eServices Deployment Guide

Genesys Engage Digital (eServices) 8.5.0

12/29/2021

Table of Contents

eServices Deployment Guide

Welcome to the eServices Deployment Guide. This document describes how to deploy your eServices 8.5 solution.

Overview

This chapter introduces you to eServices and its architecture.

Functional Components

Architecture

Deployment Planning

This chapter explains how to plan your deployment of the eServices solution.

Deployment Tasks

Using the Configuration Wizards

Third Party Application

This chapter explains how to deploy the third-party web components for the eServices solution.

Deploying Java

Deploying eServices

These topics explain how to deploy eServices.

Deploying on Windows

Deploying on UNIX

Manual Deployment

Other Information

These topics discusses other information required to run your eServices solution.

Transport Layer Security

Important

For the latest version of Interation Server Deployment Guide, see here.

For the latest version of Interaction Server Administration Guide, see here.

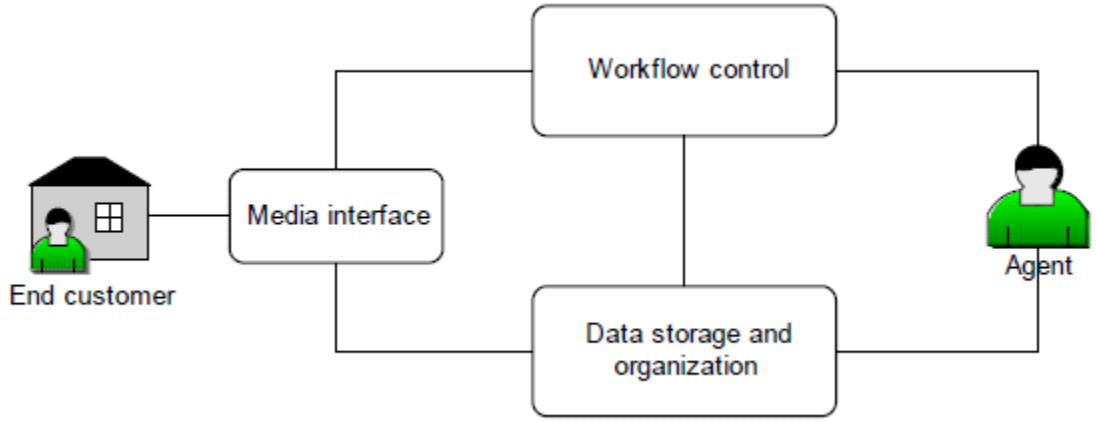
Overview of eServices

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 and their functionality
- The basic architecture and the processing of interactions

Functions and Components

The figure below shows the overall functionality of eServices with any media type. This figure shows functions only, not components.



General Functioning of eServices

The three major functions shown in the figure are:

- The **media interface,** which brings interactions into the system. It may interface with e-mail, chat, or other media.
  [+]Media Interfaces

  The media interfaces available with eServices 8.5 are E-mail Server, Chat Server, Social Messaging Server, and SMS Server.

  > **Important**
  > eServices 8.5 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; further details are available.
  - 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. More information is provided in the Social Media Solution Guide.

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.

- 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.
  [+]Data Storage: Universal Contact Server

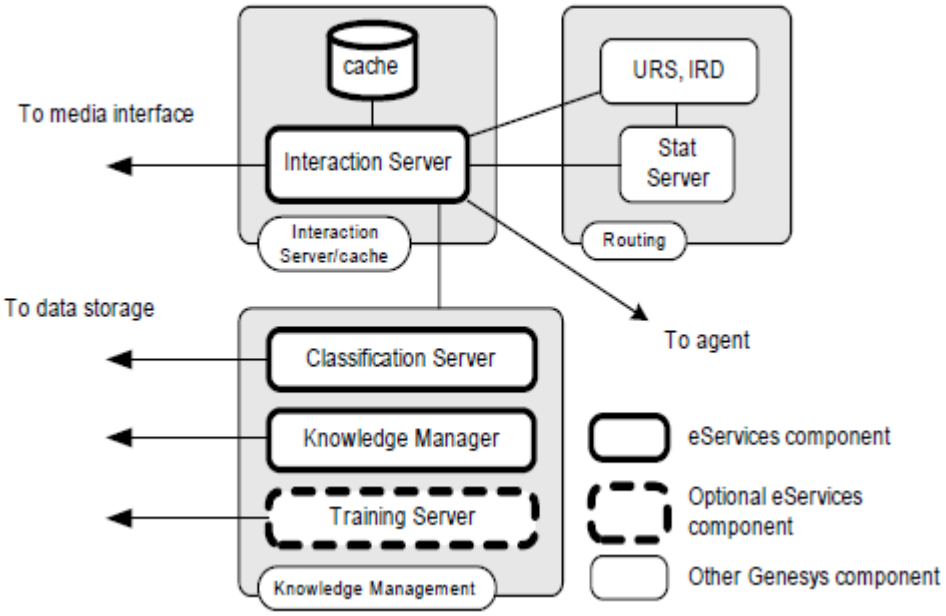
  Universal Contact Server (UCS) interfaces with a database that stores the following:

  - 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,** which determines where the interaction goes and what happens to it.
  [+]Workflow Control

  The components illustrated in the figure below handle workflow control.



Workflow Control Components

Not shown in the figure 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 (following 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 (previous section).
- **Interaction Design,** a subcomponent 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 (following 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

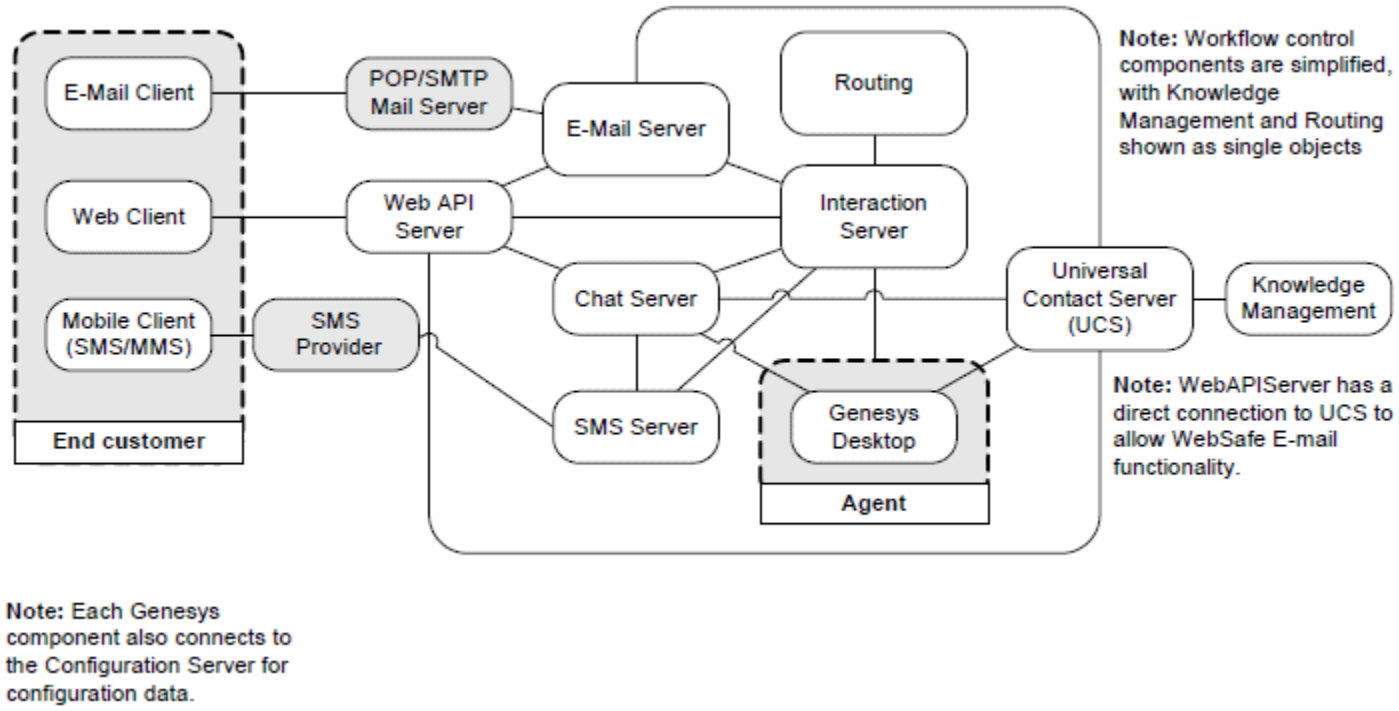
To summarize interaction flow:

- 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.

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

Architecture

The figure below shows the main eServices components.



eServices Architecture

> **Important**
>
> For simplicity, not all connections and details are included in the figure. 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 (email, chat, SMS, MMS, and web callback) are processed.

Processing E-Mail

1. E-mail interactions arrive in one of two ways:
   - If the customer sends ordinary email, the interaction arrives via the enterprise mail server.

- If the customer sends email 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. 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.
6. 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

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 emailed to the customer.

Processing SMS Messages

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 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.

Paging Mode

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.

Session Mode

This 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

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 Social Media

For information about processing Social Media, refer to the eServices Social Media Solution Guide.

Deployment Planning

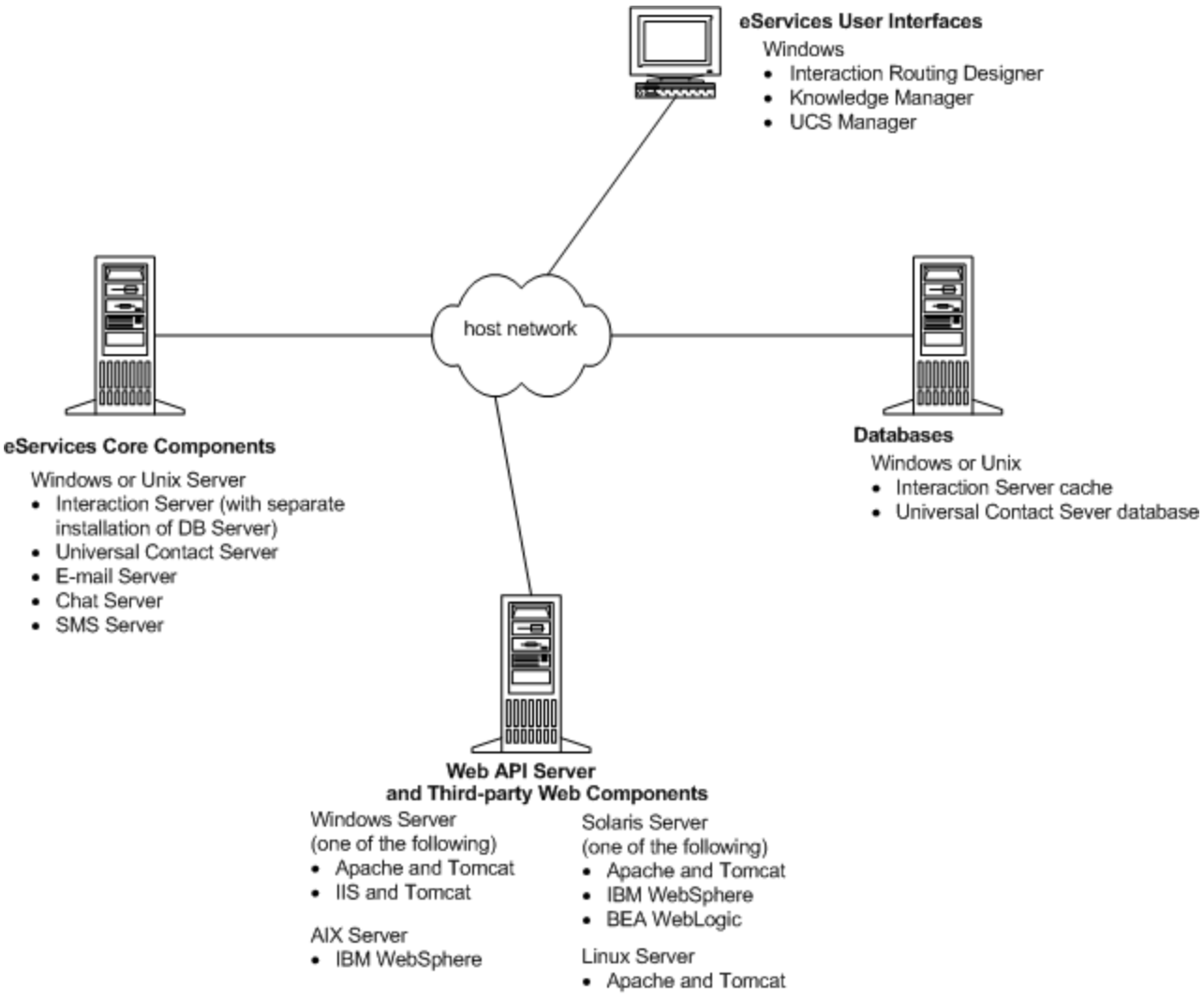
The Deploying on Windows and Deploying on UNIX sections, describe deployment in detail. Before beginning the deployment process there are several things you can think about and do to make the process easier.

Overview

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 section provides general information on deployment planning. For further information about specific issues, see also the eServices Administrator's Guide.

Component Distribution

Genesys recommends that you divide eServices and associated components among several host machines.



See this note on Web API Server and Third-party Web Components.

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

- Keep web servers for different applications on separate machines.
- Install eServices graphical user interface (GUI) applications on Windows hosts only. These GUI applications include:
  - Knowledge Manager
  - Universal Contact Server Manager
  - Interactions 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

Information about distributing eServices components across multiple hosts for load-balancing purposes is available.

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

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

Deployment Tasks

This section guides you through the various actions you must take to plan, deploy, and test your eServices solution for both Windows, and UNIX.

Deployment Summary

Windows

Windows Deployment Tasks

This section 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.

Plan your deployment

1. Choose which machines will host the eServices and various third-party components. For a sample architecture layout, see Component Distribution.
2. Choose which machines will host your databases. To help you calculate the rough size of your databases, use the formulas in Estimating Database Size.
3. Print out and complete the Configuration Worksheet with the values that fit your deployment.

Complete all prerequisite actions

1. Verify that all Prerequisites components are in place.
2. Create the Host.
3. For deployments across multiple machines, Create a Shared Directory.
4. Create UCS and Interaction Server databases.

Note on third-party software for processing Web-based interactions

Although Web API Server is not updated in the 8.5.0 release, you can use the 8.1.2 version with eServices 8.5.0.

Web API Server requires a web server and an application container; deployment of some common combinations of those items is described in the "Deploying Third-Party Web Applications" chapter of the eServices 8.1 Deployment Guide.

Configure eServices objects

1. Install the configuration wizards.
2. Create sample workflows for routing simple test interactions through the contact center to an agent.
3. Configure the eServices objects.
4. Configure Chat Server to communicate with Web API Server.
5. Create configuration objects and sample workflows for routing web callback interactions through the contact center to an agent. See Install Web Callback.

Install eServices components

Launch the Integrated Installation for all eServices components

> **Important**
>
> For deployments across multiple hosts, you need to launch the Integrated Installation one time for each host.

Deploy those eServices components not included with the wizard (Optional)

Manually deploy the eServices components that are not included in the eServices Configuration Wizard installation.

Universal Contact Server Proxy or Interaction Server Proxy

1. Create the UCS Proxy and Interaction Server Proxy application objects.
2. Install UCS Proxy and Interaction Server Proxy.
3. Configure the UCS Proxy and Interaction Server Proxy Desktop applications.

SMS Server

1. Create the SMS Server application object.
2. Install SMS Server.
3. Configure the eServices components.

Verify Connections

1. Double-check that the connections you made with the configuration wizard are in place. For a list of necessary connections, see Connections.
2. Using Genesys Administrator Extension, add a connection from Stat Server to Interaction Server (or to Interaction Server Proxy). Stat Server uses this connection to find and route interactions to available eServices agents.

For more information, see Verifying connections.

Modify the database

Run the database scripts to modify the database to process interactions.

Test the Components

1. Test the eServices servers.
2. Test the components using the Web-based TestTool.
3. Test the Genesys Desktop.

Test the Setup

1. Configure ABC Simple BP for routing e-mail interactions.
2. Configure ABC Simple Chat BP for routing chat interactions.

3. Send an e-mail to the Desktop.
4. Start a chat session.
5. Send a web form e-mail.
6. Configure WebCallback BP for routing web callback interactions.
7. Update the Interaction Server database for processing web callback interactions.
8. Send a web callback.

UNIX

1. Prepare your environment
   a. 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.
   b. Create your databases.
2. Launch the eServices Configuration Wizard from a Windows host to configure the components that will run in the Solaris, Linux, or AIX environments.
   a. Install the Configuration Wizards.
   b. Install the Interaction workflow samples.
   c. Configure the objects.
3. Run the installation package for each remaining component:
   `<component>\<platform>`
4. (Optional) Manually deploy the eServices components that are not included in the eServices Configuration Wizard installation.
   - Universal Contact Server Proxy or Interaction Server Proxy
     a. Install UCS Proxy.
     b. Install Interaction Server Proxy.
   - SMS Server: Install SMS Server.

Using the eServices Configuration 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.

This section summarizes the flow (including installing the wizards and workflow samples) 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.

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.

Configuration Wizards

Using the Wizard to Configure your eServices Components

Installing the Configuration Wizards

Prerequisites

- Interaction Management CD.
- Start the Configuration Layer.

1. Install the eServices Configuration Wizards and Genesys Wizard Manager.
   a. In the root directory of the Interaction Management CD, find `Setup.exe`.
   b. 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.

Installing and Starting the Workflow Samples

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.

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 8.1.3 > Interaction Workflow Samples > Start Interaction Workflow Samples**. This launches the Interaction Workflow Samples Wizard.
   b. Login to the wizard 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 a file "collision", since the source and target files are then identical.

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

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 or Genesys Administrator.

Launching the Configuration Wizard

To launch the Genesys Wizard Manager:

1. From the Windows taskbar, select **Start > Programs > Genesys Solutions > eServices 8.1.0 > eServices Configuration Wizards > Start Wizard Manager**.
2. Click **Log into the Configuration Layer**. This opens the main **Genesys Wizard Manager** window.
3. Select **Multimedia** from the menu on the left side of the window, and then select **Deploy Multimedia Solution in your contact center**. The **eServices Configuration Wizard Welcome** page opens.
4. Click **Next** to begin configuring eServices objects.

Naming your Solution

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

```
Name = ES85
```

Selecting the Configuration Type

Select Simple or Custom Configuration

1. At the **Configuration Process Selection** page, select one of the following:
   - Simple single-host configuration
   - Custom Configuration

> **Important**
> For this model setup, select **Custom Configuration**.

2. Click **Next**.

> **Important**
> 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.

Copying the IP to your shared directory

1. At the **Installation Package** page, click **Have Disk** and navigate to the Interaction Management CD.
2. To select a destination for the package, click **Browse** and navigate to your shared directory).
3. Click **Next**.

> **Important**
> 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.

4. At the **Installation Ready** page, confirm that your software is ready for installation and click **Next**.

Selecting or Adding Message Server

1. At the **Solution Components: Message Server** page, select or add a Message Server.
2. Click **Next**.

Creating the Component 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.

1. At the **Solution Components: Data Access Point** page, click **Add**.
2. At the **Browse for Application** page, click the **New Application** icon.
3. Create a new DAP.
4. Enter your Database Access Point information:

Component DAP	Application Name	DB Server Name	JDBC Connection	Database Information	Case Conversion
Interaction Server	ES85_Ixn_DAP	ES85_IxnDBServer	Clear the check box **Note:** It is important to clear the Enable JDBC access box. Failing to do so can lead to a configuration problem.	• DBMS Type = mssql • DBMS Name = ESHost • Database Name/SID = IxnDB • User Name = sa • Password = <password for user sa>	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.
Universal Contact Server	ES85_UCS_DAP	[NONE] **Note:** UCS connects to its database directly through JDBC. You	Select the check box, and enter the following information: • Host = ESHost • Port (for JDBC) = 1433 (Microsoft	• DBMS Type = mssql • DBMS Name ="" **Note:** If you do not clear the	any

Component DAP	Application Name	DB Server Name	JDBC Connection	Database Information	Case Conversion
		do not need to create a DB Server Application object for it.	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.	DBMS Name box, your UCS will not work properly. • Database Name/SID = Customer • User Name = sa • Password = <password for user sa>	

5. Click **Finish** to complete the Database Access Point Wizard and return to the eServices Configuration Wizard.
6. In the **Browse for Application:** page, select the Database Access Point that was just configured and click **OK**.

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

In Configuration Manager or Genesys Administrator, create a `settings` section on the **Options** tab. Create a new option, setting the option name as `prepare` and the option value to **false**.

Additional details are provided in the eServices Administrator's Guide.

Adding DB Server

1. Click the folder icon beside DB Server, and then click the **New Application** icon to run the DB Server Wizard.
2. Follow the wizard's directions and enter the following information:

Application Name	Host	Default Port	DBMS Type
ES85_IxnDBServer	ESHost	6110	mssql

3. 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**.
4. At the **Installation Ready** page, confirm that your DB Server software is ready for installation, and then click **Next**.
5. At the **Listening Ports and Transport Layer Security (TLS) Settings** page, click **Next**.

> **Important**
>
> For this model setup, we do not require any additional ports. If you want to install additional ports, you can do so in Configuration Manager or Genesys Administrator later.

6. At the **Log Configuration** page, take the defaults. Click **Next**.
7. At the **Backup Server Information** page, because no backup servers are configured, clear the checkbox and click **Next**.
8. Click **Finish** to exit the DB Server Wizard. The Database Access Point Wizard will resume.

Adding UCS and Interaction Server

1. At the **Solution Components:** page, select your component, and Click **Add**.
2. At the **Browse for Application :** page, click the **New Application** icon.

3. Follow the wizard's direction, and enter the following information:

Application Name	Type	Host	Default Port	API Port	Connections	License Connection	Login Account
ES85_UCS	N/A	ESHost	6120	Accept the default port value, or enter a port number where UCS should listen for third-party protocol connections.	• Message Server • DAP: ES85_UCS_DAP	N/A	Select the user account (Configuration Layer Person object) or Access Group that UCS is to use to log in to the Configuration Layer. UCS uses the Configuration Layer to pass some of its information back and forth to certain components. The selected account or access group must have write access to the

Application Name	Type	Host	Default Port	API Port	Connections	License Connection	Login Account
							tenant in use. Refer to Configuration Server access permissions for more information.
ES85_IxnSrv	Select `New-style`. • The eServices Configuration Wizard automatically creates a multimedia `Switch` object for you in the background. Continue at "Configure Framework Resources". • If your configuration does not include a properly configured `Multimedia` type `Switc`	ESHost	6130	N/A	• Servers for Third-Party Protocol • DAP: ES85_Ixn_DAP. If you already installed related eServices components (Universal Contact Server, E-mail Server, and Classification Server, for instance), you can connect to them now	• License Server Host = ESHost • License Server Port = 7260 • Specify the number of licenses for Interaction Server features	N/A

Application Name	Type	Host	Default Port	API Port	Connections	License Connection	Login Account
	h object, continue at "Create a Multimedia Switch object (Switch-based Interaction Server only)"				using this page.		

4. At the **Log Configuration:** page, accept the defaults and click **Next**.
5. At the **Installation Package:** page,
   - Click **Have Disk**, navigate to the Interaction Management CD, and then click **OK**.
   - Click **Browse**, navigate to your shared directory, and then click **Next**.
6. At the **Installation Ready:** page, click **Next**.
7. Click **Finish** to exit the Wizard and return to the eServices Configuration Wizard.
8. At the **Browse for Application:** page, select the the server and click **OK**.
9. Click **Next**.

Creating a Multimedia Switch Object (Switch-based Interaction Server only)

> **Important**
>
> Before attempting to run your eServices solution, check Configuration Manager or Genesys Administrator 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.

1. 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, it issues a warning. Click **Proceed** to launch the Switch Wizard.

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

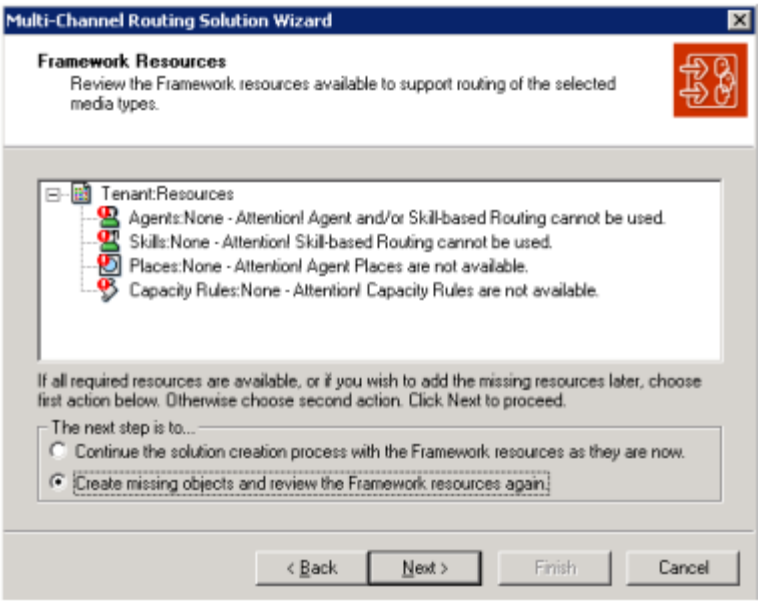
2. To create a new `Multimedia` type Switch object, provide a unique name for it and click **Next**. (The wizard allows you to create the new Switch only in folders designated to contain objects of type Switch.)
   Name = `ES85_Switch`

> **Important**
>
> 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.

3. At the **Switching Office:** page, click **Add** to create a new Switching Office object. Enter the following information:
   - **Name** = `ES85_Switching_Office`
   - **Switch Type** = `Multimedia Switch`
4. Click **Finish**.
5. At the **Switch Summary** page, review the details and click **Next**.

Configuring Framework Resources

1. If you select `New-Style Interaction Server`, the wizard opens the **Framework Resources**. This page lists the following:
   - Current Framework resources
   - Any missing resources required to support the media types served by your switch



2. To create the missing resources now, select **Create missing objects and review the Framework resources again**, and then click **Next**. For this 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**.
3. At the **Framework Objects** page, select the object type you want to create from the drop-down menu. For this model setup, create and configure the various objects as follows:

Object Type	Model Setup
Place	Create `Place` objects to facilitate the routing of multimedia interactions: • **Place Name** = `Agent1Place` **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.
Capacity Rule	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

Object Type	Model Setup
	Capacity Wizard, which is available on the Real-Time Metrics Engine CD).
Agent	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). • **Agent Name** = `Agent One` • **Employee ID** = `Agent1` • **User Name** = `Agent1` • **Password** = `Agent1` **Note:** The agent values supplied here are for general use in this 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.
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

Object Type	Model Setup
	numbering of your switch.
Skill	You do not need to create skills for this model installation.

Selecting a Stat Server

At the **Solution Components:** page, select **Stat Server:**. Click **Add** and select a Stat Server. This connects all the appropriate components in your eServices solution to this Stat Server.

> **Important**
>
> 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.

If you receive a warning message, it indicates that Stat Server does not yet have a connection to Interaction Server (or to Interaction Server Proxy). 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.

Adding the Other Components

1. At the **Solution Components:** page, select your component, and Click **Add**.
2. At the **Browse for Application :** page, click the **New Application** icon.
3. Follow the wizard's directions, and enter the following information:

Component	Application Name	Host	Default Port	Connections	Other Details
Universal Routing Server (URS)	Select the URS that you created and installed as one of the				

Component	Application Name	Host	Default Port	Connections	Other Details
	prerequisites for your eServices installation. If you receive a warning but stating that URS is not configured to monitor Interaction Server, click **Yes** to reconfigure now, or click **No** to reconfigure later. If you click **No**, make sure you add the required connections after configuration and installation are complete.				
Application Cluster	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, and E-mail Server, as well as other application clusters.				

Component	Application Name	Host	Default Port	Connections	Other Details
Web API Server					Although it may appear in the Wizard, this component is not included in the 8.5.0 release.
Universal Contact Server Manager	ES85_UCSMgr			Universal Contact Server	
E-mail Server	ES85_E-mailServer	ESHost	6150	• Message Server • Interaction Server • UCS	• POP3 server name = ESHost • POP server type = POP3 • SMTP server name = ESHost **Note:** This model installation uses ESHost 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

Component	Application Name	Host	Default Port	Connections	Other Details
					need to enter a value for the SMTP server name if it is the same server that you used for POP3 server name. • Address = <mailboxName>@ESHost • User name = <username> • Password = <password> • E-mail address default domain name = ESHost • External Agent e-mail address = external@ESHost • Enable Web Form Processing by selecting the check box, and provide a port where the Web API Server should listen for

Component	Application Name	Host	Default Port	Connections	Other Details
					requests—for example IWF Processing Port = 7777. • Enter the e-mail address that is used to supply field values in automated responses generated in the contact center. Select an existing Person object from the Configuration Layer: Auto-reply Agent = Agent1 • 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

Component	Application Name	Host	Default Port	Connections	Other Details
					1 = <mailboxName1>@ES • Customer access account 2 = <mailboxName2>@ES **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

Component	Application Name	Host	Default Port	Connections	Other Details
					before running the wizard, then no e-mail accounts are shown. You must add and configure at least one e-mail account before continuing.
Classification Server	ES85_ClassificationServer	ESHost	6160	• Message Server • UCS	• License Server Host = ESHost • License Server Port = 7260
Training Server	ES85_TrainingServer	ESHost	6170	• Message Server • UCS	
Knowledge Manager	ES85_KnowledgeManager			• Message Server • UCS	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

Component	Application Name	Host	Default Port	Connections	Other Details
					box.
Chat Server	ES85_ChatServer	ESHost	6180	• Message Server • Interaction Server • UCS	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.

4. At the **Log Configuration:** page, accept the defaults and click **Next**.
5. At the **Installation Package:** page,
   - Click **Have Disk**, navigate to the Interaction Management CD, and then click **OK**.
   - Click **Browse**, navigate to your shared directory, and then click **Next**.
6. At the **Installation Ready:** page, click **Next**.
7. Click **Finish** to exit the Wizard and return to the eServices Configuration Wizard.
8. At the **Browse for Application:** page, select the the server and click **OK**.
9. Click **Next**.

Creating Capacity Rules

1. At the **Resource Capacity Rules Deployment:** page, select a folder in the Configuration Server where eServices should store its resource capacity rules. Click **Next**.
2. At the **Resource Capacity Rules Selection:** page, 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**.

Completing the Wizard

1. At the **Solution Summary:** page, review 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.
2. At the **Completing the eServices Solution Wizard:** page, click **Finish** to confirm your solution configuration. Remember that you can change configuration details at a later time as well.

> **Important**
>
> 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.

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.

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

Interaction Server's Event Logger functionality requires its own separate database.

The following procedures describe how to estimate the amount of disk space require for the eServices databases.

Estimating UCS database size for e-mail interactions

Estimating UCS database size for e-mail interactions

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 x 90 x 25 KB or 225 MB of disk space.

> **Important**
>
> 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.

Estimating UCS database size for chat interactions

Estimating UCS database size for chat interactions

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 x 90 x 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.

Estimating the size of the Interaction Server database

Estimating the size of the Interaction Server database

1. Use the following formula:
   $(3.5 + d)m = x$ 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. Further information about the Interaction Server database is available in the eServices Administrator'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. 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.

Configuration Worksheet

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

Use the worksheet 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).
  - eServices requires a Person or Access Group object that UCS can use to log in to Configuration Server.

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 Deploying eServices on Windows.

Completing the Worksheet

Completing the Configuration Worksheet

1. Print out the worksheet.
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.

Next Steps

- Continue with Deploying eServices on Windows.

Configuration Worksheet

Configuration Worksheet

Field	Your Value	Model Setup Value	Description
Log In to Wizard Manager			
User name			User name for logging into Configuration Server.
User password			Password for logging into Configuration Server.
Application			Application object name—for example, `default`.
Host			Host of Configuration Server
Port			Port of Configuration Server host where Configuration Server listens for queries—for example, 2020.
Solution			
Name		ES85	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		ES85_Inx_DAP	Invent a name for

Field	Your Value	Model Setup Value	Description
			the DAP to use with Interaction Server. If you intend to use the Event Logger or message queue functionality you will need additional DAPs.
DB Server			
Name		ES85_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		ESHost	Host where you want DB Server installed.
Port		6110	Port where DB Server listens for queries.
SNMP Port		N/A	Port where DB Server listens for management commands (optional).
DBMS name		ESHost	Name of your database management system.
Database name		InxDB	Name of the database tablespace for Interaction Server.
User name		sa	User name for logging into the DBMS.
Password		<password for user sa>	Password for logging into the DBMS.
Universal Contact Server (UCS) Database Objects			
DB Server			

Field	Your Value	Model Setup Value	Description
Not needed because UCS used JDBC to connect to the database.			
Database Access Point (DAP)			
DAP name		ES85_UCS_DAP	Invent a name for the DAP to use with UCS.
Host		ESHost	Host name where the database s 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	Username for logging into the UCS DBMS.
Password		<password for user sa>	Password for logging into the UCS DBMS.
Universal Contact Server			
Name		ES85_UCS	Invent a name for the UCS application object.
Host		ESHost	Host where you want UCS installed.
Port		6120	Listening port for requests.
Universal Contact Server API port		Default port value	Port for Remote Method Invocation (RMI) connection to the UCS API. Must be different from stander server port.
Login account		N/A	The User Account (Person) or Access Group that UCS will use to log in to

Field	Your Value	Model Setup Value	Description
			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.
Switching Office			
Name		In the model setup, New-Style Interaction Server is selected 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.
Multimedia Switch			
Name		In the model setup, New-Style Interaction Server is selected which creates a Multimedia switch and switching office automatically in the background.	Select or create a switching office.
Interaction Server			
Name		ES85_InxSrv	Invent a name for your Interaction Server application object.
Host		ESHost	Host where you want Interaction Server installed.
Port		6130	Port where Interaction Server listens for queries.
Stat Server			
Name		In this model setup, an existing Stat Server is selected.	Select or create a Stat Server.
Host			Host where you want Stat Server installed.

Field	Your Value	Model Setup Value	Description
Port			Port where Stat Server listens for queries.
Application Cluster			
Name		N/A	Invent a name for your Application Cluster.
Web API Server			
Not included in eServices 8.5.0. However you can use Web API Server 8.1.2 with eServices 8.5.0.			
UCS Manager			
Name		ES85_UCSMgr	Invent a name for your UCS Manager application object.
E-Mail Server (for Genesys E-Mail)			
Name		ES85_E-mailServer	Invent a name for your E-mail Server application object.
Host		ESHost	Host where you want E-mail Server installed.
Port		6150	listening port for requests.
POP server		ESHost	The name of your corporate POP server.
SMTP server		ESHost	The name of your corporate SMTP server if different from the POP server.
POP3 user name and password		N/A	Address, user name and password for logging on to your POP server.
Default e-mail address domain		ESHost	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@ESHost	Address to be used as the From address in messages to external agents and as the To address in messages from

Field	Your Value	Model Setup Value	Description
			external agents. Typically, this is different from your general corporate e-mail addresses.
Integrated Web For 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 Customer Care`.
E-mail account(s) for customer access		<mailboxName1>@ESHost <mailboxName2>@ESHost	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 the do not exist.
Classification Server			
Name		ES85_ClassificationServer	Invent a name for your Classification Server application object.
Host		ESHost	Host where you want Classification Server installed.
Port		6160	Listening port for requests.
Training Server			
Name		ES85_TrainingServer	Invent a name for your Training Server application object.
Host		ESHost	Host where you want Training Server installed.

Field	Your Value	Model Setup Value	Description
Port		6170	Listening port for requests.
Knowledge Manager			
Name		ES85_KnowledgeManager	Invent a name for your Knowledge Manager application object.
Chat Server (for Genesys Chat)			
Name		ES85_ChatServer	Invent a name for your Chat Server application object.
Host		ESHost	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.
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.

Connections Table

The following table 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. In addition to the information in the table, 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 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).

> **Important**
>
> It is possible to have multiple connections to some components. In the table, these components are marked with a *.

Application	Connects to
Chat Server	Interaction Server Message Server Universal Contact Server
Classification Server	Message Server Universal Contact Server
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
SMS Server	Message Server Chat Server* Interaction Server Solution Control Server

Application	Connects to
Training Server	Message Server Universal Contact Server
Universal Contact Server	Message Server Stat Server* UCS DAPs
Universal Contact Manager	Universal Contact Server
Web API Server	Not included in release 8.5.0

Web API Server

8.5.x

Web API Server was a component of eServices through the 8.1.x release.

In the 8.5 release, it is treated as a separate product, with its own documentation.

8.1.x

You can use Web API Server 8.1.2 with eServices 8.5.

> **Important**
> Web API Server 8.1.2 does not work with Stat Server 8.1.1 and above.

Deployment of Web API Server, and of the web application and application container that it requires, is described in the eServices 8.1 Deployment Guide.

Third-Party Applications

Web Server and Application Container

Although Web API Server is not included in the 8.5 eServices release, Web API Server 8.1.2 can be used with eServices 8.5.

Web API Server requires a web server and a web application container. Information on installing and configuring common combinations of these applications can be found in the 8.1 version of the eServices Deployment Guide.

JDK and JRE

Java Software Development Kit (JDK) or Java Runtime Environment (JRE) is required for various eServices functionalities. To deploy these, determine the correct version for your system by consulting the Genesys Supported Operating Environment Reference Guide. Then,

- For Solaris and Linux, download the appropriate JRE from the Oracle website.
- For Windows, download the appropriate JDK from the Oracle website.
- For AIX, download the appropriate JRE from the IBM website.

Deploying eServices in Windows Environments

This section describes how to install the eServices components in your Windows environment.

> **Tip**
> In previous eServices releases, some components required Java Environment and Libraries for eServices and UCS (JELEU), which was supplied as part of eServices. Starting with the 8.5 release, JELEU is no longer required.

For your own deployment, be sure to use the Configuration Worksheet to help you plan the specifics of your solution.

Deployment Steps

Prerequisites

The following table describes the prerequisite components required before you install your eServices solution.

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 email server
Java Environment	Java Runtime Environment (JRE). For supported versions, see the Genesys Supported Operating Environment Reference Guide.
Database Server	For supported databases, see the Genesys Supported Operating Environment Reference Guide.
Licence Server	License Server 9.5; License File: 7260@ESHost.eServices 8.1 also supports FLEXNet Publisher v11.9 License Manager.
Management Framework Components	See the Management Framework documentation for information about installing and configuring the components.
Stat Server	See the Stat Server documentation for installation

Type of Component	Component Used
	and configuration instructions.
Universal Routing Server	See the Universal Routing documentation for installation and configuration instructions.

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 the 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. 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.

> **Important**
>
> 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.

Install and Start Interaction Workflow Samples

Install and run this component first to create various configuration objects that you will need when installing the rest of eServices.

Post Deployment Steps

Finishing the eServices Deployment on Windows

Creating Databases and Running Scripts

If you have not already done so, create the databases for Interaction Server and Universal Contact 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.

Running the Database Scripts

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

- Interaction Server—`Script`
- Universal Contact Server—`sql-scripts`

2. Run the database scripts for each component.

- UCS has two scripts. First run `ucs_drop_<dbname>.sql`; this deletes any existing tables and indexes in the UCS database. Then run `ucs_<dbname>.sql`.

> **Important**
> Only one Universal Contact Server database pair (Main and Archive) is allowed per Tenant.

- Interaction Server has a single script file.

Verifying Connections

At this point in the eServices installation process, you must update the configuration settings for some components before continuing. See the Connections Table for a list of connections that must be set in the Configuration Layer. Refer to the eServices Options Reference for information on option settings.

1. In Configuration Manager or Genesys Administrator, verify that the required connections have been made in the eServices Application objects.
2. Add a connection from Stat Server to Interaction Server, or to Interaction Server Proxy (see "Interaction Server Proxy as Reporting Bus").
3. Add a connection from Universal Routing Server to Interaction Server.

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.
- Wherever this Guide speaks of connecting Stat Server to Interaction Server, instead make the connection to Interaction Server Proxy

Testing eServices Components and the Setup

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

Next you should test your installation's functionality. This requires that you implement an interaction workflow. The workflow and its associated routing strategies (Business Processes) 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.

Components

Testing the Components

Testing the eServices Servers

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.

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. Web Server and Web Application Container for Web API Server launch
2. After each server starts, check its console window for errors.

Testing the components using the Web-based TestTool812

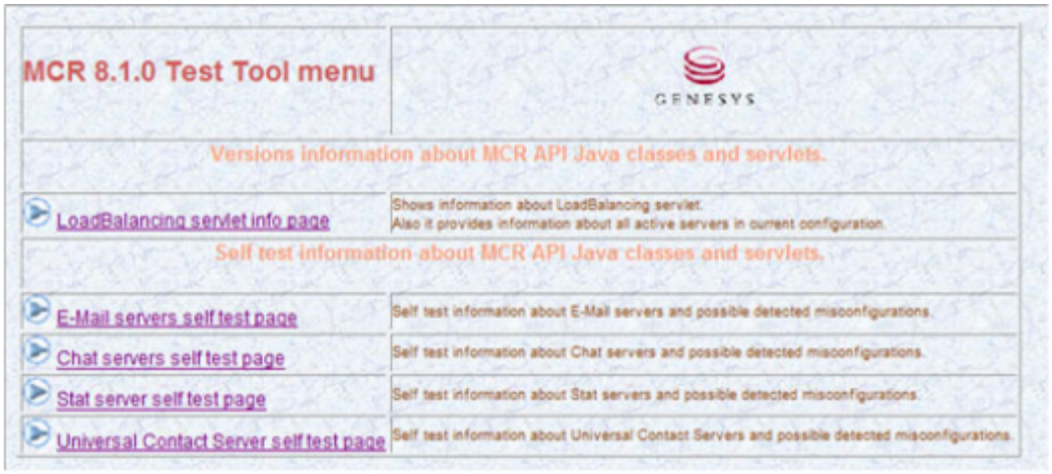
An application for testing the components, called TestTool812, is included as part of the installation of Web API Server. Use this tool to test the various Java classes and servlets that run the eServices solution.

> **Important**
>
> TestTool812 supports Internet Explorer. It does not support Mozilla Firefox.

1. Open a web browser window and enter the following URL: `http://<WebAPIServerhost>/WebAPI812/TestTool812`

   > **Important**
   >
   > The address used to access the testing application is case sensitive.

   A window similar to the following appears:



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.

Testing that the desktop can handle interactions

Test agent handling of interactions by configuring and installing Workspace Desktop Edition.

Setup

Testing the Setup

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 8.1 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.

Configuring ABC Simple BP for routing e-mail interactions

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`.

> **Important**
>
> 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 or Genesys Administrator. 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`.

Configuring ABC Simple Chat BP for routing chat interactions

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 or Genesys Administrator. 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`.

Sending a test e-mail to a Desktop agent

1. If you have not already started the necessary components, 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.

Starting a test chat session with a Desktop agent

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, 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>/WebAPI812`.
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**. Look for the arrival of the chat interaction at the Genesys Desktop.
8. 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.
   - WebAPI812 parameters, contained in the `constants.jsp` file.

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

1. If you have not already started the necessary component, do so now.
2. Open the Genesys MCR 8.1 Platform SDK Java Samples Pages by opening a browser navigating to `http://<WebAPIServerhost>/WebAPI812`.
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.

Configuring WebCallback BP for routing web callback interactions

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 or Genesys Administrator. That queue is also the value for the Web API Server `wcbNewQueue` (in the `\...\SimpleSamples812\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 IRD.
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

Updating the Interaction Server database for processing web callback interactions

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

Sending a test web callback to a Desktop agent

1. If you have not already started the necessary components, do so now.
2. Open the Genesys MCR 8.1 Platform SDK Java Samples Pages by opening a browser and navigating to `http://<WebAPIServerhost>/WebAPI812`.
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.

Building Strategies

For More Information

See the Universal Routing 8.1 Business Process User's Guide and Universal Routing 8.1 Interaction Routing Designer Help Zip for additional instructions on creating and activating/loading routing strategies.

Deploying eServices in UNIX Environments

This topic 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. It includes the following:

> **Important**
>
> You can deploy all eServices components on UNIX platforms with the exception of the following user interfaces:
>
> - Knowledge Manager
> - Universal Contact Server Manager
> - Interaction Workflow Samples

Deploying eServices on UNIX Hosts

Configuring and installing eServices on Solaris, AIX, or Linux follows the general procedure introduced in Model Configuration and Installation on Windows. You can use the eServices 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 Model Configuration and Installation on Windows: "Installing eServices Components."

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 (the 8.5 Framework Deployment Guide does not discuss configuration wizards).

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.

> **Tip**
> In previous eServices releases, some components required Java Environment and Libraries for eServices and UCS (JELEU), which was supplied as part of eServices. Starting with the 8.5 release, JELEU is no longer required.

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.

1. 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 Model Configuration and Installation on Windows, in the topic: Create a Shared Directory 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.
2. On a Windows machine, run the eServices Configuration Wizards as described in Model Configuration

and Installation on Windows, in the topic: Model Configuration and Installation on Windows. The Wizards copy the installation packages using a directory structure such as the following:

```
<Component A>
                IBM_AIX
                Linux
                Solaris
                Windows_2008


<Component B>
                IBM_AIX
                Solaris
                Windows_2008

...
aix
                ThirdPartyComponents


linux
                ThirdPartyComponents


solaris
                ThirdPartyComponents


windows
                CommonWizardComponentSet
                ThirdPartyComponents
```

3. 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.
4. 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.

Compatibility Package for Red Hat Linux

On some Red Hat Linux platforms, eServices 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.

> **Important**
>
> 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 the 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 at:

```
http://rpm.pbone.net/index.php3/stat/4/idpl/3416184/com/compat-libstdc+
+-33-3.2.3-61.i386.rpm.html
```

Important

Refer to the *Genesys Supported Operating Environment Reference Guide* for more information.

Manual Installation of eServices Components

This section provides instructions on manually deploying eServices. `

Creating the Application Object

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.
6. Click **Apply**.

Installing eServices Components on Windows

Prerequisites

- The Application object has been configured.
- The product CD for the component you are installing.

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, enable **Use Client Side Port**.
5. If you are not configuring client-side port functionality, click **Next** and continue with step 7.

> **Important**
> 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. For client-side port configuration, 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.

> **Important**
> 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.

7. Select the appropriate Application object from the list.
8. Click **Install**.

Installing eServices Components on AIX, Solaris, or Linux

Prerequisites

- The Application object has been configured.
- The product CD for the component you are installing.

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 to step 8.

> **Important**
>
> 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.

> **Important**
>
> On some Red Hat Linux platforms, eServices 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 for information about correcting this issue.

Specifics

Some components require procedures beyond the general directions on this page.

- Interaction Server, including clusters.
- Universal Contact Server (UCS).

Manual Deployment of UCS

This chapter describes procedures for manual configuration and installation of UCS and UCS Manager.

UCS Deployment

Creating or Editing the UCS 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.

1. Create the main and (optional) archive databases if you have not already done so. See also Creating Databases and Running Scripts.
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.

Creating or Editing the UCS Manager Application Object

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**.

Installing UCS on Windows

Prerequisites

- The UCS Application object has been configured.
- The Interaction Management CD.

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**.

Installing UCS on AIX, Solaris, or Linux

Prerequisites

- The UCS Application object has been configured.
- The Interaction Management CD.

1. Locate the `install.sh` file for Universal Contact Server.
2. Start the installation script with the command `>install.sh`.
3. 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.
8. Answer other questions, if required.

Installing UCS Manager

Prerequisites

- The Universal Contact Server Manager Application object has been configured.
- The Interaction Management CD.

1. Locate and double-click `Setup.exe` for UCS Manager on the Interaction Management CD.
2. Click **Next**.
3. Enter or browse to the location of the destination folder. Click **Next**.
4. Select JDK version in the list. Click **Next**.
5. Click **Install**.
6. Click **Finish**.

Manual Deployment for UCS Proxy, Interaction Server Proxy, and SMS Server

This section describes the manual deployment of UCS Proxy, Interaction Server Proxy, and SMS Server.

Introduction

Large numbers of custom desktop (ESP client) connections to Interaction Server and UCS may give rise to performance issues.

Important

- 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.

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.

Important

- 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:

- Deploying eServices on Windows for a typical eServices Solution running on Windows.
- Deploying eServices on UNIX for a sample deployment of an eServices Solution running on UNIX.
- Manual Deployment of eServices Components, for environments that are not using all eServices components, but that do require Universal Contact Server, such as Voice Callback.

Deploying UCS Proxy, Interaction Server Proxy and SMS Server

This section describes how to manually install UCS Proxy, Interaction Server Proxy and SMS Server.

UCS Proxy

UCS Proxy Deployment

Creating the Application Object

Prerequisites

- The Interaction Management CD.

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 UCS Proxy.
   b. Create a new Application object based on the template.
3. Open the **Properties** dialog box of the Application object.
4. On the **Server Info** tab:
   - In the **Host** text box, enter the name of the desired host.
   - In the **Communication Port** text box, enter the port the UCS Proxy will use for listening.

5. 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.
6. On the **Connections** tab, add connections to:
   - The primary Universal Contact Server (for UCS Proxy). This connection is mandatory.
   - Message Server (optional)
7. Click **Apply**.
8. If you would like to configure your UCS Proxy for HA, repeat this procedure for the second instance.

Installing UCS Proxy

Prerequisites

- The UCS Proxy Application object has been configured.
- The Interaction Management CD.

1. Locate the `Setup.exe` for UCS 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 UCS Proxy Application object from the list.
5. Click **Install**.

Installing UCS Proxy on AIX, Solaris, or Linux

Prerequisites

- The UCS Proxy Application object has been configured.
- The Interaction Management CD.
- Review the section Compatibility Package for Red Hat Linux.

1. Locate the `install.sh` file for UCS 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 a desired one.
7. Answer other questions, if required.

> **Important**
>
> On some Red Hat Linux platforms, eServices 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 for information about correcting this issue.

Configuring the Desktop Application to Use UCS Proxy

> **Important**
>
> UCS Proxy is 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 server.

Prerequisites

- Your desktop application is installed and configured.
- The UCS Proxy is installed and configured.

1. Login to Configuration Manager or Genesys Administrator.
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.
4. Add a connection to the UCS Proxy.
5. Click **Apply**

Interaction Server Proxy

Interaction Server Proxy Deployment

Creating the Application Object

Prerequisites

- The Interaction Management CD.

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 Interaction Server Proxy.
    b. Create a new Application object based on the template.
3. Open the **Properties** dialog box of the Application object.
4. On the **Server Info** tab:
    - In the **Host** text box, enter the name of the desired host.
    - In the **Communication Port** text box, enter the port the Interaction Server Proxy will use for listening.
5. 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.
6. On the **Connections** tab, add connections to:
    - The primary Interaction Server. This connection is mandatory.
    - E-mail Server. This connection is required in order to send outbound e-mails from Workspace Desktop Edition.
    - Message Server (optional)
7. Click **Apply**.
8. If you would like to configure your Interaction Server Proxy for HA, repeat this procedure for the second instance.

Installing Interaction Server Proxy

Prerequisites

- The Interaction ServerProxy Application object has been configured.
- The Interaction Management CD.

Procedure

1. Locate the `Setup.exe` for Interaction Server 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 Interaction Server Proxy Application object from the list.
5. Click **Install**.

Installing Interaction Server Proxy on AIX, Solaris, or Linux

Prerequisites

- The Interaction Server Proxy Application object has been configured.
- The Interaction Management CD.
- Review the section Compatibility Package for Red Hat Linux.

Procedure

1. Locate the `install.sh` file for Interaction Server 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 a desired one.

7. Answer other questions, if required.

> **Important**
>
> 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 for information about correcting this issue.

Configuring the Desktop Application to Use Interaction Server Proxy

> **Important**
>
> Interaction Server Proxy is 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 server.

Prerequisites

- Your desktop application is installed and configured.
- The Interaction Server Proxy is installed and configured.

Procedure

1. Login to Configuration Manager or Genesys Administrator.
2. Locate the desktop Application object and open its **Properties** dialog box.
3. On the **Connections** tab, remove the connection to the main Interaction Server.
4. Add a connection to the Interaction Server Proxy.
5. Click **Apply**

SMS Server

SMS Server Deployment

Creating the Application Object

Prerequisites

- The SMS Server CD.
- The following Java versions:
  - In release 8.1.300.14 and later,
    - Windows: JDK 1.7
    - UNIX: JRE 1.7
  - In release 8.1.3 prior to 8.1.300.14,
    - Windows: JDK 1.6 or JDK 1.7
    - 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

1. Create an Application object for the SMS Server if it does not already exist.
   a. Import the correct application template from the SMS Server CD.
   b. Create a new Application object based on the template.
3. Open the **Properties** dialog box of the Application object.
4. On the **Server Info** tab:
   - In the **Host** text box, enter the name of the desired host.
   - In the **Communication Port** text box, enter the port the SMS Server will use for listening.
5. 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.
6. On the **Connections** tab, add connections to:
   - The primary Interaction Server. This connection is mandatory.
   - Message Server (optional)
7. If this is for a multi-tenant environment, add the tenant(s) on the **Tenants** tab.
8. Click **Apply**.

Installing SMS Server on Windows

Prerequisites

- The SMS Server Application object has been configured.
- Java is installed on the host.
- The SMS Server CD.

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 the destination location.
8. Select JDK.
9. Click **Install**.

> **Important**
>
> Refer to the SMS Server Options Reference for additional information. You must properly configure the x-jsms-config-file option before using SMS Server.

Installing SMS Server on AIX, Solaris, or Linux

Prerequisites

- The SMS Server Application object has been configured.
- JRE x64, or x32 (depending on Operating System version) is installed.

> **Important**

- 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.

1. Locate the `install.sh` file for SMS Server.
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. Choose whether the application will use Client Side Port.
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.

Important

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 for information about correcting this issue.

Important

> Refer to the SMS Server Options Reference for additional information. You must properly configure the x-jsms-config-file option before using SMS Server. This option is required for MMS processing.

Configuring SMS Server for Chat Server (Optional)

SMS Server will work with Chat Server in session mode:

1. Add a connection to Solution Control Server.
2. Add a connection to one or more Chat Servers.

Configuring eServices Components for SMS Server

Prerequisites

- The SMS Server Application object exists.

1. In Configuration Manager or Genesys Administrator, locate the Interaction Server Application object and open its properties.
2. On the **Connections** tab, add a connection to SMS Server.
3. Click **Apply**.

Wizards

Using the eServices Configuration Wizards

For information about installing the eServices components using the wizards, see Using the eServices Configuration Wizards.

Deploying Java Components on Windows and UNIX

Windows

1. Verify that the Oracle JDK for Windows x64 of the correct version is installed. The correct versions are:
   - Version 1.6.0_26 or higher within the 1.6 family
   - Version 1.7
2. Run the component's IP (located in the Windows_x64 folder on the CD).
3. When you are prompted with a list of JDK choices, select the required version (1.6.0_26 or higher) of Oracle JDK.

UNIX

1. Verify that JRE x64 for UNIX is installed.
   - For Oracle JRE, the version must be (a) 1.6.0_26 or higher, within the 1.6 family, or (b) 1.7.
   - For IBM JRE (for AIX), the version must be (a) 1.6.0 build pap6460sr9fp2-20110627_03(SR9 FP2) or higher, within the 1.6 family, or (b) 1.7.
2. Run the component's UNIX IP. When prompted, enter the path to the Java executable (from JRE x64). Enter the path to the Java executable (from JREx64) when you are prompted to do so during the installation.

Transport Layer Security

eServices 8.5 supports Transport Layer Security (TLS) protocol to secure data exchange between components. The following tables 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 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.

TLS Support 1

TLS Support 1 Connections

Connection	Config Server	Config 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	Yes	Yes	N/A	Yes	Yes	Yes

Connection	Config Server	Config Server Proxy	Solution Control Server	Message Server	Interaction Server	Universal Contact Server
Server						
SMS Server	Yes	Yes	Yes	Yes	Yes	N/A
Social Messaging Server	Yes	Yes	N/A	Yes	Yes	Yes
Classification 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

TLS Support 2

TLS Support 2 Connections

TLS Support 2

Connection	E-mail Server	Chat Server	SMS Server	Social Messaging Server	Classification Server
Interaction Server	Yes	Yes	Yes	Yes	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	Yes		N/A	N/A
Social Messaging Server	N/A	Yes	N/A		N/A
Classification Server	N/A	N/A	N/A	N/A	

Connection	E-mail Server	Chat Server	SMS Server	Social Messaging Server	Classification Server
Web API Java	Yes	Yes	N/A	N/A	N/A
Web API .NET	Yes	Yes	N/A	N/A	N/A

TLS Support 3

TLS Support 3 Connections

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
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

Important

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 Definition

eServices supports client-side port definition between components as outlined in the following tables. 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 Security Deployment Guide for information about how to configure client-side port parameters.

Client-side Port Support 1

Client-side Port 1 Connections

Connection	Config Server	Config 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	Yes
Classification Server	Yes	Yes	N/A	Yes	N/A	No
Web API Java	Yes	Yes	Yes	Yes	No	No
Web API	Yes	Yes	Yes	No	No	No

Connection	Config Server	Config Server Proxy	Solution Control Server	Message Server	Interaction Server	Universal Contact Server
.NET						

Client-side Port Support 2

Client-side Port 2 Connections

TLS Support 2

Connection	UCS Proxy	E-mail Server	Chat Server	SMS Server	Social Messing 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	Yes		N/A	N/A
Social Messaging Server	N/A	N/A	Yes	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

Client-side Port Support 3

Client-side Port 3 Connections

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
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

> **Important**
> Training Server does not support the client-side port connection type.

Deploying an E-Mail System in Secured Mode

This section 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.

TLS/SSL

Configuring TLS/SSL for E-mail Server

This section describes procedures for configuring your E-mail Server application to work with TLS/SSL.

Generating the .truststore file

Prerequisites:

- The corporate e-mail server is configured to work in secured mode.

Steps:

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.

Modifying the E-mail Server startup command line on Windows

Prerequisites:

- The `.truststore` file has been created.

Steps:

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.

Modifying the E-mail Server startup command line on Unix

Prerequisites:

- The `.truststore` file has been created.

Steps:

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.

Configuring E-mail Server's POP, IMAP, and SMTP Ports

Prerequisites:

- The `.truststore` file has been generated and E-mail Server's startup command line has been modified.

Steps:

1. In Configuration Manager or Genesys Administrator, 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.
6. (Optional) If the application has already started, restart the application to apply the changes.

Corporate E-mail Server

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 <certicate_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

> **Important**
>
> - 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.

Multiple Interaction Servers in a Single Tenant

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.

> **Important**
>
> 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 Multiple Interaction Servers

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. For more information, see the "Last Step in Creating a Business Process" topic of Universal Routing 8.1 Interaction Routing Designer Help (control of access to other configuration objects 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:
    1. 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.
    2. 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.

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 section 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.

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 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.

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.

The following sections describe how to deploy an E-mail Solution using MIME.

Deploy MIME Solution

Deploying a MIME-compliant E-Mail Solution

Viewing a sample transformer

E-mail Server 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.

1. Open the directory where E-mail Server 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.

Creating a custom transformer

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.
- Is thread-safe.

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
   - Outgoing E-Mail Customization API
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.

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. In

Configuration Manager or Genesys Administrator, 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.

MIME Options

MIME Customization Options

The following table describes the options for the `[mime-customization]` Section of the E-mail Server application object.

Option Name	Value	Description
enable-inbound	Default Value: `false` Valid Values: `true`, `false`	If set to `true`, sends inbound e-mail messages to the specified inbound MIME transformer class.
enable-outbound	Default Value: `false` Valid Values: `true`, `false`	If set to `true`, sends outgoing e-mail messages to the specified outbound MIME transformer class.
enable-inbound-debug-log	Default Value: `false` Valid Values: `true`, `false`	If set to `true`, activates the inbound debug logger.
enable-outbound-debug-log	Default Value: `false` Valid Values: `true`, `false`	If set to `true`, activates the outbound debug logger.
inbound-class-name	Default Value: "" Valid Values: <any string>	Specifies the fully qualified name of the custom inbound transformer.
outbound-class-name	Default Value: "" Valid Values: <any string>	Specifies the fully qualified name of the custom outbound transformer.
inbound-keep-received-mime	Default Value: `false`	Controls the way E-mail Server saves the content of MIME

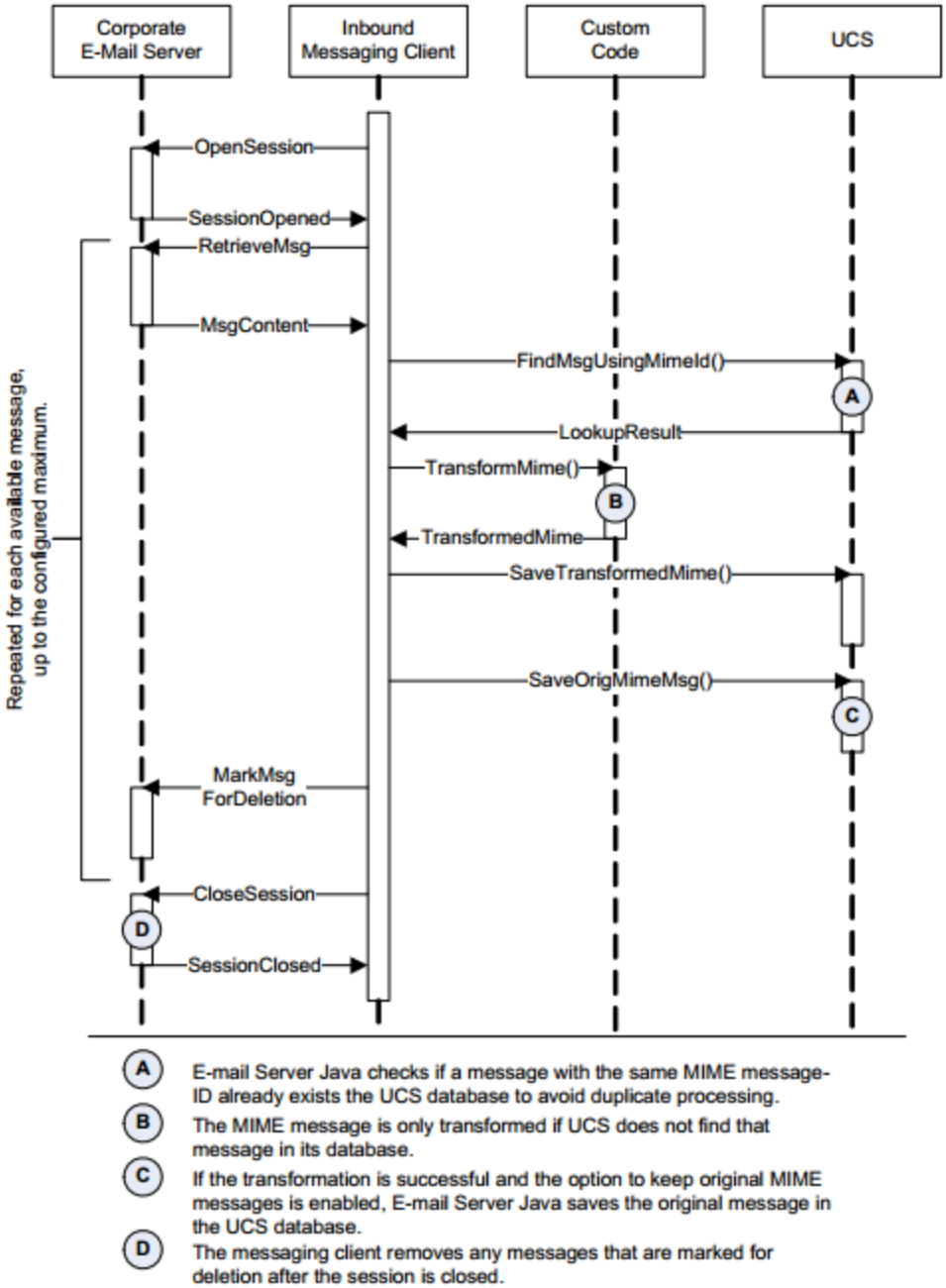
Option Name	Value	Description
	Valid Values: `true`, `false`	messages. If set to `true`, and if the message was transformed successfully, then saves the unmodified MIME content of incoming e-mails received in the UCS database (along with the transformed content).
outbound-keep-sent-mime	Default Value: `false` Valid Values: `true`, `false`	Controls the way E-mail Server saves the content of MIME messages. If set to `true`, 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 Flows

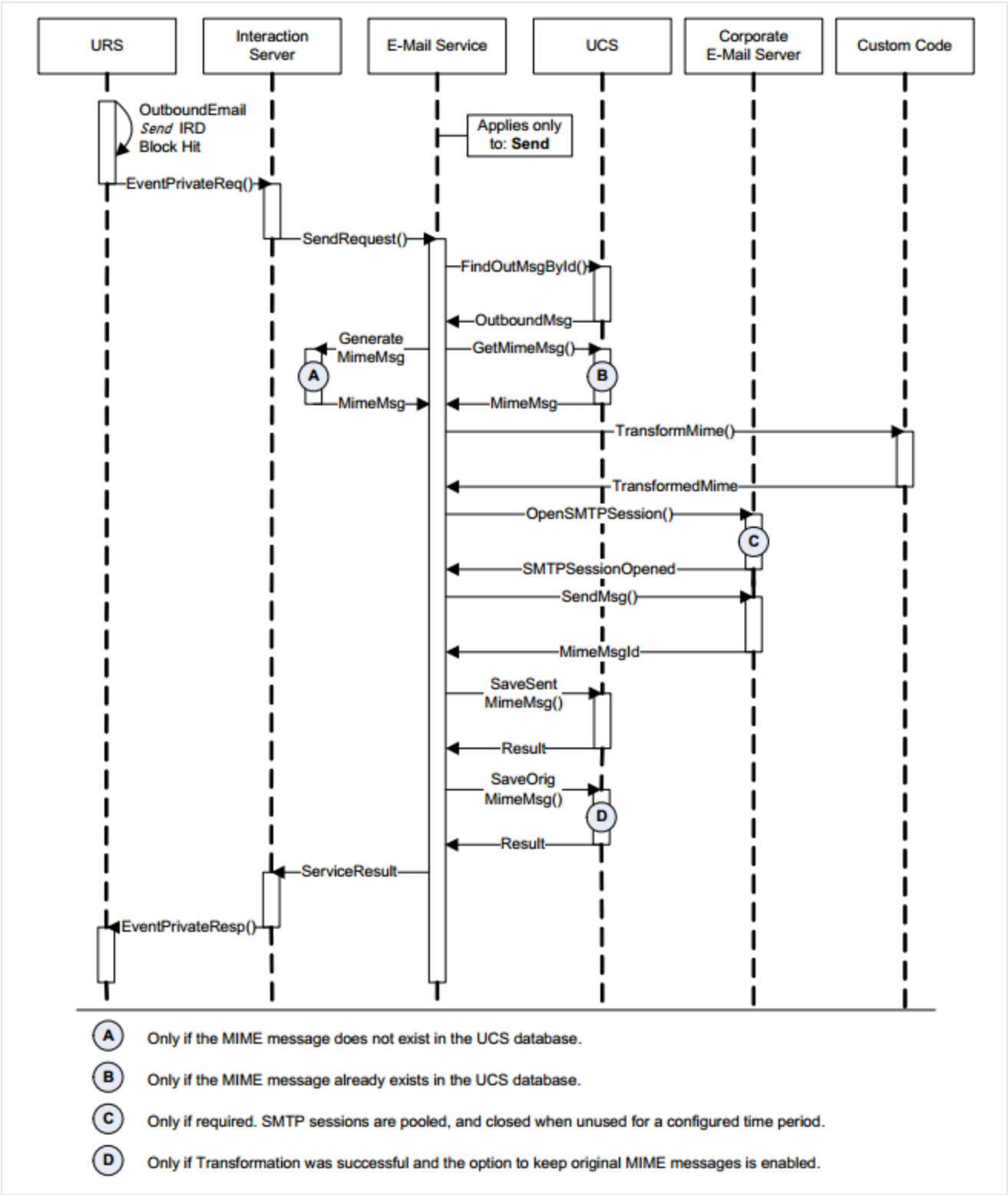
Message Flow Patterns

The following diagrams show the message patterns for e-mail messages.

Incoming E-Mail Message Flow



Outgoing E-Mail Message Flow


URS
Interaction Server
E-Mail Service
UCS
Corporate E-Mail Server
Custom Code
OutboundEmail Send IRD Block Hit
Applies only to: Send
EventPrivateReq()
SendRequest()
FindOutMsgById()
OutboundMsg
Generate MimeMsg
GetMimeMsg()
A
B
MimeMsg
MimeMsg
TransformMime()
TransformedMime
OpenSMTPSession()
C
SMTPSessionOpened
SendMsg()
MimeMsgId
SaveSent MimeMsg()
Result
SaveOrig MimeMsg()
D
Result
ServiceResult
EventPrivateResp()
A Only if the MIME message does not exist in the UCS database.
B Only if the MIME message already exists in the UCS database.
C Only if required. SMTP sessions are pooled, and closed when unused for a configured time period.
D Only if Transformation was successful and the option to keep original MIME messages is enabled.