parameter. You can use this section to define custom options and settings.
This section is not part of the default template. Add this section manually to set options for your custom transformer for incoming e-mail.
- [mime-custom-outbound-properties]—Content in this section is passed to the `EmailOutTransformer.transform()` method using the `config` parameter. You can use this section to define custom options and settings.
This section is not part of the default template. Add this section manually to set options for your custom transformer for outgoing e-mail.
- [mime-customization]—Use this section to enable or disable the MIME customization, to specify the fully qualified class names of your custom classes, or to set allow optional debugging or saving features.
See Table 17 on [page 253](#) for details about the options applicable to this section.

End of procedure

Next Steps

- You have completed the actions required to deploy a MIME-compliant e-mail solution.

MIME Customization Options Table

Table 17: MIME Customization Options Section

Option Name	Value	Description
[mime-customization] Section		
enable-inbound	Default Value: <code>false</code> Valid Values: <code>true</code> , <code>false</code>	If set to <code>true</code> , sends inbound e-mail messages to the specified inbound MIME transformer class.
enable-outbound	Default Value: <code>false</code> Valid Values: <code>true</code> , <code>false</code>	If set to <code>true</code> , sends outgoing e-mail messages to the specified outbound MIME transformer class.
enable-inbound-debug-log	Default Value: <code>false</code> Valid Values: <code>true</code> , <code>false</code>	If set to <code>true</code> , activates the inbound debug logger.
enable-outbound-debug-log	Default Value: <code>false</code> Valid Values: <code>true</code> , <code>false</code>	If set to <code>true</code> , activates the outbound debug logger.
inbound-class-name	Default Value: <code>""</code> Valid Values: <code><any string></code>	Specifies the fully qualified name of the custom inbound transformer.
outbound-class-name	Default Value: <code>""</code> Valid Values: <code><any string></code>	Specifies the fully qualified name of the custom outbound transformer.
inbound-keep-received-mime	Default Value: <code>false</code> Valid Values: <code>true</code> , <code>false</code>	Controls the way E-mail Server saves the content of MIME messages. If set to <code>true</code> , and if the message was transformed successfully, then saves the unmodified MIME content of incoming e-mails received in the UCS database (along with the transformed content).
outbound-keep-sent-mime	Default Value: <code>false</code> Valid Values: <code>true</code> , <code>false</code>	Controls the way E-mail Server saves the content of MIME messages. If set to <code>true</code> , and if the message was transformed successfully, then saves the transformed MIME content of outgoing e-mails is also saved in the UCS database (along with the initial content).

Message Flow Pattern

Incoming E-Mail Message Flow

Figure 61 shows the message pattern for incoming e-mail messages.

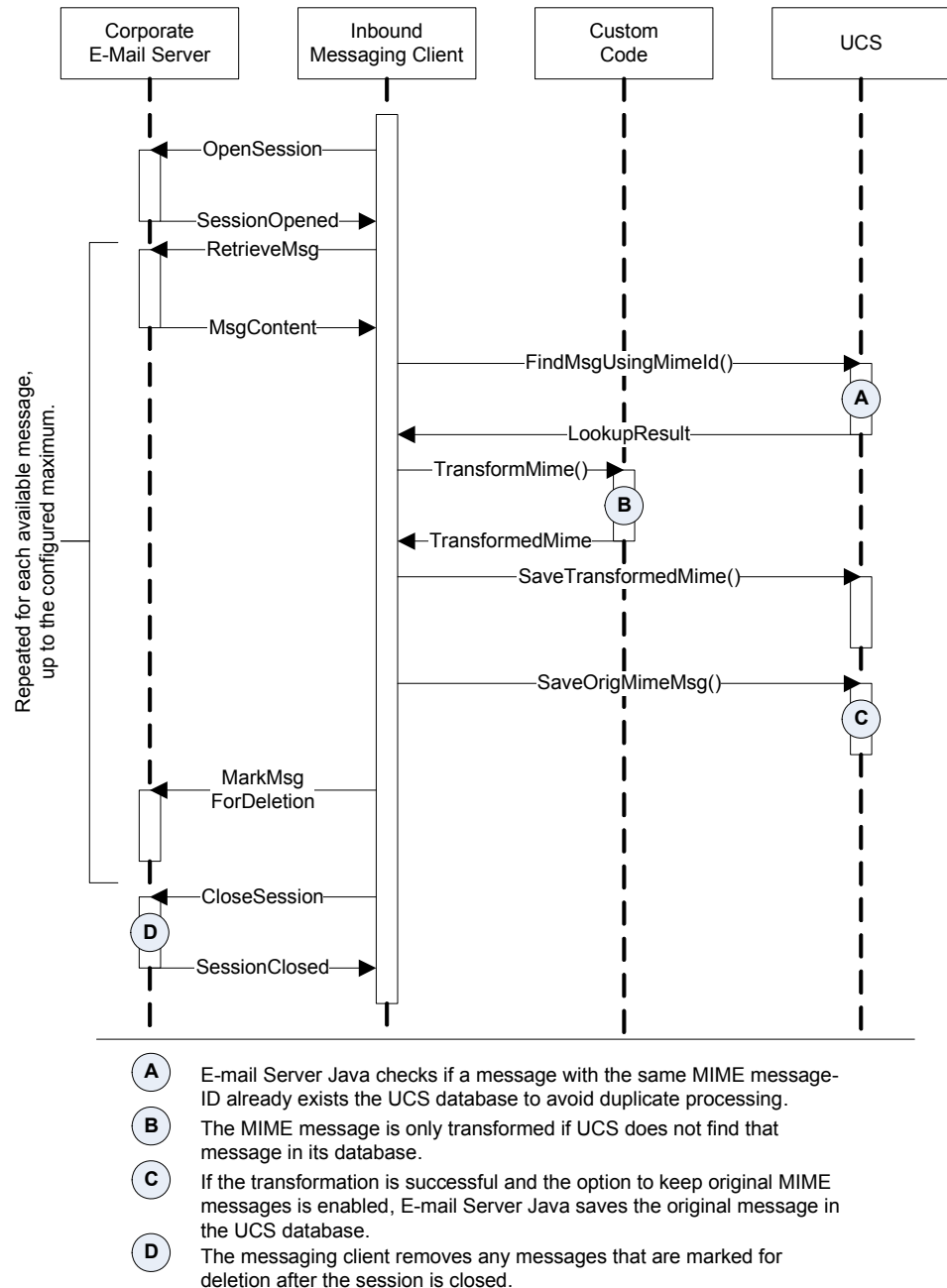


Figure 61: Message Flow for Incoming E-Mail

Outgoing E-Mail Message Flow

Figure 62 shows the message pattern for outgoing e-mail messages.

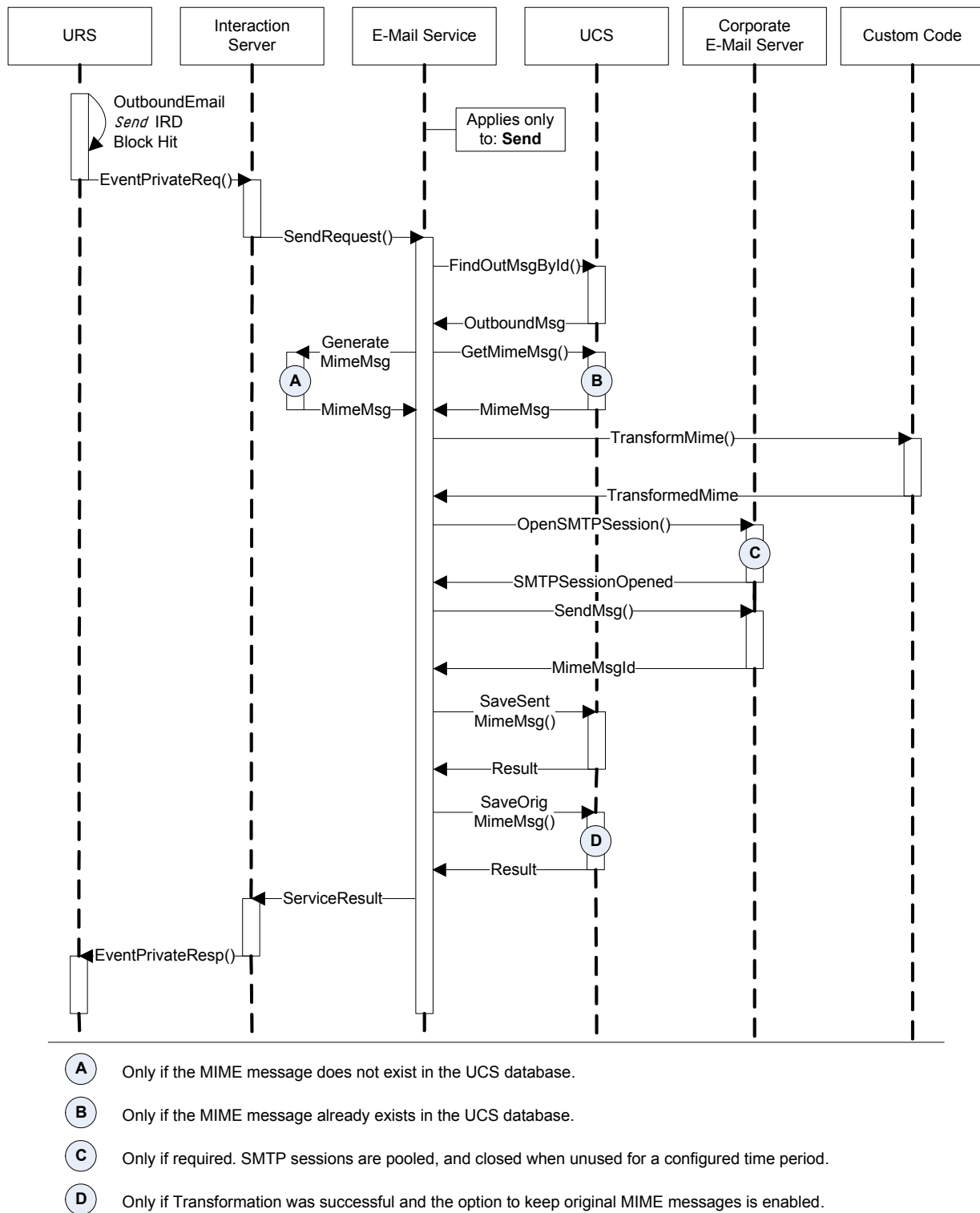


Figure 62: Message Flow for Outgoing E-Mail



Supplements

Related Documentation Resources

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

eServices

- *eServices 8.1 Reference Manual*, which provides a reference listing of all configuration options and of field codes used in standard responses.
- *eServices 8.1 User's Guide*, which provides overall information and recommendations on the use and operation of eServices.
- *eServices Social Media Solution Guide*, which provides information on deploying and using the Genesys Social Messaging Management product. It is available on the Genesys Documentation Wiki.
- *eServices 8.1 Universal Contact Server Manager Help*, which is a guide to the Universal Contact Server Manager user interface.
- *eServices 8.1 Knowledge Manager Help*, which is a guide to the Knowledge Manager user interface.
- For a listing of classes, methods, fields, and constants of the Web API portion of the Web API Server component, see:
 - *eServices 8.0 .NET Web API Reference* for the .NET Web API
 - The API References of the Platform SDK for the Java-based Web API
- *eServices 8.1 Web API Client Developer's Guide*, which describes the structure of the Web API, explains the Simple Samples, and describes procedures for customizing them.
- “eServices Log Events” in *Framework 8.1 Combined Log Events Help*, which is a comprehensive list and description of all events that may be recorded in logs.
- For the Web Collaboration option, the following documents describing design and administration for the KANA Response Live Server, which is supplied by Genesys as part of the Web Collaboration product:
 - [Hipbone Client API Reference Guide](#)

- [KANA Response Live Organization Administration Tool Guide](#)
- [KANA Response Live Server Installation Guide](#)
- [KANA Response Live System Administration Tool User Guide](#)

Genesys Desktop

- *Genesys Desktop 7.6 (or later) Deployment Guide*, which describes deployment procedures for the Genesys Desktop.
- *Genesys Desktop 7.6 (or later) Developer's Guide*, which describes customizing the Genesys Desktop.
- *Genesys Desktop 7.6 (or later) Agent Help*, which is a guide to the Genesys Agent Desktop.
- *Genesys Desktop 7.6 (or later) Supervisor's Help*, which is a guide to the Genesys Supervisor Desktop.

Universal Routing

- *Universal Routing 8.1 Reference Manual*, which contains descriptions of all routing strategy objects, including those that are specific to eServices.
- *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 supplied with eServices.
- *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 for eServices.

Genesys

- *Genesys Events and Models Reference Manual*, which includes a set of basic interaction models, showing the components involved and relevant event messages sent among them. For authoritative description of the event messages, see the next item.
- The API References of the Platform SDK, which provide the authoritative information on methods and functions for each SDK, including requests and events. The class `Message` includes all event and request messages.
- *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 Customer Care for more information.
- *Genesys 8.1 Security Deployment Guide*, which describes Genesys security features and detailed instructions for deploying them. These features provide secure data transfer between Genesys components, protection against unauthorized access, and protection against data loss in case of component failure.
- Release Notes and Product Advisories for this product, which are available on the Genesys Customer Care website at <http://genesyslab.com/support>.

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

- [Genesys Supported Operating Environment Reference Guide](#)
- [Genesys Supported Media Interfaces Reference Manual](#)

Consult these additional resources as necessary:

- The documentation on the other three members of the Genesys Customer Interaction Platform: Universal Routing, Reporting, and Management Framework.
- *Genesys Hardware Sizing Guide*, which provides information about Genesys hardware sizing guidelines.
- *Genesys Interoperability Guide*, which provides information on the compatibility of Genesys products with various Configuration Layer Environments; Interoperability of Reporting Templates and Solutions; and Gplus Adapters Interoperability.
- *Genesys Licensing Guide*, which introduces you to the concepts, terminology, and procedures relevant to the Genesys licensing system.
- *Genesys Database Sizing Estimator 8.x Worksheets*, which provides a range of expected database sizes for various Genesys products.

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

Genesys product documentation is available on the:

- Genesys Customer Care website at <http://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 18](#) describes and illustrates the type conventions that are used in this document.

Table 18: Type Styles

Type Style	Used For	Examples
Italic	<ul style="list-style-type: none"> Document titles Emphasis Definitions of (or first references to) unfamiliar terms Mathematical variables <p>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 261).</p>	<p>Please consult the <i>Genesys Migration Guide</i> for more information.</p> <p>Do <i>not</i> use this value for this option.</p> <p>A <i>customary and usual</i> practice is one that is widely accepted and used within a particular industry or profession.</p> <p>The formula, $x + 1 = 7$ where x stands for . . .</p>
Monospace font (Looks like teletype or typewriter text)	<p>All programming identifiers and GUI elements. This convention includes:</p> <ul style="list-style-type: none"> The <i>names</i> 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. <p>Also used for any text that users must manually enter during a configuration or installation procedure, or on a command line.</p>	<p>Select the Show variables on screen check box.</p> <p>In the Operand text box, enter your formula.</p> <p>Click OK to exit the Properties dialog box.</p> <p>T-Server distributes the error messages in EventError events.</p> <p>If you select true for the inbound-bsns-calls option, all established inbound calls on a local agent are considered business calls.</p> <p>Enter exit on the command line.</p>
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.	<code>smcp_server -host [/flags]</code>
Angle brackets (< >)	<p>A placeholder for a value that the user must specify. This might be a DN or a port number specific to your enterprise.</p> <p>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.</p>	<code>smcp_server -host <confighost></code>



Index

Symbols

[] (square brackets)	261
< > (angle brackets)	261

Numerics

3rd Party Media SDK	18
64-bit UNIX	165
64-bit Windows	165

A

ABC Simple Business Processes	
chat	159
e-mail	158
AIX	
configuration	27
installation on	137
Java SDK	57
WebSphere configuration	77
angle brackets	261
Apache Tomcat	55
ApacheGroup	58
Application Cluster	
configuration	124
audience, for document	14

B

brackets	
angle	261
square	261
business process	20

C

cache (Interaction Server database)	20, 23
Capacity Rules	

configuration	132
CATALINA_HOME	61
chat	
testing	159, 160
chat mode	24
Chat Server	18
in Wizard	131
installation	142
CIM Platform	12
Classification Server	20
in Wizard	129
installation	141
new-style installation	166
client-side port	169
client-side port configuration	149, 151
co-browse session	93
Co-Browsing Server	
in Wizard	132
installation	142
commenting on this document	14
configuration	
Application Cluster	124
Capacity Rules	132
DAPs and DB Server	114
E-mail Server	127
Genesys Web Collaboration	132
KANA Response Live for Windows	84
Knowledge Manager	130
Multimedia Solution	112
multimedia switch	119
Stat Server	32, 52, 124
Training Server	130
Universal Contact Server	117
Universal Routing Server	124
configuring	
Checking your WebLogic configuration file	203
Configuring E-mail Server's POP, IMAP, and SMTP ports	181
Configuring IIS to serve static content	73
Configuring KANA Response Live for Solaris or Linux	91

- Configuring KANA Response Live for Windows 84
- Configuring Multimedia Components to work with SMS Server 232
- Configuring Multimedia objects using the wizard 104
- Configuring the desktop application to use UCS Proxy or Interaction Server Proxy . 228
- Configuring Tomcat to serve multiple contexts. 74
- Configuring WebCallback BP for routing web callback interactions. 162
- Creating or editing the UCS application object 213
- Creating or editing the UCS Manager application object 214
- Creating or editing the Web API Server application object 212
- Creating the Application object 147
- Creating the KANA database schema on Solaris or Linux 90
- Creating the Proxy application object . . . 225
- Creating the SMS Server application object 229
- Generating the .truststore file 179
- Modifying the E-mail Server startup command line
 - Unix 180
 - Windows 179
- Protecting the WEB-INF Directory 74
- Running scripts for your databases 153
- conventions
 - in document 260
 - type styles 261
- Customer Interaction Management
 - See CIM Platform

D

- database
 - size 39
 - Universal Contact Server database . 19, 20, 22
 - See also cache
- Database Access Point (DAP)
 - configuration. 114
 - for Universal Contact Server 116
 - note for JDBC use. 116
- document
 - audience. 14
 - change history. 15
 - conventions 260
 - errors, commenting on 14
 - version number 260

E

- e-mail
 - incoming message flow 254
 - MIME customization 247
 - outgoing message flow 255
 - testing. 158, 159
 - testing web form 161
- E-mail CD
 - Wizard 127
- E-mail Server 18
 - configuration 127
 - in Wizard 127
 - installation. 141
 - MIME customization 251
 - starting manually 244
- enable-ssl option (E-mail Server) 181
- Estimating UCS database size for chat interactions 40
- Event Logger 39, 44, 107, 152
- Event Logger DAP. 44, 107

F

- font styles
 - italic 261
 - monospace 261
- Framework resources
 - agent 122
 - agent login 122
 - capacity rule. 121
 - place 121
 - skills. 122

G

- Genesys Content Analyzer 21

H

- https port value 125

I

- IHS_HOME 190
- IMAP. 181
- install directory
 - Apache 58
 - IBM HTTP Server 190
 - JDK 57, 61
 - Multimedia 101, 113
 - Tomcat 61
 - WebSphere 190
- installation

- Chat Server 142
- Classification Server. 141
- Co-Browsing Server. 142
- E-mail Server 141
- Interaction Server 140
- IPStarter 139
- KANA Response Live for Solaris 88
- KANA Response Live for Windows 82
- Knowledge Manager 142
- Training Server 142
- Universal Contact Server 140, 142
- Web API Server 141
- installation on
 - AIX. 137
 - Linux. 137
 - Solaris 137
- installing
 - Deploying Genesys web applications for WebSphere on AIX or Solaris 192
 - Deploying Multimedia on Solaris, AIX, or Linux from a Windows host. 187
 - Deploying your web applications 204
 - Downloading and installing the Java SDK. 56
 - Installing a new SSL certificate using IIS 80
 - Installing and Starting Interaction Workflow Samples 103
 - Installing eServices components manually
 - AIX, Solaris, or Linux. 151
 - Windows 148
 - Installing Genesys web applications for WebLogic 202
 - Installing Genesys web applications for WebSphere 191
 - Installing KANA Response Live for Windows 82
 - Installing KANA Response Live for Linux 89
 - Installing KANA Response Live for Solaris 88
 - Installing SMS Server
 - AIX, Solaris, or Linux. 231
 - Windows 230
 - Installing the Apache Web Server on Windows 58
 - Installing the JK1 Connector with Apache. 64
 - Installing the JK1 Connector with IIS 6 66
 - Installing the JK1 connector with IIS 7 69
 - Installing the Multimedia Configuration Wizards 102
 - Installing the Tomcat web application container on Windows 60
 - Installing Universal Contact Server
 - AIX, Solaris, or Linux. 219
 - Windows 218
 - Installing Universal Contact Server Manager 220
 - Installing Universal Contact Server Proxy or Interaction Server Proxy

- AIX, Solaris, or Linux 227
 - Windows. 226
- Installing Web API Server
 - Solaris or Linux 217
 - Windows. 215
- Performing an integrated installation of all Multimedia servers 138
- Verifying connections 153
- intended audience 14
- Interaction design 20
- Interaction Server 20
 - database 26, 101
 - installation. 140
- Interaction Server Proxy 154, 223
- IPStarter
 - location 139
- italics. 261

J

- Java Environment and Libraries for eServices and UCS 33, 56
- Java Environment and Libraries for eServices and UCS installation 137
- Java Runtime Environment (JRE) 26, 100
- Java SDK 57
- JAVA_HOME. 57, 61
- JDBC
 - note for Universal Contact Server. 116

K

- KANA Response Live 78
 - configuration for Windows 84
 - database schema 90
 - installation for Solaris 88
 - installation for Windows. 82
 - launching co-browse session 93
 - testing for Windows 87
- Knowledge Manager
 - configuration 130
 - in Wizard 130
 - installation. 142
 - new-style installation 166

L

- Linux
 - Apache & Tomcat configuration. 76
 - configuration 27
 - installation on 137
 - Java SDK 57
- Local Control Agent (LCA). 242

M

Management Layer	26, 100, 242
media channels	12
message flow	
incoming e-mail	254
outgoing e-mail	255
MIME customization	247
custom transformer	249
E-mail Server	251
incoming API	250
incoming message flow	254
outgoing API	250
outgoing message flow	255
sample transformer	248
transformer	247
MMS interactions	25
model	21
monospace font	261
Multimedia CDs	
Interaction Management CD	102
selecting additional CDs	127
Multimedia Configuration Wizards	98
Multimedia install directory	101, 113
multimedia routing strategy	100
multimedia switch	
configuration	119

N

New in This Release	28
---------------------	----

O

operational data	18
------------------	----

P

paging mode	24
planning	
Completing the configuration worksheet	43
Estimating the size of the Interaction Server database	41
Estimating UCS database size for chat interactions	40
Estimating UCS database size for e-mail interactions	40
Obtaining Apache documentation from the product CD	76
Obtaining Tomcat documentation from the CD and the Web	77
POP	181
POP3 server	99
port option (E-mail Server)	181

Q

queue	20
-------	----

R

routing	157
routing strategy	20

S

Sample Business Processes	157
screening rule	20
Secure Socket Layer (SSL)	59
session (chat) mode	24
SMPP	18
SMS interactions	24
SMS Server	18, 223, 228
application object	229
deploying	228
installing	218, 219, 230, 231
new-style installation	166
other Multimedia components and requirements	232, 228
SMTP	181
Social Messaging Server	18
new-style installation	166
Solaris	
Apache & Tomcat configuration	76
configuration	27
installation on	137
Java SDK	57
Solution Control Server (SCS)	242
square brackets	261
SSL Configuration for E-mail Server	177
standard response	19, 21
starting	
Starting Multimedia components as services	243
Starting Multimedia components from the command line	243
Starting Multimedia from the Windows Start menu	244
Starting Multimedia using SCI	241
Stat Server	153, 154
as part of Real-Time Metrics Engine	26, 100
configuration	32, 52, 124
stopping	
Stopping Multimedia using SCI	244
support for	
AIX	27
Linux	27
Solaris	27

T

testing

- chat 159, 160
- Configuring ABC Simple BP for routing
 - e-mail interactions 158
- Configuring ABC Simple Chat BP for routing
 - chat interactions 159
- e-mail 158, 159
- Launching a co-browsing session 93
- Multimedia 154
- Sending a test e-mail to a Desktop agent . 159
- Sending a test web callback to a Desktop
 - agent 164
- Sending a test web form e-mail to a
 - Desktop agent 161
- Starting a test chat session with a Desktop
 - agent 160
- Testing KANA Response Live for Windows . 87
- Testing that Genesys Desktop can handle
 - interactions 156
- Testing the Apache Web Server on
 - Windows 59
- Testing the components using the Web-
 - based TestTool 155
- Testing the JK1 Connector with Apache. . 65
- Testing the JK1 Connector with IIS . . . 72
- Testing the KANA installation on Solaris or
 - Linux. 92
- Testing the Multimedia servers 154
- Testing the Tomcat installation on Windows. 62
- Testing the WebLogic installation and
 - deployment 208
- Testing the WebSphere installation on AIX
 - or Solaris 199
- web form e-mail 161
- Third Party Components. 137
- third-party e-mail server 99
- TLS 169
- Tomcat 55
- Training Server 21
- configuration. 130
- in Wizard 130
- installation 142
- new-style installation 166
- Transport Layer Security 169
- type option (E-mail Server). 181
- type styles
 - conventions 261
 - italic 261
 - monospace 261
- typographical styles 260, 261

U

UCS Proxy

- See Universal Contact Server Proxy
- uninstalling
 - Removing Multimedia components 239
 - Uninstalling Multimedia using the wizards 237
- Universal Contact Server 19
- configuration 117
- database 101
- Database Access Point (DAP) 116
- in Wizard 117
- installation. 140, 142
- new-style installation 166
- note for JDBC DAP 116
- Universal Contact Server database 26
- Universal Contact Server Manager 19
- in Wizard 126
- Universal Contact Server Proxy
 - application object 225
 - deploying 169, 223
 - Genesys Desktop and 228
 - installing. 148, 151, 226, 227
 - requirements 224
- Universal Routing 100
- Universal Routing Server
 - configuration 124
- updating
 - Creating or editing the UCS application
 - object. 213
 - Creating or editing the UCS Manager
 - application object 214
 - Creating or editing the Web API Server
 - application object 212
 - Updating the Interaction Server database for
 - processing web callback interactions. . 163

V

- version numbering, document 260

W

- Web API Server 18, 23
- in Wizard 125
- installation. 141
- web callback interactions 25
- Web Callback sample 25
- Web Collaboration
 - KANA Response Live 78
- Web Collaboration CD
 - Wizard 132
- web form e-mail
 - testing. 161
- Web Media CD
 - Wizard 131
- WebSphere
 - software requirements 78

WEBSphere_HOME	190
Windows	
Java SDK	57
Wizards.	98