



Universal Routing 8.0

Deployment Guide

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

Copyright © 2010 Genesys Telecommunications Laboratories, Inc. All rights reserved.

About Genesys

Alcatel-Lucent's Genesys solutions feature leading software that manages customer interactions over phone, Web, and mobile devices. The Genesys software suite handles customer conversations across multiple channels and resources—self-service, assisted-service, and proactive outreach—fulfilling customer requests and optimizing customer care goals while efficiently using resources. Genesys software directs more than 100 million customer interactions every day for 4000 companies and government agencies in 80 countries. These companies and agencies leverage their entire organization, from the contact center to the back office, while dynamically engaging their customers. Go to www.genesyslab.com for more information.

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

Notice

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

Your Responsibility for Your System's Security

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

Trademarks

Genesys, the Genesys logo, and T-Server are registered trademarks of Genesys Telecommunications Laboratories, Inc. All other trademarks and trade names referred to in this document are the property of other companies. The Crystal monospace font is used by permission of Software Renovation Corporation, www.SoftwareRenovation.com.

Technical Support from VARs

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

Technical Support from Genesys

If you have purchased support directly from Genesys, please contact Genesys Technical Support at the regional numbers provided on [page 17](#). For complete contact information and procedures, refer to the [Genesys Technical Support Guide](#).

Ordering and Licensing Information

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

Released by

Genesys Telecommunications Laboratories, Inc. www.genesyslab.com

Document Version: 80r_dep_10-2010_v8.0.101.00



Table of Contents

List of Procedures	11
Preface	13
	About Universal Routing.....	14
	CIM Platform.....	14
	Intended Audience.....	16
	Making Comments on This Document	17
	Contacting Genesys Technical Support.....	17
	Document Change History	18
	New in Document Version 8.0.101.00	18
Part 1	Getting Started	19
Chapter 1	Routing Overview	21
	What is Routing?	22
	Routing Strategies	22
	What Is a Routing Strategy?	23
	Segmenting Interactions	24
	Routing Design Window	24
	Miscellaneous Operations	26
	Routing Targets.....	27
	Loading/Monitoring Routing Strategies	29
	Interaction Design Window.....	30
	Universal Routing Capabilities.....	32
	Database-Driven Routing	32
	Agent-Level Routing	33
	Skills-Based Routing.....	33
	Virtual Agent Groups	33
	Service-Level Routing.....	34
	Multi-Site Routing	35
	Statistical Routing	35
	Routing Interactions Across Tenants	35
	Support for Business Attributes	36

Priority Tuning (Voice Interactions)	37
Strategy Debugging	37
Sharing Strategies Between Environments	38
High Availability.....	38
Business Process Support.....	38
Specialized Interaction Processing Objects.....	39
Open Media Support.....	40
Routing To and Setting Thresholds for Non-Configured DNS	40
Routing Based on Agent Capacity.....	40
Strategy Support for Ring-No-Answer	42
Service Level Agreement Routing	42
Cost-Based Routing.....	43
Proactive Routing	43
Multi-Campaign Agents	44
Load Balancing	44
Session Initiation Protocol Support.....	44
SCXML-Based Strategy Support.....	45
URS Web Interface	45
New Features	46
Release 8.0.1	46
Release 8.0.0.....	47
Licensing	49
Security.....	50
Client-Side Port Definition.....	50
IRD Inactivity Timeout.....	51
Security Notification Banner	51
Secure Web Services Communication	51
Web Service Connections using HTTP Bridge	52
Default of No Access for New Users	52
Migration.....	53
Migrating IRD Objects.....	53

Chapter 2	Voice Routing Architecture and Call Flows	55
	Voice Routing Components.....	55
	Shared Components.....	56
	Optional Components	56
	Network Routing Component.....	56
	Management Framework.....	57
	Media Layer	57
	Configuration Layer	57
	Management Layer.....	57
	Services Layer	60
	Flow Diagram.....	60

Interaction Flow Messaging.....	61
Events and Requests.....	61
General Voice Interaction Flow.....	61
Example: Routing Steps	62
Example: The Call Flow in Detail.....	63
Enterprise Routing Architecture.....	65
Sample Interaction Flow, Enterprise Routing.....	66
Network Routing Architecture.....	67
Sample Interaction Flow, Network Routing	69
URS as a Client to LDS	71
Other Solutions and Applications	73
Chapter 3	
System Availability and Redundancy.....	75
Levels of High Availability.....	76
Base Product Offering	77
Application Failure Management	78
Redundancy Level.....	79
Warm Standby Redundancy Type	79
URS Redundancy Using Warm Standby	80
Using LDS For Redundancy.....	82
URS Options at the Redundancy Level	84
Geographically Distributed Configuration Servers.....	84
High Availability Routing.....	85
Hot Standby Redundancy Type	85
URS Hot Standby with LDS in Enterprise Routing	87
URS Options Impacting High Availability Routing.....	88
System Availability\Redundancy Planning for Network Routing.....	89
Recovering Interactions to a Routable State	90
System Recovery and Statistic Reconciliation	95
Warm Standby URS.....	95
Hot Standby URS	95
URS Options and System Availability.....	96
Failover Options.....	96
Option event_arrive	98
Agent Reservation Options.....	98
Chapter 4	
eServices Architecture and Flows	103
eServices Components	103
eServices Functions	104
Media Interfaces	105
Data Storage: Universal Contact Server.....	105
Workflow Control.....	106

Architecture	109
Processing E-mail	109
Processing Chat	110
Processing SMS Messages	111
3rd Party Media	112
eServices Documentation.....	113
Chapter 5	
Deployment Planning.....	115
Setting Up a Solution in a Contact Center.....	116
Architecture Planning.....	116
Distributed Configuration Environment	117
Private Branch Exchange (PBX) Timeout.....	118
Planning Worksheets.....	119
Enterprise Routing Worksheet.....	120
Network Routing Worksheet.....	123
Worksheet Definitions.....	127
Table Formats, Fields, and Table Access	138
Table Formats Worksheet.....	138
Security Banner Message	139
Part 2	
Deployment.....	141
Chapter 6	
Deployment Process	143
Before You Begin.....	143
How to Configure.....	144
eServices Installation Considerations.....	144
CD and Solution Packaging.....	145
Universal Routing CD	145
Management Framework CD.....	145
eServices CDs	146
Real-Time Metrics Engine CD	147
Reporting Templates CD.....	147
Genesys Info Mart CD	147
Network Routing CD	147
Enterprise Routing Component Installation Order.....	148
Order for Individual Components.....	148
What Each Component Does	149
Network Routing Component Installation Order	149
Custom Server.....	150

Chapter 7	Configuring with Wizards	151
	Important Information	151
	Using the Wizard to Configure Enterprise Routing	152
	Special Note on Adding/Creating Applications	162
	Network Routing Wizard	175
	When a Network Switch Is Not Detected	177
	Routing Solutions and Message Server	180
Chapter 8	Manually Configuring Routing	181
	How to Configure Manually	181
	Manually Configuring Enterprise Routing	186
	Configuration Order for eServices	186
	Enterprise Routing Objects Configuration Order	186
	Limitations	189
	Configuration Manager Limitations	189
	IRD Limitations	189
	Manually Configuring Network Routing	191
	Configuration Order for eServices	191
	Enterprise Routing Objects of Network Routing	191
	Network Routing Objects Configuration Order	193
	Manually Configuring URS	194
	Defining Client-Side Port Information	201
	Setting the use_agent_capacity Option	201
	Other Options That Affect URS	203
	URS Web Service Options	204
	URS Web Interface Options	207
	Communicating with Workforce Management	208
	Manually Configuring IRD	209
	Installing IRD on Desktops	211
	Configuring the IRD Inactivity Timeout	212
	IRD Access Permissions	213
	Interaction Design Shortcut Bar	213
	Creating Subfolders Under the Script Folder	214
	Running the Configuration Database Update Script	215
	Other Manual Configuration Operations	217
	Creating Business Attributes	217
	Scheduling Permissions	218
	Voice Callback Server	219
	Enterprise Routing Solution Object	219
	Network Routing Solution object	223
	Switches for Network Routing	225
	Manually Configuring Stat Server	227

	Premise T-Server for Network Routing	228
	Configuring Message Servers for Different Functions	229
	Configuring DNSs Manually	232
Chapter 9	Load Balancing	235
	Load Balancing Among Multiple URSs.....	235
	Using Load Distribution Server	236
	Enterprise Routing Versus Network Routing Load Balancing.....	236
	Load Balancing Between URSs Using LDS	237
	Load Balancing Without LDS.....	238
	Strategy Loading In Load Sharing Mode	240
	Load Balancing Between Targets	242
	Calls In Transition	243
	Load Balance Family of Statistics	245
	Reporting on Load Balancing	251
	Deciding Which Statistic to Use.....	252
	RStatCallsInTransition	253
	RStatCallsInTransitionEx	254
	Router Self-Awareness	255
	Other Ways to Load Balance.....	260
	StrTargets Function.....	260
	Utility Subroutines	261
	IVR Server Load Balancing	262
	Access Resources	263
	Option Settings	263
	Message Server for Multiple URSs.....	264
Chapter 10	Orchestration Support	265
	Configuring URS to Work with ORS	265
Chapter 11	Samples	267
	Importing the *.zcf Files	268
	Files in Samples.zcf.....	272
	Files in RLU.zcf	274
	Interaction Workflow Samples	279
	Interaction Workflow Samples Functionality	282
	Step-Numbered Business Processes	283
	How To: Business Processes	286
	Graphical Portion of a Strategy.....	287

Chapter 12	Configuring Custom Server.....	289
	Configuring Custom Server	290
	Configuring Custom Server	290
	Backup Custom Server	292
	Sample Procedure File	293
	Windows Sample Procedure File.....	293
	UNIX Sample Procedure File.....	294
	About rcp.c (R_CUSTOMER_PROCEDURE)	295
	Sample rcp.h File	296
	Sample rcp.c File	297
	Logging.....	305
	Custom Server Automatically Generated Log	305
	User-Implemented Logging	305
	Custom Server FAQs.....	307
	Can URS call methods contained in rcp.dll?	307
	Can the DLL be multi-threaded?.....	308
	How do I get the number of idle and used trunk lines?.....	308
	Does Load Distribution Server Support Custom Server?	309
	Can I run more than one Custom Server on the same host?	309
	Can Custom Server trigger an external program?	310
	Can I send a page (or e-mail) as part of a strategy?	310
Chapter 13	Installing a Routing Solution.....	311
	Installation Package Location.....	312
	Installing on Windows Operating Systems	313
	Installing on UNIX.....	328
	Installing URS on UNIX-Based Operating Systems	328
	Installing Custom Server on UNIX-Based Operating Systems	330
Chapter 14	Starting and Stopping Procedures	333
	Prestart Information	333
	Starting	334
	Starting Universal Routing Server Manually	337
	Starting Interaction Routing Designer.....	338
	Starting Network T-Server from the Command Line	339
	Starting Custom Server Manually	339
	Stopping	340
	Non-Stop Operation.....	341
	Version Identification	342

Chapter 15	Uninstalling a Routing Solution	343
	Removing Components with Genesys Administrator	344
	Removing Components Manually.....	344
	Deleting Tables	346
Supplements	Related Documentation Resources	347
	Document Conventions	350
Index	353



List of Procedures

Starting Wizard Manager	153
Creating an Enterprise Routing solution.	156
Creating a URS Application object.	163
Creating an IRD Application object.	171
Configuring Reporting, IVR Interface, and Voice Treatment Options. . .	172
Copying the installation package to a directory	173
Creating a Network Routing solution	175
Configuring a network switch using the Wizard	178
Logging into Configuration Manager	183
Creating/Configuring the URS Application object.	194
Viewing/changing the use_agent_capacity option	202
Configuring URS Web Interface Functionality	208
Configuring URS to communicate with WFM.	209
Creating and configuring the IRD Application object	210
Setting the inactivity-timeout option	212
Displaying the Interaction Design shortcut bar.	214
Creating a subfolder of object type Script.	215
Creating the ird_strategies table.	216
Creating permissions to modify strategy scheduling	218
Configuring an Enterprise Routing Solution object.	219
Configuring a Network Routing Solution object	223
Configuring a Network switch.	225
Configuring a Stat Server Application.	227
Configuring a premise T-Server Application for Network Routing	228
Configuring a Message Server for logging	230
Configuring a Message Server for monitoring only	231
Configuring Load Balancing With LDS.	237
Configuring load balancing without LDS.	239
Synchronizing strategy loading.	240

Configuring URS for Router Self-Awareness	256
Logging into Interaction Routing Designer	268
Importing the sample strategies and other objects.	270
Opening a business process in IRD's Interaction Design window	279
Viewing the path for the .RBN file.	287
Creating and configuring the Custom Server Application object	290
Modifying the sample procedure file on Windows	293
Modifying the sample procedure file on UNIX	294
Installing URS on Windows using the Installation Wizard	313
Installing IRD on Windows using the Installation Wizard	321
Installing Custom Server on Windows using the Installation Wizard	322
Installing Universal Routing Server on a UNIX platform.	328
Installing Custom Server on UNIX	330
Starting a solution with Solution Control Interface	335
Starting URS on UNIX	338
Starting Custom Server 6.5–8.x On Windows	339
Starting Custom Server 6.1	340
Stopping solutions using Solution Control Interface.	340
Stopping Interaction Routing Designer.	341
Removing Universal Routing components with Genesys Administrator	344
Manually removing Universal Routing components on Windows.	345
Manually removing Universal Routing components on UNIX-based operating systems	346



Preface

Welcome to the *Universal Routing 8.0 Deployment Guide*. The first part of this guide familiarizes you with Genesys Universal Routing features, functions, and architecture; and provides deployment-planning guidance. The second part explains how to configure, install, start, and stop Enterprise Routing and Network Routing. It also describes how to configure Custom Server.

This document is valid for all 8.0 release(s) of this product.

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

This preface contains the following sections:

- [About Universal Routing, page 14](#)
- [Intended Audience, page 16](#)
- [Making Comments on This Document, page 17](#)
- [Contacting Genesys Technical Support, page 17](#)
- [Document Change History, page 18](#)

For information about related resources and about the conventions that are used in this document, see the supplementary material starting on [page 347](#).

About Universal Routing

Genesys Universal Routing enables intelligent distribution of interactions throughout the enterprise, whether you have a single-tenant or a multi-tenant environment. Universal Routing can direct interactions from a wide variety of platforms, such as toll-free carrier switches, premise PBXs or ACDs, IVRs, IP PBXs, e-mail servers, web servers, and workflow servers. It can handle pure-voice, multimedia, and blended environments, enabling routing of each media type based on appropriate criteria. Routing strategies and business processes automate interaction routing to the most appropriate agent/resource based on factors such as the type of inquiry, the value of the customer, and the media channel.

CIM Platform

Universal Routing is a part of the Genesys Customer Interaction Management (CIM) Platform that provides the core interaction management functionality.

Universal Routing, on its own, provides voice-routing capabilities. When combined with Genesys eServices (called Multimedia in 8.0.0 and earlier), you can also route various types of non-voice media. Universal Routing and Genesys eServices work together to enable you to seamlessly route both voice and non-voice interactions.

CIM Components

Genesys CIM is the collection of core servers that enable the rest of your Genesys environment to process the thousands of interactions representing the needs of your customers. The CIM Platform consists of the following Genesys products:

- Management Framework
- Interaction Management, which in turn consists of:
 - Universal Routing
 - Interaction Workflow
 - Knowledge Management
 - Content Analysis
 - Universal Contact History
- Reporting

Figure 1 on [page 15](#) depicts CIM graphically.

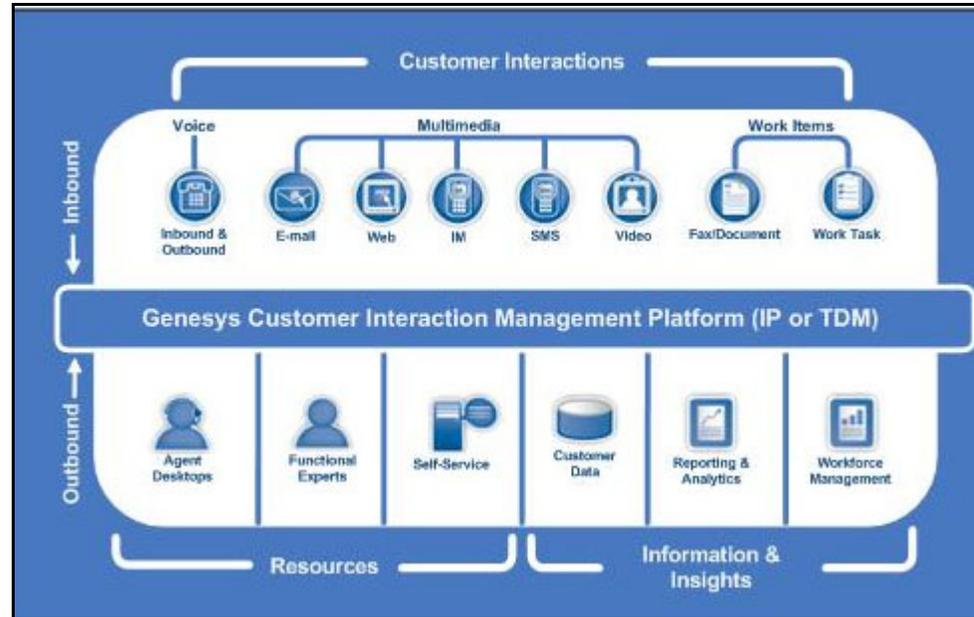


Figure 1: Customer Interaction Management Platform

eServices

The CIM Platform can handle various media channels. Genesys eServices (formerly Multimedia) in [Figure 1](#) encompasses those Genesys components that work together to manage interactions whose media is something other than traditional telephonic voice (for example, e-mail or chat).

eServices includes some parts of the Genesys Customer Interaction Management (CIM) Platform, plus certain of the media channels that run on top of the Platform:

- From the CIM Platform, all of Interaction Management except for Universal Routing:
 - Interaction Workflow—centralized handling of interactions irrespective of media type
 - Knowledge Management—creation and maintenance of standard responses and screening rules
 - Content Analysis—optional enhancement to Knowledge Management, applying natural language processing technology to categorize interactions
 - Universal Contact History—storage of data on contacts and on interactions (linked as threads)
- From the media channels, at least one of the following:
 - Genesys E-mail
 - Genesys Chat (formerly Genesys Web Media)
 - Genesys SMS (Short Message Service)
 - Genesys MMS (Multimedia Messaging Service)

Ideally you will have taken Genesys University routing courses, such as Routing Installation and Configuration, Building Basic Routing Strategies, and Advanced Routing Strategies Workshop.

Making Comments on This Document

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

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

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

Contacting Genesys Technical Support

If you have purchased support directly from Genesys, contact Genesys Technical Support at the following regional numbers:

Region	Telephone	E-Mail
North America and Latin America	+888-369-5555 (toll-free) +506-647-6767	support@genesyslab.com
Europe, Middle East, and Africa	+44-(0)-1276-45-7002	support@genesyslab.co.uk
Asia Pacific	+61-7-3368-6868	support@genesyslab.com.au
Malaysia	1-800-814-472 (toll-free) +61-7-3368-6868	support@genesyslab.com.au
India	000-800-100-7136 (toll-free) +91-(022)-3918-0537	support@genesyslab.com.au
Japan	+81-3-6361-8950	support@genesyslab.co.jp
Before contacting technical support, refer to the <i>Genesys Technical Support Guide</i> for complete contact information and procedures.		

Document Change History

This section lists topics that are new or that have changed significantly since the first release of this document.

New in Document Version 8.0.101.00

The document has been updated to support Universal Routing release 8.0.1. The following topics have been added or significantly changed since document version 8.0.001.00

- In the Overview chapter, new features have been listed for Universal Routing 8.0.1. (See “New Features” on [page 47](#)).
- Information about using HTTP Bridge for Web Services connections has been added. See “Web Service Connections using HTTP Bridge” on [page 53](#).



Part

1

Getting Started

Part One of the *Universal Routing 8.0 Deployment Guide* introduces you to Genesys routing. The information in Part One is divided into the following chapters:

- Chapter 1, “Routing Overview,” on [page 21](#)
- Chapter 2, “Voice Routing Architecture and Call Flows,” on [page 55](#)
- Chapter 3, “System Availability and Redundancy,” on [page 75](#)
- Chapter 4, “eServices Architecture and Flows,” on [page 103](#)
- Chapter 5, “Deployment Planning,” on [page 115](#)



.Chapter

1

Routing Overview

Genesys Universal Routing can be configured for both Enterprise Routing and Network Routing:

- Enterprise Routing—Connects to your premise switch(es). Supports one or multiple sites. A typical Enterprise Routing installation includes T-Servers, Stat Server, Message Server, DB Server, Universal Routing Server (URS), and Interaction Routing Designer (IRD).
- Network Routing—Partially housed within the telephone carrier network (Network T-Server). Connects directly to the network carrier to provide call-routing instructions while the call is still in the carrier's network. Enables immediate load balancing across multiple sites and saves trunking charges between sites. A typical Network Routing installation includes the same components as an ERS except that it requires both network T-Servers and premise T-Servers.

Universal Routing can be used in a pure voice environment. You can also use Universal Routing with Genesys eServices (called Multimedia in 8.0.0 and earlier), which enables you to route non-voice interactions based on IRD-designed business processes (in addition to voice interactions).

This chapter includes the following sections:

- [What is Routing?, page 22](#)
- [Routing Strategies, page 22](#)
- [Routing Design Window, page 24](#)
- [Loading/Monitoring Routing Strategies, page 29](#)
- [Interaction Design Window, page 30](#)
- [Universal Routing Capabilities, page 33](#)
- [New Features, page 47](#)
- [Licensing, page 50](#)
- [Security, page 51](#)
- [Migration, page 54](#)

What is Routing?

In the simplest terms, *routing* is the process of sending an interaction to a target, for example, sending an incoming telephone call or an incoming e-mail message to an agent.

In practice, many steps must be taken between the arrival of an interaction and the selection and use of a target. Not all interactions should go to the same target; choices must be made in order to determine the best target for each interaction.

Each choice-point is an opportunity to make a decision based on the current situation—with the goal of getting the interaction delivered to the right target.

Routing Strategies

An integral part of Universal Routing are routing strategies. In Universal Routing 8.0, strategies are created using Interaction Routing Designer (IRD), which creates strategies in the Genesys IRL routing language. There are two ways to create routing strategies:

1. Using Interaction Routing Designer (IRD), which creates strategies in the Genesys IRL routing language. Figure 3 on [page 23](#) shows the IRD main window.

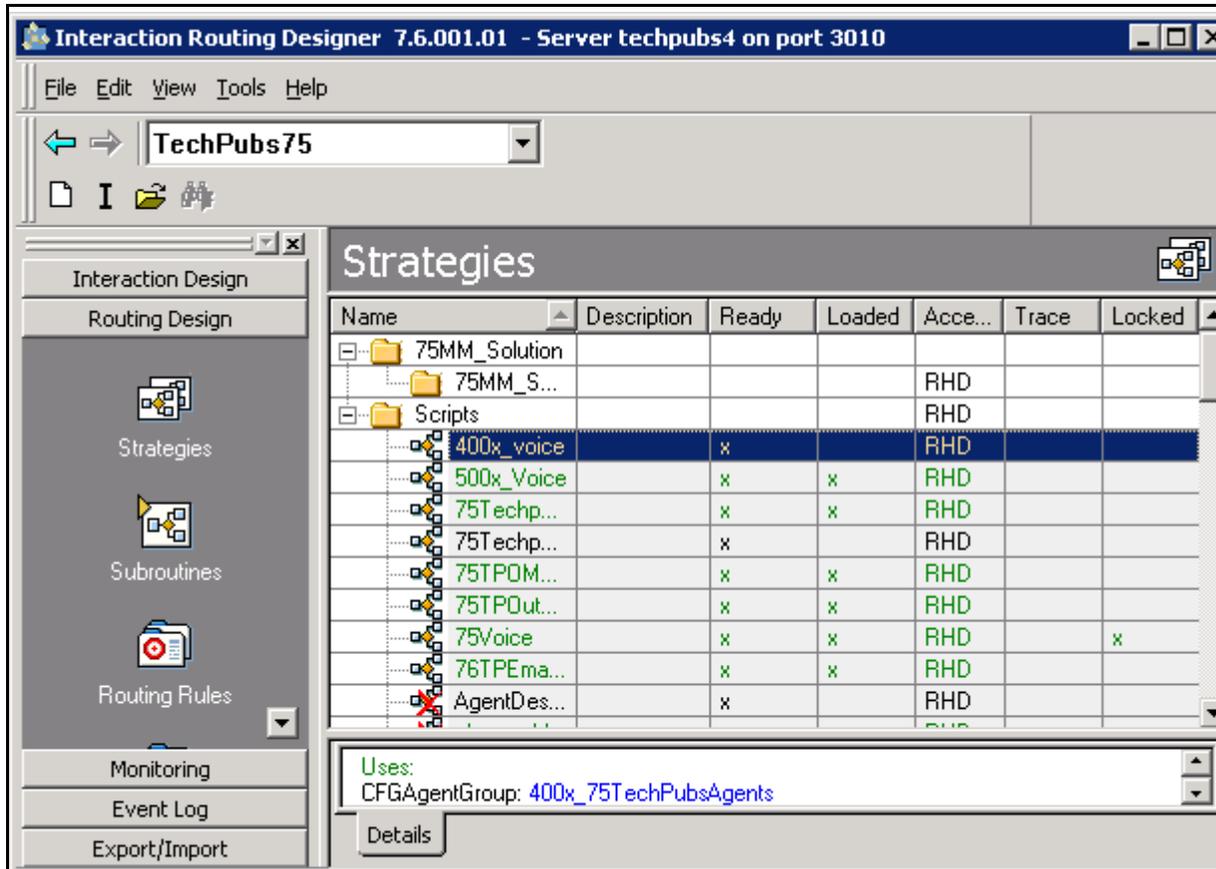


Figure 3: IRD Main Window

- Using Genesys Composer, which lets you create SCXML-based strategies. To execute SCXML-based strategies, Orchestration Server (ORS) is required. For more information on these types of strategies, see Chapter 10, “Orchestration Support,” on page 265 and the *Orchestration Server 8.0 Deployment Guide*.

What Is a Routing Strategy?

A routing strategy is a set of decisions and instructions that tell URS how to handle and where to direct interactions under different circumstances.

Think of a strategy as a structured set of choice-points, each of which analyzes some aspect of the current interaction. The data that a strategy uses to analyze an interaction includes facts related to the interaction itself, the customer initiating the interaction, the state of the contact center, the particular point in time, and so on.

At any given choice-point, only one of several possible outcomes can be optimal. URS determines which outcome is optimal and sends the interaction along a specified route accordingly.

Typically, a choice-point is represented graphically in a strategy by an *object* with one yellow entry port, one red error port, and one or more green exit ports. Figure 4 shows how these ports appear in IRD's Interaction Design window).

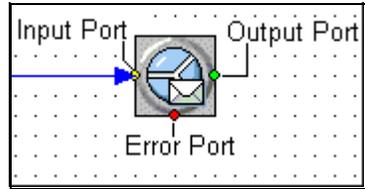


Figure 4: Yellow (Input), Green (Exit), and Red (Error) Ports

Segmenting Interactions

Routing strategies often begin with a Segmentation object that is used to separate incoming interactions and send them on different paths in the strategy. IRD enables you to segment incoming voice interactions based on:

- The date, time, or day of week that the interaction arrives.
- The originating phone number (ANI).
- The number dialed (DNIS).
- Customer information, like account numbers or origin of the interaction.
- Information stored in a database.
- Codes returned by the interaction classification process.
- A combination of different decision criteria.
- A series of logical expressions called *business rules*.

Routing Design Window

Figure 5 on [page 25](#) shows a voice routing strategy in IRD's Routing Design window that uses the Business Segmentation object, which enables you to segment based on business rules.

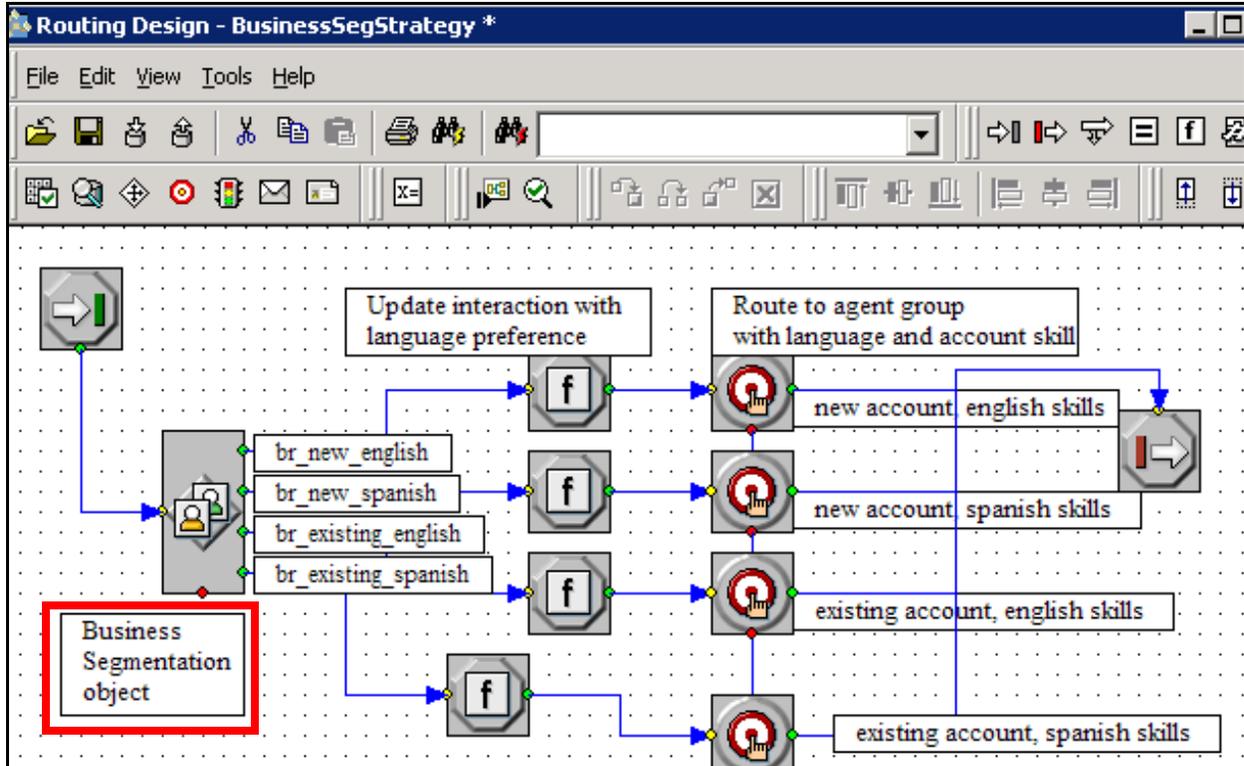


Figure 5: Example Routing Strategy in Routing Design Window

You first create the business rules in IRD (see Figure 6).

Business Rules					
Name	Value	Description	Details	Access	
br_existing_english		Existing Accounts, English		RHD	
att_acct_type	val_existing		UData['acct_type'] = '2'		
att_language	val_english		UData['language'] = '1'		
br_existing_spanish		Existing Accounts, Span...		RHD	
att_acct_type	val_existing		UData['acct_type'] = '2'		
att_language	val_spanish		UData['language'] = '2'		
br_investment_existing				RHD	
br_investment_new				RHD	

Figure 6: Example Business Rules

You can select already-created business rules in the Business Segmentation Properties dialog box (see Figure 7 on page 26), which opens when you double-click the Business Segmentation object.

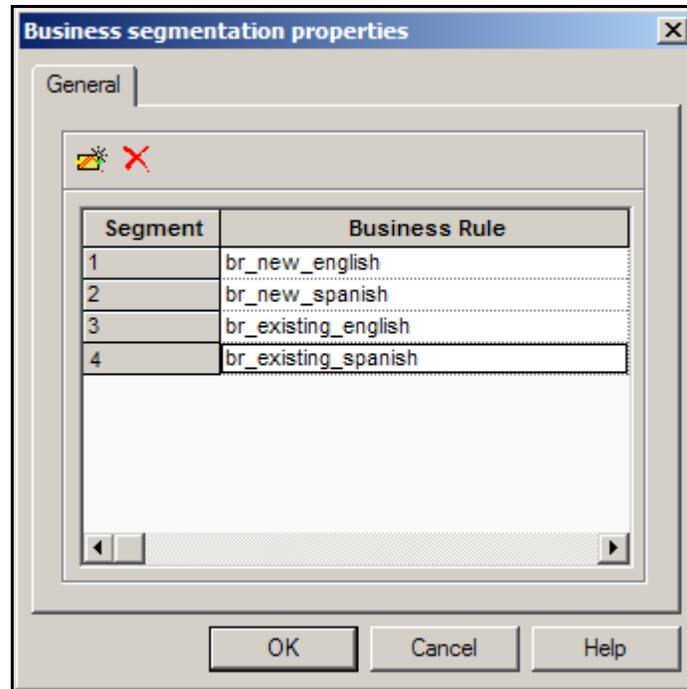


Figure 7: Business Segmentation Properties Dialog Box

Each business rule you select in the Business Segmentation Properties dialog box creates its own green exit port in the strategy object (see Figure 5 on [page 25](#)).

For more information on segmenting incoming interactions, see the Segmentation Objects section in the *Universal Routing 8.0 Reference Manual*.

Miscellaneous Operations

Once an interaction is segmented along a path, the strategy can instruct URS to perform other operations on the interaction, such as:

- Assigning values to one or more variables.
- Calling a subroutine from within the strategy.
- Using a function to perform an operation (see Figure 8 on [page 27](#)), with the option of assigning the output to a variable (variable not shown in the example below).
- Creating an If expression and basing subsequent strategy actions on whether the If statement is true or false.
- Using a macro.
- Attaching data to the interaction or updating data in the interaction.
- Handling errors.

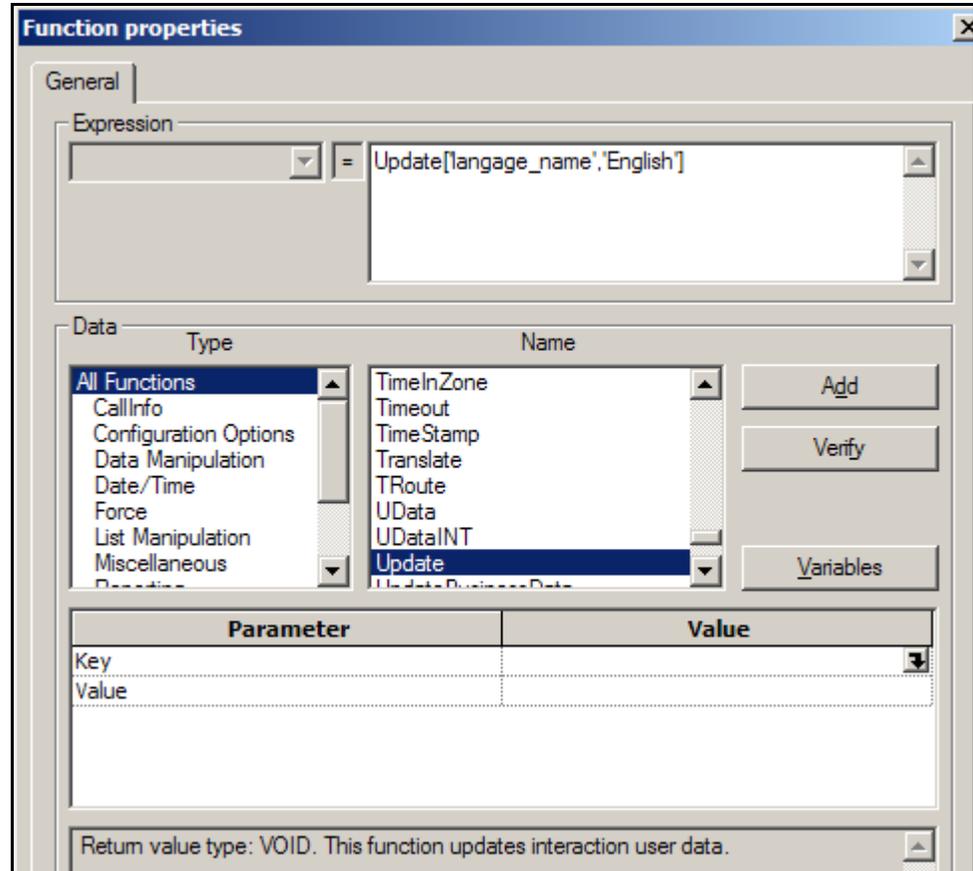


Figure 8: Function Object Updating Interaction Data

For more information on these operations, see the Miscellaneous Objects section in the *Universal Routing 8.0 Reference Manual*.

Routing Targets

During the last phase of the strategy, the interaction is routed to a target. Possible voice routing targets include:

- Agents
- Agent Groups
- Campaign Group
- Destination Label
- Places (desks)
- Place Groups
- ACD Queues
- Queue Groups
- Routing Points
- Skills (agents with certain skills or skill levels)

- Variables
- An object that is the result of applying a Routing Rule, which can allocate based on Load Balancing, Percentage, Service Level, the value of a Statistic, or Workforce considerations.

Possible multimedia routing targets include:

- Agents
- Agent Groups
- Campaign Group
- Places (desks)
- Place Groups
- Queues
- Variables
- Workbins (associated with an Agent, Place, or Place Group)
- Skills

The Selection routing object, used for voice routing, enables you to specify the target for a call in the properties dialog box under Type and Name (see [Figure 9](#)).

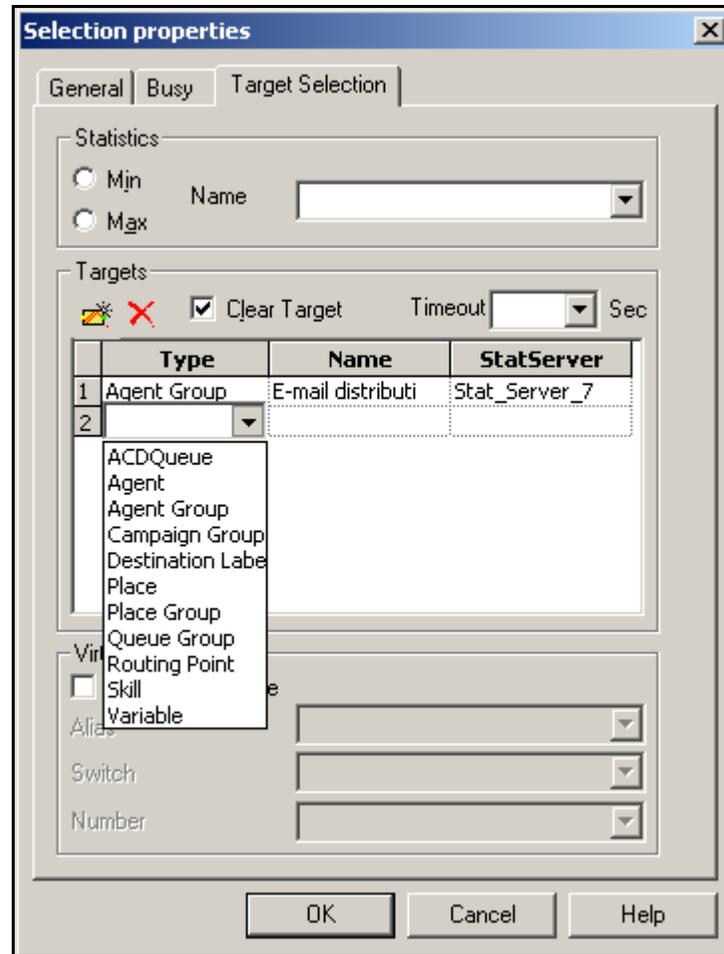


Figure 9: Available Voice Target Types in Selection Properties Dialog Box

- For more information on Routing objects, see the Routing Objects section in the *Universal Routing 8.0 Reference Manual*.
- For step-by-step instructions for creating routing strategies, see the *Universal Routing 8.0 Interaction Routing Designer Help*, accessed from the IRD Help menu.
- Also see *Universal Routing 8.0 Strategy Samples*, which contains examples of both multimedia and voice routing strategies.

Loading/Monitoring Routing Strategies

After you create a routing strategy, you load it on a routing point in IRD's Loading (or Group Loading) view. [Figure 10](#) shows the shortcut menu for loading a voice strategy in IRD's Loading view.

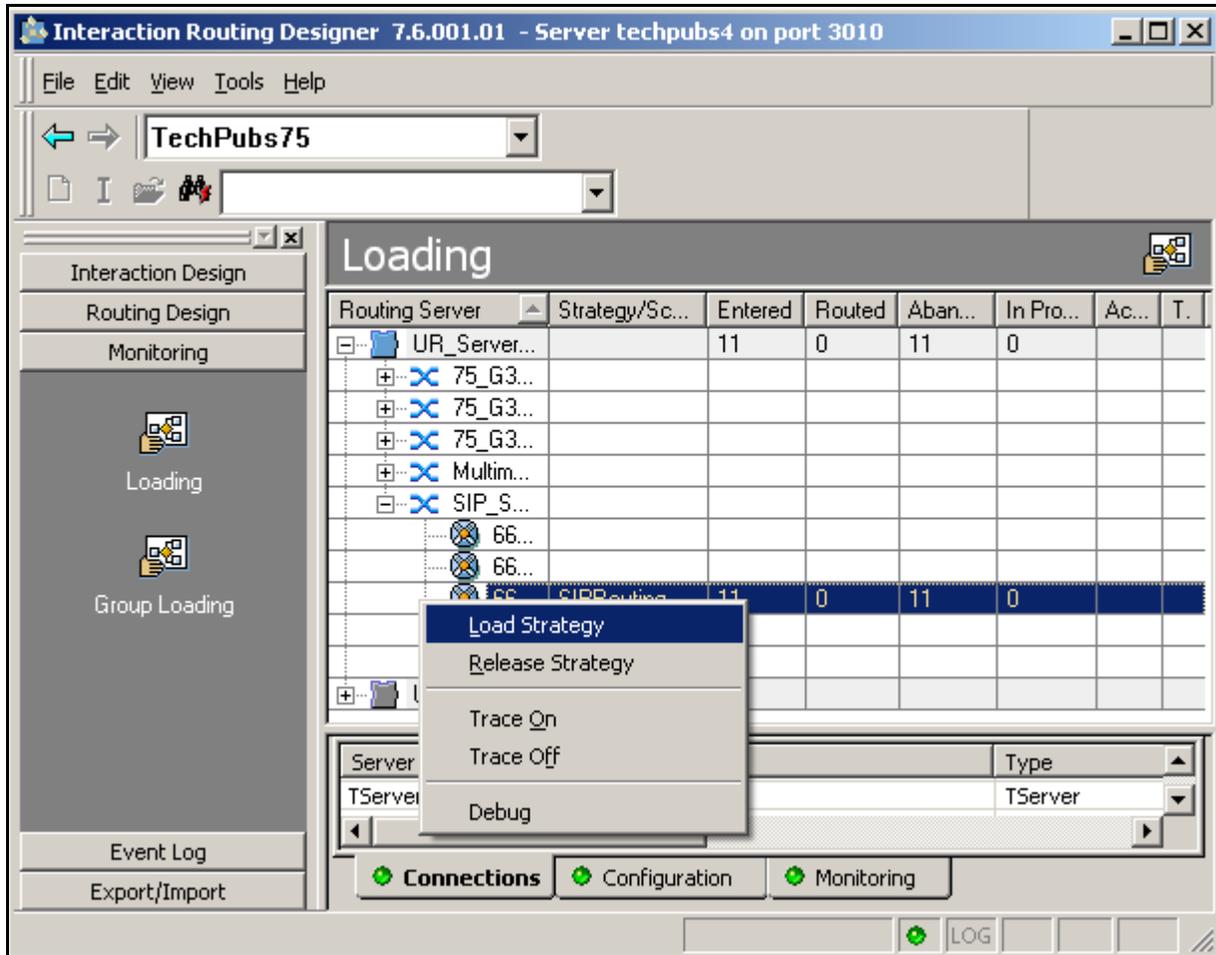


Figure 10: Loading a Voice Routing Strategy

Interaction Design Window

You can create workflows (consisting of business process objects that are connected via queues) in IRD for processing non voice interactions, if you have the Genesys eServices (called Multimedia in 8.0.0 and earlier) components installed.

Note: When IRD starts up, it checks for an eServices solution that is installed by the eServices Configuration Wizard. If none is found, the IRD main window does not contain an Interaction Design shortcut bar. You cannot navigate to the Business Processes list pane or open the Interaction Design window. To change the default, use the Views tab in Routing Design Options, which opens from the Tools menu.'

To access the interface used for creating business process objects:

1. In the IRD main window, click the Interaction Design shortcut bar.

2. Click the Business Processes icon.

Existing business processes display in the list pane (see [Figure 11](#)).

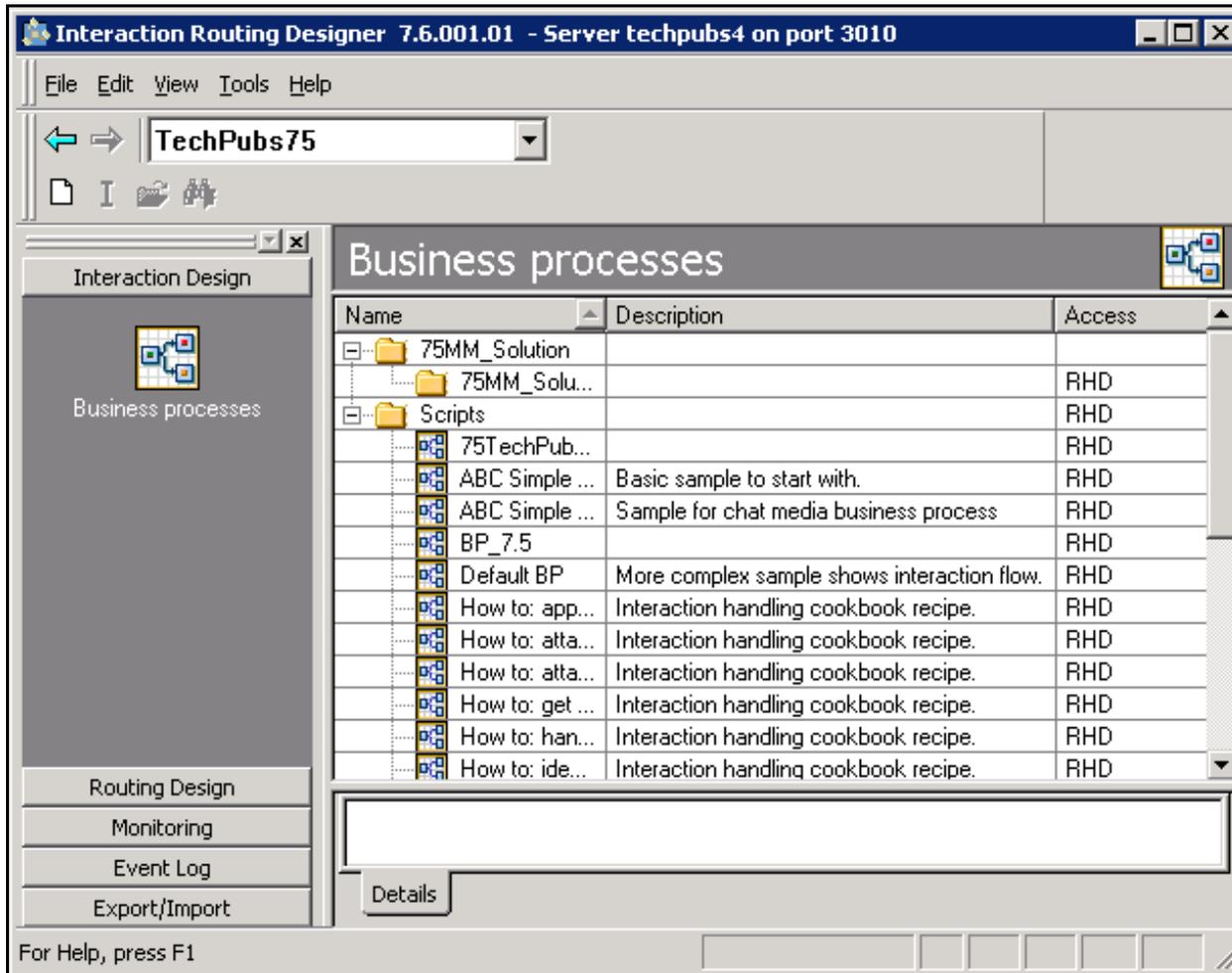


Figure 11: Business Processes List Pane

3. To create a new business process, select New from the File menu.

4. To view/edit an existing business process, double-click it on the list pane.

Either action opens the Interaction Design window. [Figure 12](#) shows the window after opening an existing business process.

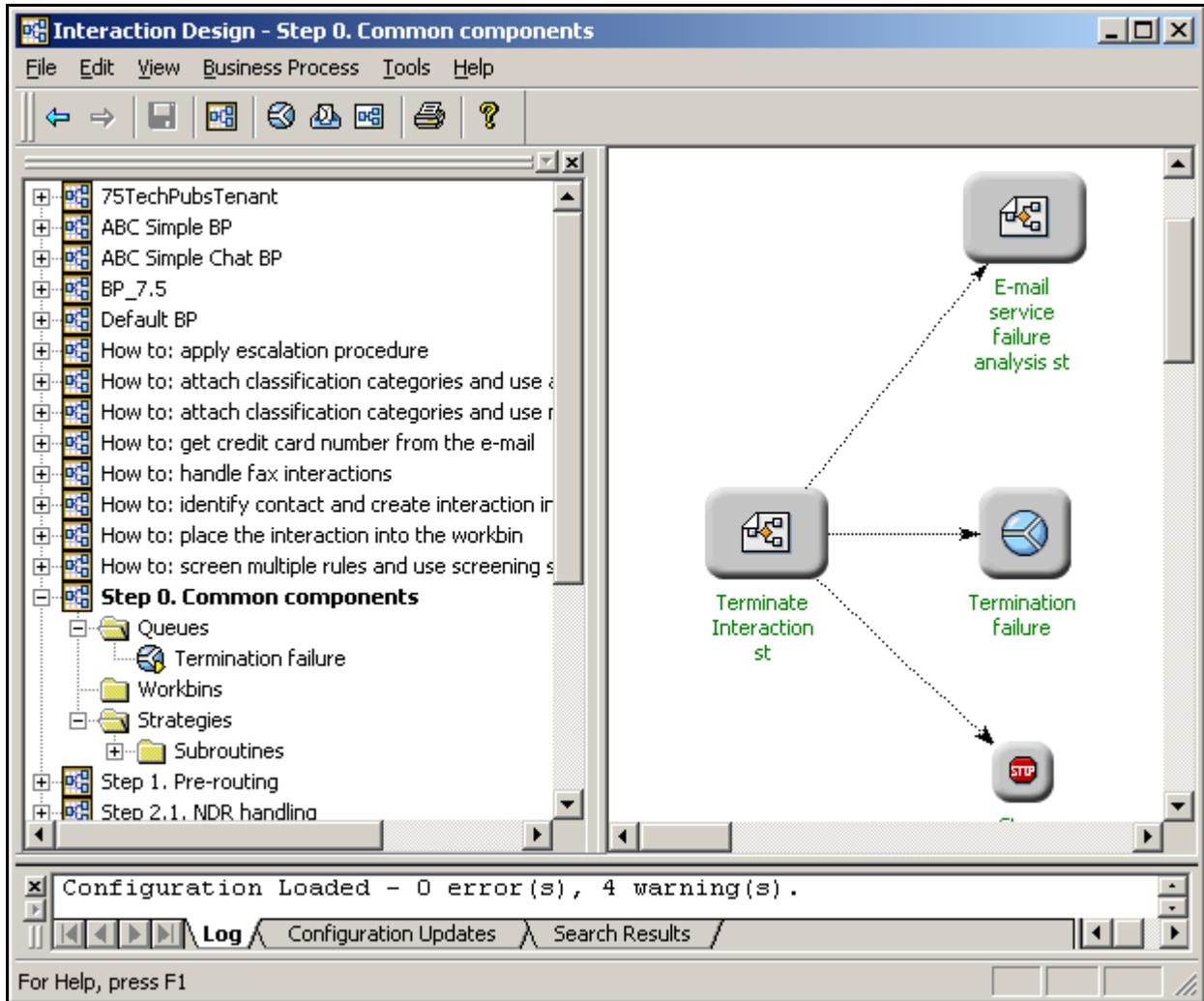


Figure 12: Interaction Design Window For Creating Workflows

As can be seen in [Figure 12](#), the Interaction Design window has three panes:

- An object browser (dockable) on the left displays queues, views, routing strategies, and targets in a tree style.
- A workflow viewer (static) on the right graphically displays the selected interaction flow.
- A log viewer (dockable) displays a running list of status and event messages.

Note: For more information on the Interaction Design window and configuring business process objects, see the *Universal Routing 7.6 (or later) Business Process User's Guide*.

Universal Routing Capabilities

Important Universal Routing functionality includes:

- Database-driven routing ([page 33](#)).
- Agent-level routing, including workforce routing ([page 34](#)).
- Skills-based routing ([page 34](#)).
- Virtual agent groups with priority routing across virtual queues ([page 34](#)).
- Service-level routing ([page 35](#)).
- Multi-site routing ([page 36](#)).
- Statistical routing ([page 36](#)).
- Routing voice interactions across tenants ([page 36](#)).
- Business Attribute assignment ([page 37](#)).
- Priority tuning for voice interactions ([page 38](#)).
- Strategy debugging ([page 38](#)).
- Sharing strategies between environments ([page 39](#)).
- High availability of Enterprise Routing and Network Routing ([page 39](#)).
- Support for business processes ([page 39](#)).
- Specialized interaction processing objects ([page 40](#)).
- Open media support ([page 41](#)).
- Routing to and setting thresholds for non-configured DNs ([page 41](#)).
- Routing using agent capacity information ([page 41](#)).
- Strategy support for ring-no-answer situations ([page 43](#)).
- Service level agreement routing ([page 43](#)).
- Cost-based routing ([page 44](#)).
- Support for proactive routing ([page 44](#)).
- Support for agents participating in multiple outbound campaigns ([page 45](#)).
- Support for load balancing ([page 45](#)).
- Session Initiation Protocol (SIP) support ([page 45](#)).
- SCXML-based routing strategy support ([page 46](#)).
- URS Web Interface ([page 46](#)).

Each function is briefly discussed in the following sections.

Database-Driven Routing

Database-driven routing means that an interaction is routed based on database information. You can build a strategy that instructs URS to:

1. Query a database for any available information, including customer name and language, but also for account level, ID of the last agent who handled the customer, and so on.
2. Attach this information to the interaction.
3. Route the interaction to the appropriately skilled agent.
4. Request services from Web-based applications.

URS supports relational database management systems such as Oracle, Microsoft SQL Server, Sybase, Informix, and DB2. SELECT statements or stored procedures enable data to be read from or written to a database.

URS can also access data from mainframe or non relational databases through Custom Server. For information on Custom Server, see Chapter 12 on [page 289](#).

Agent-Level Routing

Agent-level routing enables interactions to be routed to a specific agent (see Figure 9 on [page 29](#)). You can design a strategy that stores the agent ID with the customer information in a database or attaches it to the interaction. When the customer calls, sends e-mail, or initiates contact by whatever media you might use, URS routes this interaction to the agent with that ID. For a sample strategy, see *Universal Routing 8.0 Strategy Samples*.

When creating strategies called by business processes (see Figure 12 on [page 32](#)), you can send interactions to temporary storage areas associated with an agent, agent group, place (desk) or place group. For example, a workbin associated with an agent may contain draft e-mail responses.

Another type of agent-level routing, workforce routing, routes interactions according to agents' scheduled activities.

Skills-Based Routing

You can also develop skills-based routing (also called agent-based routing) strategies, so that interactions are routed to the most appropriately skilled agent based on the needs of the customer. For more information, see the “Skills-Based Routing” chapter in the *Universal Routing 8.0 Routing Application Configuration Guide*.

Virtual Agent Groups

Virtual agent groups group together agents with similar skills or responsibilities regardless of their work location. Stated another way, a group of agents is considered to be *virtual* if agents do not permanently belong to the group. Instead, Stat Server assigns an agent to the group when an agent meets the criteria specified by the virtual group's definition. Stat Server adds agents

to, or removes them from, the group if agent parameters that affect eligibility change or if the specified criteria are modified.

Stat Server currently supports virtual group functionality with three types of agent parameters:

- A skill configured for an agent
- An ACD queue to which an agent is logged in
- A switch into which an agent is logged in

You can simultaneously specify these types of parameters in an expression for a single virtual group.

Note: For more information on the skill, ACD queue, and switch agent parameters as well as configuring virtual agent groups in the Annex tab of the Agent Group object, see the *Framework 8.0 Stat Server User's Guide*, specifically the chapter on Virtual Agent Groups.

Agents can be in more than one group and URS can determine the availability of an agent across all agent groups using Stat Server. You can add or remove agents from agent groups as needed. URS and Genesys reporting applications are immediately notified of the changes.

Service-Level Routing

Service-level routing enables interactions to be routed according to specified service-level requirements for service types and customer segmentation. For example, a business might have two service-level requirements:

- Ninety percent of interactions received from premier customers must be answered in 10 seconds.
- Eighty percent of all interactions must be answered in 20 seconds.

URS can expand or contract the number of agents available to maintain these service-level requirements.

This type of routing also supports a more advanced form of skills-based routing, which rates a skill or skill set according to its importance level. In this scenario, URS searches all agents in the virtual contact center (or within an agent group if specified) that have the corresponding skills and lists these agents as potential targets according to the skill's importance level.

Agents with the most ideal skill set are placed at the beginning of the list. URS compares the specified service-level goal to the currently achieved service level in the contact center to determine whether the specified service level is maintained. Based on this comparison, URS automatically adjusts the target group size, if necessary, to achieve the service-level goal.

See the Routing Objects section of the *Universal Routing 8.0 Reference Manual* for more information about service-level routing.

Multi-Site Routing

Genesys supports routing across the contact center. Multiple sites—and all available agent resources—serve as one virtual contact center. Interactions are routed according to business criteria and best available agent, regardless of location. Genesys T-Servers support switches that transfer interactions between sites, so you can route interactions to contact centers at different sites and between different departments at different sites. This results in reduced administrative expense and increased workforce flexibility.

Statistical Routing

You can route interactions based on the value of a statistic. Commonly used statistics are:

- `StatAgentLoading` for distributing interactions among agents evenly within an agent group. This predefined statistic can be used to select an agent based on two criteria: Least number of busy DN's, then Longest time in Ready state.
- `StatAgentOccupancy` for distributing interactions among agents fairly within a group in a voice contact center. This predefined statistic can be used as agent selection criteria in an agent surplus scenario.
- `StatAgentLoadingMedia` for distributing interactions works like `StatAgentLoading` with one difference: `StatAgentLoadingMedia` returns data about the current media (not all the agent's medias).
- In 7.6, Universal Routing extended the set of statistics oriented on load balancing with a new “[Load Balance Family of Statistics](#)” as described on [page 245](#). Included are statistics that account for calls in transition from the network to the contact center.

For more information on statistics mapping and the suitable routing target types, see the chapter on routing statistics in the *Universal Routing 8.0 Reference Manual*.

Routing Interactions Across Tenants

The Switch-to-Strategy object uses a routing rule to tell URS to route the interaction to the starting point of another strategy. When using this object, you can select the Tenant name associated with the switched-to strategy.

For more information on the Switch-to-Strategy object, see the Routing Objects section of the *Universal Routing 8.0 Reference Manual*.

Support for Business Attributes

Business attributes are interaction attributes that are used in different ways by various Genesys applications. [Figure 13](#) shows Configuration Manager with the Business Attributes folder expanded.

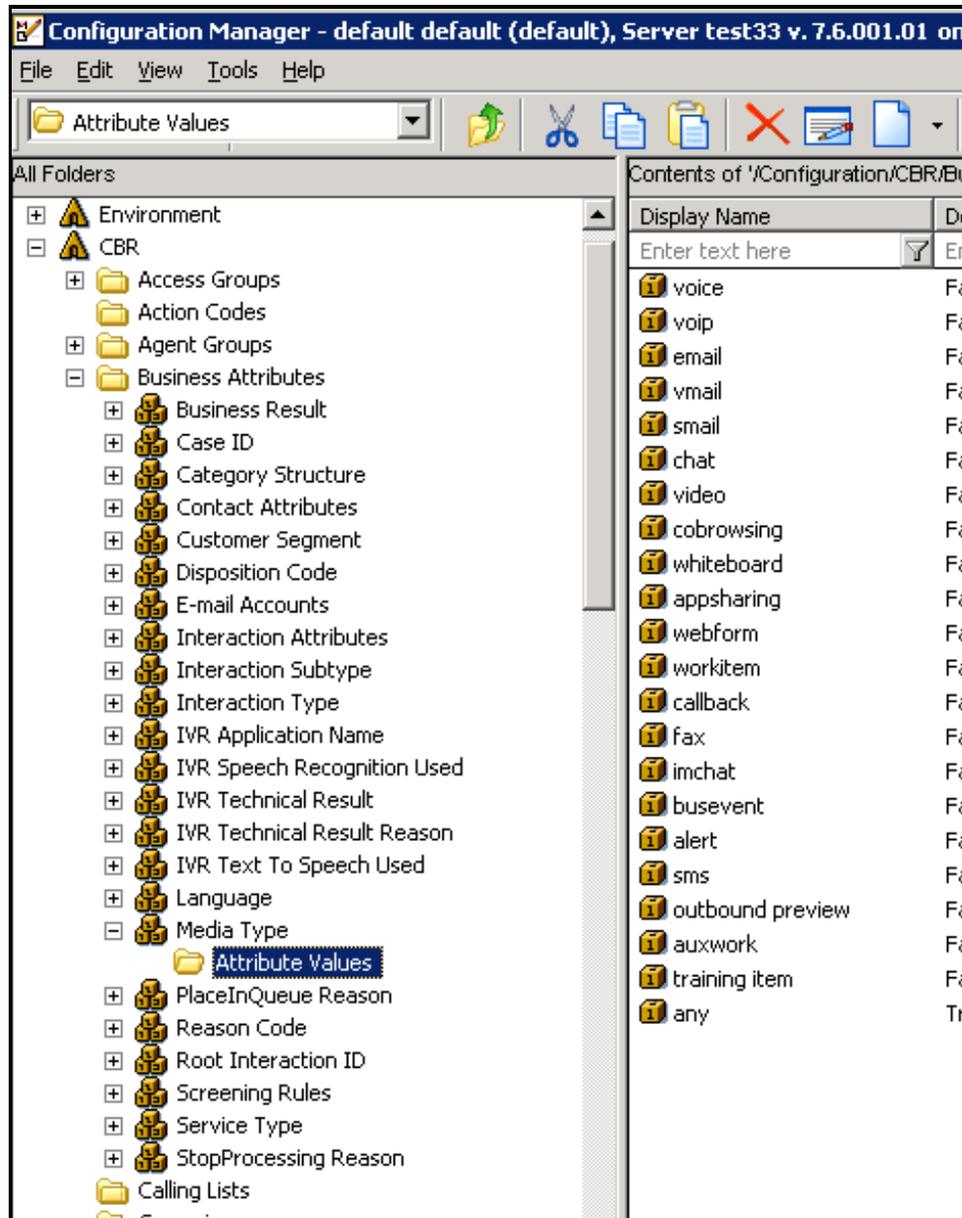


Figure 13: Business Attributes, Media Type Folder

The origin of Business Attributes varies:

- Some Business Attributes are predefined by Genesys and can be expanded.
- Other Business Attributes are customer-defined in Configuration Manager.

- Other Business Attributes are initially defined in Knowledge Manager and then carry over to Configuration Manager.

Notes: See the *eServices (Multimedia) 8.0 User's Guide* for important information on the Business Attributes that can be updated. For information on using Business Attributes when routing, see *Universal Routing 7.6 (or later) Business Process User's Guide*. Also see the routing strategy samples in *Universal Routing 7.6 (or later) Strategy Samples*.

Priority Tuning (Voice Interactions)

In an interaction-surplus scenario, URS selects the most appropriate interaction across virtual queues when a skilled agent target becomes available. An interaction-surplus scenario happens when a skilled agent handles interactions from multiple virtual queues because of membership in multiple agent groups or in a group based on a skill expression. In an interaction-surplus scenario when routing voice interactions, URS can select the most appropriate interaction according to the following priority tuning factors:

- Highest assigned priority number
- Longest relative wait time such as wait time in queue, age of interaction, or what-if wait time
- Greatest service objective risk factor (a weighted value of relative wait time to service objective of the interaction)

For more information on business-priority routing, see the *Universal Routing 8.0 Routing Application Configuration Guide*.

For more information on the PriorityTuning function, see the chapter on functions in the *Universal Routing 8.0 Reference Manual*.

Strategy Debugging

The strategy debugging tool makes it easy to locate and resolve syntax errors and logic problems in your strategies. The debugger verifies the syntax of each strategy statement, tests it on a fictitious interaction, and evaluates every function used. You can also view and edit the value of variables.

The debugger's gives you the advantage of being able to test a strategy without connecting to outside applications. The debugger simulates data and statistics that it would get from Stat Server in a real environment.

See *Universal Routing 8.0 Interaction Routing Designer Help* for information on using the strategy debugger.

Sharing Strategies Between Environments

With IRD, you can share strategies between configuration environments. To do that successfully, you must provide the destination environment with all of the strategy-related objects used in the source strategy.

IRD provides an automated process to package all strategy-related objects that IRD created in the source environment and to extract the same objects in the destination environment. To do this, IRD creates a strategy-moving file to move objects from a source environment to a destination environment.

However, you must use Configuration Manager to manually create objects such as fields, formats, and tables that IRD is not responsible for creating.

After exporting and importing, IRD generates export and import log files that contain detailed information about the objects that were or were not successfully packaged and extracted between configuration environments.

Note: For more detailed information about the automated process of packaging and unpacking a strategy, export and import log files, and access permissions, see *Universal Routing 8.0 Interaction Routing Designer Help* and the section access control in the *Universal Routing 8.0 Reference Manual*.

High Availability

High availability refers to the percentage of time that a routing solution is available to process routing interaction services according to a specified performance criteria above a given performance threshold which depends on system architecture.

Universal Routing supports high availability with three product offerings or levels. See Chapter 3 on [page 75](#) for more information.

Business Process Support

If you are using Genesys eServices (formerly Multimedia), IRD provides an *Interaction Design* window for creating business processes (see [Figure 12 on page 32](#)). A *business process* directs incoming interactions through various processing objects with the goal of delivering an appropriate response to the customer.

- In the case of an e-mail interaction, an appropriate response might be an e-mail message that answers the customer's questions.
- In the case of a chat interaction, an appropriate response might be mailing product brochures to the customer.
- In the case of a fax interaction, an appropriate response might be an e-mail message that states the requested materials had been received.

Genesys supplies sample business processes in its eServices Interaction Workflow Samples component. These include sample queues, views, and routing strategies (see “Interaction Workflow Samples” on [page 279](#)).

Note: For more information, see *Universal Routing 7.6 (or later) Business Process User’s Guide*.

Specialized Interaction Processing Objects

Interaction Routing Designer (IRD) contains specialized objects that enable strategy designers to create non-voice routing strategies that are called by eServices business processes. These objects can:

- Generate an acknowledgement e-mail message.
- Generate an autoresponse e-mail message.
- Request Classification Server to classify interactions for relevance to site-defined categories and return classification codes that can be used for routing.
- Manually assign classification categories.
- Segment incoming interactions on classification codes.
- Screen interactions for certain words or word patterns.
- Request E-mail Server to attach a chat transcript to an e-mail message.
- Request E-mail Server to forward (with the expectation of getting a response back) e-mail to other agents or experts.
- When an external resource reply comes back as a result of a forwarded e-mail message, take the reply e-mail message as input, extract the reply text from it, and create a customer reply outbound e-mail message.
- Enable you to interact with Web-based applications (Web Services) outside of Genesys applications. You specify request parameters, what type of data you expect to be returned, and what to do with the returned data.
- Request E-mail Server to redirect (no expectation of a response back) interactions to other agents or experts.
- Place an interaction in an internal queue.
- Route an interaction to a specified target type (Agent, Agent Group, Campaign group, Place, Place Group, Variable, or Skill).
- Route an interaction to a workbin, a temporary storage area associated with an Agent, Agent Group, Place, or Place Group.
- Send an interaction waiting in a queue to a customer or another agent.
- Stop interaction processing and (optionally) notify Universal Contact Server or a third-party server (through Interaction Server) that interaction processing has stopped for a specified reason.

- Create a new outbound e-mail message, a notification message, or an e-mail message that can be translated into Short Message Service (SMS) format.
- Identify one or more contacts in the UCS database that match data attached to an interaction or, if no such contact exists, create a new contact using the attached data.
- Update contact information by overwriting data in the UCS database with that attached to an interaction. You can also use the Update Contact object to add more data fields to an existing contact.

Open Media Support

To support Open Media, IRD supplies general-purpose objects for use in routing strategies that are applicable to all media types. These include objects for sending interactions to queues, attaching classification categories, and stopping interaction processing.

There is also an External Service object that can be used for all media types that you can use to route strategies called by business processes. This object can be used to exchange data with third-party (non-Genesys) servers that use the Genesys Interaction SDK or any other server/application that complies with Genesys Interaction Server communication protocol.

In addition to the above objects, other IRD objects can also be used for both voice and non-voice interactions. See *Universal Routing 7.6 (or later) Business Process User's Guide* for more information.

Routing To and Setting Thresholds for Non-Configured DNs

You can count the number of calls at DNs that are not configured in the Configuration Database and use this count for routing decisions (`RStatCallsInQueue` statistic). To count calls at these DNs, you must configure the URS `call_monitoring` option as `true`. You must also create a strategy in IRD using the `SetThresholdEx` and, if necessary, the `NMTExtractTargets` function (which is used as a helper function).

Note: If you want to have multiple service numbers distributed to the same non-configured (non-monitored) DN, you must align the SN to DN table, the network switch, and URS so that all calls that are sent to a non-configured DN are processed by the same URS.

Routing Based on Agent Capacity

Capacity rules assigned to agents (`Person` objects) define a resource's ability to handle multiple interactions. An agent-capacity rule is a set of logical

expressions that specify the boundaries of a resource’s ability to handle one interaction or more than one simultaneous interaction of differing media types. By default, URS is required to use agent-capacity information supplied by Stat Server when routing non-voice interactions. For example, URS must use agent-capacity information in a blended environment (such as when routing e-mail plus voice or e-mail plus chat) or when routing other media types (such as when routing only e-mail or only chat).

Agent-capacity rules, configured using the Genesys Resource Capacity Wizard (see Figure 14), provide information about whether an agent is available for routing.

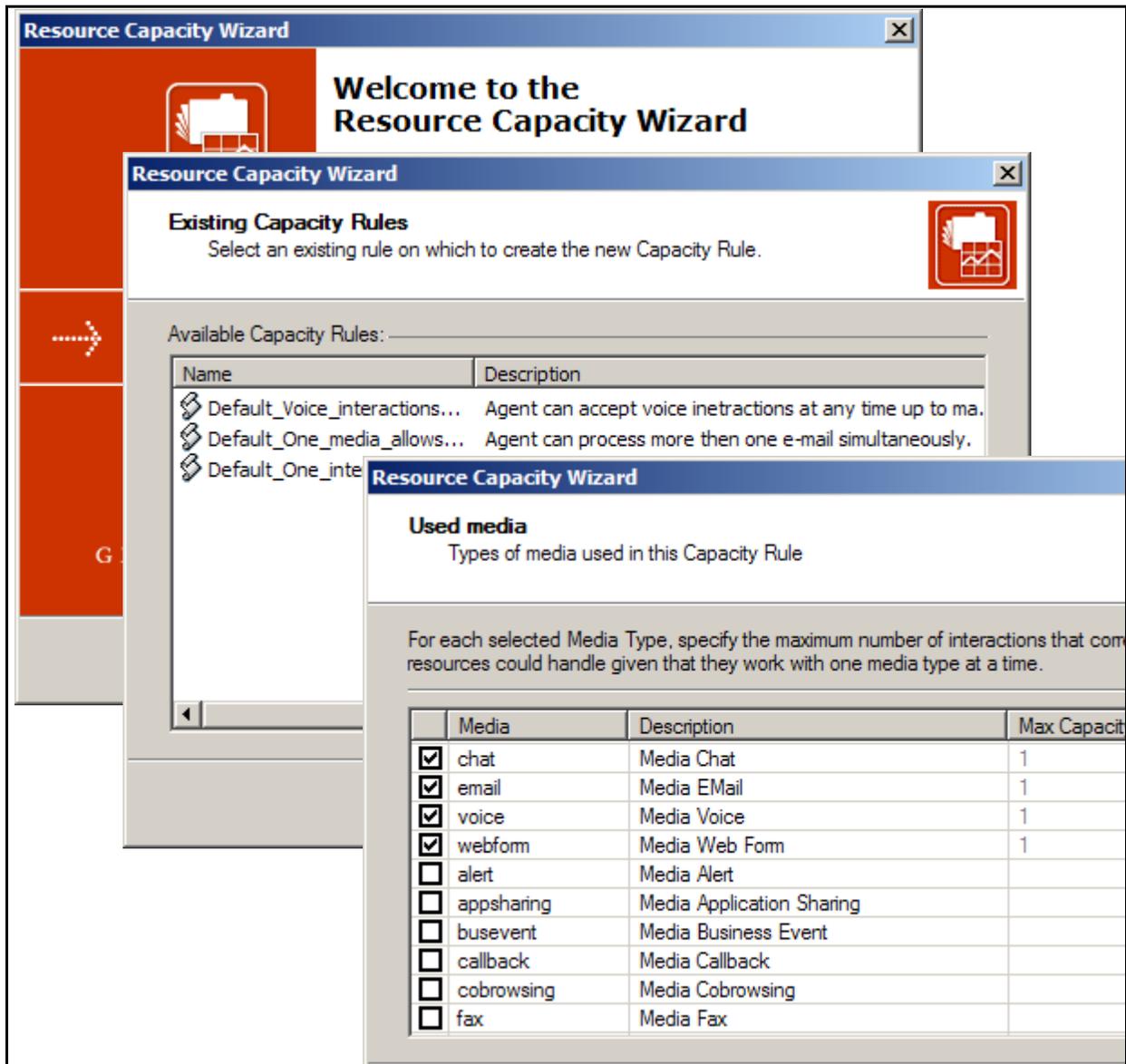


Figure 14: Genesys Resource Capacity Wizard

After compiling a complete list of available agents (taking agent-capacity rules into consideration, if configured), URS applies the routing selection criteria specified in the strategy objects.

For more information on routing based on agent-capacity rules, see [page 202](#). Also see the option `use_agent_capacity` and the function `CheckAgentState` in the *Universal Routing 8.0 Reference Manual*. The *Genesys 8.0 Resource Capacity Planning Guide* explains the Resource Capacity Wizard.

Notes: By default, the `use_agent_capacity` option in the URS `Application Template` is set to `true`. If you do not want to route based on the agent-capacity model, change the option to `false`. For information on setting the option, see “Setting the `use_agent_capacity` Option” on [page 201](#).

The latest available version of the Resource Capacity Wizard is 7.1. You can install it from the Real-Time Metrics Engine 8.0 CD using the instructions documented in the *Genesys 8.0 Resource Capacity Planning Guide*.

Strategy Support for Ring-No-Answer

A ring-no-answer scenario occurs when URS sends an interaction to an agent because the agent has indicated that he or she is in a Ready state. However, in reality, the agent has stepped out without changing his or her status into Not Ready. In this case, a ring-no-answer scenario occurs. Some PBXs can detect this scenario and re-route the call.

In Universal Routing 8.0, ring-no-answer is handled via:

- A function, `SendRequest`, that makes it possible to send various requests to T-Server, including `RequestAgentNoReady` and `RequestRedirectCall`.
- Predefined macros that use the `SendRequest` function, as well as other T-Library functions.

For more information on handling ring-no-answer, see the `SendRequest` function and the Macro sections in the *Universal Routing 8.0 Reference Manual*.

Service Level Agreement Routing

Share Agents by *Service Level Agreement* (SLA) routing uses cross-trained agents. It enables a business user that manages multiple business lines to define the triggering conditions and constraints that allow agents to be shared among business lines. With Genesys share agents by SLA routing:

- You can define a specific set of activation conditions for the main business line to invoke shared agents (that is, to borrow agents).

- You can also define the “Guarding” conditions so that the shared agents who are eligible to receive calls/interactions from non-primary business lines do not affect their own business goals negatively.

For more information, see the *Universal Routing 8.0 Routing Application Configuration Guide*.

Cost-Based Routing

Cost-based routing (CBR) enables Universal Routing Server to:

- Calculate the cost of routing to any target based on configuration information, statistical values, and its own data.
- Use the cost of routing to target as additional target selection criteria.
- Use information contained in strategies to activate/de-active cost-based routing during target selection.
- Automatically attach to interactions information required for cost-based routing reporting.

You have the option of implementing various levels of CBR, from simple (cost defined only as an agent property) to advanced (based on both Infrastructure cost (cost to transfer an interaction from Site A to Site B) and Resource cost. A wizard guides you through the process of defining Infrastructure and Resource Cost.

When defining Resource cost, you define Interaction Types (comprised of Media Type + Service Type + Customer Segment). For each Interaction Type, you can specify processing based on volume or a variable rate. For more information on CBR, see the *Universal Routing 7.6 (or later) Cost-Based Routing Configuration Guide*.

Proactive Routing

The IRD Routing Design window provides strategy-building objects that support Genesys Outbound Contact, a product for creating, modifying, running, and reporting on outbound campaigns for proactive customer contact. You can use the Outbound strategy-building objects to automate the building of customer Calling Lists, finish processing a Calling List record, reschedule a customer call, and update a Calling List record. You can also add customer records to Do Not Call lists from within a strategy. The generic External Server IRD object can be used to process other requests for Outbound proactive customer contact, such as Record Cancel and Record Reject.

For more information, see the *Genesys 7.6 (or later) Proactive Routing Solution Guide*. Also see the section on Outbound objects in the “Interaction Routing Designer Objects” chapter of the *Universal Routing 8.0 Reference Manual*.

Multi-Campaign Agents

To support Genesys Outbound Contact and agents participating in multiple Outbound Campaigns, Campaign Group is a target type in the Routing Selection and Route Interaction strategy-building objects. Functions SelectDN and SData add Campaign Group target as a parameter.

In addition, the Universal Routing 8.0 installation package includes strategy bytecode designed to automatically route outbound calls/interactions to Campaign Groups. Because a pure Genesys Outbound Contact customer does not have the rights to edit strategies, this guide provides instructions for loading the bytecode as described in “OutboundMultiCampaign.ooo” on [page 273](#).

A strategy option instructs URS to run the strategy. For ERS/NRS inbound voice customers, Universal Routing 8.0 provides an editable strategy of the same content as well as utility and sample strategy/subroutines that can be imported, edited, and used as customers see fit.

Load Balancing

As described in the chapter on load balancing (see [page 235](#)), URS supports load balancing for both Enterprise Routing and Network Routing with or without Load Distribution Server, as well as IVR Server load balancing (see [page 262](#)). You can load balance among multiple URSs or load balance between routing targets using statistics. Starting with 7.6, Universal Routing extends the set of statistics oriented on load balancing with a new “[Load Balance Family of Statistics](#)”. Included are statistics that account for calls in transition from the network to the contact center. As an alternative to statistics, function StrTargets (see [page 260](#)), makes it easy to specify targets during load balancing. In addition, the Universal Routing 8.0 installation package contains several sample utility subroutines (see [page 267](#)) that can be used for load balancing.

Session Initiation Protocol Support

Genesys SIP Server is a combined T-Server and a call-switching component, in which the call-switching element functions as a Session Initiation Protocol (SIP) Back-to-Back User Agent. In concrete terms, this means that call switching and control is performed by Genesys—no third-party PBX or ACD system is required.

Universal Routing Server can work with SIP Server to support both chat and instant messaging (IM) sessions between agents and customers. Chat calls can be received from federated public networks via Microsoft’s Live Communication Server (LCS), which is currently the only Genesys-supported third party component available for integrating with SIP server instant messaging. Chat calls can be sent to Microsoft’s Office Communicator (via LCS) or routed to a Genesys SIP endpoint.

For more information, see [page 202](#) and “Agent Capacity and Instant Messaging” on [page 202](#). Also see the *Genesys 7.6 Instant Messaging Solution Guide*.

SCXML-Based Strategy Support

Starting with 8.0, Universal Routing takes a step toward “routing openness” by providing the ability to run both existing IRD-defined routing strategies and routing strategies that are created outside of IRD written in State Chart EXtensible Markup Language (SCXML). A new component, Orchestration Server, has been introduced in 8.0 to provide this functionality. See [Figure 15](#).

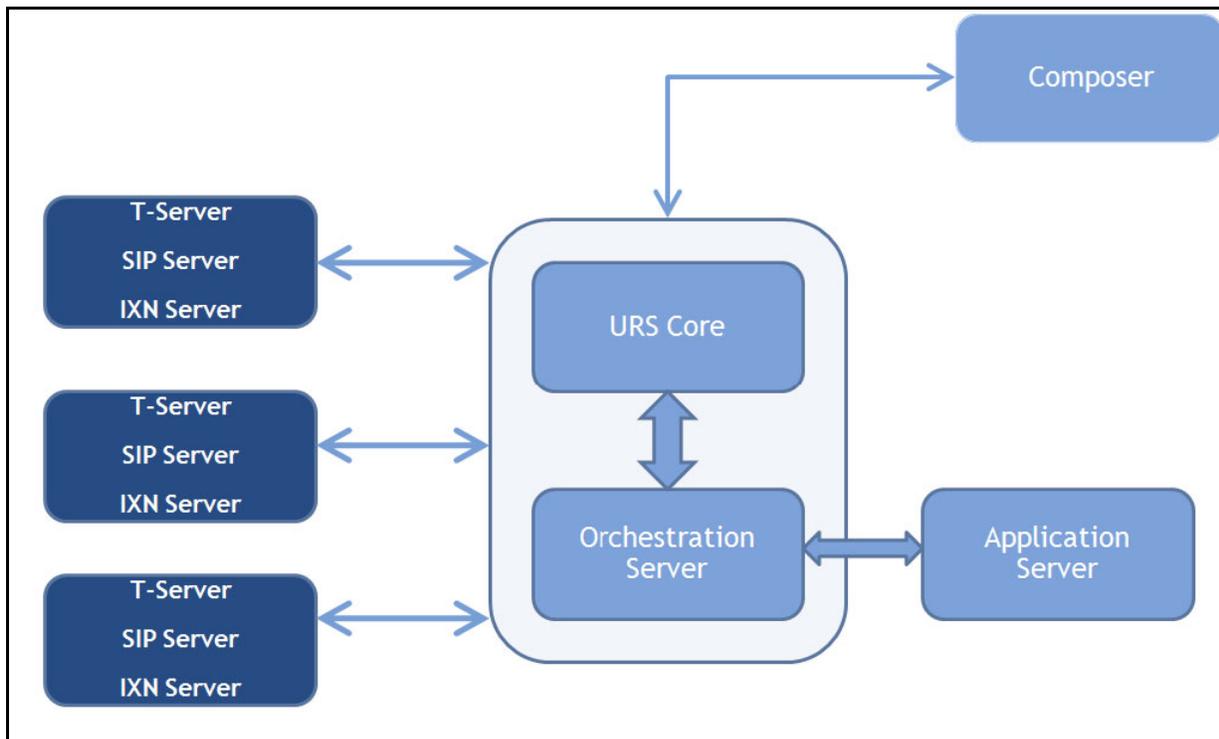


Figure 15: Orchestration Support

For more information, see Chapter 10, “Orchestration Support,” on [page 265](#) and the *Orchestration Server 8.0 Deployment Guide*.

URS Web Interface

In another step toward “router openness,” Universal Routing 8.0 provides a URS Web Interface, which extends URS’s routing functionality to external Web Services/application servers/third party applications.

The URS Web Interface is a “T-Server-less” way for those users who do not have Genesys installed to communicate with URS and use its resource

allocation service. As a result, a routing request can be delivered to URS in one of the following ways:

- Through T-Server.
- Through URS Web Interface.

Figure 16 illustrates the URS Web Interface.

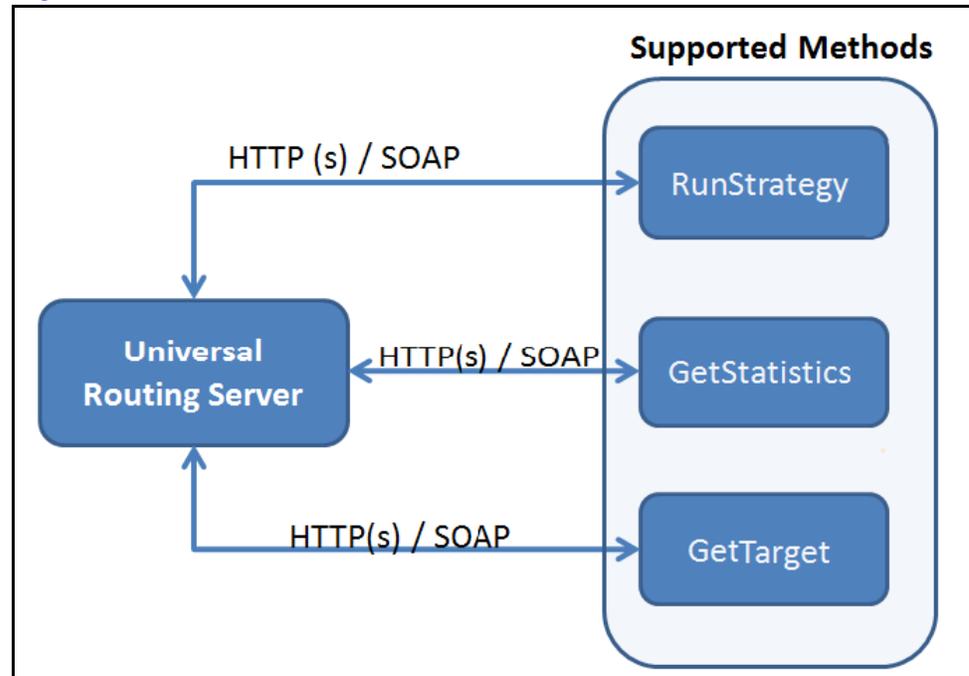


Figure 16: URS Web Interface

HTTP or SOAP requests can be made directly to URS to get statistics, visible targets, or to run a strategy. For configuration information, see “Configuring URS Web Interface Functionality” on [page 208](#).

New Features

For features introduced in earlier versions of Universal Routing, see the “Overview” chapter in the applicable *Deployment Guide* (or for releases prior to 7.1, the *Getting Started Guide*) for the relevant release.

Release 8.0.1

- URS web services functionality supports nested Web Services Description Language (WSDL) access.
- URS will periodically check for updates to intelligent Workload Distribution (iWD) interaction priority, thereby augmenting the iWD solution. The frequency of these checks is configurable.

- Interaction Routing Designer (IRD) now supports ESP rendering of SMS in the eServices solution.
- URS 8.0.1 supports the `Submit New` function, which is used to create a new interaction in Interaction Server.
- URS 8.0.1 supports the `DistributeCustomReportingEvent` function, which provides delivery of special event to reporting clients and the logging database.
- Universal Routing now supports the use of strategy variables in `Condition` expressions in the `FindInteractions` object.
- IRD now provides a new option `Don't Send User Data`, applicable only to multimedia objects—`External Service` and `Submit New`. Users will have the option either to enable or disable sending user data when using these objects. Other objects like `Find Interaction`, `Update Interaction` and all objects from the “Workflow and Resource Management” group will not send user data by default now, without the possibility of turning off the function. This helps to minimize network traffic between URS and Interaction Server.
- URS and IRD 8.0.1 have support for rendering message content for eServices interactions.
- URS 8.0.1 provides functionality to route inbound voice interactions to Agents who could be assigned for Outbound interactions.
- IRD now provides additional flexibility with the `FindInteractions` object by allowing strategy variables to be used in `condition` expressions.
- IRD now allows you to use strategy variables as `Interaction Parameter Names` in `Update Interaction` and `Update UCS Record` objects.
- IRD 8.0.1 allows `Capture Point` applications as `Media Servers` in `Business Process` diagrams within IRD. `Capture Point` is a new application object defined in `Management Framework 8.0.3` specifically to be used with `iWD`.
- IRD provides special events (`Distribute Custom Event` object), such that Interaction Server reporting clients and the `Logging` database are able to capture the same events, thereby improving reporting details.
- IRD now supports installation on the Windows 2008 and Windows 7 operating systems.

Release 8.0.0

Support for External Calls

- Built-in as part of the IP, Universal Routing now provides a new `Web (HTTP)` interface in addition to `HTTP bridge` service. The new interface facilitates `SOAP` and `HTTP(s)` calls into `Universal Routing Server (URS)`. For configuration information, see [page 207](#).

- The URS Web interface supports the following methods through HTTP or SOAP calls:
 - `RunStrategy`: Runs an existing IRL strategy with this request, resulting in independently running a strategy.
 - `GetTarget`: Provides information on an available target based on statistics, priority, and media type.
 - `GetStatistics`: Enables you to request statistics from URS, based on a list of targets such as agents, queues, and so on. Note that the statistics that are retrieved are specific to URS.
- URS provides functional modules that cater to Orchestration Server requests that are based on SCXML applications. Orchestration Server (ORS) is a new component for Universal Routing. With ORS, Universal Routing can now execute routing strategies that are written in SCXML. You can create SCXML-based strategies manually (for example, by using Notepad or your favorite text editor) or by using the Genesys Composer interface. For more information, see Chapter 10, “Orchestration Support,” on [page 265](#) and the *Orchestration Server 8.0 Deployment Guide*.

New Features in Universal Routing Server (URS) 8.0.0

- Universal Routing Server (URS) 8.0 supports Genesys Administrator for provisioning.
- All connections that originate from URS can now be secured through firewalls by using client-side port definition. You can define these ports when you connect other Genesys applications to URS.
- URS now enhances UUI data readability by providing functions to convert it into different data types.
- URS now provides Transport Layer Security (TLS). For more information on this feature, see the *Universal Routing 8.0 Reference Manual*.
- URS has new options to mask ring-back on transferred calls. You can now continue call treatment on the transfer leg. This functionality applies to all T-Servers.
- URS now provides a new function, `StrNextToIndex`, to help the `StrAsciiTok` function manage the indexing sequence when using two or more variables.
- URS now provides alarms and log messages when it discovers issues that occur during external routing of interactions. These alarm messages are propagated to Message Server once across many calls.
- Genesys Universal Routing 8.0 is now supported on Linux 5, VMWare, IBM LPAR.
- Genesys Universal Routing 8.0 now supports SQL Server 2008 and 2005 clustered DB; it also supports Oracle 11g, Windows 32/64 bit 2008, and IBM AIX 32/64 bit version 6.1.

New Features for Interaction Routing Designer (IRD) 8.0.0

- Interaction Routing Designer (IRD) 8.0.0 does not support SCXML applications, but it will continue to support existing IRL strategies that were developed using previous versions of IRD.
- IRD now provides Jump to Strategy functions that can move control across tenants based on tenant parameters and the target strategy name.
- IRD is now supported in Citrix environments—specifically Citrix XenApp Version 4.0 and 4.5.
- IRD provides the `CallUUID[]` function to attach the TEvent Attribute of the `CallUUID` as attached data.
- Using IRD 8.0.0, you can now use Division Arithmetic Operators within the `SetTargetThreshold` function expression and on the `Threshold` tab.
- IRD now supports installation of multiple versions of IRD 8.0 on the same computer. Note that you cannot have both IRD 8.0 and an earlier version of IRD on a single computer.
- IRD now provides functionality to exit call treatment by using different options based on caller entered digits.

Licensing

Note: You must set up licensing before you start to install and configure Universal Routing. Starting with Management Framework 8.0.2, Genesys 8.x products that run on the Windows operating system work with FLEXlm 11.7 and 9.5. Windows 2008 requires FLEXlm 11.7.

Along with its software, Genesys supplies its customers with software licenses. Licenses manifest the customers' legal rights to use the features that Genesys software provides.

Note: Development or sales of third party tools and applications which require or utilize Genesys Universal Routing or Genesys Orchestration Server functionality, including access to Genesys Universal Router web interface to provision Target information, Statistics, invoking session, etc., requires a Router Connector License. Please contact your account executive to get more information on its applicability and pricing.

Genesys licenses its applications using the FLEXlm License Manager, produced by Macrovision. At startup, all licensed Genesys servers establish a client connection to License Manager, providing a computer host ID or IP

address along with various information about the application. If the application has a valid license, License Manager allows the application to start and run properly. Note that the Framework Management Layer can control and monitor License Manager as a third-party application but not as a Genesys server application.

Genesys recommends that you configure and install License Manager and license files before configuring and installing Framework components (which precede installing of Universal Routing components). For information about which products require what types of licenses as well as on the installation procedure for License Manager, refer to the *Genesys Licensing Guide*, which is available on the Genesys Documentation Library DVD.

Note: If you are planning to deploy redundant configurations for any Genesys servers, you must have a special high-availability (HA) license. Otherwise, the Management Layer does not perform a switchover between the primary and backup servers.

Security

As described in the *Genesys 8.0 Security Deployment Guide*, Genesys uses the Transport Layer Security (TLS) protocol that is based on the Secure Sockets Layer (SSL) 3.0 protocol. TLS uses digital certificates to authenticate the user as well as to authenticate the network (in a wireless network, the user could be logging on to a rogue access point).

You have the ability to secure all communications (SSL/TLS) between Genesys components, which includes authentication and authorization (certification validation). This functionality is configurable so that you can secure all connections, selected set of connections, or none of the connections.

Note: Summary information on Universal Routing 8.0 security is presented below. For detailed information on how to implement security within Genesys, see the *Genesys 8.0 Security Deployment Guide*. For information on deploying a third party authentication system to control access to Genesys applications, see the *Framework 8.0 External Authentication Reference Manual*.

Client-Side Port Definition

The client-side port definition feature of Genesys security enables a client application (of server type) to define its connection parameters before connecting to the server application. This enables the server application to control the number of client connections. In addition, if the client application is

located behind a firewall, the server application will be able to accept the client connection by verifying its predefined connection parameters.

Universal Routing provides the flexibility and security of defining the client-side port of the client/server connection. [Table 1](#) indicates where client-side port configuration is supported for other servers.

Table 1: Component Support for Client-Side Port Security

Clients	Config. Server/ Config. Server Proxy	Custom Server	T-Server	DB Server	Inter-action Server	Stat Server	URS
URS 8.0	Yes, 7.6+ required	Yes, 7.6+ required	Yes, 7.6+ required	Yes, 7.6+ required	Yes, 8.0 required	Yes, 8.0 required	--
Custom Server 8.0	Yes, 7.6+ required	--	NA	NA	NA	NA	NA

Notes: NA = Not Applicable.

Client-side port configuration is not supported for Load Distribution Server or Message Server.

For detailed information on client-side port configuration, see the “Client-Side Port Definition” chapter of the *Genesys 8.0 Security Deployment Guide*. Also see pages [315](#) and [323](#) in this guide.

IRD Inactivity Timeout

Option `inactivity-timeout` allows you to require that users log back into IRD after a specified period of user inactivity. This application level option allows you to override the Genesys default, which is hardcoded into IRD. For more information on this option, see the *Universal Routing 8.0 Reference Manual*.

Security Notification Banner

The IRD installation process enables you to configure one or more security banner messages. Users can accept or reject these messages, which can be configured to display every time they log into IRD (as well as other Genesys applications) or on a one-time basis. For more information on this feature, see [page 139](#).

Secure Web Services Communication

To secure communications through the HTTP connections that URS uses to access Web Services, IRD extends the Web Service object property dialog box.

The Security tab (introduced in 7.6) allows you to provide parameters that can be used to authenticate (if necessary) requests during URS communication with the specified Web Service. For information on this function and the Web Service object, see the *Universal Routing 8.0 Reference Manual*.

Web Service Connections using HTTP Bridge

To support https Web Service connections using HTTP Bridge, the following information need to be specified: `Certificate`, `Certificate Key` and `Trusted CA`. If the URL that is passed from URS to HTTP Bridge starts with `https://`, HTTP Bridge will try to extract these attributes from the corresponding HTTP Bridge options: `def_certificate`, `def_certificate_key`, `def_trusted_ca`. These options are specified in URS's web folder (which is where all other HTTP Bridge options are specified).

HTTP Bridge reads the attributes and passes them to the connection library. The connection library is responsible for providing secure communications.

On Windows, only the `Certificate` attribute needs to be specified; on UNIX, all three are required for proper operation.

Below are some examples of values for these attributes.

- Windows:
Certificate: 21 32 33 63 74 95 ... 01
- UNIX:
Certificate: /home/gcti/.security/certificate.pem
CertificateKey: /home/gcti/.security/certificate-key.pem
TrustedCA: /home/gcti/.security/ca.pem

For more information about how to use certificates (such as how to create and deploy them), refer to the *Genesys 8.0 Security Deployment Guide*.

Default of No Access for New Users

The `Environment` and `Tenant` objects have an option that defines the access permissions for a new `Person` object created in Configuration Manager. The default is for a new `Person` object to have no access permissions unless you specify assign them in the `Members of` tab of the `Person Properties` dialog box. As a result, in order to use IRD 7.6 or 8.0, each `Person` object must have one or more `Access Groups` assigned. Any user without an assigned `Access Group` can open IRD, but can only view the IRD GUI.

Migration

Note: All 8.0 Universal Routing software components are backward compatible with previous versions. All configuration options from previous versions are supported and have the same meaning in 8.0.

As described in the *Genesys Migration Guide*, you can migrate any 6.x and 7.x routing solutions and options to 8.0. Migration includes:

- Converting data in the Configuration Database to the 8.0 format.
- Upgrading the IRD and Custom Server software components by using an automated procedure as described in the “[Migrating IRD Objects](#)” section that follows.
- Migrating routing strategies and routing strategy objects (used within those strategies or independently).

Universal Routing software components (defined as *Interoperable Components* in the chapter on Maintenance Level Interoperability in the *Genesys 8 Interoperability Guide*) are compliant with 7.x environments and operate in 6.x compatibility mode in 6.x environments (Configuration Layer 6.x) for existing installations as hot fixes.

Migrating IRD Objects

For a high level overview, see the section on migrating strategies and other objects in the *Universal Routing 8.0 Reference Manual*.



Chapter

2

Voice Routing Architecture and Call Flows

This chapter describes the components that make up Universal Routing, the general architecture for routing voice interactions, and how Universal Routing voice routing components relate to Framework components. It also provides two sample interaction flows with explanations of how components function together when a customer contacts a business that uses ERS or NRS.

Note: For information on the components involved in routing multimedia interactions, see Chapter 4 on [page 103](#)

This chapter includes these sections:

- [Voice Routing Components, page 55](#)
- [Management Framework, page 57](#)
- [Interaction Flow Messaging, page 61](#)
- [General Voice Interaction Flow, page 61](#)
- [Enterprise Routing Architecture, page 65](#)
- [Network Routing Architecture, page 67](#)
- [URS as a Client to LDS, page 71](#)
- [Other Solutions and Applications, page 73](#)

Voice Routing Components

The components that make up Enterprise Routing and Network Routing are almost identical with one exception. Network Routing also uses one or more Network T-Servers.

Shared Components

Components shared by Enterprise Routing and Network Routing include:

- **Interaction Routing Designer (IRD)**—the user interface for creating, editing, loading, debugging, and monitoring routing strategies. IRD communicates with URS through Configuration Server. URS uses the strategies designed in IRD but stored in the Configuration Database to route interactions. IRD also receives real-time routing information about interactions, server status, and routing points from URS through Message Server.
- **Universal Routing Server (URS)**—the server that executes the rules specified in strategies created in IRD. URS creates a list of destinations or targets based on a strategy, determines which target is most appropriate, and instructs T-Server where to route the interaction. URS takes strategy information created in IRD from Configuration Server. (Strategies are designed in IRD but stored in the Configuration Database.) When you monitor strategies, URS sends real-time routing information about interactions, server status, and routing points to IRD through Message Server.

Optional Components

Universal Routing can also include the following optional components:

- **Custom Server**—enables URS to retrieve and send data from non-SQL databases using a customized stored procedure. The IRD interface for defining a custom stored procedure is identical to the interface used for a SQL stored procedure.
- **Voice Callback Server**—when wait times are high, gives customers the choice to request a callback ASAP, request a callback at a specific time, or to continue to hold. For more information on the Voice Callback Server, see *Voice Callback 7.1 Getting Started Guide, Deployment Guide, and Reference Manual*.

Network Routing Component

Network Routing also requires a Network T-Server:

- **Network T-Server**—a server that enables URS to communicate with the network carrier and to direct interactions from the network to the appropriate premise switch and T-Server. URS can connect to multiple Network T-Servers at once, enabling URS to handle interactions from multiple carriers simultaneously.

For more information on a specific Network T-Server, see the reference manual for your specific Network T-Server.

Management Framework

Universal Routing relies on various Management Framework Layers.

Media Layer

The Media Layer enables solutions to communicate with a variety of media including traditional telephony systems. This layer also provides the mechanism for creating attached data and distributing it within the Genesys environment. The Media Layer consists of T-Server and Network T-Server.

- T-Server provides an interface with traditional telephony systems. As the center of any Genesys environment, T-Server keeps track of interactions from start to finish, receiving and distributing messages as directed by other Genesys components.
- Network T-Server is a specialized T-Server that communicates with network carriers and provides the interface between these carriers and the Genesys environment.

Configuration Layer

The Framework Configuration Layer includes Configuration Server, Configuration Manager, and DB Server.

- Configuration Server provides centralized access to the Configuration Database, based on permissions set for any user to any configuration object. Configuration Server also maintains the common logical integrity of configuration data, including strategy objects, and notifies applications of changes made to the data.
- Configuration Manager is the user interface that provides access to configuration data used by routing solutions and for setting user permissions for solution functions and data.

For more information on Configuration Server and Configuration Manager, see *Framework 8.0 Deployment Guide*, *Framework 8.0 Configuration Options Reference Manual*, *Framework 8.0 Configuration Manager Help File*, *Framework 8.0 External Authentication Reference Manual*, and *Framework 8.0 Configuration Layer Error Messages Help File*.

Management Layer

The Framework Management Layer includes Local Control Agent, Message Server, Solution Control Server, and Solution Control Interface. It may also include Load Distribution Server.

- Local Control Agent (LCA) is used internally to start and stop applications.

- Message Server provides centralized processing and storage of every application's maintenance events. It also provides a way for IRD and URS to communicate. You can configure multiple Message Servers for different purposes. See "Configuring Message Servers for Different Functions" on [page 229](#) for more information.

Note: The information passed between URS and IRD through Message Server for monitoring interactions, server status, and routing points does not appear in the Message Server log. That information is internal to Message Server and is not designed for readability.

- Solution Control Server (SCS) is the processing center for Management Layer.
- Solution Control Interface (SCI) provides the following:
 - The status of all installed Genesys components and LCA on hosts
 - Information about the hosts and each active alarm condition for components
 - Requests to start and stop components and solutions to SCS

Together, SCI and SCS provide a central point of administration.

Figure 17 on [page 59](#) shows how an Enterprise Routing Solution appears in Solution Control Interface.

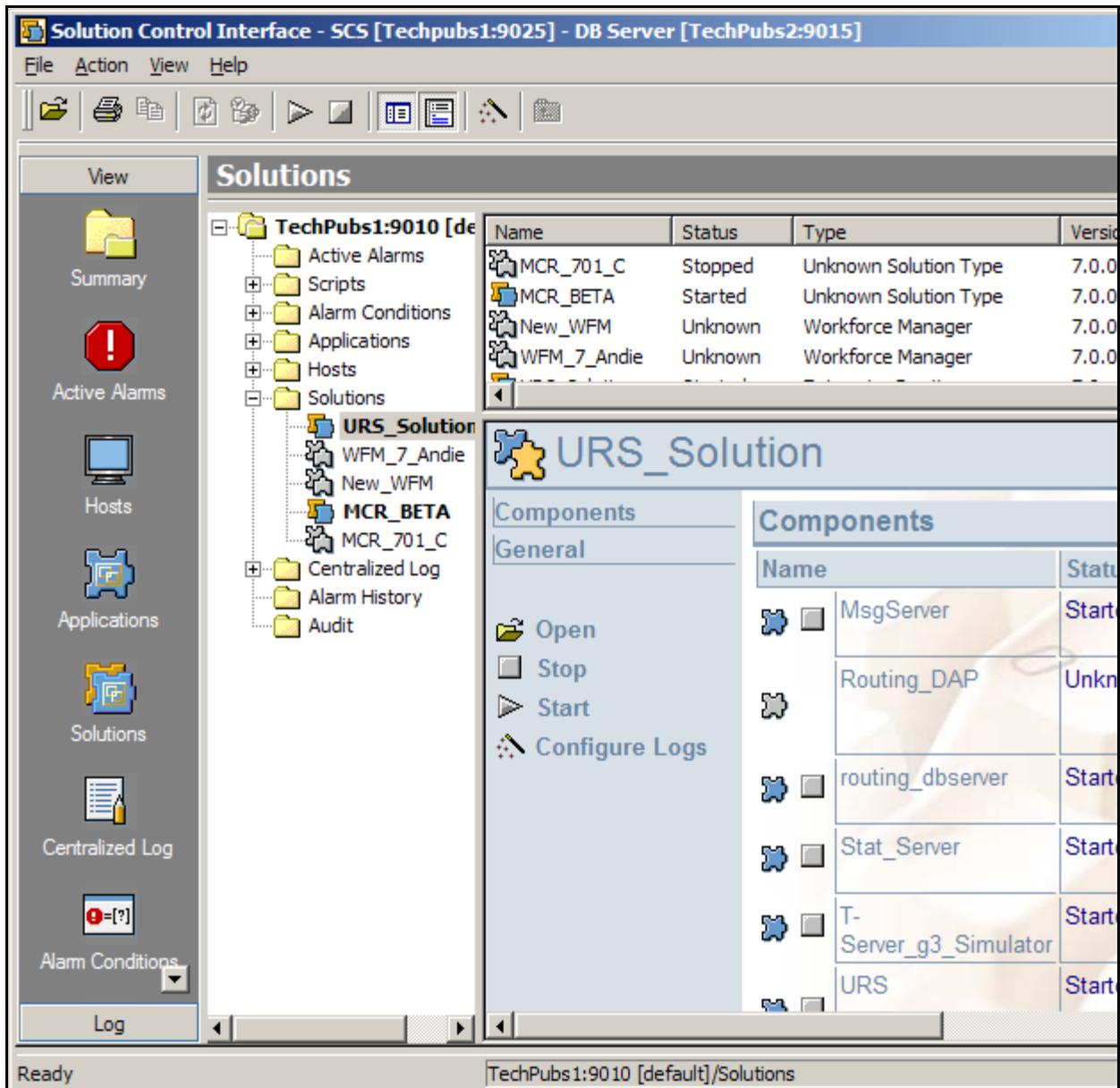


Figure 17: Solution Control Interface

- Load Distribution Server (LDS) mediates between one or more T-Servers and URS instances in order to evenly balance the load of interactions. For more information, see “Using Load Distribution Server” on [page 236](#).
For more information, see the *Framework 7.2 Load Distribution Server User’s Guide*. For specific routing-related information on Load Distribution Server and an architecture diagram, you can also see “Using Load Distribution Server” on [page 236](#).

Services Layer

The Framework Services Layer includes Stat Server and DB Server.

- Stat Server tracks the real-time status of the contact center. For example, Stat Server monitors the Place that each Agent is logged into along with the status of each (Ready or Not Ready). Stat Server also collects performance statistics. For example, Stat Server knows the length of time each Agent has been in the Ready state. These statuses and statistics can be used by URS to make intelligent real-time routing decisions.
- DB Server provides the URS with access to enterprise customer data stored in standard database management systems (DB2, Informix, Microsoft SQL, Oracle, Sybase).

Flow Diagram

Figure 18 depicts the information flow of components belonging to the various Management Framework Layers.

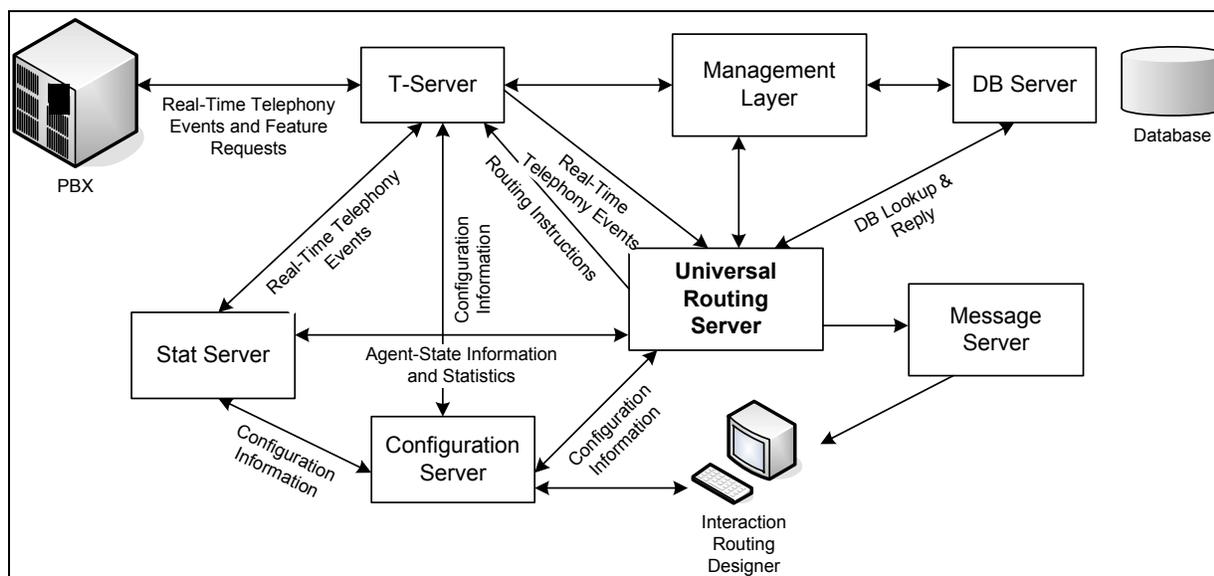


Figure 18: Components in Management Framework Layers

For more information on these components, see the *Framework 8.0 DB Server User's Guide* and the *Framework 8.0 Stat Server User's Guide*.

Interaction Flow Messaging

In order to understand routing interaction flow, it is essential to understand Genesys-specific client-server messaging.

Events and Requests

Events are generated by the PBX such as the ringing of an agent's phone. T-Server distributes Events to other Genesys applications and can also generate Events.

Requests are generated when a CTI application requests interaction-handling or a related task. One example is the message sent by an agent desktop application requesting that the switch hang up a call.

A typical flow of a set of messages looks like this:

- The PBX sends event messages to T-Server via the CTI link.
- T-Server distributes event messages to its clients.
- T-Server clients send request messages to T-Server.
- T-Server sends request messages to the PBX via the CTI link.

T-Server Messaging

T-Server has standard event and request messaging to communicate with its clients (T-Library). These standard messages enable developers to integrate applications with T-Server. For more information on event and request messaging, see the *Genesys Developer Program T-Library SDK Developer's Guide*.

T-Server communicates with switches in the language of their dedicated CTI-links, translating between CTI link messaging (switch-dependent) and Genesys client messaging (T-Library).

In some cases, T-Server creates an event not related to the PBX or does not relay a request to the PBX when the messaging doesn't apply to switch functionality.

General Voice Interaction Flow

Figure 19 on [page 62](#) shows voice interaction flow messaging and the components of the Media, Management, Configuration, and Services Layers that are involved.

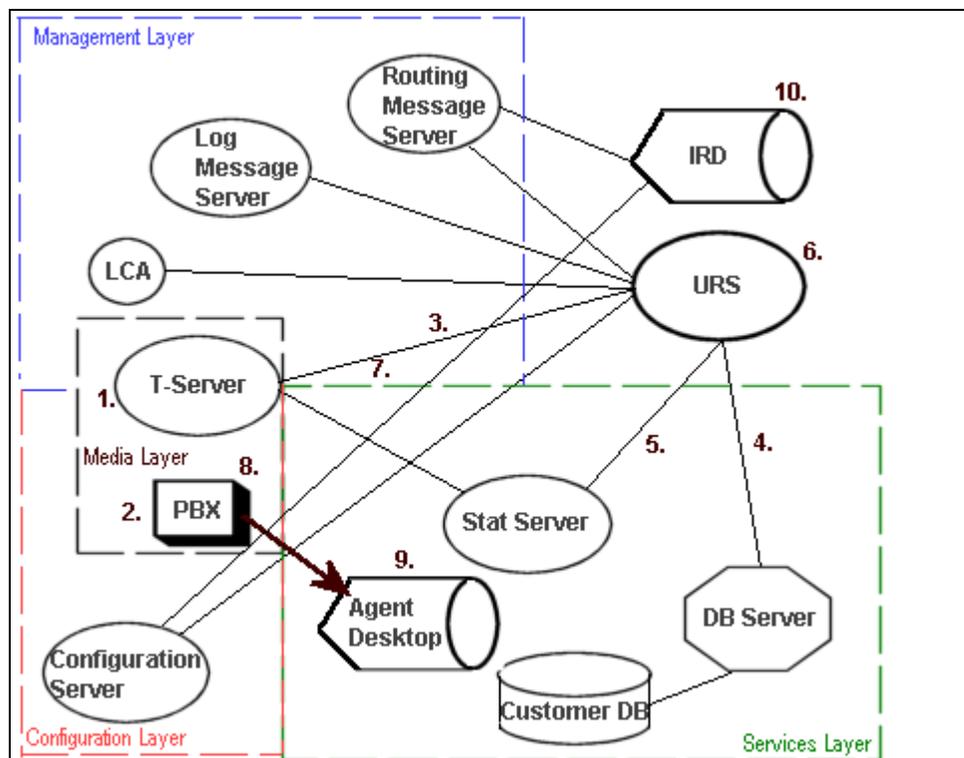


Figure 19: IRD and URS In Relation to Framework Layers

This example includes steps, such as collection of customer-entered digits (CED) and a database lookup, that are not necessarily part of every call flow but are included to give a sense of the kind of interaction processing you might choose to use.

Note: For information on multimedia interaction flows, see Chapter 4 on [page 103](#).

Example: Routing Steps

To illustrate routing architectural functionality, the flow diagram in [Figure 19](#) breaks routing down into ten basic steps:

1. Call Arrival
2. Collect Digits (CED) (optional)
3. Route Request
4. Database Lookup (optional)
5. Target Sub-List
6. Target Selection
7. Request Route Call

8. Call Routed
9. Screen Pop
10. Monitor Strategy

Example: The Call Flow in Detail

The steps in Figure 19 on [page 62](#) are explained in detail below.

1. Call Arrival

As soon as a call comes into the switch, T-Server creates a unique record for the call which it maintains until the call is released. T-Server tracks the status of every call through each step of the process and provides updated information to all relevant client applications.

2. Collect Digits (optional)

In this example, assume the switch has been programmed to send all incoming calls to an interactive voice response unit (IVR) for the collection of customer entered digits (CED). The call has not yet hit a Genesys routing point, so the switch is still controlling the processing of the call. Assume also that the IVR prompts for the customer's account number then sends the call back to a route point (6660) on the switch.

3. Route Request

Since 6660 is our special route point, the switch sends a message that gets interpreted by T-Server as an `EventRouteRequest` message. T-Server attaches the CED to the call record and passes the route request message on to all the client applications that registered to receive messages regarding DN 6660. Since URS registered DN 6660, as a client of T-Server it receives the message it has been waiting for, `EventRouteRequest`.

4. Database Lookup (optional)

Assume the strategy instructs URS to route each call according to the caller's customer service level, derived from the customer database. Through a DB Server connection to the customer database, URS looks up the customer's record and evaluates certain fields specified in the strategy. In our example, the `AcctNum` field is compared to the Caller Entered Digits to look up the customer's record and then the `Tier` field is used to determine the customer service level (`Platinum`, `Regular`, or `null`). Assume the caller has a customer service level of `Regular`.

5. Target Sub-List

Based on the results of the database lookup, URS proceeds to the next step of the strategy, target selection. Assume the strategy specifies that callers with a `Regular` customer service level must be routed to Agent Group B. URS determines the subset of qualified targets. Having limited the possible targets to the agents of Group B, URS requests the status of those agents (e.g., `WaitForNextCall` or `NotReadyForNextCall`) and receives the status messages from Stat Server.

In addition, URS can request statistical values from Stat Server in its quest for the best agent. In our example, we want to select the agent who has been available for the longest period of time, so URS requests the statistical value (Maximum) `StatTimeInReadyState` for all the agents in available status. Stat Server provides those values.

6. Target Selection

In order to select a specific target, URS uses strategy instructions and relies on the information it receives from Stat Server. URS evaluates each call independently and applies the status information and the statistical values it requested from Stat Server to determine the best target. In this example, there are only two agents available and only one of them is a member of Agent Group B. In a real world agent-group routing scenario, URS would take advantage of the complex capabilities of both the strategy and the data provided by Stat Server to route the call to the best agent.

URS selects Agent 2 as the best target.

7. RequestRouteCall

Once it has selected a target, URS sends a `RequestRouteCall` message to T-Server. This message includes the target specification (DN). T-Server forwards the request to the switch.

8. Call Routed

The switch routes the voice portion of the call to the DN specified by T-Server (which was selected by URS). In this example, it is the extension DN where Agent 2 is logged in, DN 6002. T-Server distributes the call detail, including the `TEvent` structure, to all the client applications that registered for DN 6002.

9. Screen Pop

One of the applications that registered for messages about DN 6002 is Agent 2's Desktop Application. Depending on the functionality of Agent 2's soft phone, the `TEvent` structure can be converted into a screen pop which arrives at the same time as the call. The screen pop can contain all the customer data T-Server attached to the call including the customer's account number, name, address, customer service level, phone number, and so on.

10. Monitor Strategy

From the IRD Loading List (accessed from the Monitoring shortcut bar), you can monitor Universal Routing Server and routing points. You can also open a Trace View window from either the Loading list or the Strategies list and view call counts and percentages for each object in a strategy. For instance, you could see at a glance how many `Platinum` calls have been routed, or what percentage of all calls are being routed to a queue.

Enterprise Routing Architecture

Genesys Enterprise Routing provides the environment for routing voice interactions at the enterprise premise level which can be single-site (switch) or multi-site (switches) routing. The diagram in [Figure 20](#) illustrates the general architecture for Enterprise Routing.

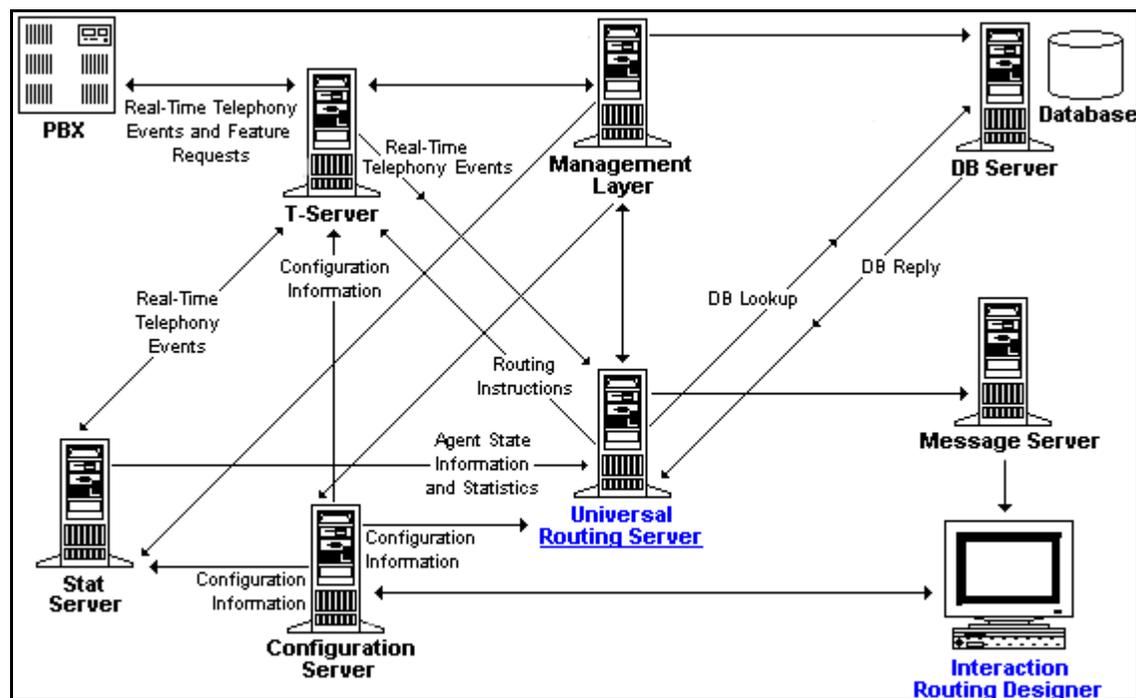


Figure 20: Enterprise Routing Architecture

As the diagram in [Figure 20](#) shows, T-Server is the only component that connects to the PBX. Management Layer, Universal Routing Server, Configuration Server, and Stat Server connect to T-Server. The remaining components connect to these servers as necessary to perform their tasks. Communication can be either one way or two ways between components.

For an example of how these components function when a call or interaction arrives at a PBX, see “Sample Interaction Flow, Enterprise Routing” on [page 66](#).

Note: In the diagram, the Message Server, which is part of Management Layer, is shown outside of Management Layer to illustrate how it functions in relation to Universal Routing Server and Interaction Routing Designer. For more information about Message Server, see the *Framework 8.0 Deployment Guide*.

Sample Interaction Flow, Enterprise Routing

This sample interaction flow explains how components function with each other when a customer contacts a business that uses Enterprise Routing. Figure 21 on [page 67](#) illustrates these steps.

Customer Contact

1. A customer contacts a business and the interaction enters the premise PBX switch system.
2. The PBX switch requests routing instructions from the T-Server, which creates a unique identifier for the interaction.
3. T-Server converts the request to a TCP/IP message (the language understood by Genesys applications) and passes the request to Universal Routing Server (URS).

URS Execution

4. URS executes the routing strategy designed in Interaction Routing Designer. The strategy includes instructions to collect customer-entered information.
According to the sample flow, URS first sends the interaction to an IVR to prompt the customer to enter an account number and language preference.
5. Customer enters the account number and a French language preference and the IVR attaches this data to the interaction.
IVR routes the interaction back to URS.
6. URS queries the customer database through the DB Server to determine the identity of the customer.
7. The strategy attaches the database information about the customer to the interaction.
8. URS queries Stat Server to determine which French-speaking agent is available.
9. URS learns from Stat Server that one or more agents are available and determines the agent to whom the interaction should be routed.
10. URS sends a `RequestRouteCall` message to the T-Server. The message contains the agent information (DN, Connection ID) and the collected customer information.
11. T-Server tells the switch to route the interaction to the Agent DN.
12. The PBX routes the interaction to the agent while T-Server provides the attached customer information to the agent's desktop application.

The diagram in Figure 21 on [page 67](#) illustrates the numbered steps of the Sample Interaction Flow.

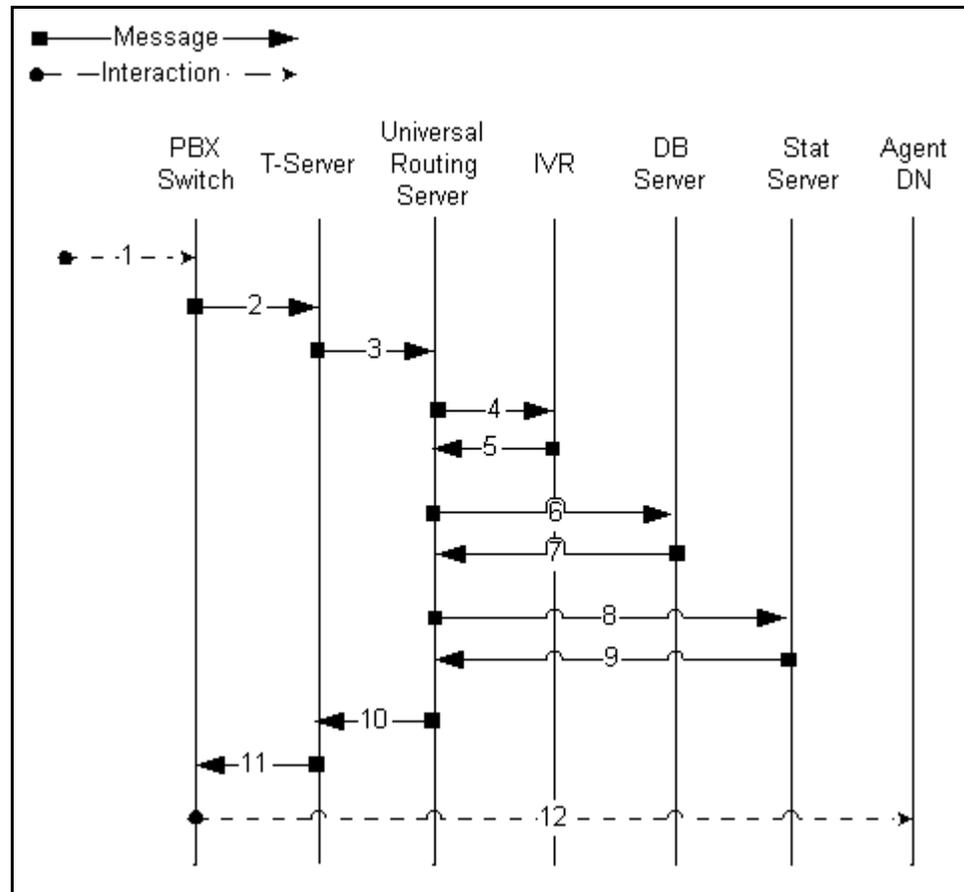


Figure 21: Sample Interaction Flow, Enterprise Routing

Network Routing Architecture

Genesys Network Routing provides the environment for routing interactions from a toll-free network to enterprise premise switches. Figure 22 on [page 68](#) illustrates the general architecture for Network Routing.

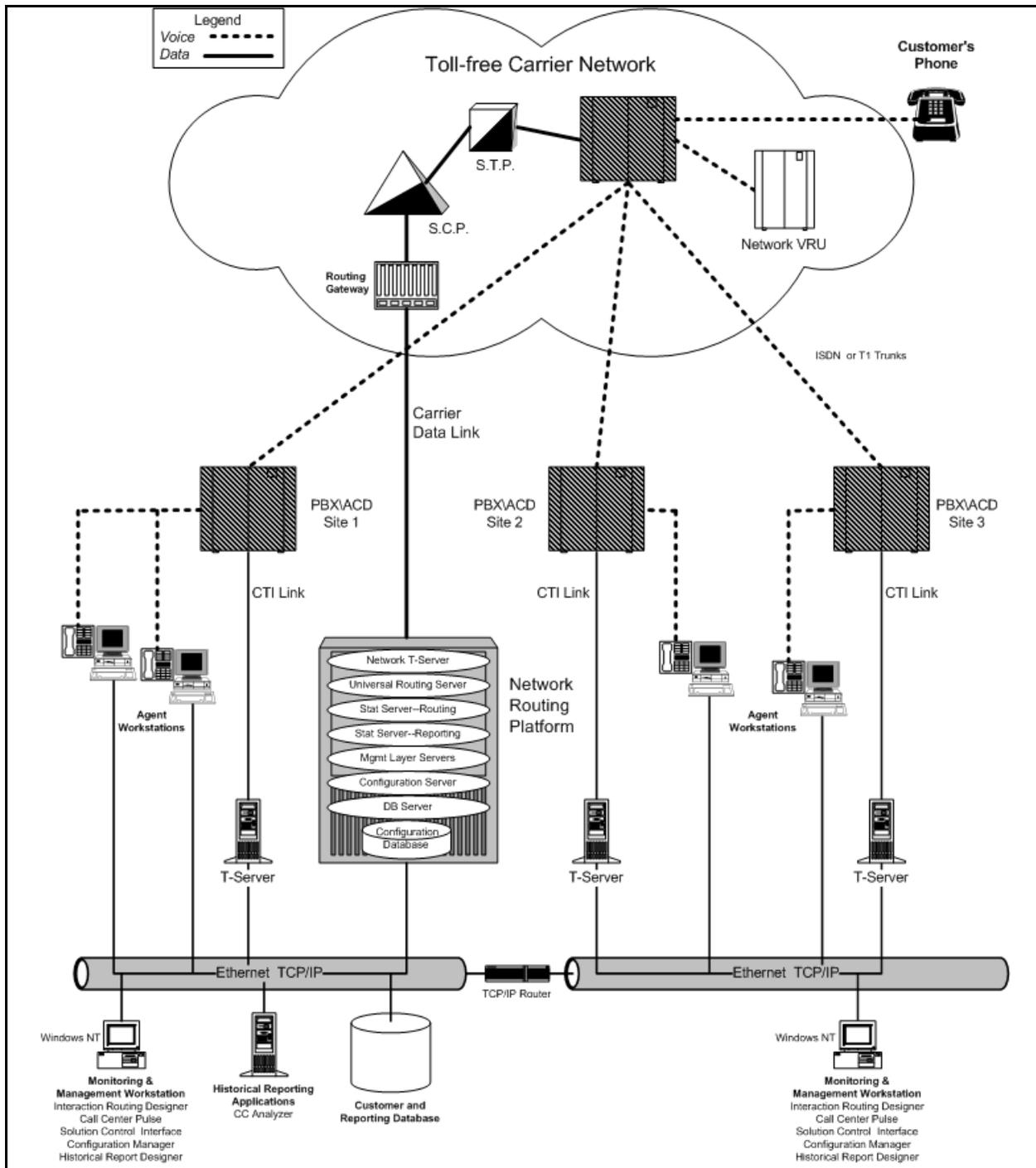


Figure 22: Network Routing Architecture

T-Servers connect through CTI links to PBX/ACD sites. T-Servers also connect to the Ethernet TCP/IP which, in turn, connects to the Configuration Database, DB Server, Configuration Server, Management Layer Servers, Stat Servers, Universal Routing Server, and Network T-Server. The additional connections required for Network Routing are connections between URS and

Network T-Server and between Stat Servers and Network T-Server. The remaining components connect to these servers as necessary to perform their tasks. Communication can be either one way or two ways between components.

For an example of how these components function when a call or interaction arrives at a PBX, see “[Sample Interaction Flow, Network Routing](#)” below.

For more information about the Genesys Framework servers, see the *Framework 8.0 Deployment Guide*.

Sample Interaction Flow, Network Routing

This sample interaction flow explains how components function together when a customer contacts a business that uses Network Routing.

Figure 23 on [page 71](#) illustrates these steps.

Customer Contact

1. A customer contacts a business through a Network Switch.
2. An interaction notification enters the Service Control Point (SCP) from a Network Switch.
3. Based on the Service Number dialed, the SCP determines where to trigger a routing request. The SCP generates a routing request to Network T-Server. The ANI, CED (caller-entered-digits), and DNIS are passed to the T-Server in the request along with other attached interaction data.
4. Network T-Server sends an `EventRouteRequest` message to Universal Routing Server (URS).

URS executes the strategy. The ANI, CED, DNIS, and other attached data are passed in the request.

URS and Target-Selection Analysis

5. URS consults Stat Server to determine the availability of the targets based on the skill and proficiency level requirements of the routing strategy.
6. URS learns from Stat Server that one or more of the targets are available, for example, an agent with DN 4444 on premise switch 2.
7. URS selects a target and from the data returned by Stat Server determines which premise T-Server will need to be consulted for an external routing point.

URS passes interaction information (Agent DN, Connection ID, User Data) directly to the Network T-Server.

8. The Network T-Server communicates with the premise T-Server to determine which external routing point, for example, 9999, should be reserved to receive the interaction and returns that DN to the Universal Routing Server.
9. URS then consults the Switch Access Code tables from the network switch to premise switch 2 for target type Target Agent. The table has a value of [DN.DL] in the Destination Source field. URS consults the DN group associated with the external routing point supplied by the premise T-Server in Step 7 (DN 9999). The DN group contains a single Network Destination type DN defined for the network switch. This completes the network number translation process.

Interaction Routing

10. URS sends a `RequestRouteCall` message to the Network T-Server for the derived destination. The `RequestRouteCall` message contains the DL produced by URS in Step 9.
11. The Network T-Server tells the Service Control Point to send the interaction to the derived destination for the target by the trunk that the Network Destination points to.
12. The Service Control Point instructs the Network Switch to route the interaction using the given Network Destination.
13. The interaction arrives at External Routing Point 9999 on the destination premise switch.
14. The premise switch sends the interaction arrival message to the premise T-Server. The rest of the interaction flow is now entirely controlled by the premise T-Server.
15. Since the premise T-Server knows the target agent DN, the premise T-Server instructs the switch to route the interaction to agent DN 4444.
16. The interaction is routed to the agent DN.

The diagram in Figure 23 on [page 71](#) illustrates the numbered steps of the Sample Interaction Flow in a Network Routing environment.

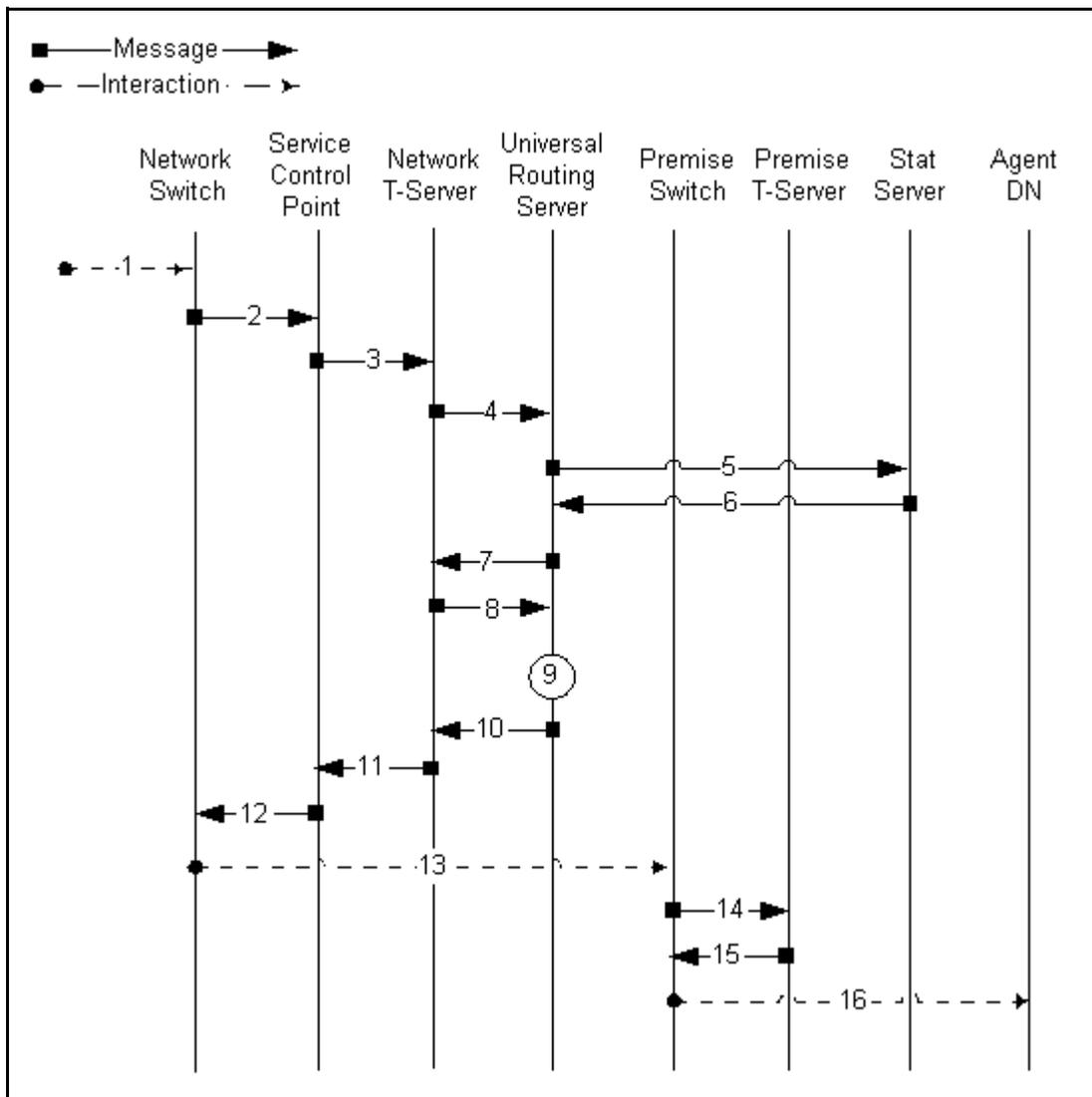


Figure 23: Sample Interaction Flow, Network Routing

URS as a Client to LDS

Note: This section presents general information on Load Distribution Server (LDS). For complete information on LDS, including high availability configuration, see the *Framework 7.2 Load Distribution Server User's Guide*.

For load distribution, you can use different configurations between T-Servers, LDS, and URS. For example, you can connect T-Servers to one LDS or to multiple URSs, as shown in Figure 24 on [page 72](#).

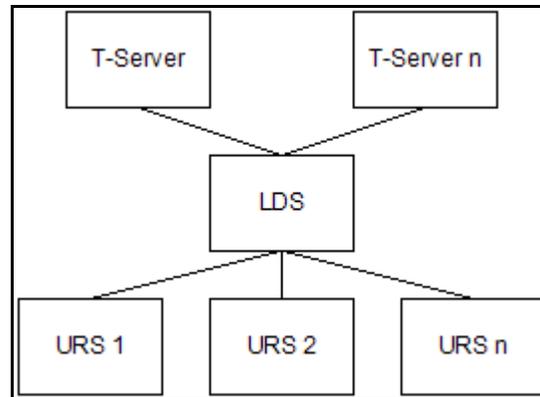


Figure 24: T-Servers, LDS, and Multiple URSs

This configuration can be used when the transaction rate (interactions/second) on the switch/T-Server pair is higher than a single instance of URS can handle. LDS will distribute routing requests (balance the load) among URSs.

- You can put one or multiple instances of URS on the same computer with multiple processors or distribute URSs on several computers when additional processing power is required.
- You can also connect multiple T-Servers to one LDS and any number of single instances of URS to meet the specified load. Or you can connect multiple T-Servers to multiple LDSs to any number of single instances of URS.
- LDS is application-type specific which means that URS(s) can only connect to LDS(s) that are assigned to it. For example, URS cannot connect to an LDS that is already assigned to Interaction Concentrator.
- LDS determines its type dynamically during run time by the first successful client connection to LDS. Therefore, there are no explicit options setting required. During deployment planning, identify each LDS functional type by defining the necessary connection between URS and LDS.
- URS can receive and process messages from multiple T-Servers of any media type through the same LDS (including T-Server for PBX and Network T-Server).
- LDS and URS do not need to exist on the same computer. However, you can strategically position LDS and multiple URSs on the same computer to optimize network traffic.

For information on configuring this and other types of load balancing, see Chapter 9, “Load Balancing,” on [page 235](#).

Other Solutions and Applications

In addition to working with Genesys eServices as described in Chapter 4 on [page 103](#), a routing solution can work with other Genesys solutions or applications to provide additional routing capability including Workforce Management.

Workforce Management unifies forecasting, employee scheduling and calendar management, monitoring of real-time agent-adherence and intra-day contact-center performance, and historical reporting into a contact center resource-planning application.

For more information, see “Configuring URS to communicate with WFM” on [page 209](#). Also see the *Workforce Management 7.6 Administrator’s Guide*.



Chapter

3

System Availability and Redundancy

Note: To use the High Availability functionality described in this chapter, you need High Availability licensing. For more information on this licensing feature, see the *Genesys Licensing Guide*.

This chapter provides information on system availability and redundancy. It also describes High Availability options for Universal Routing associated with three levels of product offerings: the base product, the Redundancy level, and the High Availability Routing level.

Universal Routing supports High Availability of contact center services through mechanisms in Management Layer such as automatic switch-over and recovery in case of failure, redundant configurations using backup components in Warm or Hot Standby, load distribution and sharing and, at the CTI level, HA Proxy and backup T-Server (premise) in Hot Standby.

This chapter contains the following topics:

- [Levels of High Availability, page 76](#)
- [Base Product Offering, page 77](#)
- [Redundancy Level, page 79](#)
- [Using LDS For Redundancy, page 82](#)
- [High Availability Routing, page 85](#)
- [System Availability/Redundancy Planning for Network Routing, page 89](#)
- [Recovering Interactions to a Routable State, page 90](#)
- [System Recovery and Statistic Reconciliation, page 95](#)
- [URS Options and System Availability, page 96](#)

Levels of High Availability

High availability refers to the percentage of time that a solution is available to process routing interaction services according to a specified performance criteria above a given performance threshold which depends on system architecture.

Universal Routing supports high availability with three product offerings or levels: the base product, the Redundancy level, and the High Availability Routing level. The first or base product offering is a prerequisite to the Redundancy level which, in turn, is a prerequisite to the High Availability Routing level. Each level of availability requires the purchase of additional features or components. The level of high availability that you choose depends on business criteria and routing capabilities required for your particular routing solution.

The base product, Redundancy level and High Availability Routing level include different high availability options:

1. The **base product** is the standard product offering available with no additional charge. This offering has built-in support for basic high availability such as exception handling (Non-Stop Operation), Advanced Disconnect Detection Protocol (ADDP), alarms, logs, security, dynamic configuration updates, consistency checking, and central solution control through Management Layer. Apart from the Management Layer and the Configuration Layer, the base product has no additional redundancy capabilities.

The base product also has no Genesys-controlled application redundancy. If a component server fails, the service interruption results in the loss of active interactions, data, and prevents the routing of new requests until service restoration. In the base product offering, failed instances of components can be automatically or manually restarted.

2. The **Redundancy level** includes the high availability mechanisms of the base product offering plus primary and backup processes (primary and backup URS in Warm Standby). The Redundancy level also includes failover control by Management Layer, options for load sharing and redundancy through Load Distribution Server (LDS), and geographically distributed Configuration Servers.

Service recovery occurs through automatic Management Layer failover processes. The addition of LDS architecture protects against service degradation because new interactions will be sent to surviving instances of URS.

3. The **High Availability Routing level** includes the base product offering, the Redundancy level, plus the option for CTI level routing through HA Proxy and Hot Standby (premise) T-Server, load sharing, and high

availability through combinations of URS and LDS. (To use the High Availability Routing level, you must purchase HA Proxy and backup T-Server (premise) in Hot Standby.)

The High Availability Routing level is offered for business-critical services requiring advanced high availability capabilities. See “High Availability” on [page 39](#) for more information.

If a component fails, service may be on hold briefly, which could result in non-routable interactions that are recoverable. Virtually no loss of data or interactions occur for affected interactions processed by the failed URS. Only a routing delay occurs for these interactions.

In any level of High Availability, interactions are not lost. Only advanced routing services provided by Enterprise Routing or Network Routing are temporarily interrupted. (See “Recovering Interactions to a Routable State” on [page 90](#) for more information.)

For information on planning for Network Routing, see “System Availability\Redundancy Planning for Network Routing” on [page 89](#).

Note: If you need high availability functionality (more than the base product provides) then for voice interactions (no matter whether pure voice or a mixed deployment), either Redundancy level or High Availability level routing is required. For pure multimedia deployments (no voice), this not required. High availability is achieved with Interaction Server, which can support pairs of standby URSs.

Base Product Offering

The base product is most appropriate for single site contact centers. As the standard product offering, the base product supports high availability through basic mechanisms in Management Layer and Configuration Layer.

The base product offering supports high availability using application failure management in Management Layer and Configuration Layer including:

- Basic built-in support for exception handling (NSO).
- Advanced Disconnect Detection Protocol.
- Alarms, logs, security, dynamic configuration updates, consistency checking, and central solution control through Management Layer.

For more information on alarms, logs, security, dynamic configuration updates, consistency checking, and central solution control through Management Layer, see the *Framework 8.0 Management Layer User's Guide*.

Although redundancy is not offered at the base product level, a form of load balancing can be achieved using multiple primary URSs.

Recovery includes the automatic restart of failed instances of applications.

PBX routing capability can compensate for URS service interruption.

Application Failure Management

Application failure management involves minimizing the impact of service failure (the loss or reduced availability of components to function in a solution), or interruptions to service as a result of internal or external events.

If an instance of URS fails, Management Layer restarts URS automatically if you selected `Autorestart` in the `Start Info` tab of the `URS Application` object. After a URS is restarted, URS begins to route new interactions submitted after startup is complete. Existing interactions pending on the routing points before URS restarts are ignored.

This setup is generally used by a single site contact center where PBX/ACD routing is used as the redundant routing mechanism

Note: Whether URS processes only newly-submitted interactions after restart as the primary server or also processes existing routing requests is controlled by the option `pickup_calls`. By default, `pickup_calls` is set to `false`. In the base product offering, where only a primary URS is supported, `false` is the only valid value for `pickup_calls`.

Exception Handling

Exception handling (Non-Stop Operation or NSO) is a basic high availability mechanism that enables URS to continue running even though an internal defect occurs that interrupts the normal flow of the application.

An exception-handling function enables servers to continue to run even though an illegal operation is attempted. This function monitors the exceptions and attempts to prevent server termination by skipping the program block from which the exception originated.

An exception is an interruption to the normal flow of a server. Exceptions are generated by the operating system when a program attempts to perform an illegal operation and may cause the server to stop working.

This process uses functionality called Non-Stop Operation. For information on setting up Non-Stop Operation, see “Non-Stop Operation” on [page 341](#). Also, see the *Framework 8.0 Management Layer User’s Guide*.

Advanced Disconnect Detection Protocol

Advanced Disconnect Detection Protocol (ADDP) is used URS. ADDP helps detect a connection failure on both the client and server side. After a communication failure is detected, the application repeatedly attempts to reconnect to the appropriate resource. If a redundant process is not configured (as in the base product offering), the reaction is a repeated attempt to restore the communication session with the same process.

For more information on application failure management functions, see the *Framework 8.0 Management Layer User's Guide*.

Redundancy Level

In addition to the mechanisms for high availability offered with the base product, the Redundancy level of high availability includes:

- URS Redundancy using Warm Standby.
- Load distribution.
- Smart Registration.
- Geographically distributed Configuration Server.

Note: URS redundancy with backup URS in Warm Standby or load distribution using URS requires additional pricing.

The Redundancy level of high availability is most appropriate for contact centers that need multiple instances of URS to achieve routing requirements.

Like the base product offering, with the Redundancy level you can also achieve load balancing using two or more URSs ([page 237](#)).

Warm Standby Redundancy Type

Warm Standby means that a backup application server is initialized and ready to take over the operation of the primary server. Figure shows how you designate a URS as Primary or Backup in the

Using this redundancy type, you can reduce to a minimum the time interval between the detection of a primary server failure and the failover to the backup component. The time interval between the primary server failure and the activation of the backup server as the primary is less than the time interval required for system initialization of a server during the startup process.

The backup server in Warm Standby recognizes its role as a backup and does not process client requests until its role is changed to primary by Management Layer. When a connection is broken between the primary server and the Local Control Agent (LCA) running on the same host, LCA reports to Management Layer on the failure of the primary process and notify the backup process to

change its role from backup to primary and to start processing all new routing requests.

Figure 25 shows how Redundancy Type is specified in the URS Application object.

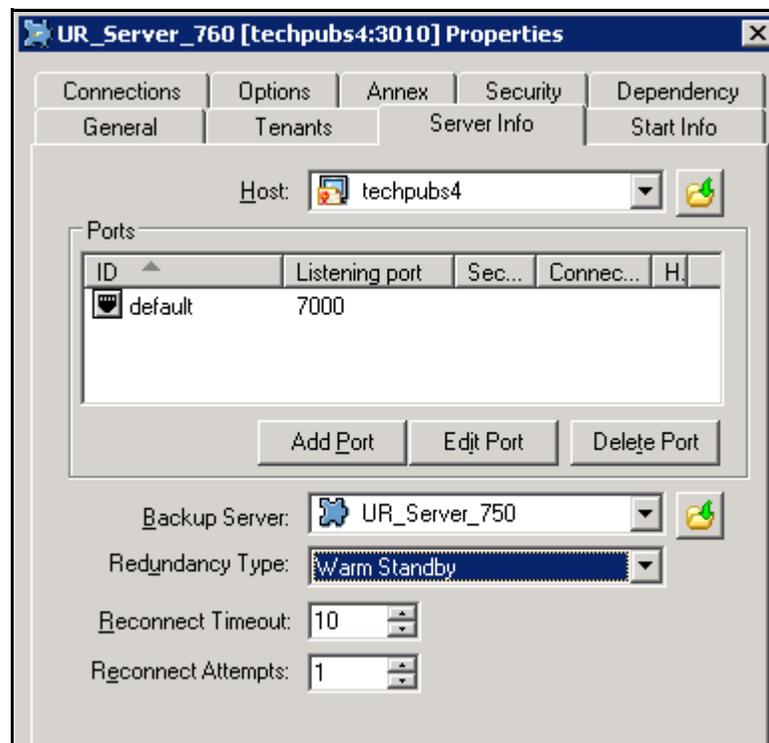


Figure 25: URS Application, Specifying Backup Server and Redundancy

URS Redundancy Using Warm Standby

A backup URS in Warm Standby is started, with all appropriate connections, during the solution startup. It has all of the appropriate connections to servers (including T-Server, Stat Server, DB Server, Configuration Server, and Message Server) and registers all DNs.

This backup URS neither processes routing requests nor monitors the state of interactions. It does not maintain statistics received from Stat Server or request Stat Server to open new statistics when new routing target resources are added.

Both the backup URS in Warm Standby and the primary URS receive information from T-Server and log interaction information. When strategies are loaded to the primary URS, they are also automatically loaded to the backup URS.

System redundancy is achieved using primary and Warm Standby URS pairs. Once the failover is complete, the backup URS, now the primary URS, processes newly-submitted routing requests. Figure 26 on page 81 illustrates this. (*T-Server n* refers to the total number (*n*) of T-Servers connected to the primary and backup URSs.)

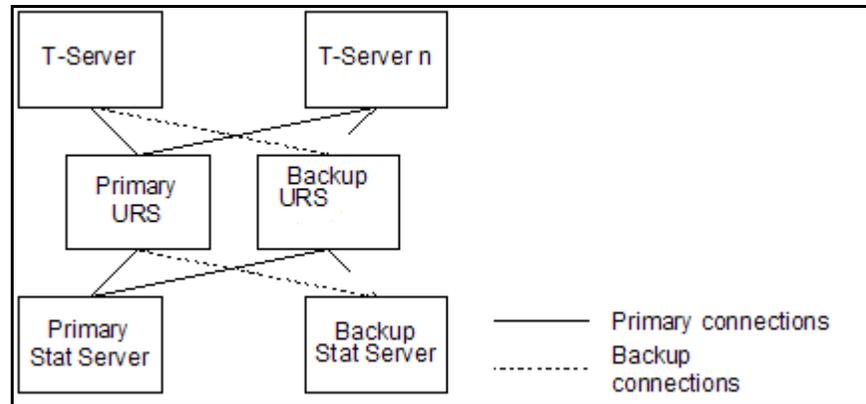


Figure 26: Redundancy Using URS Warm Standby

This form of redundancy is most often used in contact centers where interaction volume can be handled by one URS. Redundancy is added with one backup URS in Warm Standby mode. The Warm Standby server speeds up system recovery time without waiting for a manual or automatic restart of the primary.

Note: Backup servers should be configured with identical options and settings to primary servers to help ensure that backup servers process in the same way as primary servers.

URS and Failover

If the primary URS unexpectedly terminates, the LCA in Management Layer notifies the backup URS in Warm Standby to become the primary server.

The duration of the failover is defined as the start time of Management Layer detecting that a URS is down to the time the Warm Standby server starts routing interactions.

After failover is complete, the new primary server starts processing new submitted routing requests. It does not replay any strategy for routing requests submitted before the failover, which means that all routing processing handled by the former primary server and during failover is lost.

Note: *URS non-routable interactions* does not mean that the interactions are unrecoverable. It means that URS cannot take control of those interactions and attempt to route them until after failover is complete. For information on recovering interactions, see “Recovering Interactions” on [page 90](#).

Important

For media type such as e-mail, if `pickup_calls` is turned on, eServices can potentially submit thousands of routing requests for processing and cause serious network traffic problems. When URS is used for routing multimedia interactions from Genesys eServices and you have configured a backup URS in Warm Standby, set the `pickup_calls` option to `false`.

When URS Warm Standby mode is configured, URS:

- Ignores the `on_router_activated` option setting if you set the value of this option is set to `route`. Stated another way, when URS Warm Standby mode is configured, the valid values for the `on_router_activated` option are `default` and `ignore`.
- Routes accordingly if the value for the `on_router_activated` option is set to `ignore` (ignore old interactions) or `default` (send interactions to the default destination).

If you do not specify a value for `on_router_activated`, URS in Warm Standby sends old interactions to the default destination.

If you set up the switch to act as a backup mechanism, then the switch routes any pending interactions. (Default routing is configured at the switch. For example, default routing is configured to an ACD routing point.)

Notes: If you install the URS in Warm Standby mode on a different computer from the primary URS, the URSs (primary and backup) will not both be impacted in case of a hardware failure on one of the two computers.

Two `EventDiverted` messages can sometimes be sent for one `EventQueued` during switchover from a primary to a backup URS.

Using LDS For Redundancy

Note: This section presents general information on Load Distribution Server (LDS). For complete information on LDS, including high availability configuration, the Weighted Round Robin distribution mode, and the `start-calc-threshold` option, see the *Framework 7.2 Load Distribution Server User's Guide*. For information on using LDS for load balancing, see [page 236](#).

LDS can be used to distribute requests among multiple URSs and combine their processing power to increase total throughput.

When you use LDS, you can achieve redundancy in a configuration with any number of single instances of URS to meet a specified load without the use of Warm Standby URS. This is only if the number of instances of URS is equal to an $n+1$ configuration, where n is the minimum number of URS instances that

are actually needed to process the projected load. If only n instances are used there would be a risk of service degradation in case of a failure.

Primary URS plus backup URS in Warm Standby is an active/passive configuration. LDS plus URS is an active/active configuration.

In this configuration, when one URS fails, LDS directs new routing requests to the remaining URSs. After the failed URS is restarted through Management Layer, it is included again in the LDS load distribution.

Redundant configurations using LDS with multiple single instances of URS offer a growing path for Enterprise Routing and Network Routing. For example, you can add URS or LDS components to an existing infrastructure to handle extra loads in a configuration with LDS and any number of single instances of URS required to meet a specified load.

In addition to this, URS has the ability to support a mixture of connections between LDS and T-Server.

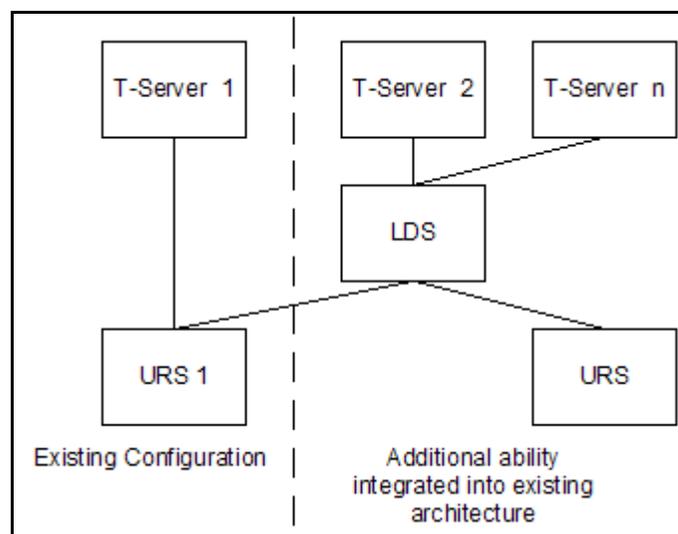


Figure 27: Phased-In Approach when Introducing LDS

There are two reasons to support this configuration:

- Within a large system, you could have a grouping with n_1 T-Server(s) connected to URS₁, n_2 T-Server(s) connected to URS₂, and so on. The mix of connections enables you to take a phased-in approach for introducing LDS. Ideally, you maintain the original URS configurations and introduce new ones with LDS usage. In case of problems, you can switch back to the original configuration.
- This configuration enables other media type T-Servers that require a connection to URS to submit routing requests.

You can incorporate LDS into a new contact center without disturbing the existing architecture.

For detailed information on LDS including a standby configuration, see the *Framework 7.2 Load Distribution Server User's Guide* (the latest published version).

URS Options at the Redundancy Level

Two URS options that you can use at the Redundancy level of high availability are `pickup_calls` and `on_router_activated`.

- The option `pickup_calls` is set to `false` as the default setting during installation. When set to `true`, this option uses the attribute in the TEvent set called `Smart Registration` to recognize interactions already existing on routing points even before URS registers these routing points during startup.
- The option `on_router_activated` is ignored by a backup URS in Warm Standby only if the option is set to `route`. Pending interactions submitted before failover is complete are not subject to a replay of the strategy and URS does not route these interactions to the default destination. These pending interactions are routed by the switch (if the switch is set up as a backup mechanism).

For more detailed information see, “URS Options and System Availability” on [page 96](#).

Geographically Distributed Configuration Servers

Management Layer supports a geographically-distributed configuration environment. For example, you might have an environment in which multiple Configuration Servers act as proxies to one master Configuration Server. In this environment, the master Configuration Server provides information to the Configuration Server Proxies. Changes to objects in the master Configuration Server are communicated to the Configuration Server Proxies immediately. Configuration Server Proxies receive information from the master Configuration Server and provide information to Configuration Server clients. Applications that have a read-only relationship to Configuration Server, such as URS, can be connected to Configuration Server Proxy. Since the Configuration Server Proxy contains information on all configuration objects, interaction processing continues uninterrupted.

Note: If a Configuration Server Proxy unexpectedly terminates, the master Configuration Server *does not* act as a backup to applications connected to the proxy.

Termination of the master Configuration Server does not interrupt application services such as routing services. However, URS bases routing decisions according to the last known Configuration Layer objects.

For more information, see the section “Distributed Configuration Environment” on [page 117](#).

High Availability Routing

In addition to the high availability options of the base product level and the Redundancy level, the High Availability Routing level offering includes the following:

- HA CTI level routing through HA Proxy and backup T-Server (premise) in Hot Standby. (You must purchase HA Proxy and backup T-Server (premise) in Hot Standby for the High Availability Routing level.)
- High availability routing through the Hot Standby setting in backup URS
- Load sharing and high availability through combinations of backup URS and LDS in the Hot Standby mode

High Availability Routing is achieved by using redundancy to eliminate single points of failure on both the routing level and CTI level.

For information on HA Proxy installation and architecture, see the *Framework 8.0 Deployment Guide*.

For specific configuration instructions for an HA Proxy associated with a T-Server, see the appropriate T-Server *Reference Manual*.

Hot Standby Redundancy Type

Note: URS checks for any update of licensing feature `router_ha_option` upon startup.

Hot Standby is a term that describes the redundancy type of a backup application server that remains initialized with clients connected to both the primary and the backup servers at startup. In this redundancy type, the primary server always synchronizes its state with the backup. Data synchronization and existing client connections to the backup ensure higher service availability.

In Enterprise Routing, Hot Standby redundancy type is available at the High Availability Routing level in the pricing structure. This level of routing requires the purchase of Hot Standby (premise) T-Server or HA proxy. (See “Levels of High Availability” on [page 76](#).)

Note: Hot Standby is not available at the network level of Network Routing. However, it is available at the premise level using Enterprise Routing.

URS and Hot Standby

Backup URS in Hot Standby is started during solution startup. This Hot Standby URS has all of its connections to appropriate servers (T-Server, Stat Server, Message Server, Configuration Server, DB Server). It registers all DN's. URS in Hot Standby does not process routing requests but monitors, in parallel with its paired primary URS, the state of interactions waiting to be routed at the routing point. It also tracks the location of interactions such as interactions currently waiting on an IVR port or other treatment devices for busy treatments.

When the backup URS in Hot Standby becomes the primary server based on LCA notification on failover, the backup URS becomes active and starts processing new interactions submitted after failover is complete. For interactions pending on routing points (those submitted before failover and not yet routed), URS replays the strategy and attempts to route the interactions as follows:

- URS replays the strategy and attempts to send interactions that are already at Interactive Voice Response (IVR) ports (for example, `EventEstablished` received) for a busy treatment directly from the IVR to the agent.
- URS replays the strategy and attempts to transfer the interactions that are at an Automated Call Distribution (ACD), Vector DN (VDN), or Control DN (CDN) for a busy treatment to agents.

The success or failure of the transfer depends on the switch because, for some switch types, interactions are not allowed to be transferred while they are set at the ACD, VDN, or CDN level. In such cases, configure the switch ACD, VDN, or CDN for busy treatments to default route the interaction to a controlled routing point.

For more information on recovering interactions, see “Recovering Interactions to a Routable State” on [page 90](#).

Note: Backup servers should be configured with identical options and settings to primary servers to help ensure that backup servers process in the same way as primary servers.

URS Hot Standby without LDS in Enterprise Routing

A high availability system is achieved by eliminating a single point of failure in a system configuration. Based on this, the user deploying Enterprise Routing should consider redundancy not only on the routing level but also on the CTI level. A backup URS in Hot Standby should therefore go hand-in-hand with backup T-Server and HA proxy in Hot Standby.

This form of high availability routing using a backup URS in Hot Standby without LDS is most likely used in contact centers with an interaction volume that can be handled by one URS.

High availability is achieved by adding one backup URS in Hot Standby mode to a system configuration. The Hot Standby URS replays the strategy for pending interactions waiting at routing points and other treatment devices (DNs) that URS monitors after failover is complete.

Figure 28 illustrates a system configuration with HA Proxies, primary T-Server and backup T-Server in Hot Standby, and primary URS and Hot Standby backup URS, Stat Server and backup Stat server.

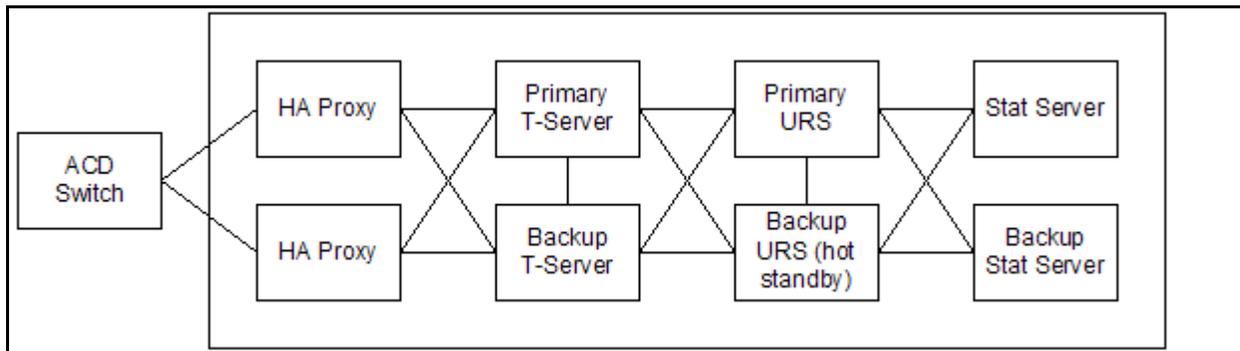


Figure 28: HA Routing with Backup URS in Hot Standby

Note: Hot Standby T-Server and HA Proxy are offered on premise T-Servers only.

Figure 29 shows a system configuration using a backup URS in Hot Standby and multiple (n) T-Servers. After failover is complete, URS processes new routing requests, replays the strategy and attempts to re-route interactions last known waiting at the routing point and other treatment devices (DNs) that URS monitors.

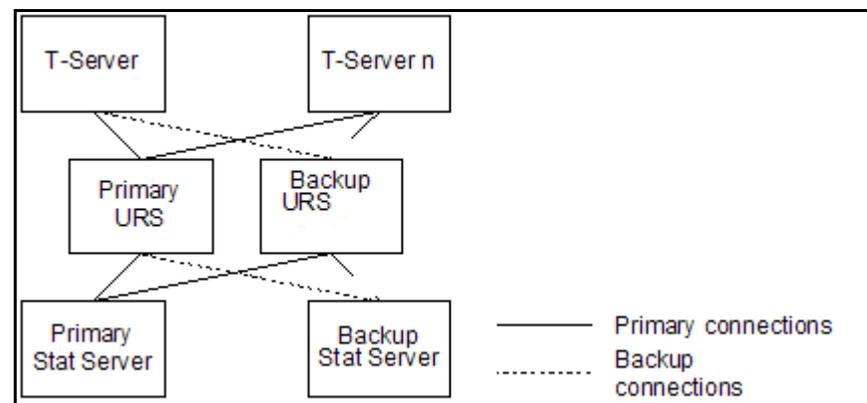


Figure 29: HA Routing with Backup URS (Hot Standby)

URS Hot Standby with LDS in Enterprise Routing

High availability routing using primary and backup URS Hot Standby pairs and LDS is used in contact centers requiring Enterprise Routing and multiple

URSs to handle a high volume of interactions. It is also appropriate for a combination of Network Routing and Enterprise Routing using the same URS. When LDS is used, high availability can be achieved by using a $2(n+1)$ URS configuration (primary and backup) where n refers to the number of single instances required to meet the specified load. (Since it is nearly impossible to scale a system exactly to the projected load, a $2(n+1)$ architecture is a good practice.)

Figure 30 illustrates one possibility. (For purposes of readability in Figure 30, the connections between the backup URSs and the primary and backup Stat Servers are not shown.)

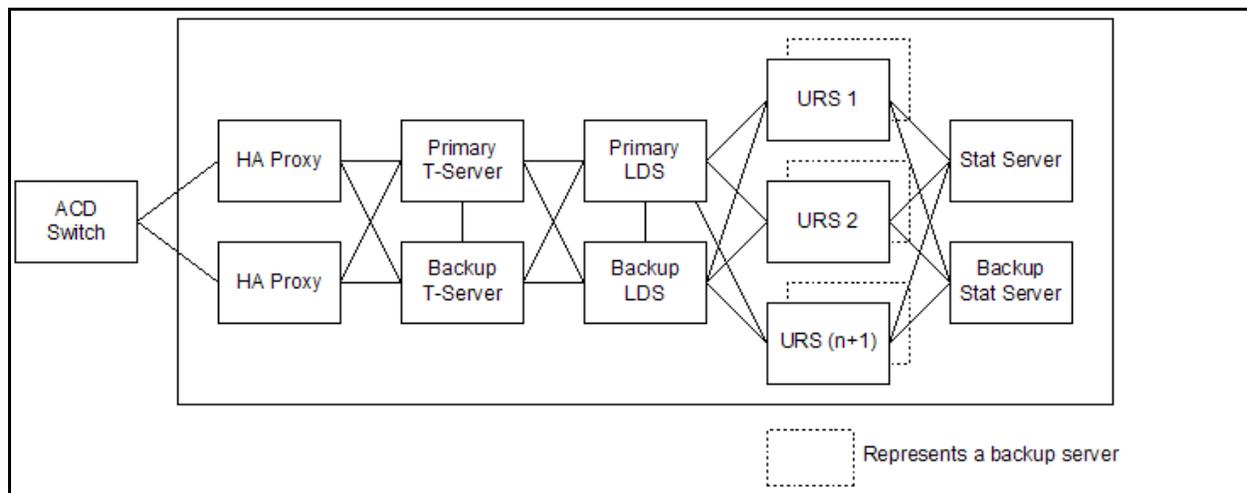


Figure 30: HA Routing with LDS and Backup URS (Hot Standby)

When one primary URS shuts down, new routing requests are redirected to the remaining servers until failover is complete. After the backup Hot Standby URS becomes the primary URS, it replays the strategy for the pending interactions waiting in the routing point and other treatment devices (DNs) that URS monitors.

In this instance, LDS distributes the load the same way it would to multiple single URSs. But, in this case, all events for a given interaction are sent to both primary and backup URS pairs.

URS Options Impacting High Availability Routing

In addition to the licensing feature (see [page 75](#)), two URS options impact high availability routing: `pickup_calls` and `on_router_activated`.

- The option `pickup_calls` can be set to `on` or `off` according to business criteria for routing interactions. This option should be used for high availability routing and voice media only. It should not be set at the virtual routing point level. When set to `on` this option uses the attribute in the

The event set called Smart Registration to recognize interactions already existing on routing points even before URS registers these routing points during startup.

If the option `pickup_calls` is set to on, an excessive number of e-mails waiting at the virtual routing point for processing can cause eServices to submit an unlimited number of routing requests for these e-mails and create network traffic problems.

- The option `on_router_activated` is ignored by the backup URS in Hot Standby. URS replays the strategy for pending interactions on routing points.

For more information on URS options, see the *Universal Routing 8.0 Reference Manual*.

System Availability\Redundancy Planning for Network Routing

System availability and redundancy planning for Network Routing requires an understanding of routing concepts related to the following three categories:

1. Network Routing involves routing interactions from the carrier network to the premise PBX at the ACD/CDN level, then the PBX/ACD routing of interactions to the agent level.
2. The Network Parking Platform allows interactions to be staged at the network level before routing to the premise PBX at the ACD/CDN level (This is similar to the first category except with the Network Parking Platform.)

Network Routing with a Network Parking platform can be achieved using the same methods as described for Enterprise Routing with Hot Standby URS and LDS. See “URS Hot Standby with LDS in Enterprise Routing” on [page 87](#).

3. Network Routing from the carrier to the agent level uses a two stage approach. In the first stage, interactions are routed from the carrier to the premise PBX/CDN level. In the second stage, interactions are routed to the agent level using skill based routing functionality as in Enterprise Routing. In this scenario, the same instance of URS service is for both the network and enterprise (premise) level of routing.

Availability and redundancy planning for Network Routing Category 1 can follow the planning as described in “For information on configuring this and other types of load balancing, see Chapter 9, “Load Balancing,” on [page 235](#).” on [page 72](#). This is because Network Routing using backup URS in Hot Standby without Network Parking Platform is impractical due to strict routing time requirements defined by the network carrier (usually falling within 250 to 500 ms). Routing decision time is measured from the time that the network

carrier sends the routing request to URS to the time that the routing request is set up at the routing target which is usually a CDN or ACD on a premise switch. (Failover time to Hot Standby backup URS plus the time to replay the strategy in most cases exceeds the routing time constraints of the network.)

Availability and redundancy planning for Network Routing (category 2 and 3 above), can follow the planning as described in “Redundancy Level” on [page 79](#) and in “High Availability Routing” on [page 85](#).

Enterprise (premise) Routing (as described in category 3) and Network Routing with the Network Routing Platform can benefit from backup URS in Hot Standby and LDS features. Such set ups vary case by case depending on business needs, the network carrier and the Network Parking Platform time requirements.

Note: Network T-Server does not support Hot Standby. Therefore, the High Availability Routing level is designed for premise-level routing (enterprise routing single site and/or multi-site) or a combination of premise and network routing but not network routing alone.

Recovering Interactions to a Routable State

Different levels of URS service interruption can take place regardless of which level of high availability is used (base, Redundancy level, or High Availability Routing level). The impact of service interruption varies depending on the level with the least impact at the High Availability Routing level.

During failover time (the time interval between the detection of the failure of the primary URS and the completion of the failover process to the backup server), URS is unable to control or route interactions. Interactions already in progress and interactions arriving during failover are in a non-routable state.

In any level of high availability, the interactions are not lost. Only advanced routing services provided by Enterprise Routing or Network Routing are temporarily interrupted.

Note: A low percentage of interactions arriving after the completion of failover can have a non-routable state. For example, interactions arriving after failover is complete are in a non-routable state if a switch was set to a default time out and the interaction was removed from the routing point.

To recover non-routable interactions (to a routable state), you have two methods including using only PBX as a simple backup mechanism and a combination of PBX and URS routing-point pairs. As with any backup mechanism, the backup requirement and corresponding solution varies

according to user business needs. The decision that the user finally makes is a balance between the cost of the infrastructure and software versus the cost of downtime or reduced service quality.

Using PBX as a Backup Mechanism

Use PBX as a backup mechanism to change non-routable interactions to a routable state.

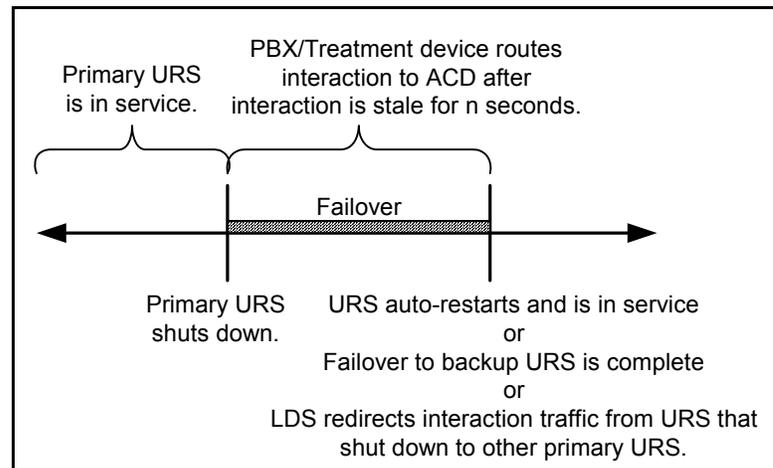


Figure 31: PBX as a Backup Mechanism

In this recovery method, design the PBX's routing point, IVR (when used as a busy treatment platform), and other treatment devices that URS monitors, to route those interactions that URS has lost control of during the failover period to the ACD where the agent is logged in. [Figure 31](#) illustrates this. As indicated, when failover is complete, interactions are again routed by a URS.

If you are using this PBX backup mechanism, set the switch to route interactions at the routing point, treatment DN, or IVR port to the ACD where the agent is logged in when the interaction is stale for a time interval that is between the average wait time in queue and the maximum wait time in queue (or average failover time, whichever is greater).

Depending on business needs, this time period should be tuned so that the PBX and URS would not compete on controlling the calls under a normal operating environment.

Using PBX and a Backup Routing Point

Use PBX and a backup routing point as a mechanism to change non-routable interactions to a routable state.

Backup Routing Point with URS in Warm Standby

In this case, the recovery method requires using a pair of routing points, one primary and one backup. The primary routing point is used for URS routing under normal conditions. The secondary or backup routing point is used for routing interactions under failover conditions. On both the primary and backup routing points, load the same routing strategy. Figure 32 on page 92 illustrates this.

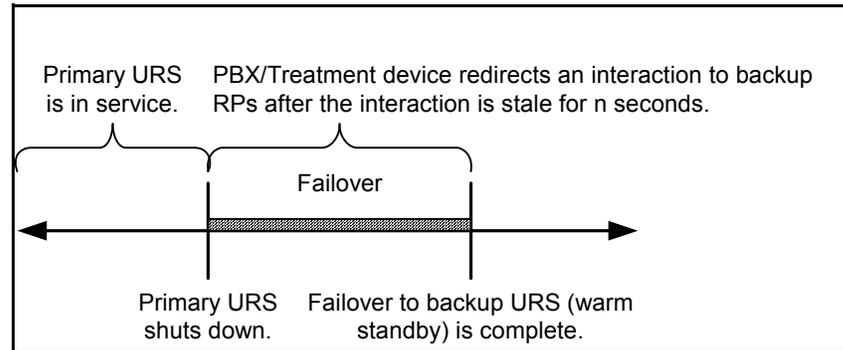


Figure 32: Backup URS with Warm Standby Only

During failover time, stale interactions at primary routing points and related busy treatment DN's (ACD, IVR/VTO) are redirected by the switch to backup routing points where URS service is resumed until failover is complete, a new URS is automatically restarted, or an existing primary URS is in service to take over the additional load. After failover is complete, the new primary URS will resume routing service and also route interactions from backup routing points. If you are using this backup mechanism, set the switch to route interactions at the routing point, treatment DN, or IVR port to backup routing points when the interaction is stale for a time interval that is between the average wait time in queue and the maximum wait time in queue (or average failover time, whichever is greater).

LDS with Multiple Primary URSs

In this case, the recovery method requires a set up at the switch which is similar to Figure 32 (backup URS with Warm Standby only) except that the stale period before the switch redirects the interaction to a backup routing point is between the average wait time in queue and the maximum wait time in queue (or LDS switch over to redirect traffic to a remaining primary URS).

Figure 33 illustrates this.

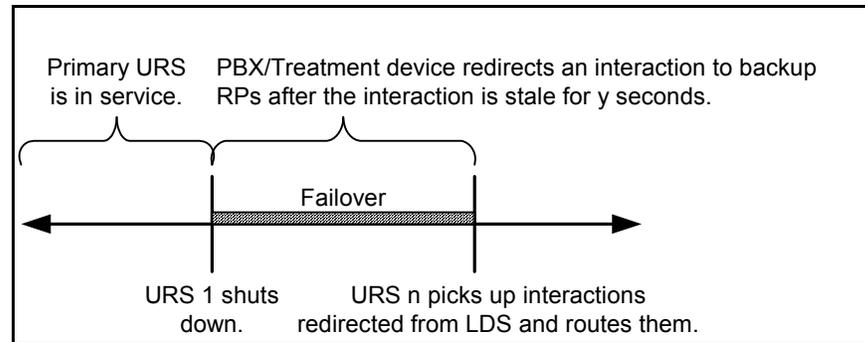


Figure 33: LDS with Multiple Primary URSs

After LDS recognizes the fail status on one URS, LDS redirects interactions to the remaining n URSs. These primary URSs route interactions from the backup routing points.

LDS with Multiple Primary URSs and Backup URS in Hot Standby

In this case, the recovery method requires a backup URS in Hot Standby working in parallel with its paired primary URS that monitors the interaction state for interactions waiting to be routed at the routing point. It also tracks the location of interactions (for example, interactions currently waiting on the IVR ports for busy treatment). [Figure 34](#) illustrates this.

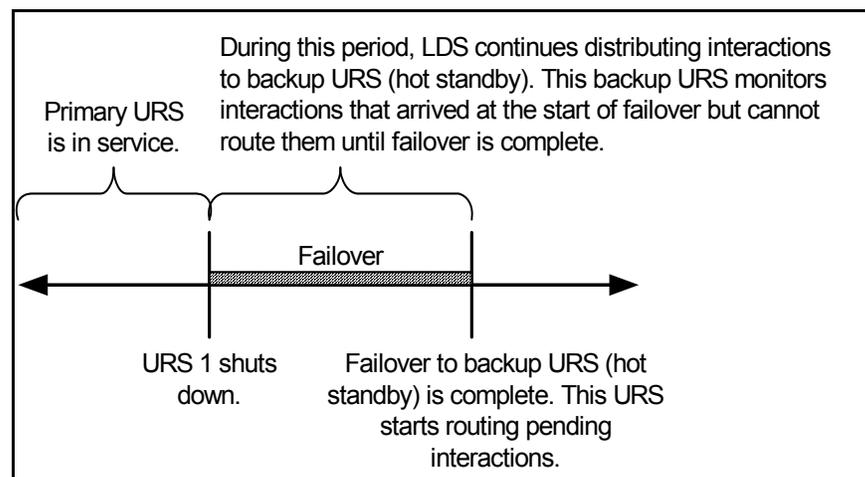


Figure 34: LDS with Multiple Primary URSs and Backup URS in Hot Standby

After failover is complete, URS replays the strategy and attempts to route interactions pending on the routing point and busy treatment DNs (those interactions submitted before and during failover that are not yet routed) and continue to route new submitted routing requests. Usually, the failover time is minimal and setting up the switch to redirect interactions to backup routing points as described previously is optional.

Note: For both recovery methods (using PBX as a backup mechanism and using PBX plus URS controlled routing point pairs), adjust the maximum wait time in queue to avoid competition between PBX and URS routing and to provide maximum protection during the failover time.

Formula for Non-Routable Interactions for URS Warm Standby Definition

In the URS Warm Standby definition, when the backup URS becomes the primary after failover is complete, the new primary server starts processing new submitted routing requests. It does not replay any strategy for routing requests submitted before the failover, which means that any routing requests and processing handled by the former primary server and during failover are non-routable. The number of URS non-routable interactions is calculated based on the following formula:

$$\text{URS non-routable interactions} = n \text{ interactions/seconds} * \text{the duration of failover time in seconds}$$

The duration of failover time is the start time Management Layer detected a URS is down to the time the Warm Standby backup server starts routing interactions. The duration of failover time in seconds can range from one second to about fifteen seconds depending on the hardware or software component failure scenario and the ADDP setting.

For information on backup URS in Warm Standby, see “URS Redundancy Using Warm Standby” on [page 80](#).

Formula for Non-Routable Interactions for URS Hot Standby Definition

Using URS Hot Standby with LDS continuing to send routing requests to the paired URSs (primary and Hot Standby) and replaying results in a lower number of URS non-routable interactions than using a URS Warm Standby configuration or load balancing with an LDS configuration alone.

URS non-routable interactions are calculated based on the following formula:

When used with Hot Standby configuration:

$$\text{URS non-routable interactions} \leq n \text{ interactions/seconds} * \text{duration of failover time in seconds}$$

When used with LDS and Hot Standby configuration:

$$\text{URS non-routable interactions} \leq 1/N (n \text{ interactions/seconds}) * \text{duration of failover time in seconds}$$

The duration of failover time is the start time Management Layer detected a URS is down to the time Hot Standby backup server starts routing

interactions. The duration of failover time in seconds can range from one second to about fifteen seconds depending on the hardware or software component failure scenario and the ADDP setting.

For information on URS in Hot Standby, see “URS and Hot Standby” on [page 86](#).

Note: URS non-routable interactions does not mean that interactions are not recoverable. It means that URS cannot take control of these interactions or attempt to route them until after failover is complete.

System Recovery and Statistic Reconciliation

Neither Hot nor Warm Standby backup URSs run a strategy in parallel.

Warm Standby URS

For a Warm Standby URS, statistics are not requested from Stat Server and no statistics are kept in the backup URS memory. After the failover of the standby server is completed, URS then starts requesting statistics from Stat Server for the potential routing of interactions to targets. This is a limitation that only becomes apparent in a scenario in which the wait time set in routing rules or the Select Target object in a strategy is zero and the backup URS has just completed failover and begins routing interactions. In this case, interactions are default routed by the new primary URS. Default routing ceases after URS receives the necessary statistics for all of its potential routing targets. These statistics are immediately available to the new primary URS once failover is complete.

Hot Standby URS

URSs in Hot standby mode synchronize the set of opened statistics between each other. The primary URS sends a notification to the backup URS each time a statistic is opened so the backup URS can open this statistic as well. This enables the backup URS to be synchronized with the primary URS.

Only the Hot Standby URS reconciles statistics collected for a virtual queue. After failover is complete, the Hot Standby backup URS sends an `EventDiverted` message with attribute `Redirected` for all pending interactions at the virtual queue. When Stat Server receives this event, Stat Server recognizes this event as the `ClearCall` action. This statistic is displayed in CCPulse+ and CC Analyzer reports.

URS Options and System Availability

Some URS options that can impact system availability include:

- Failover options (pickup_calls, on_router_activated)
- event_arrive
- Agent Reservation options

Failover Options

Two URS options affect failover, pickup_calls and on_router_activated. URS handles interactions differently depending on how each option is set.

pickup_calls

The URS pickup_calls option is specifically for use in scenarios where a backup server for URS is defined. This option can be set at the routing point or virtual routing point level. When turned on, this option allows URS to use a new attribute in the T-Event set called *Smart Registration*. There are two design purposes for this option:

1. The pickup_calls option is used to minimize the gap in acknowledging pending interactions on routing points when there are time differences between primary and backup URS start up so that primary and backup URS will have a synchronized view on the number of interactions on routing points waiting to be distributed.
2. The pickup_calls option is used in a single primary URS configuration (as a base product offering) so that URS, upon startup, picks up interactions existing at the routing point even before URS registers the routing points and attempts to route interactions.

Since Genesys eServices has safety mechanisms that resubmit all pending transactions to the backup URS after the failover is complete, the synchronizing method of the option pickup_calls for the backup server is not needed.

Note: The pickup_calls option should be used for High Availability Routing for voice media only. That is, pickup_calls should not be set at the virtual routing point level. For media types such as e-mail, there could be thousands of e-mails waiting at the virtual routing point to be processed. If pickup_calls is turned on, eServices could potentially submit thousands of routing requests for these e-mails.

on_router_activated

The option `on_router_activated` has three settings.

1. When set to `route`, the backup URS after failover replays the strategy and attempts to route any pending interactions that it recognizes.
2. When set to `default`, the backup URS after failover attempts to route any pending interactions that it recognizes to the DN defined in the `Default_DN` option.
3. When set to `ignore`, the backup URS after failover ignores the old interactions.

Option Support In Product Offerings

The options `pickup_calls` and `on_router_activated` are supported in Universal Routing as follows:

- Base product:
 - `pickup_calls = false` (this is the default setting during installation)
 - `on_router_activated` is irrelevant since it concerns only a failover to a backup server scenario
- Redundancy:
 - `pickup_calls = false` (this is the default setting during installation)
 - `on_router_activated` setting is ignored by URS backup in Warm Standby only if the setting for this option is `route`
- High Availability routing:
 - `pickup_calls = true/false` (user can turn it on or off according to business needs; for voice media only)
 - `on_router_activated` setting is ignored by backup URS in Hot Standby

URS Behavior, Default Settings For Both Options

Figure 35 illustrates how URS functions when the default setting for both options is used

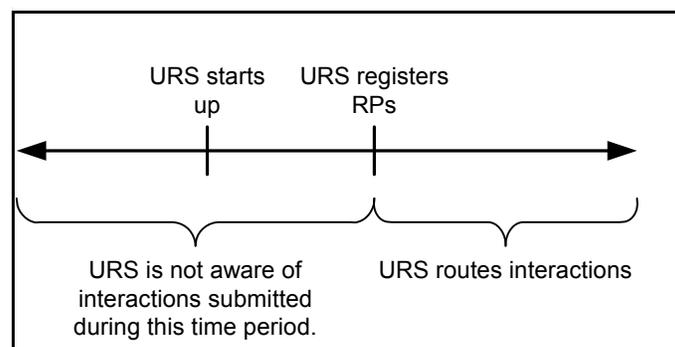


Figure 35: Options with Default Setting

Figure 36 illustrates how the options `pickup_calls` and `on_router_activated` change URS behavior.

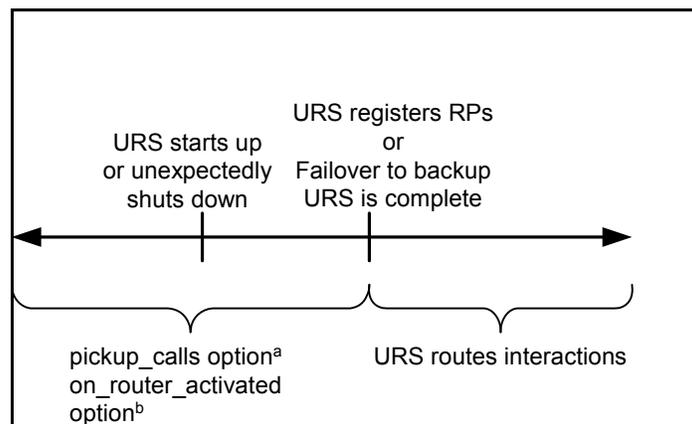


Figure 36: Options change URS behavior

- a. Option `pickup_calls` determines if URS should be aware of interactions submitted before URS startup until URS registers routing points.
- b. Option `on_router_activated` determines how URS treats these pending interactions between the time primary URS unexpectedly terminates and failover to the backup server is complete.

Option `event_arrive`

URS registers all routing points except those having option `event_arrive` configured and set to `none` (this means that if this option is not configured at all, URS registers this routing point) at the routing point or the virtual routing point. If URS is not registered to a routing point, URS does not receive routing requests at all because T-Server will not monitor this routing point. Within a given LDS configuration URS uses this mechanism to subscribe to different routing points.

Note: When LDS is used, the external routing functionality related to Enterprise Routing and Network Routing should not be affected.

Agent Reservation Options

Note: When the agent reservation feature is used in an LDS/multi-URS environment, Genesys recommends that all URSs come from the same family (all 7.5, or all 7.6 for example), and that you configure them all with the same set of options.

Using LDS to achieve redundancy or load distribution always requires two or more URSs to route interactions from multiple T-Servers. In a configuration

with LDS, the probability of having two URSs requesting the same routing target increases. As a result, you should set the `agent_reservation` option to `true` when using LDS.

The following options in the `URS Application` object are related to the agent reservation feature:

- `agent_reservation`

This option requires URS to send a `reserve-agent` request to T-Server and wait for confirmation from T-Server before routing interactions to an agent.

Note: If Router Self-Awareness (see [page 255](#)) is on, then every URS will block an agent for routing as soon it receive notification that this agent was selected by some other URS. The assumption here is that other URS's notification can arrive much sooner than the agent will be reported as busy by Stat Server. This can save URS from the necessity of doing a reserving request that will unconditionally fail.

- `reduced`

This option switches off features that are not used in order to increase the efficiency of URS. A value of `32` disables checking the reservation state of the target.

- `transition_time`

This option determines the time interval during which an agent is reserved. The agent cannot be reserved again until the reservation time has expired. This option is specifically used to address “races” between two or more routing requests for the same target.

Note: This option does not work for multi-direct (for example, direct transfers between agents) or multi-ACD interactions.

- `reservation_pulling_time`

This option temporarily eliminates the regular two-second pause cycle for URS to select each routing target. With this option on, URS sends a reservation request for another ready agent immediately after a negative response to the preceding request. This option intensifies network traffic.

- `treatment_delay_time`

This option delays treatment if `agent_reservation` is used.

For more information on the above options, see the chapter on options in the *Universal Routing 8.0 Reference Manual*.

Agent Reservation Options in T-Server

The following options are related to the agent reservation feature at the T-Server Application level:

- `request-collection-time`
Specifies the interval that agent reservation requests are collected before a reservation is granted. During this interval, agent reservation requests are delayed, in order to balance successful reservations between client applications (for example, Universal Routing Servers).
- `reservation-time`
Specifies the default interval that an Agent DN is reserved to receive a routed call from a remote T-Server. During this interval, the agent cannot be reserved again.
- `reject-subsequent-request`
Specifies whether T-Server rejects subsequent requests, from the same client application, for an agent reservation for the same Agent object that is currently reserved.

For more information on the above three options, see the section on T-Server common configuration options in the deployment guide for your T-Server.

Agent Reservation Types

There are three “flavors” of the agent reservation feature:

1. Standard Agent Reservation

Only URS option `agent_reservation` set to `true` is required. URS issues requests to reserve agents for a T-Server where agents are logged in (URS must be connected to this T-Server).

In some environments, it is possible to have agents who log into multiple T-Servers. This can potentially cause the scenario where two URSs may request different T-Servers for agent reservation. In this case, the second type of agent reservation described below may be the recommended type.

2. Agent reservation through a centralized (dedicated) T-Server.

In this case, you need to select a T-Server that will deal with all possible reservation requests from all URSs. To configure this type of agent reservation (in addition to setting `agent_reservation` to `true` on the URS Application level), configure the selected (dedicated) T-Server Annex tab to have sections corresponding to all URS Application names and with option `agent_reservation` set to `true`.

3. Agent reservation through ISCC. This is an improved version of the standard agent reservation. It combines agent reservation requests with regular URS requests (`RequestGetAccessNumber` or `RequestRouteCall`) in such way that there is one less request during the routing process of each interaction. As a result, overall URS performance is improved.

To switch on this type of agent reservation, set URS option `agent_reservation` to `implicit`.



Chapter

4

eServices Architecture and Flows

Universal Routing 8.0 can work with Genesys eServices (called Multimedia in 8.0.0 and earlier) software components, enabling you to route non-voice interactions based on IRD-designed business processes, as well as voice interactions.

As part of the CIM platform, eServices is a group of Genesys components that enable you to process, manage, and archive customer/agent interactions in the media of your choice. This chapter introduces the architecture and flows behind Multimedia. For more information than presented here, see *eServices (Multimedia) 8.0 Deployment Guide*.

Note: For architecture and flow information on routing voice calls, see Chapter 2 on [page 55](#).

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

- [eServices Components, page 103](#)
- [eServices Functions, page 104](#)
- [Architecture, page 109](#)
- [3rd Party Media, page 112](#)
- [eServices Documentation, page 113](#)

eServices Components

The eServices components are of three general types:

1. **Interaction Management.** This is the basic layer of components, providing core functionality. It requires at least one media option (next item).

2. Media options. Interactions arrive and take place via some medium. In a traditional contact center, the medium is PSTN voice. With Multimedia, you choose Genesys E-mail, Genesys Chat, or SMS.
3. Other option: Genesys Content Analyzer is an enhancement to Genesys E-mail.

Note: Support for media not listed above may be added through the Genesys Developer Program.

eServices Functions

Figure 37 shows the overall functioning of eServices, which applies to any media type. This figure shows functions only, not components.

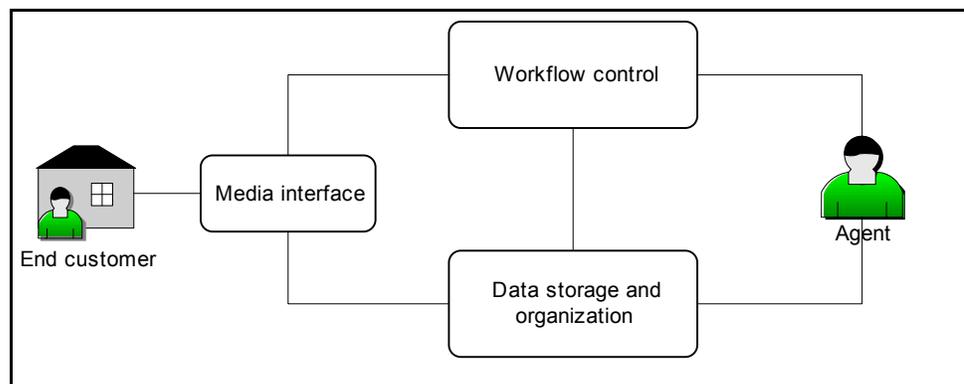


Figure 37: General Functioning of Genesys eServices

The three major functions shown in Figure 37 are:

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

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

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

Media Interfaces

The media interfaces available with eServices 8.0 are E-Mail Server, Chat Server, and SMS Server.

Note: eServices 8.0 also supports the processing of open media interactions in conjunction with the Genesys Platform SDK and Interaction SDK products. See the *Genesys Developer Program* documentation for those products for more information.

- E-Mail Server interfaces with your 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 messages sent from a mobile client. SMS Server uses SMPP v3.4 protocol. If you want to use a different protocol you can use an SMS gateway, such as Kannel. For more details see the “SMS Server” part of the “Administration” section of the “Ongoing Administration and Other Topics” chapter of the *eServices (Multimedia) 8.0 User’s Guide*.

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 workflow control components.

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

Data Storage: Universal Contact Server

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

- 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.

As part of the Universal Contact Server installation package, your eServices installation 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.

IRD provides a Create Interaction object for use in strategies that enables you to create interaction records in the UCS database. For more information on this object, see *Universal Routing 8.0 Reference Manual* or *Universal Routing 8.0 Interaction Routing Designer Help*.

The Identify Contact object locates one or more contact profiles in the UCS Database based on criteria you specify. If no matching contact is found in the database, the Identify Contact object can create a new contact profile.

The Update Contact object is used to update contact profiles in the UCS Database. You can change existing information or add fields to an existing record.

Workflow Control

The components illustrated in [Figure 38](#) handle workflow control.

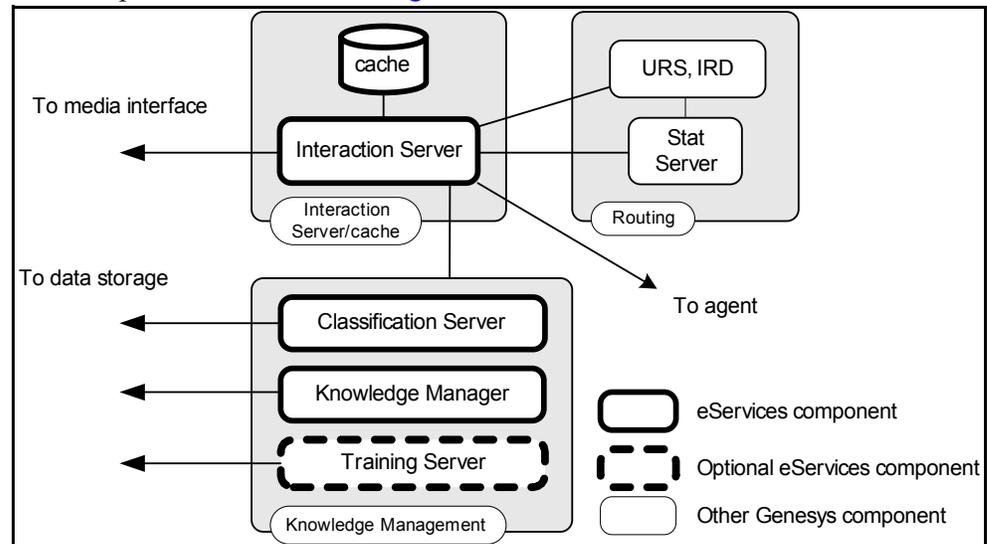


Figure 38: Workflow Control Components

Not shown in [Figure 38](#) 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* 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 Universal Routing components to route interactions according to interaction workflows and routing strategies (see the next section).
- It provides the means for agents to log in and indicate readiness.

URS/IRD

Routing components in Figure 38 on [page 106](#) include the following:

- Interaction Routing Designer (IRD) is where you design *routing strategies*, which trigger functions such as automatic responses and screening; apply logic (segmentation and conditional branching) to the interaction flow; and ultimately deliver the interaction to an agent or other target. Routing strategies are one of the two main types of objects used in interaction workflows (see the previous item). Universal Routing Server (URS) executes routing strategies, which can be contained in business processes (see next item).
- The Interaction Design window (see Figure 12 on [page 32](#)), a part of IRD, creates and displays *business processes*, which plot an overall path for interactions. Groups of business processes connected via queues create interaction workflows, which map a route for the interaction between contact center objects, principally queues and routing strategies (see the next item). Interaction Server executes business processes. For more information, see the *Universal Routing 7.6 (or later) Business Process User's Guide*.
- 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 they are handling, 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 the contents of text-based interactions. The results of these queries can then be used in 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. 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.

- Manage field code Custom Variables that may have been associated with the standard responses in order to personalize them.

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 pre-classified interactions. Setting up and scheduling these training sessions is another function of Knowledge Manager.

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. Among other possibilities, strategies can:
 - Send an Acknowledgement or an automatic reply.
 - Apply a screening rule.
 - Apply content analysis (with Content Analyzer option only).
 - Forward or redirect an e-mail.

Architecture

This section presents the architecture for eServices 8.0. [Figure 39](#) shows its main components.

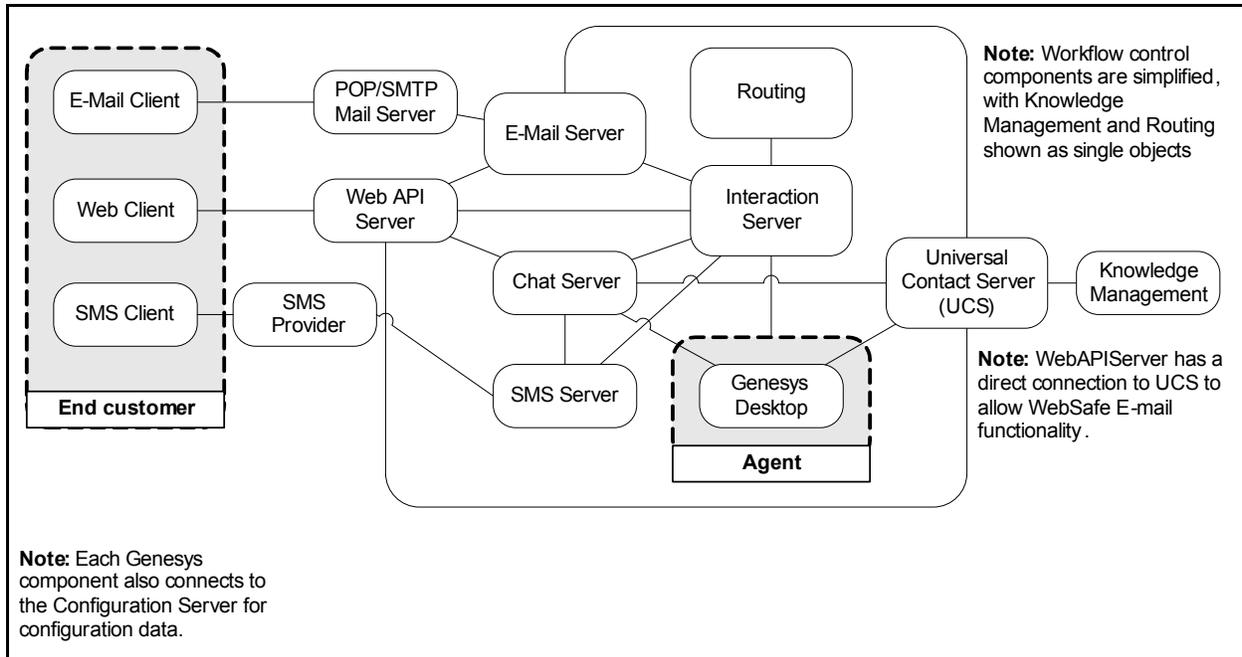


Figure 39: eServices Architecture

[Figure 39](#) simplifies the workflow control components by showing Knowledge Management and Universal Routing as single objects. Interaction processing generally proceeds according to the type of interaction.

Processing E-mail

1. An e-mail arrives. E-mail interactions arrive in one of two ways:
 - If the customer sends ordinary e-mail, the interaction arrives via the enterprise mail server.
 - If an external agent sends a reply to an e-mail previously processed by Genesys, the interaction can also arrive via the enterprise mail server.
 - If the customer sends e-mail from a web site (by filling out a web form), the interaction arrives via the Web API Server.
2. E-mail Server sends operational data on the interaction to Interaction Server and simultaneously stores the body of the interaction in the Universal Contact Server database.

In the case of a reply from an external agent, E-mail Server extracts the reply text from the e-mail, creates an outbound e-mail reply to the customer, and submits the outbound e-mail to Interaction Server using the

queue specified in the Reply From External Resource strategy-building object.

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

A supervisor can intervene at various points using Ad Hoc Management for 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 Universal Routing components.

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

Processing Chat

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 process the chat interaction in a number of ways, 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 accepts the invitation and connects to the chat session.
7. Agent and customer conduct a chat session.
8. The chat session ends.
9. Chat Server writes the content of the chat session to the Universal Contact Server database.
10. Postprocessing occurs; for example, a transcript of the chat session is e-mailed to the customer.

Processing SMS Messages

This section describes how SMS interactions are processed.

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

For **outbound session mode** SMS Server:

- Creates an SMS session on a request from an agent.
- Joins the session to a Chat Server session initiated by the agent.
- Forwards incoming and outgoing messages between the mobile client and the Chat Server session (agent).

3rd Party Media

The 3rd Party Media interaction channel for the CIM platform enables you to apply the sophisticated capabilities of the CIM platform to process interactions and tasks generated by third-party applications and systems. Genesys 3rd Party Media provides an entry point for non-Genesys interactions into the Genesys environment, enabling:

- Custom integration with third-party applications and systems, supporting many types of media, such as FAX, workflow items, eLearning, and so on.
- Customers to submit a custom webform via Genesys Web API Server.
- Utilization of CIM Platform interaction management capabilities, routing, and reporting services across all media channels.
- Consistent management capabilities and interaction history throughout the interaction lifecycle.

The 3rd Party Media capability is implemented through Universal Routing IRD objects used in strategies. Objects that are specifically designed to handle 3rd Party Media, as well as Genesys-based interactions include:

- Acknowledgement
- Attach Categories
- Autoresponse
- Chat Transcript
- Classify
- Create E-mail Out
- Create Interaction
- Create Notification
- Create SMS
- External Service
- Forward E-mail

- Identify Contact
- Multi-Screen
- Queue Interaction
- Redirect E-mail
- Reply From External Resource
- Route Interaction
- Screen
- Send E-mail
- Stop Interaction
- Update Contact
- Web Service
- Workbin

In addition to the above objects that can handle various types of 3rd Party Media interactions, IRD supplies the following objects for proactive routing of interactions, which routes through the outbound preview 3rd Party Media channel:

- Add record
- Do not call
- Processed
- Update record
- Reschedule

For more information on the objects that perform these functions, see the chapter on IRD objects in the *Universal Routing 8.0 Reference Manual*.

eServices Documentation

For a list of the Genesys eServices documentation that is most helpful in understanding eServices routing, see “Related Documentation Resources” on [page 347](#).

For a detailed examination of voice and multimedia interaction flows, see the *Reporting Technical Reference Guide for the 7.2 Release*.



Chapter

5

Deployment Planning

This chapter provides an overview of the planning required before you set up your Universal Routing solution and includes worksheets that help you identify what information you need to collect before you start configuring your Universal Routing Applications.

Before proceeding with the deployment of Universal Routing, be sure that Framework has already been configured and installed. If it has not, refer to the *Framework 8.0 Deployment Guide*. Routing-specific planning is described in this document to prepare for configuring and installing Enterprise Routing or Network Routing.

To facilitate configuration when running with eServices (see Chapter 4 on [page 103](#)), it is important to configure components in the following order:

1. Framework components
2. Universal Routing components
3. eServices components

For information about deploying one of the Genesys Reporting tools with Universal Routing, see the Reporting documentation.

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

- [Setting Up a Solution in a Contact Center, page 116](#)
- [Planning Worksheets, page 119](#)
- [Enterprise Routing Worksheet, page 120](#)
- [Network Routing Worksheet, page 123](#)
- [Worksheet Definitions, page 127](#)
- [Security Banner Message, page 139](#)

Setting Up a Solution in a Contact Center

When setting up a solution in a contact center for routing voice interactions, consider the system architecture that you will use, the distribution of components, and whether to use redundant components.

For an Enterprise Routing solution, you need to consider the routing point of the Private Branch Exchange (PBX). The PBX routes the interaction if the interaction is not routed by the backup URS (if configured) within a specified time.

For a Network Routing solution, you need to consider the settings on the service number of the Service Control Point (SCP). The SCP routes the interaction if the interaction is not routed by the URS within a specified time.

Architecture Planning

When planning a routing solution in a contact center, you need to make decisions about the system architecture, the distribution of components, and the redundancy of components. You also need to make decisions and plan according to the type of system architecture that you need for your solution. Consider the following points:

Application Issues

- Configure a backup URS (if purchased) running in parallel with a primary URS for high availability. (See “Redundancy Level” on [page 79](#).)
- Message Server processing speed can be affected by the number of log files produced from a server supplying log messages or from the level of real-time monitoring of IRD clients.

By default, Message Servers are configured to handle both logging operations for servers and real-time monitoring of IRD clients. If there is a high demand for logging operations at the same time that many IRD clients are conducting real-time monitoring, the performance can degrade for not only Message Server but also URS and IRD. To prevent this degradation in performance, configure Message Servers to handle only one function, logging or IRD monitoring. Dedicate one Message Server to logging. Dedicate several others to IRD monitoring.

The ability to configure multiple Message Servers according to their function enables you to design your contact center architecture to better manage messaging and monitoring conditions.

For information on configuring Message Servers, see Chapter 8 on [page 181](#).

- Windows platforms draw large amounts of data. The current network recommendation is for at least 100 MB Ethernet. Back-end servers that are not on the same machine as T-Server should have a switched Ethernet hub.
- Consider separating Applications by function: telephony with telephony, database with database. For example, install your database and DB Server on one machine. Install T-Server, Framework, and other Genesys servers on another machine. And install client GUIs on a different machine.

General Issues

- Machines with server Applications work more efficiently with two processors; large installations should use four-processor machines.
- Verify network performance and operation before performing installations. Check for proper routing, network lags, DNS, duplicate IP addresses, and other common network issues.
- If disk access is a concern, then a RAID subsystem with controller board and disks provides the best operation.
- If NT Server Genesys install uptime is a prime consideration, then a hardware-redundant NT platform (memory lock-stepped architecture) is required.
- GUI performance depends on the quality of video boards installed.
- Set up at least two machines for server tasks, each running at 166 MHz minimum.
- Back-end Genesys servers collecting data should be located on the same machine or network segment as the database server.
- Main memory data bandwidth becomes a problem when GUI clients are used on the same machine or network segment as the database server.
- Stat Server starts faster when it is on the same machine as T-Server but this could affect T-Server performance depending on available system memory.

For the latest information on system requirements, see the *Genesys Supported Operating Environment Reference Manual*, which is available on the Genesys Technical Support web-site.

Distributed Configuration Environment

In Universal Routing, you can have a solution in a configuration environment with multiple Configuration Servers acting as proxies to one master Configuration Server that provides information to the Configuration Server proxies. Changes to objects in the master Configuration Server are communicated to the Configuration Server proxies immediately. Configuration Server proxies receive information from the master Configuration Server and provide information to Configuration Server clients.

To increase performance capabilities of an Enterprise Routing or Network Routing solution and to reduce the load on Configuration Server, all Applications that have a read-only relationship (for example, URS) to Configuration Server can be connected to Configuration Server proxy. Since the Configuration Server proxy contains information on all configuration objects, interaction processing continues uninterrupted.

Note: If a Configuration Server proxy unexpectedly terminates, the master Configuration Server *does not* act as a backup to Applications connected to the proxy.

Termination of the master Configuration Server does not interrupt Application services such as routing services. However, URS bases subsequent routing decisions according to the last known Configuration Layer objects.

In a LAN environment, proxies can be located remotely from a master Configuration Server. If the LAN connection unexpectedly terminates between the master Configuration Server and Configuration Server proxy, the Applications connected to the remote proxy can start because all Configuration Server information is stored in the Configuration Server proxy. However, when the connection is interrupted, the remote Configuration Server proxy does not receive information about changes to objects in a master Configuration Server until the connection to Configuration Server is re-established.

URS can be connected to proxies but IRD must connect to the master Configuration Server because it has to write to Configuration Server. For purposes of URS operations, Configuration Server proxy is invisible. URS behaves as if it is connected directly to Configuration Server.

For more detailed information, see the following:

- *Framework 8.0 Deployment Guide* for information on using a Configuration Server instance as a proxy, including architectural concepts and use case recommendations.
- *Framework 8.0 Deployment Guide* for information on configuration and installation procedures for a proxy Configuration Server.
- *Framework 8.0 Management Layer's User Guide* for information on how Management Layer supports proxy Configuration Servers.

Private Branch Exchange (PBX) Timeout

In a configured environment, URS normally routes interactions. However, as a backup procedure when URS fails to route the interaction, the backup URS (if you have one) routes the interaction. If the interaction is not routed by the backup URS within a specified time—the Private Branch Exchange (PBX) timeout, a setting on the routing point of a PBX—the PBX routes the interaction.

To avoid having URS and PBX compete to route calls, set the PBX timeout slightly higher than the maximum waiting time of the interactions in the queue. Set the default destination of the PBX timeout to either an Automated Call Distribution (ACD) queue where agents are logged in or to another routing point from which the PBX will eventually deliver interactions to agent logins.

Note: Do not confuse the PBX timeout with the timeout settings for strategies and servers. When the timeout setting of a strategy routing rule expires, strategy processing continues as directed by the objects connected to the red port on the routing rule object. If there is no additional routing rule object or default routing object connected to the red port, URS routes the interaction according to the default option setting in URS.

See also “Using PBX as a Backup Mechanism” on [page 91](#).

Planning Worksheets

Before configuring and installing an Enterprise Routing or Network Routing Solution, collect all necessary information about the contact center. Use the worksheets in this document to collect the information. Definitions and suggested values can be found in the pages that follow the worksheets. Make an extra copy of the worksheets for recording the information.

The worksheets presume Framework is already configured.

Note: If you are running Genesys eServices, add the Interaction Server component (see [page 133](#)) to the Connections list for URS after you have configured Interaction Server, as described in the *eServices (Multimedia) 8.0 Deployment Guide*. The eServices Wizard enables you to do this.

After completing the installation, keep the worksheet as a troubleshooting reference. If you need help from Technical Support, fax the worksheet to your support engineer in order to provide complete information about the system configuration.

Some Framework components, like Stat Server and DB Server, can be added during the configuration of Enterprise Routing and Network Routing.

The Enterprise Routing worksheet begins below. The Network Routing worksheet begins on [page 123](#).

Note: All Application names must be unique. Names can contain alphanumeric characters and underscores.

Enterprise Routing Worksheet

Use the worksheet in [Table 2](#) to plan for Enterprise Routing. See [Table 4](#) on [page 127](#) for definitions of the listed items.

Table 2: Enterprise Routing Worksheet

Enterprise Routing Worksheet	
Logging into Wizard Manager	14. Cost-Based
1. user name:	15. Workforce Management
2. user password:	16. Other:
3. Application:	
4. host:	
5. port:	Stat Server Application
	17. name:
Solution	18. Log configuration:
6. name:	19. T-Server connection(s):
	20. Debug level:
Tenant	21. host:
7. name:	22. port:
Switches	Customer Database
8. names:	23. DAP name:
	24. DBMS name:
Routing Features (optional)	25. DBMS type:
9. Queue-based	26. Database name:
10. Agent-based	27. user name:
11. Skill-based	28. password:
12. Database lookups	29. case conversion:
13. Voice Treatment	30. DB Server name:

Table 2: Enterprise Routing Worksheet (Continued)

Enterprise Routing Worksheet (continued)	
Workforce Database (optional)	Universal Routing Server Application
31. DAP name:	48. name:
32. DBMS name:	49. Log configuration:
33. DBMS type:	50. host:
34. Database name:	51. port:
35. user name:	52. T-Server or LDS connections:
36. password:	53. Stat Server connection(s):
37. case conversion:	54. DAP connection(s):
38. DB Server name:	55. default destination:
	56. agent transition time:
	57. route consult calls:
DB Server Application	58. Message Server connection(s):
39. name:	59. Custom Server connection (optional):
40. Log configuration:	60. Interaction Server connection (if using eServices components)
41. DBMS type:	
42. host:	Backup Universal Routing Server (for Redundancy or High Availability routing levels)
43. port:	61. name:
	62. host:
	63. port:
Message Server Application(s)	64. redundancy:
44. name:	Load Distribution Server
45. Log configuration:	65. name:
46. host:	66. host:
47. port:	67. port:

Table 2: Enterprise Routing Worksheet (Continued)

Enterprise Routing Worksheet (continued)	
68. T-Server connections:	85. type: Customserver
69. Message Server connection (optional):	
	Backup Custom Server (optional)
Interaction Routing Designer (IRD)	86. name:
70. name:	87. host:
71. Message Server(s):	88. port:
72. host:	89. type: Customserver
73. port:	90. redundancy: Not Specified
Table Access	Interactive Voice Response Option
74. name:	91. IVR:
75. description:	
76. Database table:	VoIP Option
77. Database Access point (DAP):	92. Gateway:
78. format:	93. Voice Terminal Endpoints:
Reporting Options	Solution Control Interface
79. real-time:	94. user name:
80. historical:	95. password:
	96. Application:
Custom Server (optional)	97. host:
81. name:	98. port:
82. Log configuration:	
83. host:	Application Server
84. port:	99. type:

Table 2: Enterprise Routing Worksheet (Continued)

Enterprise Routing Worksheet (continued)	
Composer	
100. Port:	

Network Routing Worksheet

Use the worksheet in [Table 3](#) to plan for a Network Routing solution. See [Table 4](#) on [page 127](#) for definitions of the items that are listed in the Network Routing worksheet.

Table 3: Network Routing Worksheet

Network Routing Worksheet	
Logging into Wizard Manager	10. Type
1. user name:	11. Service Numbers
2. user password:	12. Network Destinations
3. Application:	Routing Features (Optional)
4. host:	13. Queue-based:
5. port:	14. Agent-based:
	15. Skill-based:
Solution	16. Database lookups:
6. name:	17. Voice Treatment:
	18. Cost-Based
Tenant	19. Workforce Management
7. name:	20. Other
Network Switching	Network T-Server Application
8. name:	21. name:
	22. Log configuration:
Network Switching Office	23. host:
9. name:	24. port:

Table 3: Network Routing Worksheet (Continued)

Network Routing Worksheet (continued)	
Stat Server Application	45. case conversion:
25. name:	46. DB Server name:
26. Log configuration:	
27. T-Server connection(s):	DB Server Application
28. Debug level:	47. name:
29. host:	48. Log configuration
30. port:	49. DBMS Type
	50. host
Customer Database	51. port
31. DAP name:	
32. DBMS name:	Message Server Application(s)
33. DBMS type:	52. name:
34. Database name:	53. Log configuration:
35. user name:	54. host:
36. password:	55. port:
37. case conversion:	
38. DB Server name:	Universal Routing Server Application
	56. name:
Workforce Database (optional)	57. Log configuration:
39. DAP name:	58. host:
40. DBMS name:	59. port:
41. DBMS type:	60. T-Server or LDS connections:
42. Database name:	61. Stat Server connection(s):
43. user name:	62. DAP connection(s):
44. password:	63. default destination:

Table 3: Network Routing Worksheet (Continued)

Network Routing Worksheet (continued)	
64. agent transition time:	Load Distribution Server
65. route consult calls:	82. name:
66. Message Server connection(s):	83. host:
67. Custom Server connection (optional):	84. port:
68. Interaction Server connection (if running with e-Services)	85. T-Server connections
	86. Message Server connection (optional)
Backup Universal Routing Server (for Redundancy or High Availability routing levels)	
69. name:	IRD Application
70. Log configuration:	87. name:
71. DBMS type:	88. Message Server:
72. host:	89. host:
73. port:	90. port:
Database for Database Lookups	
Premise Switching Office	
74. DBMS name:	91. name:
75. DBMS type:	92. type:
76. Database name:	
77. user name:	Premise T-Server Application
78. password:	93. name:
	94. host:
DB Server for Database Lookups	
79. name:	95. port:
80. host:	Premise Switches
81. port:	96. name:

Table 3: Network Routing Worksheet (Continued)

Network Routing Worksheet (continued)	
Premise Switch DN Types	
97. External Routing Points:	
DN Groups	Backup Custom Server (optional)
98. Network Ports:	112. name:
99. Service Numbers:	113. host:
	114. port:
Table Access	115. type: Customserver
100. name:	116. redundancy: Not Specified
101. description:	
102. Database table:	Interactive Voice Response Option
103. Database Access Point (DAP):	117. IVR:
104. format:	
	VoIP Option
Reporting Options	118. Gateway:
105. real-time:	119. Voice Terminal Endpoints:
106. historical:	
	Solution Control Interface
Custom Server (optional)	120. user name:
107. name:	121. user password:
108. Log configuration:	122. Application:
109. host:	123. host:
110. port:	124. port:
111. type:	

Table 3: Network Routing Worksheet (Continued)

Network Routing Worksheet (continued)	
Table Formats	
See “Table Formats, Fields, and Table Access” on page 138 .	

Worksheet Definitions

[Table 4](#) contains definitions of the items listed in the Enterprise Routing worksheet on [page 120](#) and the Network Routing worksheet on [page 123](#).

In the table below, the numbers in the columns for Enterprise Routing and Network Routing correspond to the numbered items in the Enterprise Routing worksheet and the Network Routing worksheet.

Table 4: Worksheet Definitions

Enterprise Routing	Network Routing	Worksheet Definitions
Logging into Wizard Manager		
1	1	user name: the user name for the Wizard Manager login
2	2	password: the user password for the Wizard Manager login
3	3	Application: the instance of Configuration Manager, as registered in the Configuration Database. Enter the Application name, which is <code>default</code> , if logging in for the first time after the Configuration Database is installed. If this is not your first time logging in, <code>default</code> should already appear in the field.
4	4	host: the host name of the computer running Configuration Server
5	5	port: the IP port number for connecting to Configuration Server
Solution		
6	6	name: the name you assign to this solution
Tenant		
7	7	name: the name of the tenant for which you want to create this solution
Switches		

Table 4: Worksheet Definitions (Continued)

Enterprise Routing	Network Routing	Worksheet Definitions (Continued)
8	See Premise Switches (Item 91 on Page 135)	<p>names: the names of the switches used in routing interactions for this solution</p> <p>Note: Switches are set up during Framework configuration. If you want to set up additional switches, you need the following: information on the numbers and types of all DNs to be monitored and controlled through the CTI link, agent login ID codes to be assigned to agents for run-time associations between DNs and agents, information on how the switch DNs will be arranged into groups, and information on how agents will be arranged into groups.</p>
Network Switch		
Not Applicable	8	name: a logical name for the switch using the Network T-Server
Network Switching Office		
Not Applicable	9	name: the name you will assign to the switching office on the network side
Not Applicable	10	type: the type of switching office in the carrier network. Among the switching office types listed in the Configuration Layer, the following are network switching: WorldCom 800 Gateway, AT_T 800 ICP Gateway, Sprint SiteRP Gateway, Alcatel SCP Gateway, Bell Atlantic ISCP Gateway, Concert 800 Gateway, Alcatel DTAG SCP Gateway, KPN Network Gateway, Alcatel Telecom Italia SCP Gateway, Alcatel BT SCP Gateway, 3511 Protocol Interface, NGSN, GenSpec, Verizon ISCP Gateway.
Network Switch DN Types		
Not Applicable	11	Service Numbers: this is essentially a routing point on the network switch controlled by Universal Routing Server. It usually corresponds to a toll-free number.
Not Applicable	12	Network Destinations: this DN type corresponds to numbers obtained from the network provider that controls the trunks that deliver interactions to DNs on a premise switch
Routing Features (optional)		
9	13	Queue-based: routing to ACD queues
10	14	Agent-based: routing to individual agents
11	15	Skill-based: routing based on agent skills and proficiency

Table 4: Worksheet Definitions (Continued)

Enterprise Routing	Network Routing	Worksheet Definitions (Continued)
12	16	Database lookups: routing based on database lookups for gathering additional information about the interaction or customer
13	17	Voice Treatment: routing based on voice treatment with IVR
14	18	Cost-based: routing based on the least expensive target
15	19	Workforce Management: routing based on agent schedules and activities
16	20	Other, such as Multi-Site, Service Level, Statistical (see “Universal Routing Capabilities” on page 33)
Network T-Server Application		
Not Applicable	21	name: the name you will assign to the T-Server Application on the network side Note: Do not include a dash or a space in the Application name. Use an underscore to replace these characters if necessary.
Not Applicable	22	Log configuration: the configuration type and level Note: For type, specify network, console, or file. For level, specify All, Trace, Standard, or None.
Not Applicable	23	host: the host name of the machine where T-Server will run
Not Applicable	24	port: the TCP/IP port number through which T-Server will communicate
Stat Server Application		
17	25	Collect this information if you want to provide statistics for this solution. name: the name you will assign to the Stat Server Application
18	26	Log configuration: the configuration type and level Note: For type, specify network, console, or file. For level, specify All, Trace, Standard, or None.
19	27	T-Server connection(s): the T-Server(s) that Stat Server connects to
20	28	Debug level: the level of log events specified in the log file Note: You have five options: write messages during initialization process, write content of all messages sent to clients, write content of all messages received from other servers, write action list for objects, and write status for objects.

Table 4: Worksheet Definitions (Continued)

Enterprise Routing	Network Routing	Worksheet Definitions (Continued)
21	29	host: the host name of the machine where Stat Server will run
22	30	port: the TCP/IP port number through which Stat Server will communicate
Customer Database		
23	31	<p>This database information is used by the DB Server setup in items 36 through 40 for Enterprise Routing and items 44 through 48 for Network Routing.</p> <p>Warning! The Configuration Database must be installed before proceeding with the configuration and installation of Enterprise Routing and Network Routing. See the “Configuration and Installation” chapter in the <i>Framework 8.0 Deployment Guide</i> for more information.</p> <p>DAP name: the name you will assign to the Database Access Point (DAP) for the customer database</p> <p>Note: If you intend to use the Workforce Management (WFM) Database, see the WFM Database section.</p>
24	32	DBMS name: the name or alias identifying the Database Management System that handles the database
25	33	DBMS type: the type of Database Management System (Oracle, Microsoft SQL Server, Sybase, Informix, DB2)
26	34	Database name: the name of the Customer Database
27	35	user name: the user name for SQL server login
28	36	password: the user password for SQL server login
29	37	<p>case conversion: the conversion-case method that corresponds to the DBMS settings</p> <p>Note: You have three settings: any for no conversion, upper for converting key fields into uppercase, and lower for converting key fields into lowercase.</p>
30	38	DB Server name: the name that you will assign to the DB Server

Table 4: Worksheet Definitions (Continued)

Enterprise Routing	Network Routing	Worksheet Definitions (Continued)
Workforce Management (WFM) Database (Optional)		
31	39	<p>This database information is used by the DB Server setup in items 36 through 39 for Enterprise Routing and items 43 through 46 for Network Routing. Collect this information only if using Workforce Management software and a Workforce object in a strategy.</p> <p>Warning! The Configuration Database must be installed before proceeding with the configuration and installation of Universal Routing. See the “Configuration and Installation” chapter in the <i>Framework 8.0 Deployment Guide</i> for more information.</p> <p>DAP name: the name you will assign to the Database Access Point (DAP) for the WFM database. Note: This type of configuration is required only if 6.x Workforce Management is used and if the Workforce object is going to be used in a strategy.</p>
32	40	DBMS name: the name or alias identifying the Database Management System that handles the database
33	41	DBMS type: the type of Database Management System (Oracle, Microsoft SQL Server, Sybase, Informix, DB2)
34	42	Database name: the name of the WFM Database
35	43	user name: the user name for SQL server login
36	44	password: the user password for SQL server login
37	45	<p>case conversion: the conversion case method that corresponds to the DBMS settings</p> <p>Note: You have three settings: any for no conversion, upper for converting key fields into uppercase, and lower for converting key fields into lowercase.</p>
38	46	DB Server name: the name you will assign to the DB Server
DB Server Application		
39	47	name: the name you will assign to the DB Server Application
40	48	<p>Log configuration: the configuration type and level</p> <p>Note: For type, specify network, console, or file. For level, specify All, Trace, Standard, or None.</p>
41	49	DBMS type: the type of Database Management System (Oracle, Microsoft SQL Server, Sybase, Informix, DB2)

Table 4: Worksheet Definitions (Continued)

Enterprise Routing	Network Routing	Worksheet Definitions (Continued)
42	50	host: the host name of the machine where DB Server will run
43	51	port: the TCP/IP port number through which DB Server will communicate
Message Server Application(s)		
44	52	<p>Message Server is needed in order for IRD to monitor the flow of interactions through URS and to send messages from URS to IRD. Message Server can also be used for IVR Server load balancing (see page 264) as well as the Router Self-Awareness feature described on page 255.</p> <p>Note: Configure multiple Message Servers for monitoring IRD and a separate one for logging if desired.</p> <p>For each Message Server, collect the following information:</p> <p>name: the name you will assign to the Message Server Application</p>
45	53	<p>Log configuration: the configuration type and level. You can configure one Message Server for logging and others for monitoring.</p> <p>Note: For type, specify network, console, or file. For level, specify All, Trace, Standard, or None.</p>
46	54	host: the host name of the machine where Message Server will run
47	55	port: the TCP/IP port number through which Message Server will communicate
Universal Routing Server Application (URS)		
48	56	<p>Note: While creating the Universal Routing Server (URS) Application using the Wizard, you can click the Advanced button to change the URS options.</p> <p>name: the name you will assign to the URS Application.</p>
49	57	<p>Log configuration: the configuration type and level</p> <p>Note: For type, specify network, console, or file. For level, specify All, Trace, Standard, or None.</p>
50	58	host: the host name of the machine where URS will run
51	59	port: the TCP/IP port number through which URS will connect
52	60	T-Server or LDS connection(s): the T-Server(s) or LDS(s) that URS connects to
53	61	Stat Server connection(s): the Stat Server(s) specified in item 21 for Enterprise Routing and in item 29 for Network Routing

Table 4: Worksheet Definitions (Continued)

Enterprise Routing	Network Routing	Worksheet Definitions (Continued)
54	62	DAP connection(s): the database access point(s) name specified for the Customer Database (item 20 for Enterprise Routing, item 28 for Network Routing) and the WFM Database (item 28 for Enterprise Routing, item 36 for Network Routing). A DAP for the WFM Database is only required if the Workforce object will be used in strategies.
55	63	default destination: the system-wide default destination in case of a routing error*
56	64	agent transition time: the interval, in seconds, between an agent's ready state and the agent's unavailable state*
57	65	route consult calls: the option that determines whether routing will occur as soon as the consult interaction reaches a routing point or after the transfer is complete. Selecting this option routes the interaction as soon as it reaches a routing point. <i>See the Universal Routing 8.0 Reference Manual for more information.</i>
58	66	Message Server connection(s): the Message Server(s) that URS connects to
59	67	Custom Server connection (optional): the Custom Server that URS connects to
60	68	Interaction Server connection: the server that executes eServices business processes and, in doing so, communicates with URS. <i>See the section on Interaction Server Options in the Universal Routing 8.0 Reference Manual.</i>
Backup Universal Routing Server (for Redundancy or High availability Routing levels)		
Note: Backup URS should have the same component connections the primary URS.		
61	69	name: the name you will assign to the URS Application
	70	Log configuration: the configuration type and level
	71	DBMS type: the type of database management system.
62	72	host: the host name of the machine where the Backup URS will run
63	73	port: the TCP/IP port number through which the Backup URS will connect
64		redundancy: warm standby setting for the backup URS for redundancy level, hot standby setting for the backup URS for the High Availability option. Note: URS does not support Not Specified standby mode.

Table 4: Worksheet Definitions (Continued)

Enterprise Routing	Network Routing	Worksheet Definitions (Continued)
Database for Lookups		
Not Applicable	74	Warning! The Configuration Database must be installed before proceeding with the configuration and installation of Network Routing. See the “Configuration and Installation” chapter in the <i>Framework 8.0 Deployment Guide</i> . DBMS name: the name or alias identifying the Database Management System that handles the database or more information.
Not Applicable	75	DBMS type: the type of Database Management System (Oracle, Microsoft SQL Server, Sybase, Informix, DB2)
Not Applicable	76	Database name: the name of the database
Not Applicable	77	user name: the user name for SQL server login
Not Applicable	78	password: The user password for SQL server login
DB Server for Database Lookups		
Not Applicable	79	name: the name you will assign to the DB Server Application
Not Applicable	80	host: the name of the machine where DB Server will run
Not Applicable	81	port: the TCP/IP port through which DB Server will listen for clients attempting to establish communications
Load Distribution Server (LDS) (for Redundancy or High Availability Routing levels)		
65	82	name: the name you will assign to the LDS Application
66	83	host: the host name of the machine where the LDS will run
67	84	port: the TCP/IP port number through which LDS will connect
68	85	T-Server connections: the T-Server(s) that LDS connects to
69	86	Message Server connection(s): the Message Server(s) that LDS connects to
Interaction Routing Designer Application (IRD)		
70	87	name: the name you will assign to your IRD Application

Table 4: Worksheet Definitions (Continued)

Enterprise Routing	Network Routing	Worksheet Definitions (Continued)
71	88	Message Server(s): the Message Server(s) used to monitor URS using IRD
72	89	host: the host name of the machine
73	90	port: the TCP/IP port number
Premise Switching Office		
Not Applicable	91	name: the name you will assign to the switching office for the premise side
Not Applicable	92	type: the type of switching office for the premise side
Premise T-Server Application		
Not Applicable	93	Note: Do not include a dash or a space in your Application name. Use an underscore to replace these characters if necessary. name: the name you will assign to your T-Server Application for the premise side
Not Applicable	94	host: the host name of the machine where T-Server will run
Not Applicable	95	port: the TCP/IP port number through which T-Server will communicate
Premise Switches		
Not Applicable	96	names: logical names for the switches controlled by the premise T-Server
Premise Switch DN Types		
Not Applicable	97	External Routing Points: Dedicated DNs on the premise switch that receive interactions routed from another switch. Physically, they are the same as routing points, but external routing points are not controlled by Universal Routing Server. They are controlled by T-Server, which redirects the interactions arriving to an external routing point according to target information submitted during the external routing point exchange.

Table 4: Worksheet Definitions (Continued)

Enterprise Routing	Network Routing	Worksheet Definitions (Continued)
DN Groups		
Not Applicable	98	Network Ports: the names of DN Groups to which you will assign Network Destination DNs. Each group includes one or more Network Destination DNs and each group is associated with a single premise switch DN.
Not Applicable	99	Service Numbers: this is essentially a routing point on the Network Switch controlled by Universal Routing Server. It usually corresponds to a toll-free number.
Table Access		
74	100	name: the name you will assign to the table access. This links the Table Format, Database Access Point, and Database Table.
75	101	description: a description for the table access (optional)
76	102	Database table: the name of the table in your database
77	103	Database Access Point (DAP): the DAP name specified in item 20 for Enterprise Routing and item 28 for Network Routing for the Customer Database
78	104	format: the table format. Use one of the formats you defined.
Reporting Options		
79	105	real-time: the option to use CCPulse+ for real-time reporting
80	106	historical: the option to use CC Analyzer or CCPulse+ for historical reporting Note: Before selecting these options, the installation package for reporting options and wizards must be installed.
Custom Server (optional)		
81	107	For systems that do not use standard SQL databases, Custom Server allows you to make database queries. Custom Server translates the request for non-SQL databases and values returned by the database into a format that Universal Routing Server can read. name: the name you will assign to the Custom Server
82	108	Log configuration: the configuration type and level Note: For type, specify network, console, or file. For level, specify All, Trace, Standard, or None.

Table 4: Worksheet Definitions (Continued)

Enterprise Routing	Network Routing	Worksheet Definitions (Continued)
83	109	host: the host name of the machine where Custom Server will run
84	110	port: the TCP/IP port number through which Custom Server will run
85	111	type: the type of server (Customserver only)
Backup Custom Server (optional)		
86	112	name: the name you will assign to the backup Custom Server
87	113	host: the host name of the machine where the backup Custom Server will run
88	114	port: the TCP/IP port number through which the backup Custom Server will run
89	115	type: the type of server (Customserver only)
90	116	redundancy: the standby mode for the backup Custom Server. Choose Not specified only. Because Custom Servers do not communicate with each other, the only backup mode available for Custom Servers is not specified. Warm and hot standby are not available for the backup Custom Server.
Interactive Voice Response Option		
91	117	IVR: the option to configure an IVR
VoIP Option		
92	118	VoIP Gateway: the type of gateway and the type of interface
93	19	Voice Terminal Endpoints: the type of voice terminal
Solution Control Interface		
94	120	user name: the user name for the Solution Control Interface login
95	121	user password: the user password for the Solution Control Interface login
96	122	Application: the name of the Application
97	123	host: the host name of the computer running Solution Control Interface
98	124	port: the TCP/IP port through which Solution Control Interface will communicate
99		type

Table Formats, Fields, and Table Access

To access a table in your database, you need to set up `Table Formats`, `Fields`, and `Table Access` to that database through Configuration Manager.

`Table Formats` define what fields and database properties are accessed from the database. You can define as many formats as you need. Each format consists of the database table and the fields from that table that you want to access. You do not need to use all table fields.

`Table Access` defines how the database information is accessed and formatted. Like `Table Formats`, you define as many `Table Accesses` as you need. Each `Table Access` consists of the name, database table, database access point, and format. You do not need to use all tables in your database.

First, define the format. Then, define each table access, which includes the selection of one of your previously defined formats.

Warning! Formats cannot be associated with any `Table Access` if you want to edit them. To edit a format already associated with a `Table Access`, you must first delete the associated `Table Access(es)`.

Table Formats Worksheet

Print and complete this worksheet ([Table 5](#) below and [Table 6](#) on [page 139](#)) for each format you want to define.

Field information must match database information. Field name, Database Type, Length, and the Primary, Unique, and Nullable settings must be specified *exactly* as they are in the database.

Important!

Field information must match database information. Field name, Database Type, Length, and the Primary, Unique, and Nullable settings must be specified *exactly* as they are in the database.

Table 5: Table Information

Format	Name	
	description:	
	field:	

Table 6: Field Information

Field Names	Description	Attach Field Data (check box)	Database Properties	
			data type: length:	default value: constraint:
			data type: length:	default value: constraint:
			data type: length:	default value: constraint:
			data type: length:	default value: constraint:
			data type: length:	default value: constraint:
			data type: length:	default value: constraint:
			data type: length:	default value: constraint:
			data type: length:	default value: constraint:

Security Banner Message

Genesys Windows-based GUI Applications, such as IRD, use the `uni login.dll` to provide common GUI functionality. Starting with 7.6, IRD installation gives the option of configuring a security banner message that displays when users log into IRD. You specify the message content by specifying a URL pointing to a document. You may specify multiple URLs to achieve redundancy in case a URL cannot be retrieved. For information on configuring a security banner message, see the *Genesys 8.0 Security Deployment Guide*.



Part

2

Deployment

Part Two of the *Universal Routing 8.0 Deployment Guide* describes how to configure, install, start, and stop Enterprise Routing and Network Routing (which together comprise Universal Routing). The information in Part Two is divided into the following chapters:

- Chapter 6, “Deployment Process,” on [page 143](#)
- Chapter 7, “Configuring with Wizards,” on [page 151](#)
- Chapter 8, “Manually Configuring Routing,” on [page 181](#)
- Chapter 9, “Load Balancing,” on [page 235](#)
- Chapter 10, “Orchestration Support,” on [page 265](#)
- Chapter 11, “Samples,” on [page 267](#)
- Chapter 12, “Configuring Custom Server,” on [page 289](#)
- Chapter 13, “Installing a Routing Solution,” on [page 311](#)
- Chapter 14, “Starting and Stopping Procedures,” on [page 333](#)
- Chapter 15, “Uninstalling a Routing Solution,” on [page 343](#)



Chapter

6

Deployment Process

The deployment process involves the configuration and installation of the applications needed for a functional setup of Universal Routing, which can include Enterprise Routing, Network Routing, or both.

Note: Universal Routing 8.0 works with Genesys eServices and enables you to route non-voice interactions based on business processes created in IRD. Universal Routing 8.0 also works without eServices in pure voice environments.

This chapter includes the following topics:

- [Before You Begin, page 143](#)
- [How to Configure, page 144](#)
- [eServices Installation Considerations, page 144](#)
- [CD and Solution Packaging, page 145](#)
- [Enterprise Routing Component Installation Order, page 148](#)
- [Network Routing Component Installation Order, page 149](#)
- [Custom Server, page 150](#)

Before You Begin

Before deploying Universal Routing, investigate the sizing, security, availability, and performance required for the specific environment of your contact center.

Use the worksheets in “Planning Worksheets” on [page 119](#) to:

- Specify the objects to be configured within Enterprise Routing or Network Routing
- Assign specific machines and directories on the hard drives to the components

- Identify the number of applications of the same type to provide redundancy
- Ensure that applications that require licenses be licensed properly (see the *Genesys Licensing Guide*). The licensing requirements section of this document is summarized below for your convenience.

If running with eServices, use the worksheet in the *eServices (Multimedia) 8.0 Deployment Guide* to specify information about eServices objects, including Interaction Server and E-mail Server.

How to Configure

This document describes how to manually install Universal Routing.

This involves:

- Creating objects in Configuration Manager, one for each Universal Routing software component.

Note: Objects can also be created and configured in Genesys Administrator. Refer to the *Framework 8.0 Genesys Administrator Help* for information.

- Giving the objects the proper settings for options and other attributes. If you install manually, you will need the option configuration information in the *Universal Routing 8.0 Reference Manual*. See Chapter 8 on [page 181](#).

eServices Installation Considerations

If running Universal Routing with Genesys eServices (called Multimedia in 8.0.0 and earlier), you must first install Genesys Framework and Universal Routing before installing eServices as described in the *eServices (Multimedia) 8.0 Deployment Guide*. Then, during the eServices install:

- When the eServices Configuration Wizard prompts for an existing routing solution, you can select one. You also have the option of additional URS configuration at this time, such as adding Interaction Server to the URS Connections list.
- When the eServices Configuration Wizard prompts for an existing IRD Application object for creating business processes, you can select one.
- When you install the eServices Samples as described in the *eServices (Multimedia) 8.0 Deployment Guide*, you can select a destination directory that is near the directory used for routing strategies (or you can select the StrategyFiles directory created during installation of the Samples).

Also see “Enterprise Routing Component Installation Order” on [page 148](#).

- When IRD starts, it checks for eServices solutions installed by the eServices Configuration Wizard. If not found, the IRD main window or the window shown in Figure 11 on [page 31](#) does not display an Interaction Design shortcut bar and you cannot open the Interaction Design window (shown in Figure 12 on [page 32](#)). To change the default, use the Views tab in Routing Design Options, which opens from the Tools menu.

If you manually configured the eServices components in the Configuration Database, use the workaround provided in the Universal Routing 8.0.x Release Advisory in order to display the Interaction Design shortcut bar.

CD and Solution Packaging

Enterprise Routing is packaged on the Universal Routing CD. You will need the Framework CD, the Universal Routing CD, and the Real-Time Metrics Engine CD.

Universal Routing CD

The Universal Routing CD contains the following components:

- Universal Routing Server
- Interaction Routing Designer
- Custom Server
- Application Templates
- Configuration Database update scripts
- Network Routing components listed on [page 147](#)

The Universal Routing CD contains the following solution templates:

- Enterprise Routing Solution
- Network Routing Solution

Management Framework CD

Before configuring and installing Enterprise Routing or Network Routing, you must install the Framework components. The Framework components include such applications as Management Framework Configuration Wizard, DB Server, Configuration Server, Configuration Manager, License Manager, Management Layer (Local Control Agent, Solution Control Interface, Message Server, Solution Control Server), Stat Server, and T-Server. Enterprise Routing and Network Routing rely on Framework components to function as a solution. Media Configuration Wizard for T-Servers is located on the Media CD.

Note: Network T-Server is considered a Network media interface, not a Framework component and is located in a different Network Media Interface CD (see “Network Routing CD” on [page 147](#)).

For information on installation and configuration of Framework components, see the *Framework 8.0 Deployment Guide* and the T-Server document specific to your T-Server.

Chapter 2, “Voice Routing Architecture and Call Flows,” on [page 55](#) lists the Framework components, Enterprise Routing components, and Network Routing components.

eServices CDs

The Genesys E-Mail CD contains Genesys E-Mail 8.0, a separately packaged media channel for the CIM Platform. Genesys E-Mail is a highly flexible and unified e-mail management solution with extensive real-time and historical reporting capability.

The Interaction Management CD contains the components for Genesys Interaction Management, the core functionality of eServices on the CIM Platform

The Knowledge Management CD contains Genesys Knowledge Management. Knowledge Manager, the user interface component of Genesys Knowledge Management, is used to administer content across the CIM Platform for routing decisions and by the CIM Platform knowledge base for self and assisted services.

The Chat CD contains Genesys Chat Server 8.0.

The Genesys Web Collaboration CD contains Genesys Web Collaboration 8.0, a separately-packaged option for the CIM Platform. Web Collaboration enables agents and customers to view and navigate web pages together. This enables agents to provide superior customer service by assisting customers in using capabilities of web sites such as making purchases, completing forms, finding information, and so on. Genesys Web Collaboration must be deployed in conjunction with the Genesys Interaction Management Platform, which delivers the core capabilities of eServices.

The SMS CD contains Genesys SMS Server 8.0, a separately-packaged media channel for the CIM Platform. Genesys SMS Server is a highly flexible and unified SMS management solution. As a part of the CIM Platform, Genesys SMS Server integrates seamlessly with related products such as Universal Routing, Genesys Supervisor Desktop, Genesys Agent Desktop, and CIM Platform Knowledge Management. Genesys SMS Server must be deployed in conjunction with Genesys Interaction Management, which delivers the core capabilities of eServices.

Real-Time Metrics Engine CD

Stat Server has its own installation CD (Real-Time Metrics Engine), which contains both the Stat Server Wizard and the Resource Capacity Wizard used for configuring agent capacity rules.

Reporting Templates CD

The Genesys Reporting Templates CD contains ERS real-time and historical reporting templates (.xml files + storage) as well as CC Analyzer ERS historical reporting templates (.bqy files). For configuration and installation of Reporting components, (which are contained on the Genesys Reporting CD), refer to the Genesys Reporting documentation.

Notes: Universal Routing 8.0 is compatible with 7.2 (or later) reporting templates, which can be found on the Reporting Templates CD for the relevant version, in the ers directory.

Genesys recommends setting up separate reports for voice and non-voice interactions. Do not mix voice and non-voice interactions on the same report.

Genesys Info Mart CD

Genesys Info Mart produces a data mart containing several star schemas you can use for contact center historical reporting. This includes detailed reporting on Genesys e-mail, chat, and virtual queue interactions, as well as support for reporting on interactions involved in basic Network Routing call flows.

Network Routing CD

Network Routing is packaged on two CDs, the Universal Routing CD (see [page 145](#)), and the Network Media Interfaces CD. Both the Universal Routing and the Network Media Interfaces CD contains the following Network Routing components:

- Network T-Servers
- Network T-Server templates
- Network Routing Solution template

Enterprise Routing Component Installation Order

To facilitate configuration when running with eServices, it is important to configure component groups in the following order:

1. Framework components (including Stat Server Wizard and Resource Capacity Wizard)
2. Enterprise Routing components (see below)
3. eServices components

Note: The eServices Configuration Wizard assumes that Universal Routing configuration has been completed.

Order for Individual Components

The individual components for deploying Enterprise Routing must be installed in the following order:

1. DB Server (if not already installed with other Framework components)
2. Configuration Server (if not already installed with other Framework components)
3. Configuration Manager (if not already installed with other Framework components)
4. License Manager
5. T-Server
6. Stat Server (if not already installed with other Framework components)

Note: Stat Server has its own Wizard, which is not part of the Common Wizard Set used by Genesys Wizard Manager. If planning to route based on the agent capacity rules, you must also install the Resource Capacity Wizard component. Both the Stat Server Wizard and the Resource Capacity Wizard are located on the Real-Time Metrics Engine CD.

7. Message Server(s) (if not already installed with other Management Layer components)
8. Universal Routing Server
9. Interaction Routing Designer
10. Custom Server

Note: At a minimum, to install and launch Interaction Routing Designer and Universal Routing Server, you must have already installed DB Server, Configuration Server, and Configuration Manager.

11. After installation of the above components, install eServices as described in the *eServices (Multimedia) 8.0 Deployment Guide*.
12. The eServices Configuration Wizard gives the option of using the Resource Capacity Wizard for setting up agent-capacity rules. For more information on this wizard, see the *Genesys 8.0 Resource Capacity Planning Guide*. For summary information on agent-capacity rules and how they can affect routing, see “Setting the use_agent_capacity Option” on [page 201](#).

What Each Component Does

Below is a summary of what each component does.

- T-Server: Generates events and receives requests.
- URS: Executes routing strategies.
- Stat Server: Indicates agent availability.
- Configuration Server: Provides contact center objects.
- Message Server: Parses messages for strategy-monitoring.
- DB Server: Enables access to the Genesys database.
- Custom Server: Enables access to customer database's if using a DBMS not supported by Genesys.
- Interaction Routing Designer: Enables users to create strategies.
- PBX: Routes a call.
- eServices components: See the *eServices (Multimedia) 8.0 Deployment Guide*.

Network Routing Component Installation Order

Network Routing includes both the network side and the enterprise side of a solution. The necessary components for deploying a Network Routing solution must be installed in the following order:

1. DB Server (if not already installed with other Framework components)
2. Configuration Server (if not already installed with other Framework components)

3. Configuration Manager (if not already installed with other Framework components)
4. License Manager
5. Network T-Server
6. T-Server
7. Stat Server (if not already installed with other Framework components)
8. Message Server(s) (if not already installed with other Management Layer components)
9. Universal Routing Server
10. Interaction Routing Designer
11. Custom Server

The one component necessary for the enterprise side is a premise T-Server connected to each physical switch. For installation and configuration information, see the *Framework 8.0 Deployment Guide* and the T-Server documentation for your particular T-Server.

Custom Server

For information on manually configuring and installing the server used for the customer's database, Custom Server, see Chapter 12 on [page 289](#).



Chapter

7

Configuring with Wizards

This chapter provides instructions for configuring and installing Enterprise Routing and Network Routing using the Universal Routing Configuration Wizard. This is the preferred method over manual configuration since it automates configuration and installation.

Note: For information on the tasks that the Universal Routing Configuration Wizard automates, see Chapter 8, “Manually Configuring Routing,” on [page 181](#).

This chapter includes the following topics:

- [Important Information, page 151](#)
- [Using the Wizard to Configure Enterprise Routing, page 152](#)
- [Network Routing Wizard, page 175](#)
- [Routing Solutions and Message Server, page 180](#)

Important Information

- To facilitate configuration when running with Genesys eServices, it is important to configure components in the following order:
 - Framework components
 - Stat Server (if planning to route based on agent capacity rules, you will also need to install the Resource Capacity Wizard)
 - Universal Routing components
 - eServices components

Note: The eServices Configuration Wizard assumes that Universal Routing configuration has been completed. To simplify the entire installation process, Genesys recommends configuring components in the order listed above. See “eServices Installation Considerations” on [page 144](#) for more information.

- Configure Enterprise or Network Routing using the Universal Routing Wizard as described in this chapter. At the end of the configuration process, the Wizard creates a configuration installation package. You then install the configuration installation package as described in Chapter 13 on [page 311](#).
- After configuration and installation, any changes made to the properties of an application or a configuration object may delay URS processes for a few seconds.
- If you are using a non-SQL database and plan to include database lookups in your routing strategy, you can also configure and install Custom Server using the Custom Server Wizard invoked through the Universal Routing Configuration Wizard. You will need to specify the name of the Custom Server, the tenant Custom Server belongs to, and the host, port, and options including standard, log, and custom options.
- You can configure multiple Enterprise Routing or Network Routing solutions using the Universal Routing Wizard. When you select Enterprise Routing or Network Routing in the first configuration screen of Wizard Manager, the name of every solution of that type previously configured appears on the first screen of the Wizard.
- If you wish to create capacity rules while creating a routing solution, first install the Resource Capacity Wizard, which is available on the Real-Time Metrics Engine CD. You can create and assign capacity rules to a newly created Tenant, Place, or Person.

Using the Wizard to Configure Enterprise Routing

Before you configure Enterprise Routing, complete the Framework configuration and installation. To install Management Framework Configuration Wizard, run `setup.exe` from the root directory of the Management Framework 8.0 CD. To install the Media Configuration Wizards for T-Server applications, run `setup.exe` from the root directory of the Media 8.0 CD. To install Stat Server Wizard, run `setup.exe` from the root directory of the Real-Time Metrics Engine CD. When through with these activities, collect the solution information described in Deployment Planning in [page 115](#).

You start the Enterprise Routing and Network Routing wizards through the Wizard Manager. You are already familiar with Wizard Manager since you used it to install and configure Framework.

Task Summary: Configuring Enterprise Routing with the Wizard

Objective	Related Procedures and Actions
Start Wizard Manager	Procedure: Starting Wizard Manager, on page 153
Create an Enterprise Routing solution with the Wizard	<ol style="list-style-type: none"> 1. Procedure: Creating an Enterprise Routing solution, on page 156 2. Create new or add existing applications to the solution. See “Special Note on Adding/Creating Applications” on page 162 3. Procedure: Creating a URS Application object, on page 163 4. Optional: Procedure: Configuring Reporting, IVR Interface, and Voice Treatment Options, on page 172 5. Procedure: Copying the installation package to a directory, on page 173

Procedure: Starting Wizard Manager

Purpose: To open the Wizard used for Enterprise Routing.

Start of procedure

1. Insert the Universal Routing CD.
2. Double-click `setup.exe` in the root directory to install the Wizards that are specific to Universal Routing. The Wizard Manager doesn't know how to configure Enterprise Routing or Network Routing until you run this file.

Note: The steps below assume you have already installed Framework Configuration Wizard, Media Configuration Wizard for T-Servers, and the Stat Server Wizard.

3. After installing the Wizards, open Genesys Wizard Manager. From the Start menu, the default is:
Programs > Genesys Solutions > Routing > Universal Routing Configuration Wizard > Start Wizard Manager

One of two windows opens:

- If you did not select the option below, the Welcome to Wizard Manager window opens (see [Figure 40](#)).
- The login window shown in [Figure 41](#) opens if you previously selected: Next time I start Wizard Manager, display the login window immediately. Continue with step 5.



Figure 40: Welcome to Wizard Manager

4. Click the log into the Configuration Layer link. The Wizard Manager login dialog box opens. [Figure 41](#) shows the dialog box after expanding it by clicking the Details button.



Figure 41: Configuration Layer Login

5. Enter the login information for Configuration Manager (see [page 183](#)) and click OK. The Framework screen opens (see [Figure 42](#)).

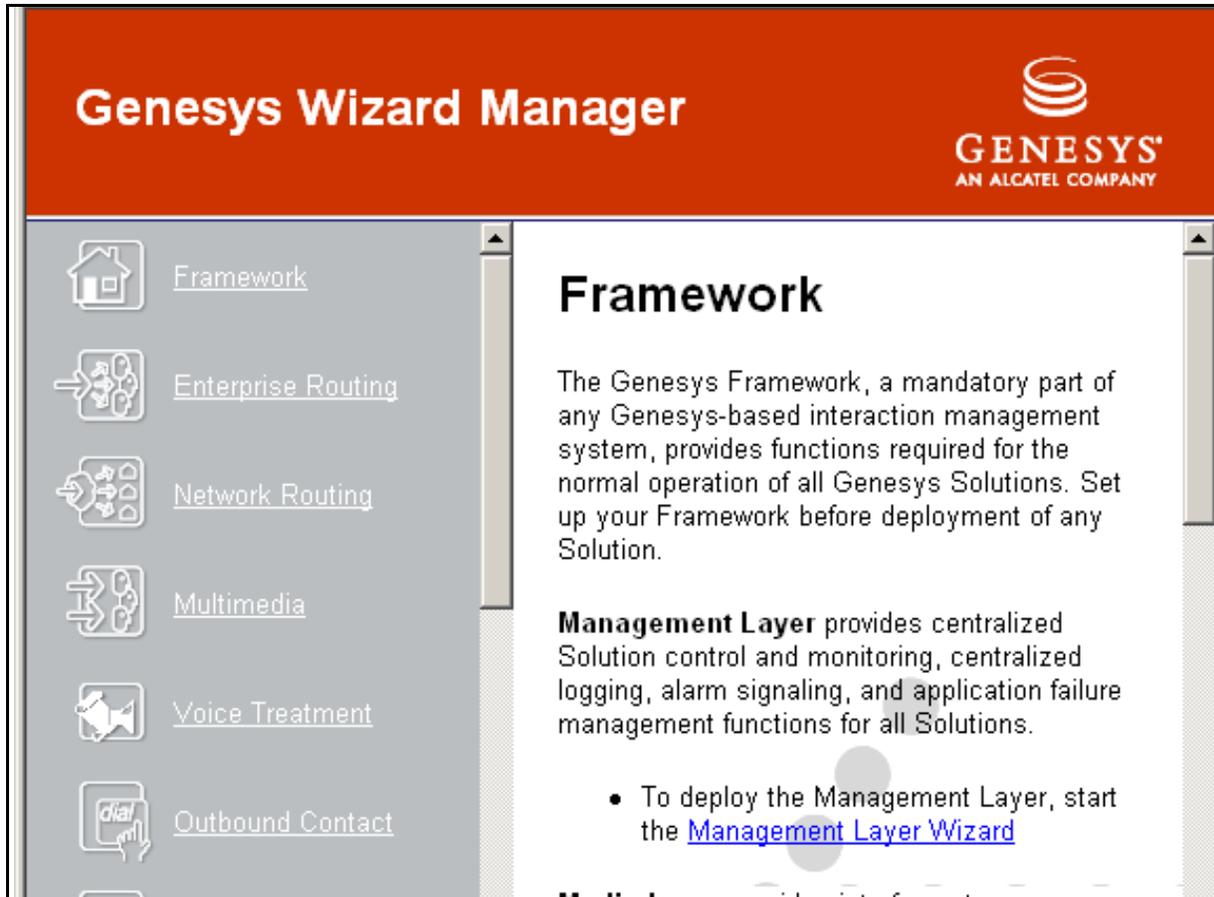


Figure 42: Genesys Wizard Manager, Framework

End of procedure

Procedure: Creating an Enterprise Routing solution

Start of procedure

1. Click the Enterprise Routing link on the left side of the window shown in [Figure 42](#) to open the Enterprise Routing window.
 - If you previously installed a solution, the Enterprise Routing window lists currently installed solutions. In this case, click the link to deploy another solution in your contact center. Continue with step 2.
 - If you have not yet installed a solution, click the link: [Deploy Enterprise Routing Solution](#) your contact center (see [Figure 43](#)).



Figure 43: Genesys Wizard Manager, Enterprise Routing

The Welcome to the Universal Routing Wizard screen opens (see [Figure 44](#)).



Figure 44: Welcome to the Universal Routing Wizard

2. Click Next to continue. The Solution Name screen appears (see [Figure 45](#)).

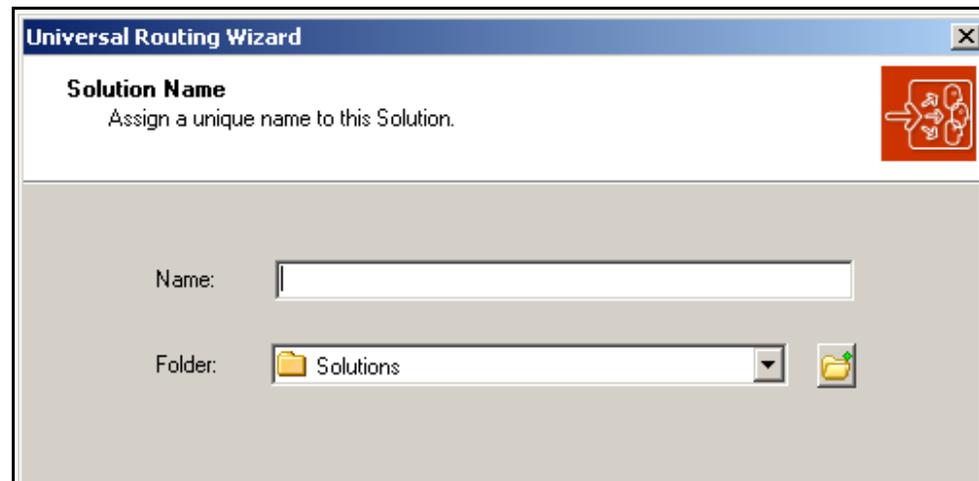


Figure 45: Universal Routing Wizard, Solution Name

3. Name the solution, select a folder (or keep the default) to store the solution executable, and click Next (button not shown in [Figure 45](#)). The Tenant screen appears (see [Figure 46](#)).

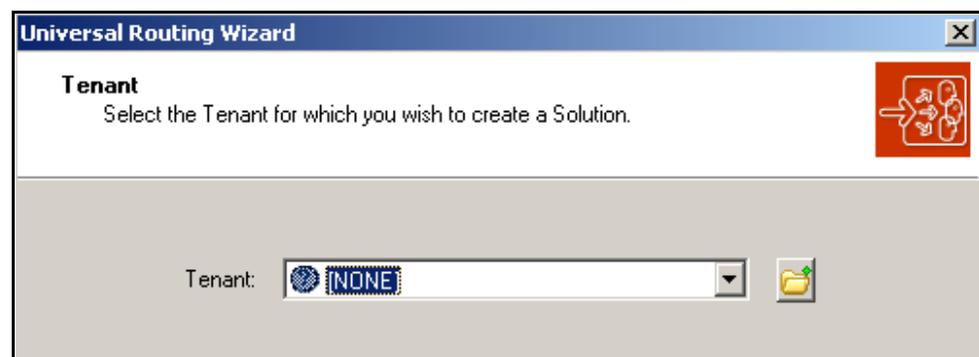


Figure 46: Universal Routing Wizard, Tenant

4. If Single-Tenant, leave the default entry of NONE. If Multi-Tenant click the folder icon to browse for and select a Tenant. In both cases, click Next.
 - If one or more Switches for the selected Tenant are not configured correctly, the Wizard instructs to use the Framework Configuration Wizard to correct configure a Switch.
 - If Switches are configured correctly, the Switches screen appears. [Figure 47](#) shows a sample; your screen will list different switches.

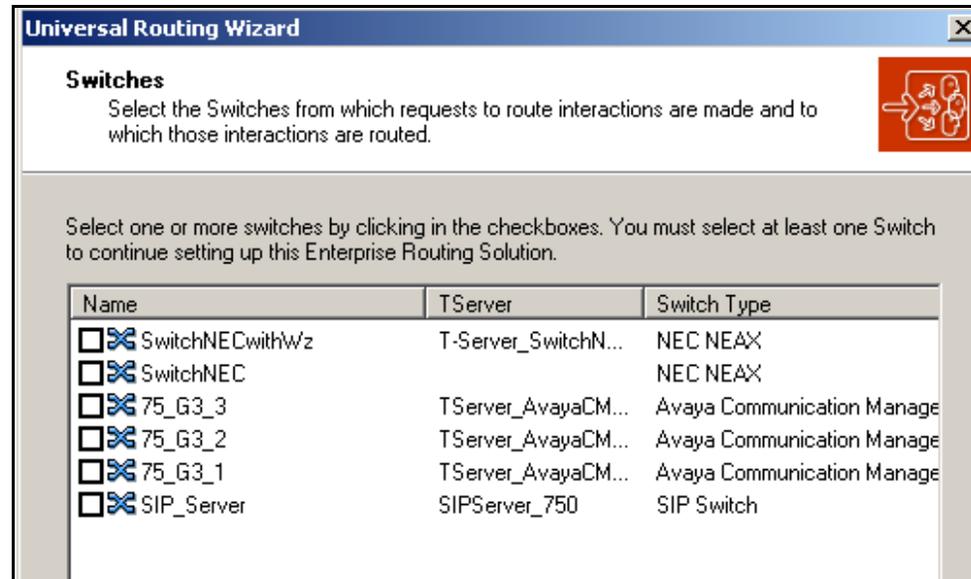


Figure 47: Universal Routing Wizard, Switches

5. Select one or more Switches and click Next. The Routing Features screen appears listing features available for configuration in the Wizard (see Figure 48).

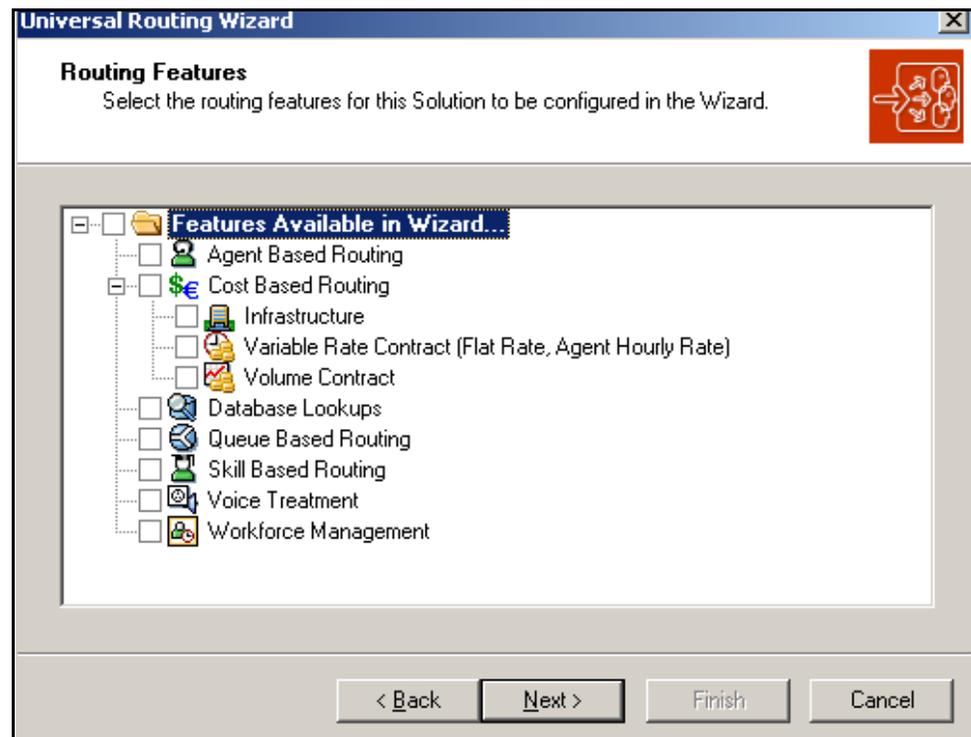


Figure 48: Universal Routing Wizard, Routing Features

Note: If you select Cost Based Routing, Genesys recommends that you send an e-mail to CBRSolution@genesys.lab.com. Depending on the number of features you implement, cost-based routing can be complex to configure for the first-time user. Because of this, and in order to ensure successful implementations, Genesys wants to track customers implementing this feature via contact with Genesys Technical Support or by e-mailing CBRSolution@genesys.lab.com.

6. Select the routing features you wish to implement, and click Next.

After clicking Next, the Framework Resources screen appears. The content of the screen varies depending on what you previously selected. Figure 49 shows an example screen after selecting Agent Based Routing.

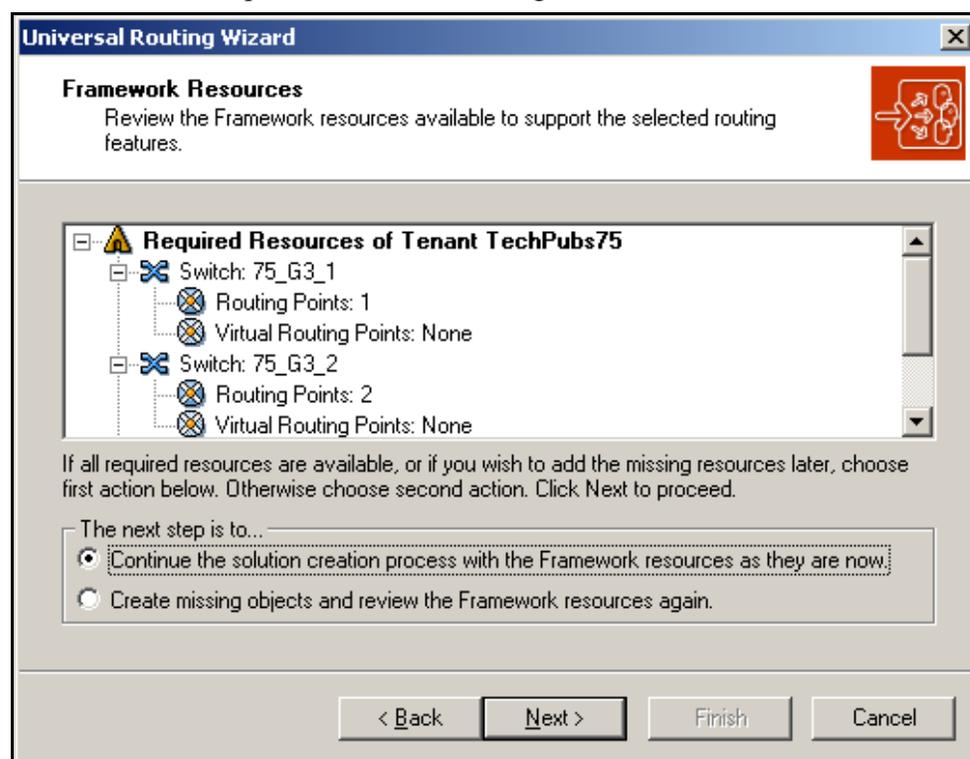


Figure 49: Universal Routing Wizard, Framework Resources

Note: The upper half of your Framework Resources screen reflects what you previously selected in the Routing Features screen (see Figure 48 on page 159), as well as your existing Configuration Environment.

7. Note whether None appears after any configuration objects. If None appears (and is not followed by Attention!), you may want or may not want to configure those objects.

- For example, in [Figure 49](#), the Framework Resources screen shows None opposite Virtual Routing Points. While the screen shows that no Routing Points have been configured, you may or may not wish to configure them. In this case, click the button opposite Create missing objects and review the Framework resources again followed by Next.

If one or more configuration objects are required for a routing feature you selected (see [Figure 48](#) on [page 159](#)) and your Configuration Environment does not contain those objects, then None - Attention! appears on the Framework Resources screen. [Figure 50](#) shows an example.

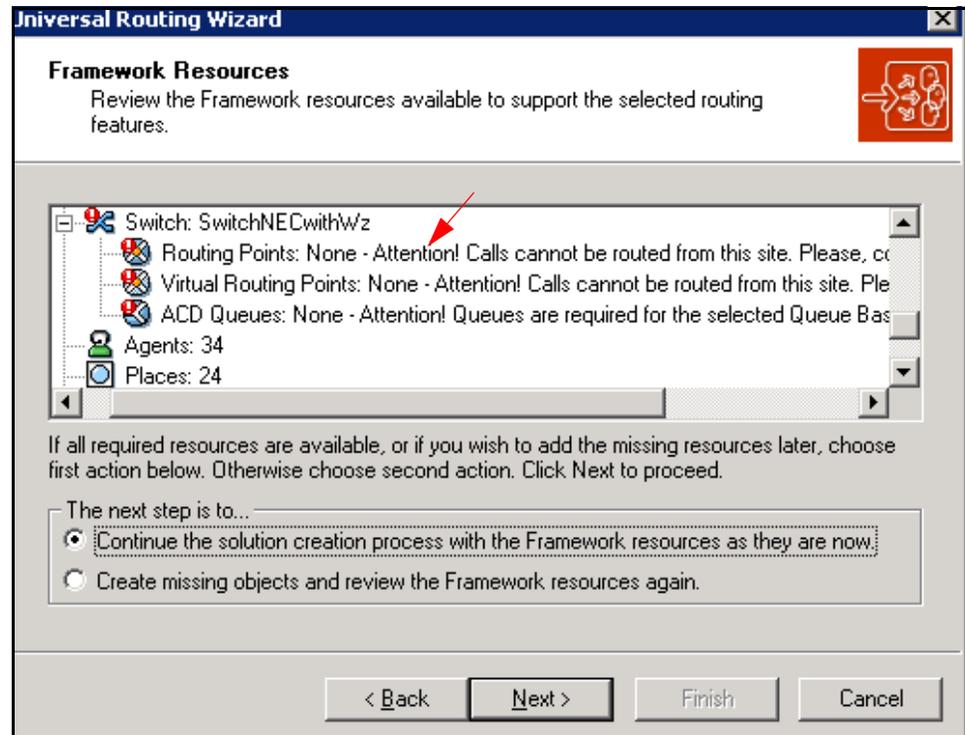


Figure 50: Framework Resources, Use of None and Attention

- Based on the above information, click the appropriate option button on the Framework Resources screen followed by Next.
- Follow the steps in the Wizard to create an ERS, using the information in your deployment planning worksheets (see “Planning Worksheets” on [page 119](#)) to complete the required fields.

End of procedure

Next Steps

- See “Creating a URS Application object” on [page 163](#).

Special Note on Adding/Creating Applications

The Wizard lets you browse for an existing Application object or create a new Application object. To do either action, you must first click the Add button.

Figure 51 shows the Add button in an example Solution Components window for Universal Routing Server.

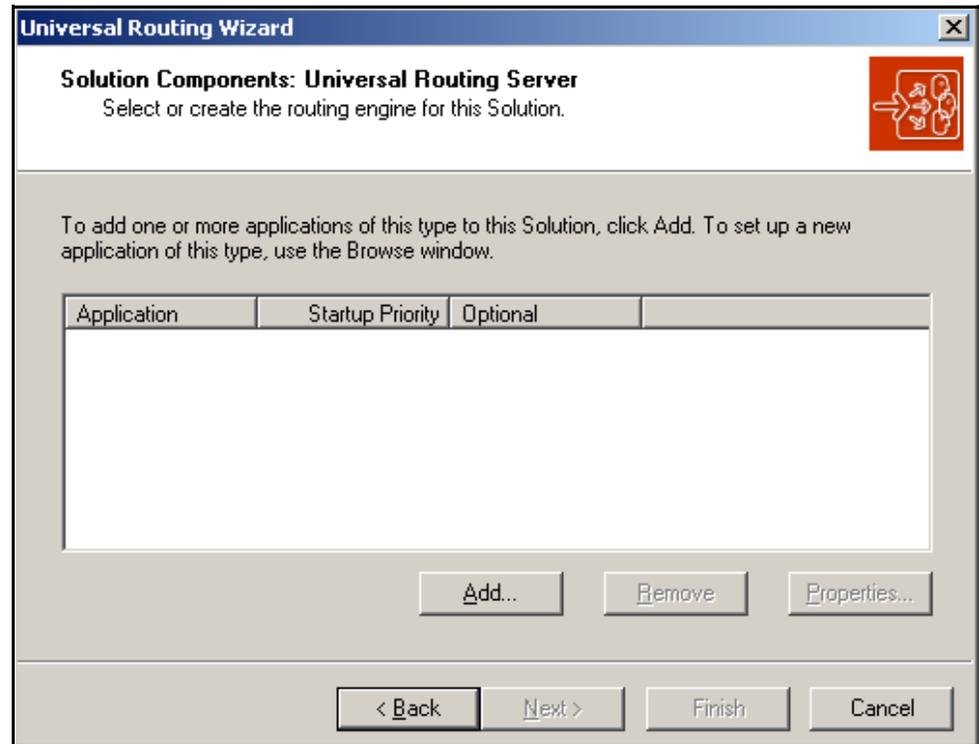


Figure 51: Solution Components

Clicking Add in Figure 51 opens the Browse for Application dialog box.

Figure 52 shows an example and the location of the button for creating a new Application button; your dialog box may be empty of existing Applications.

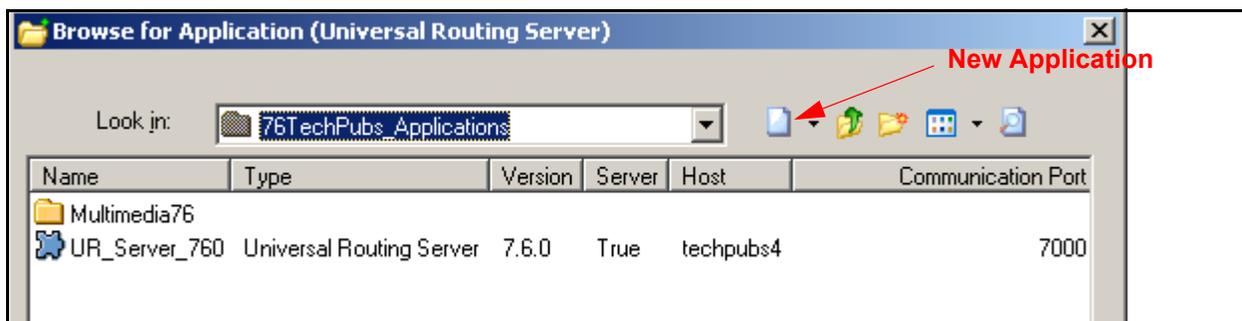


Figure 52: New Application Button in Browse for Application Dialog Box

- To create a new Application object, click the button for a new application in the dialog box (see Figure 52).

- To use an existing Application object, select it under Name in the dialog box (see [Figure 52](#)) or browse for it in another folder by clicking the up arrow.

Procedure: Creating a URS Application object

After you complete the Universal Routing Wizard screens for Stat Server, DB Server/Access Point, Message Server, Custom Server (optional), and WFM Server (optional), the Solution Components: Universal Routing Server screen appears as shown previously in [Figure 51](#) on [page 162](#).

Start of procedure

1. Click Add in this screen to bring up the Browse for Application (Universal Routing Server) dialog box shown in [Figure 52](#) on [page 162](#).
2. Click the icon for New Application shown in [Figure 52](#) on [page 162](#). This brings up the Welcome to Universal Routing Server Wizard screen (see [Figure 53](#)).



Figure 53: Welcome to Universal Routing Server Wizard

3. Click Next. The Universal Routing Server Name screen appears. [Figure 54](#) shows the screen with example entries.

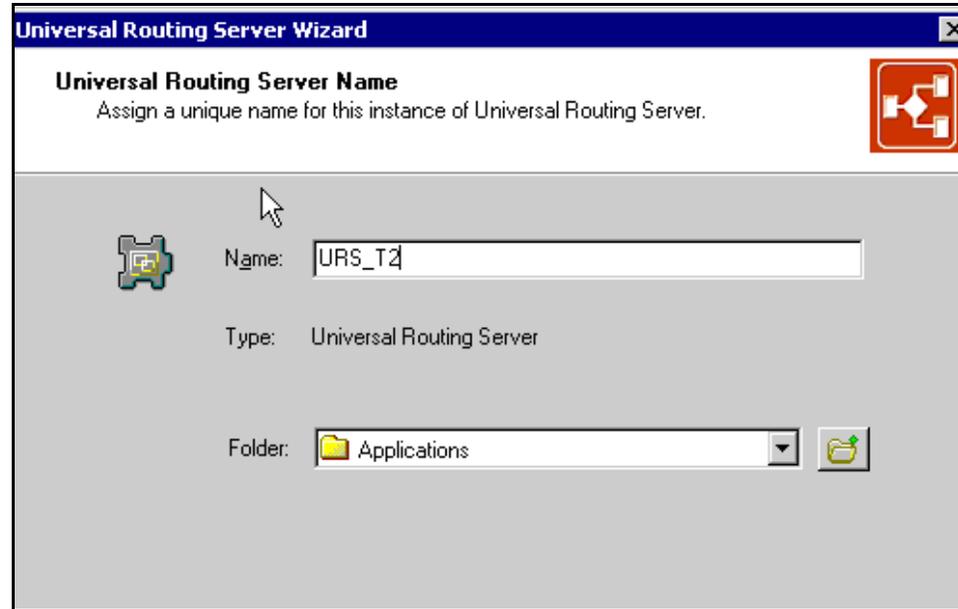


Figure 54: Universal Routing Server Name

4. Enter a name for the URS Application object, select a folder, and click Next (button not shown). The Log Configuration screen appears (see [Figure 55](#)).

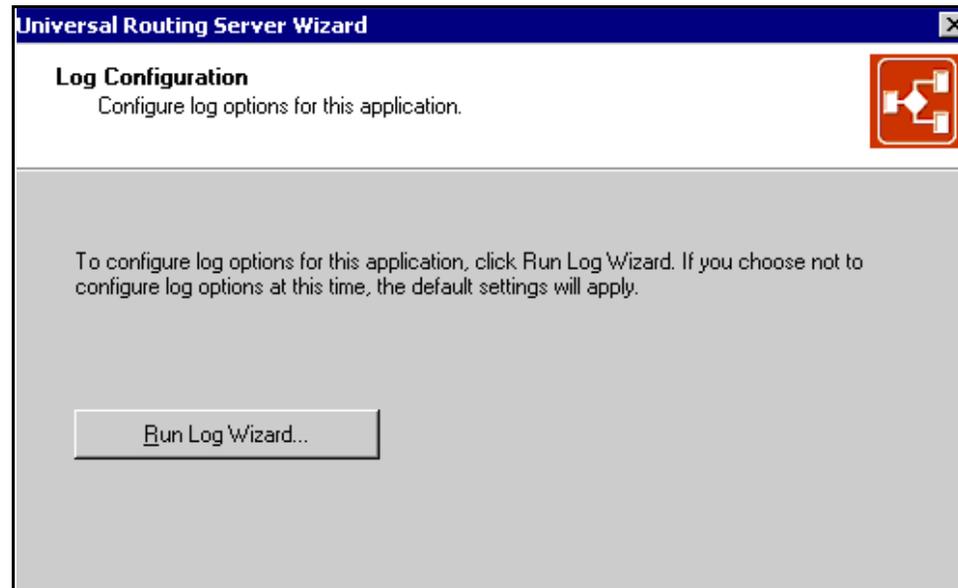


Figure 55: Log Configuration

5. Assuming you choose to configure log options now, click the Run Log Wizard button. The Log Level screen appears (see [Figure 56](#)).

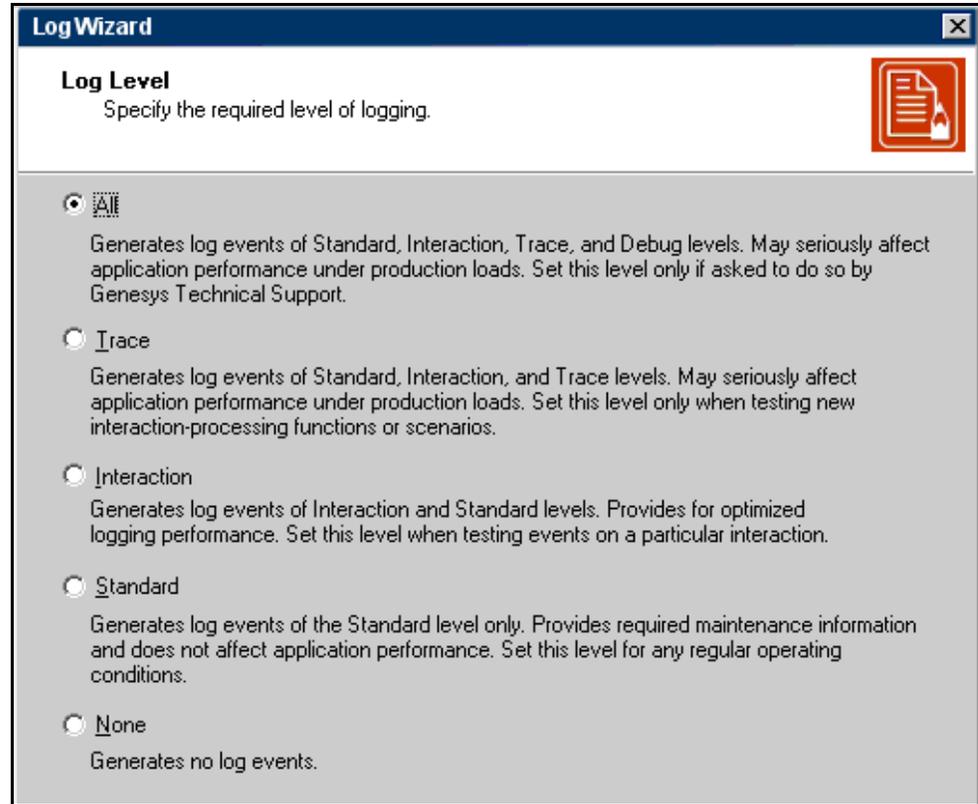


Figure 56: Log Level

- Click a radio button to specify the type of log events that the log file should include and then click **Next** (button not shown). The **Log Outputs** screen appears. [Figure 57](#) shows the screen with example entries.

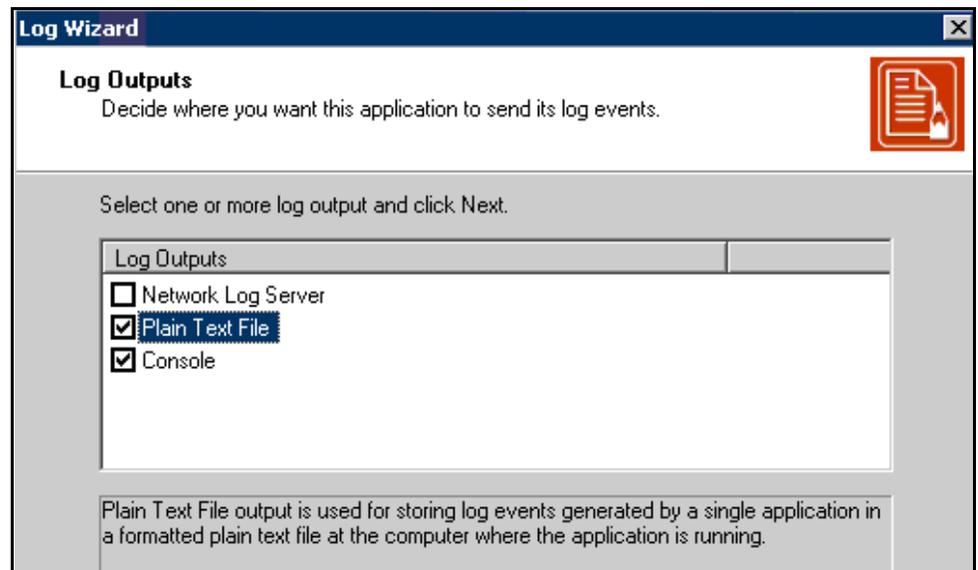


Figure 57: Log Outputs

- Indicate where you want URS to send its log events and then click Next (button not shown). The Log Output Adjustment screen appears. [Figure 58](#) shows the screen with example entries.

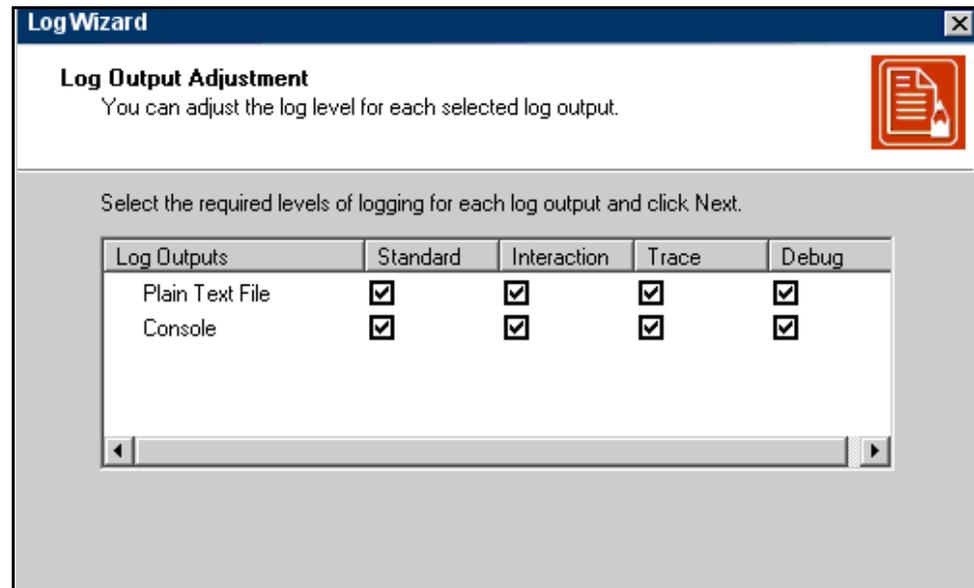


Figure 58: Log Output Adjustment

- Adjust the logging level for each output type and then click Next (button not shown). The Log File Name screen appears (see [Figure 59](#)).

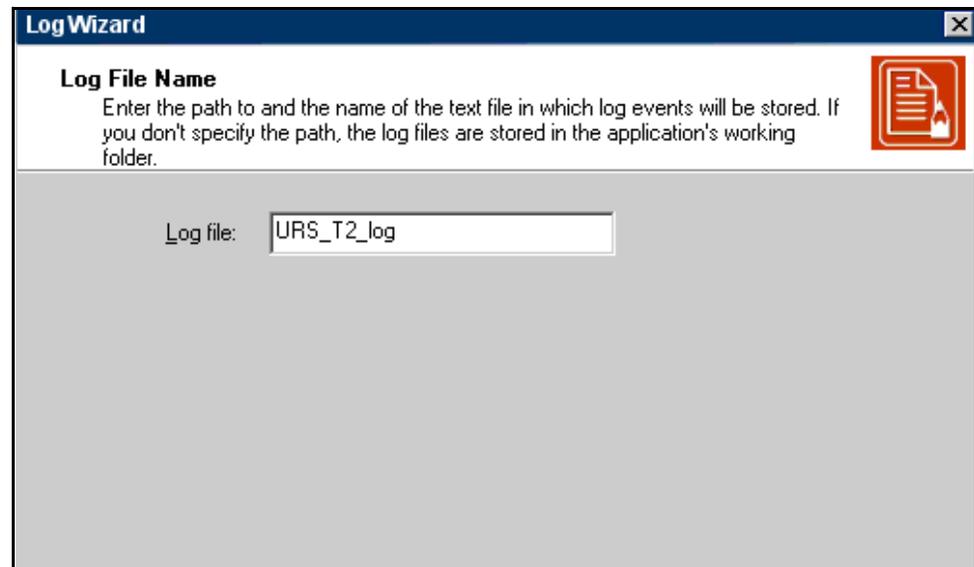


Figure 59: Log File Name

- Enter the path and/or name of the file for storing log events and click Next (button not shown). The Log Segmentation File screen appears. [Figure 60](#) shows the screen with example entries.

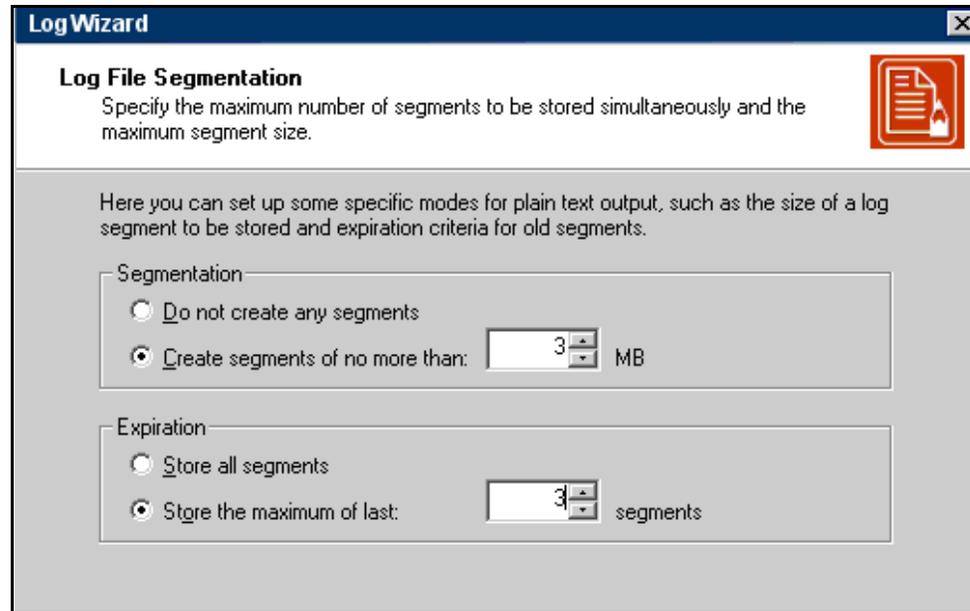


Figure 60: Log File Segmentation

10. Specify Segmentation and Expiration information if needed and click Next (button not shown). The Log Options Summary screen appears. [Figure 61](#) shows an example.

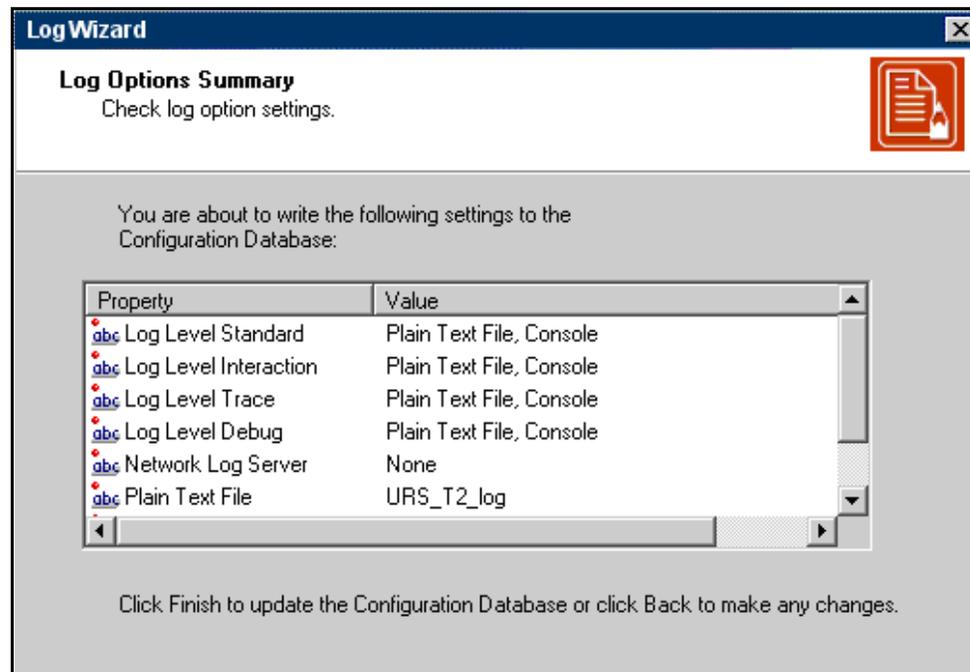


Figure 61: Log Options Summary Screen

11. Click Finish if no more changes. Click Back if you wish to make changes. Assuming you click Finish, the Log Configuration screen re-appears as shown previously in [Figure 55](#) on [page 164](#).

12. Click **Next** on the **Log Configuration** screen to bring up the **Server Information** screen.
13. Select the **Host** where this URS should be installed and specify the port number when this URS acts as a client to Configuration Server. [Figure 62](#) shows example entries.

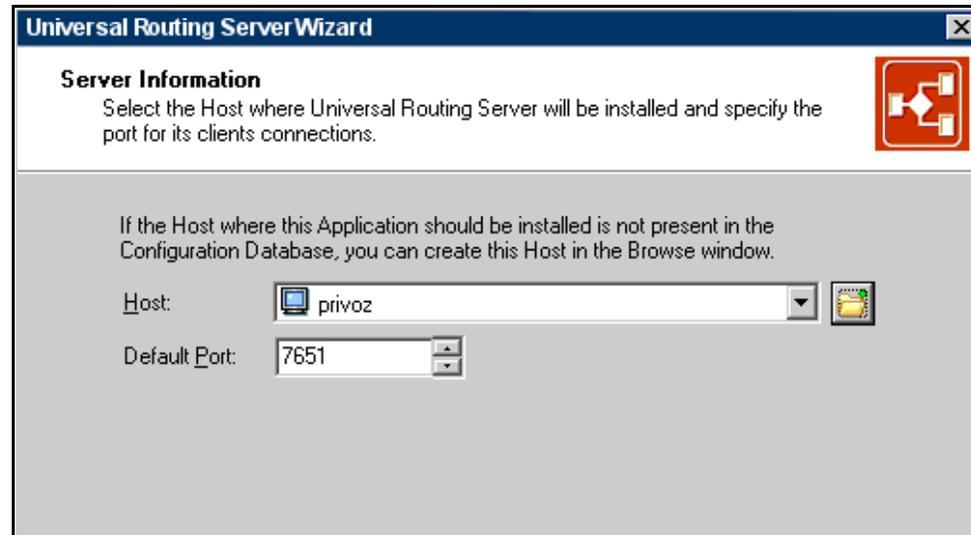


Figure 62: Server Information

14. Click **Next** (button not shown) to bring up the **Application Connections to T-Server** screen (see [Figure 63](#)).

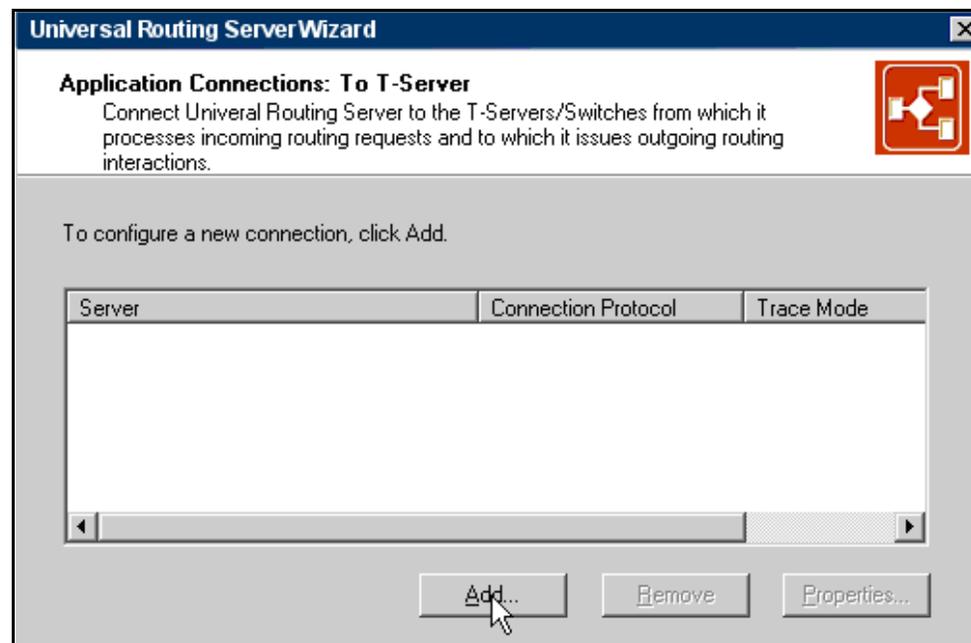


Figure 63: Application Connections: To T-Server

15. Click **Add** to bring up the **Connection Wizard** listing available T-Servers.

16. Select a T-Server or click **New**. [Figure 64](#) shows an example screen after selecting a T-Server.

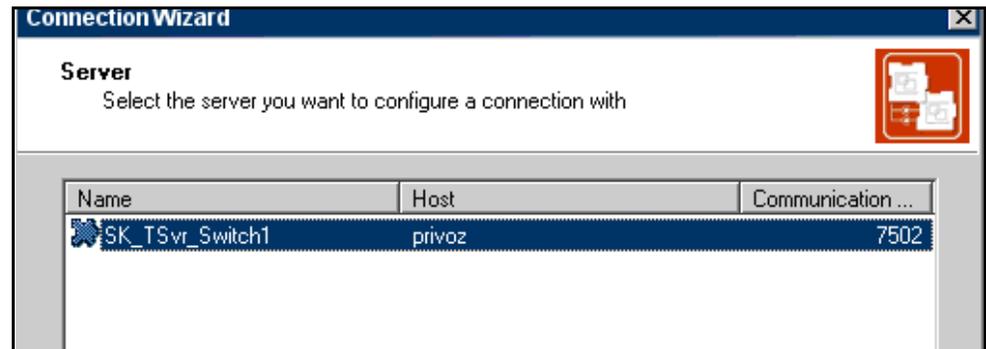


Figure 64: Connection Wizard, Selecting a T-Server

17. Assuming you have selected a T-Server, click **Next** (button not shown) to bring up the **Port and Connection Attributes** screen, which can be used for client-side port definition. (see [Figure 65](#)).

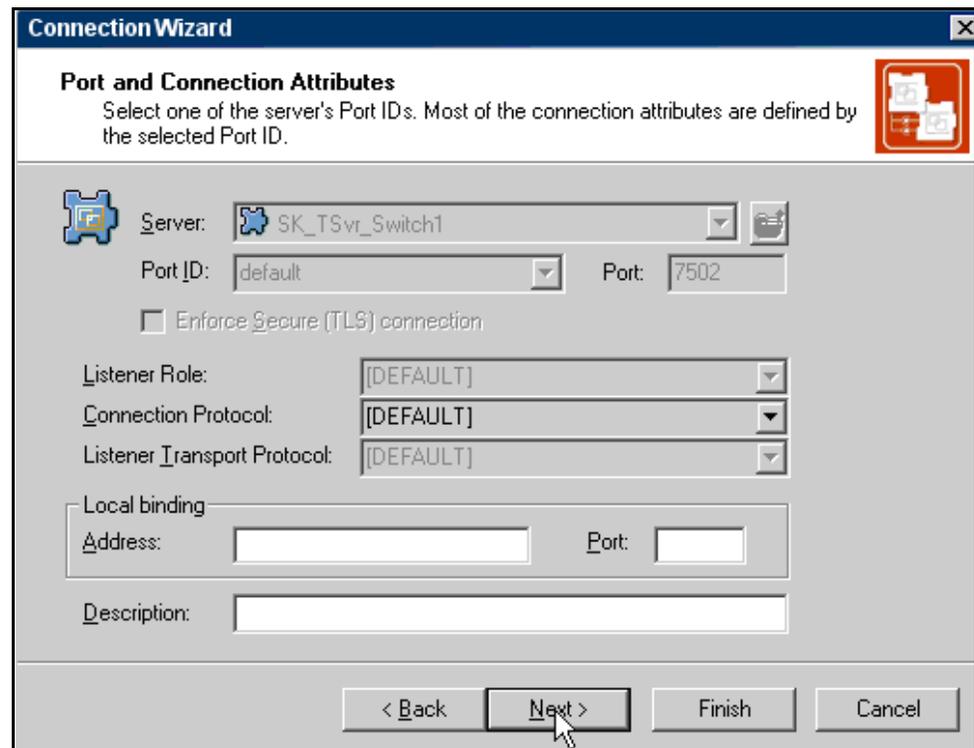


Figure 65: Port and Connection Attributes

18. For the T-Server you previously selected, specify the connection attributes. [Figure 66](#) shows a screen with example entries.

Figure 66: Port and Connection Attributes

Assume that you defined the Client-Side Port Definition data shown above.

- Click Next. The Advanced Settings screen appears as shown in [Figure 67](#).

Figure 67: Advanced Settings

- After entering parameters on the Advanced Settings screen, click Finish (button not shown). The Application Connections To: Stat Server screen appears (see [Figure 68](#)).

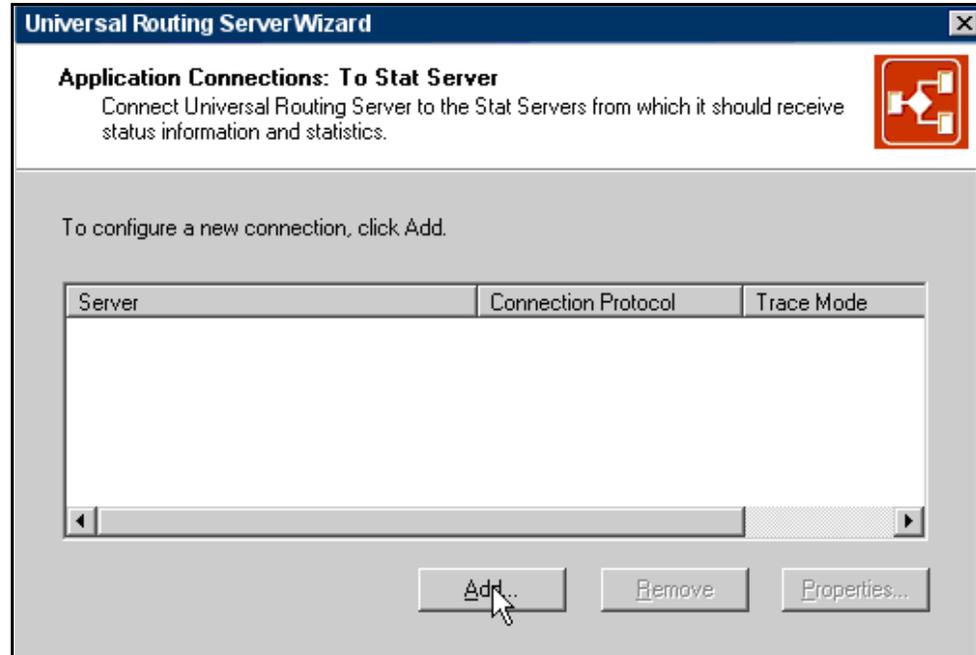


Figure 68: Application Connections: To Stat Server

21. Click Add to bring up a Connection Wizard Server screen similar to that shown in Figure 64 on [page 169](#).
22. Follow the Wizard prompts to configure Stat Server Application connections using the steps described above in the T-Server example as a guide.
23. Continue adding all other Applications as needed.

Note: Client-side port definition is not supported for all Genesys Applications and is optional.

End of procedure

Procedure:

Creating an IRD Application object

After you complete the Wizard screens for Stat Server, DB Server/Access Point, Message Server, Custom Server (optional), WFM Server (optional), and Universal Routing Server, the following screen appears: Solution Components: Interaction Routing Designer (see [Figure 69](#)).

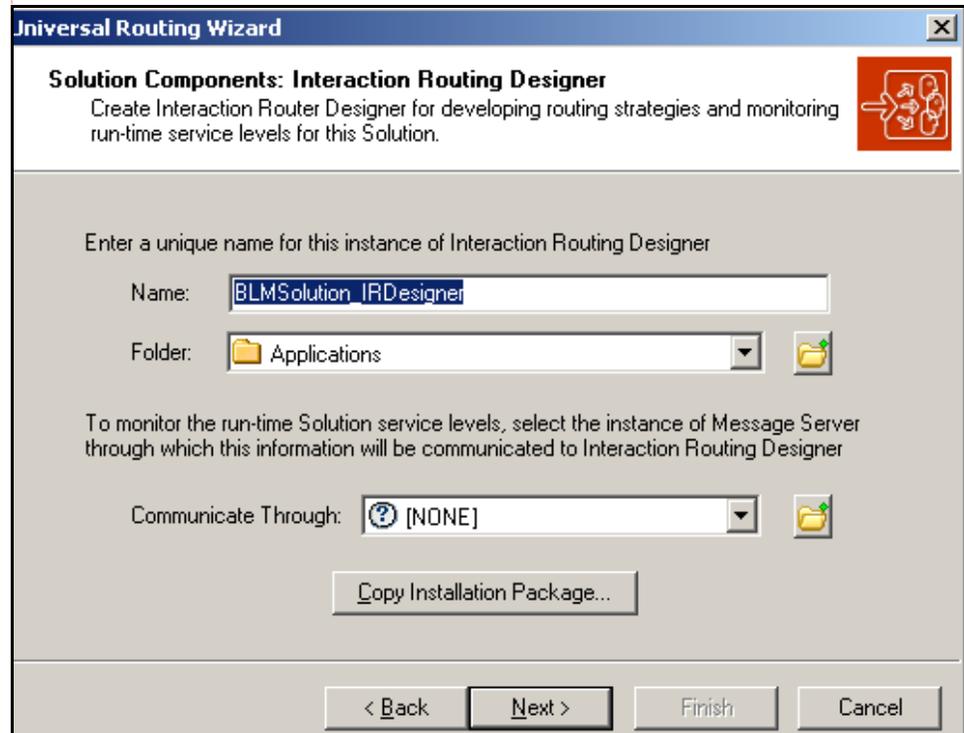


Figure 69: Solution Components Screen, Copy Installation Package Button

Start of procedure

1. Change or keep the Wizard name for this instance of Interaction Routing Designer.
2. Opposite Folder, keep the default or select a different storage location for this instance of IRD.
3. Opposite Communicate Through, select a Message Server or create a new Message Server for communication of IRD's monitoring information (use the button for creating a new Application as shown in Figure 52 on [page 162](#)).

End of procedure

Procedure: Configuring Reporting, IVR Interface, and Voice Treatment Options

Note: The procedure below is optional. You can always choose to configure these options at a later time.

Start of procedure

1. To configure one of the above Solution Options, click Next on the Solution Components: Interaction Routing Designer screen shown in Figure 69 on page 172. The Solution Options screen appears (see Figure 70).

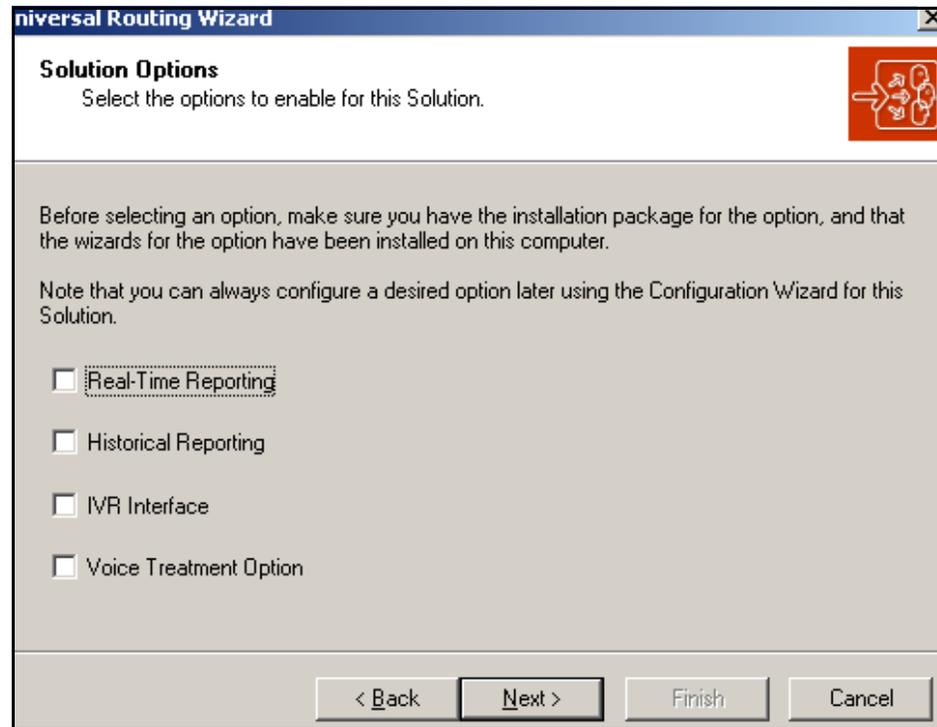


Figure 70: Universal Routing Wizard, Solution Options

2. Select one or more of the following:
 - Real-Time Reporting
 - Historical Reporting
 - IVR Interface
 - Voice Treatment Option
3. Click Next.

End of procedure

Procedure:

Copying the installation package to a directory

Purpose: To place the solution installation package in a directory so you can run the setup file later as described in Chapter 13, “Installing a Routing Solution,” on page 311.

Start of procedure

1. Click **Copy Installation Package** in Figure 69 on [page 172](#). The **Welcome to the Interaction Designer Wizard** screen appears.
2. Click **Next** on the **Welcome to the Interaction Designer Wizard** screen. The **Installation Package** screen appears (see [Figure 71](#)).

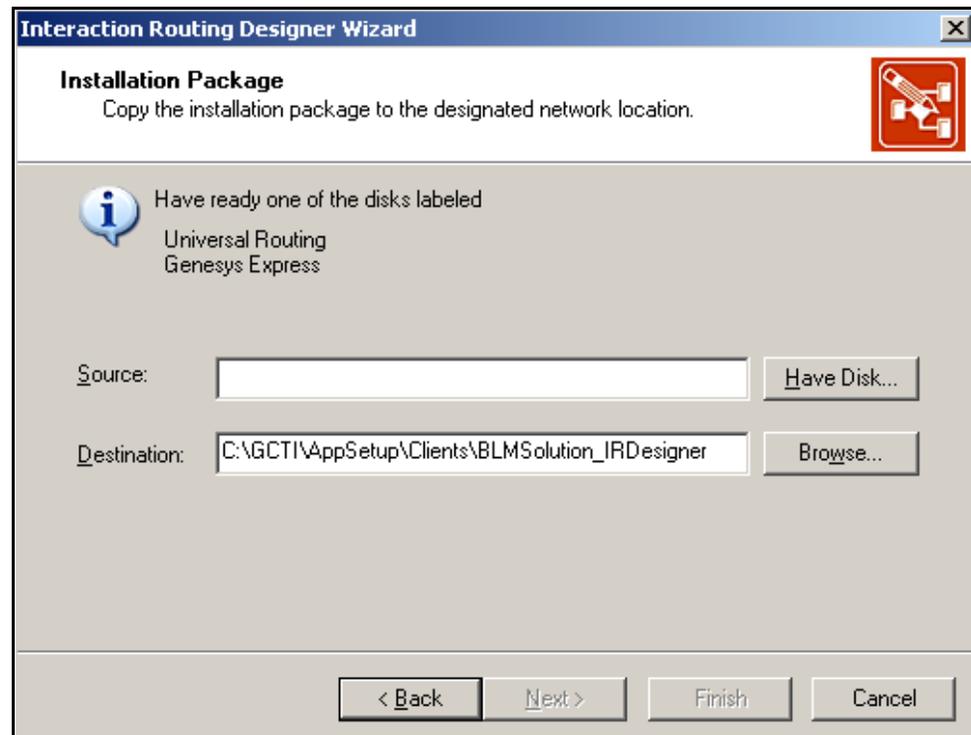


Figure 71: Universal Routing Wizard, Installation Package

3. Opposite **Source**, click **Have Disk**.
4. Select the CD-ROM drive of the computer containing the **Universal Routing CD**. In the **Browse for Folder** dialog box, locate the root of the CD or the CD image. The **OK** button becomes enabled when you reach the appropriate directory.
5. For **Destination**, specify the destination directory where the installation package should be copied.
6. Make a note of the destination directory.
7. Click **Finish**.

End of procedure

Next Steps

As stated on [page 152](#), you first use the Wizard to configure a solution and create an installation package. You then must install the package as described in Chapter 13, “Installing a Routing Solution,” on [page 311](#).

Network Routing Wizard

Note: Universal Routing includes a Network Routing Wizard.

Start the Network Routing Wizard through the Wizard Manager (see Figure 42 on [page 156](#)). You are already familiar with Wizard Manager since you used it to install and configure Framework.

Task Summary: Configuring Network Routing with the Wizard

Objective	Related Procedure and Actions
Install the Network Routing Wizard and configure a solution	Procedure: Creating a Network Routing solution, on page 175
If a network switch is not detected, configure a network switch	Procedure: Configuring a network switch using the Wizard, on page 177

Procedure: Creating a Network Routing solution

Start of procedure

1. Insert the Network Media Interfaces CD.
2. Double-click `setup.exe` in the root directory to install the Configuration Wizard for Network T-Servers which enables you to configure Network T-Servers during the configuration of a solution.
3. Insert the Universal Routing CD into the CD-ROM drive.
4. Double-click `setup.exe` in the root directory to install the Universal Routing Wizard that enables you to configure Network Routing solutions.
5. Open Genesys Configuration Wizard Manager. From the Start menu, the default is:

Programs > Genesys Solutions > Genesys Wizard Manager

Note: Your site may have installed Genesys Wizard Manager in another directory.

The Welcome to Wizard Manager screen opens (see Figure 40 on [page 154](#))

6. Click the `log into the Configuration Layer` link. The Wizard Manager login dialog box opens. Figure 41 on [page 155](#) shows the dialog box after expanding it by clicking the `Details` button.

7. Enter the login information for the Configuration Manager application at your site and click OK. The Framework screen opens (see Figure 42 on page 156).
8. Click the Network Routing link on the left side of the screen shown in Figure 42 on page 156. The Network Routing window opens.



Figure 72: Genesys Wizard Manager, Network Routing

9. Click the link: Deploy [Network Routing](#) Solution in your contact center.

End of procedure

Next Steps

- Follow the steps in the Wizard, using the information in your deployment planning worksheets, to complete the required fields. The starting steps are similar to those shown in the procedure “Creating an Enterprise Routing solution” on [page 156](#).

Note: Be sure to note the directory containing the installation package. As stated on [page 152](#), you first use the Wizard to configure a solution and create an installation package. You then install the configuration package as described in Chapter 13 on [page 311](#).

- As the installation proceeds, the Wizard searches for a properly configured network switch. This switch is required for the solution.
 - If a switch is detected, the component installation continues.
 - If one is not detected, a dialog box appears informing you of the need for a network switch. Before proceeding any further in the installation process, you need to correctly configure the network switches.

When a Network Switch Is Not Detected

When installing Network Routing, after selecting a Tenant, the Wizard detects whether a network switch has been configured. If one has not been configured, the message shown in [Figure 73](#) appears.

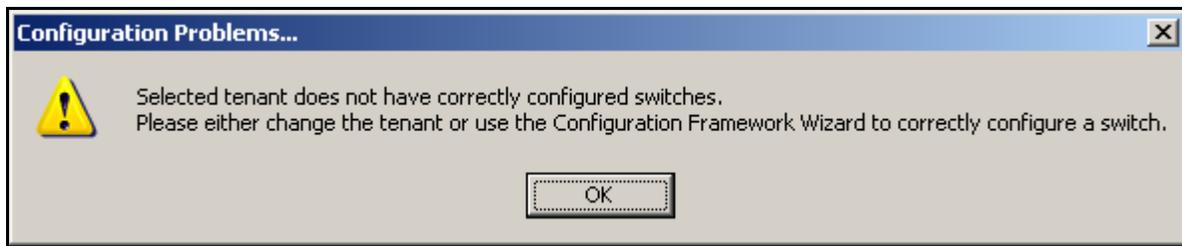


Figure 73: Configuration Problems Dialog Box

If a configured network switch is not detected and you wish to configure one, proceed as follows:

Procedure:

Configuring a network switch using the Wizard

Purpose: To create a network Switch and associated T-Server so you can finish creating the solution.

Start of procedure

1. When the warning dialog box (see [Figure 73](#)) appears noting that the selected tenant does not have correctly configured switches (in this case, meaning no network switch), click OK.
2. On the Network Routing screen, click the Framework link (see [Figure 72](#) on [page 176](#)).
3. On the Framework screen, click the link: Create a new Switch object and deploy associated T-Server. The Switch Wizard is invoked.
4. Click Next.
5. Select a Tenant and click Next.
6. In the Switch Name dialog box, enter a unique name for the Switch or Service Control Point (SCP) and click Next.
7. In the Switching Office dialog box, create the Switch Office to which the Switch connects and specify the Switch Office type. [Table 7](#) lists the Switching Office types for each 8.0 Network T-Server:

Table 7: 8.0 Network T-Server and Switching Office Types

Network T-Server	Switching Office Type
Network T-Server for AT&T	AT&T 800 ICP Gateway
Network T-Server for GenSpec	GenSpec
Network T-Server for MCI	WorldCom 800 Gateway
Network T-Server for NGSN	NGSN
Network T-Server for Sprint	Sprint SiteRP Gateway
Network T-Server for DTAG	Generic Switch or Unknown Switch Type
Network T-Server for CRSP	Generic Switch or Unknown Switch Type

8. In the Switch Summary dialog box, confirm your switch information. Once you do this, the Switch Wizard advances you to the Network T-Server Wizard.
9. Respond to the Network T-Server Wizard prompts:
 - For the carrier Name and Folder location.
 - For the Host name and Default Port.

The wizard then asks you to copy the Network T-Server installation package to a designated computer (see [Figure 74](#)).

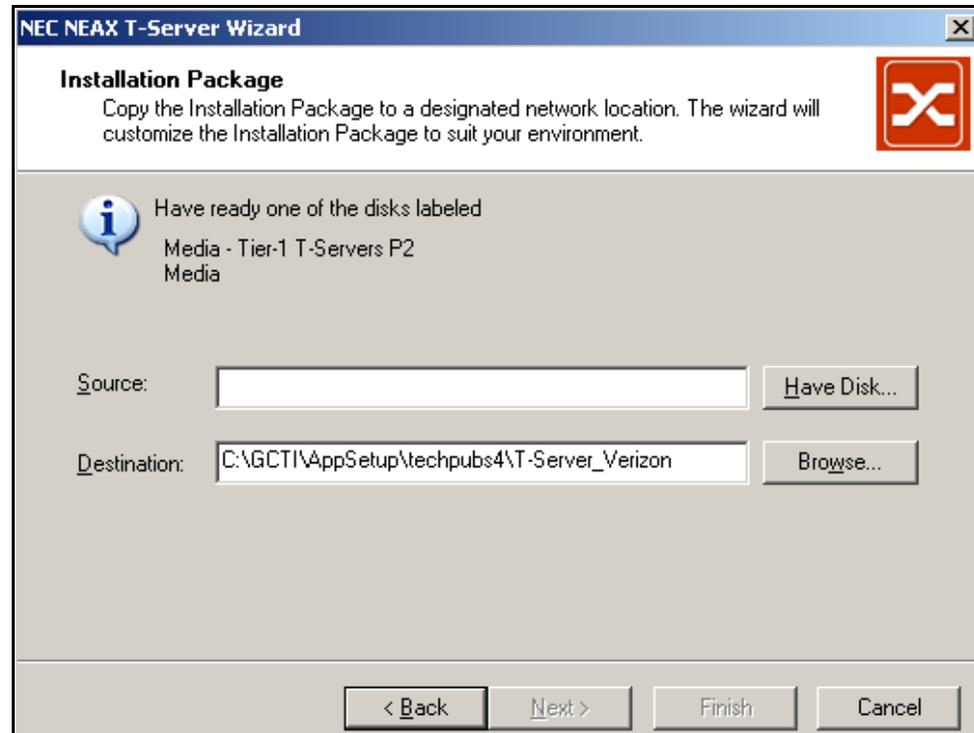


Figure 74: T-Server Wizard, Installation Package Dialog Box

10. Click **Have Disk**. Select the CD-ROM drive of the computer where the Network Media Interfaces CD is placed. In the **Browse for Folder** dialog box, locate the root of the CD or the CD image. The **OK** button becomes enabled when you reach the appropriate directory.
11. For **Destination**, specify the destination directory to which the installation package will be copied; this must be a directory on the host where Network T-Server will be installed.

End of procedure

Next Steps

- After the Network T-Server installation package is copied, make sure to manually install Network T-Server on the assigned computer.

Warning! This does not happen within the Wizard.

- When the Network T-Server is configured, the installation of Network Routing can continue. Follow the Wizard instructions to install and configure the remaining components.

Note: For more information on configuring a Network T-Server, refer to the Network T-Server documentation for your specific T-Server.

Routing Solutions and Message Server

When setting up a routing solution, you can configure one or more Message Servers to perform different functions:

- **Logging:** a Message Server configured for logging collects and queues server log messages and then commits the log messages to the log database through a DB Server.
- **IRD Monitoring:** a Message Server configured for monitoring serves as a communication channel between Interaction Routing Designer (IRD) and Universal Routing Server (URS). This Message Server controls the delivery of messages between the applications to provide IRD users with a real-time view of the status of interaction routing.
- **Router Self-Awareness** (see [page 255](#))

When using the Message Server Wizard to configure Message Servers, the servers are automatically set up to handle both logging and monitoring tasks. To specify a Message Server for logging only, you must do so manually. See “Configuring Message Servers for Different Functions” on [page 229](#) for more information.



Chapter

8

Manually Configuring Routing

Before you can install the routing components as described in Chapter 13 on [page 311](#), they must be configured. This chapter describes the configuration process for network and enterprise components and includes the following topics:

- [How to Configure Manually, page 181](#)
- [Manually Configuring Enterprise Routing, page 186](#)
- [Limitations, page 189](#)
- [Manually Configuring Network Routing, page 191](#)
- [Manually Configuring URS, page 194](#)
- [Manually Configuring IRD, page 209](#)
- [Other Manual Configuration Operations, page 217](#)

See Chapter 12 on [page 289](#) for information on configuring Custom Server.

How to Configure Manually

Manual configuration is done entirely in Configuration Manager within the specific Genesys applications. It involves setting up properties of each application in a dialog box and adding properties to servers to which the application connects.

Note: Universal Routing 8.0 can also be configured using Genesys Administrator. Refer to the *Framework 8.0 Genesys Administrator Help* for more information.

Task Summary: Manual Configuration

Objective	Related Procedures and Actions
Log into Configuration Manager	Procedure: Logging into Configuration Manager, on page 183
Create the Universal Routing Server Application object	Procedure: Creating/Configuring the URS Application object, on page 194
Configure the use_agent_capacity option	Procedure: Viewing/changing the use_agent_capacity option, on page 202
Configure Web Interface functionality	Procedure: Configuring URS Web Interface Functionality, on page 208
Configure Universal Routing Server to communicate with Workforce Manager	Procedure: Configuring URS to communicate with WFM, on page 209
Create the Interaction Routing Designer Application object	Procedure: Creating and configuring the IRD Application object, on page 210
Configure the inactivity-timeout option	Procedure: Setting the inactivity-timeout option, on page 212
Display the Interaction Design shortcut bar	Procedure: Displaying the Interaction Design shortcut bar, on page 214
To save strategies in a custom folder, create a Script subfolder.	Procedure: Creating a subfolder of object type Script, on page 215
To save strategies in a database, create the ird_strategies table	Procedure: Creating the ird_strategies table, on page 216
Set permissions to modify strategy schedules	Procedure: Creating permissions to modify strategy scheduling, on page 218
Configure a Solution object	Procedure: Configuring an Enterprise Routing Solution object, on page 219 or Procedure: Configuring a Network Routing Solution object, on page 223
For a Network Routing Solution, configure a network switch	Procedure: Configuring a Network switch, on page 225
Configure a Stat Server Application object	Procedure: Configuring a Stat Server Application, on page 227

Task Summary: Manual Configuration (Continued)

Objective	Related Procedures and Actions
For a Network Routing Solution, configure a premise T-Server Application object	Procedure: Configuring a premise T-Server Application for Network Routing , on page 228
Configure Message Servers	<ol style="list-style-type: none"> 1. Procedure: Configuring a Message Server for logging, on page 230 2. Procedure: Configuring a Message Server for monitoring only, on page 231
Configure DNSs	“Configuring DNSs Manually” on page 232

Procedure: Logging into Configuration Manager

Purpose: To manually configure the Application objects (Applications folder in [Figure 76](#)) associated with Universal Routing and other Genesys products.

Start of procedure

1. Open Configuration Manager. From the Start menu on your PC, the default path is Start > Genesys Solutions > Framework > Configuration Manager > Start Configuration Manager. The Configuration Manager login dialog box opens with the last entries. [Figure 75](#) on [page 184](#) shows an example.



Figure 75: Configuration Manager Login Dialog Box

2. Use the information in [Table 8](#) to complete the login dialog box.

Table 8: Configuration Manager Login Dialog Box

Field	Description
User name:	Name of Person object defined in Configuration Manager.
User password:	Password of Person object defined in Configuration Manager.
Application:	Enter the name of the Configuration Manager Application object or default.
Host name:	Name of machine where Configuration Server is running.
Port:	Port number used by Configuration Server.

3. Click OK in the login dialog box to open Configuration Manager.

End of procedure

Figure 76 on [page 185](#) shows Configuration Manager for a Multi-Tenant environment.

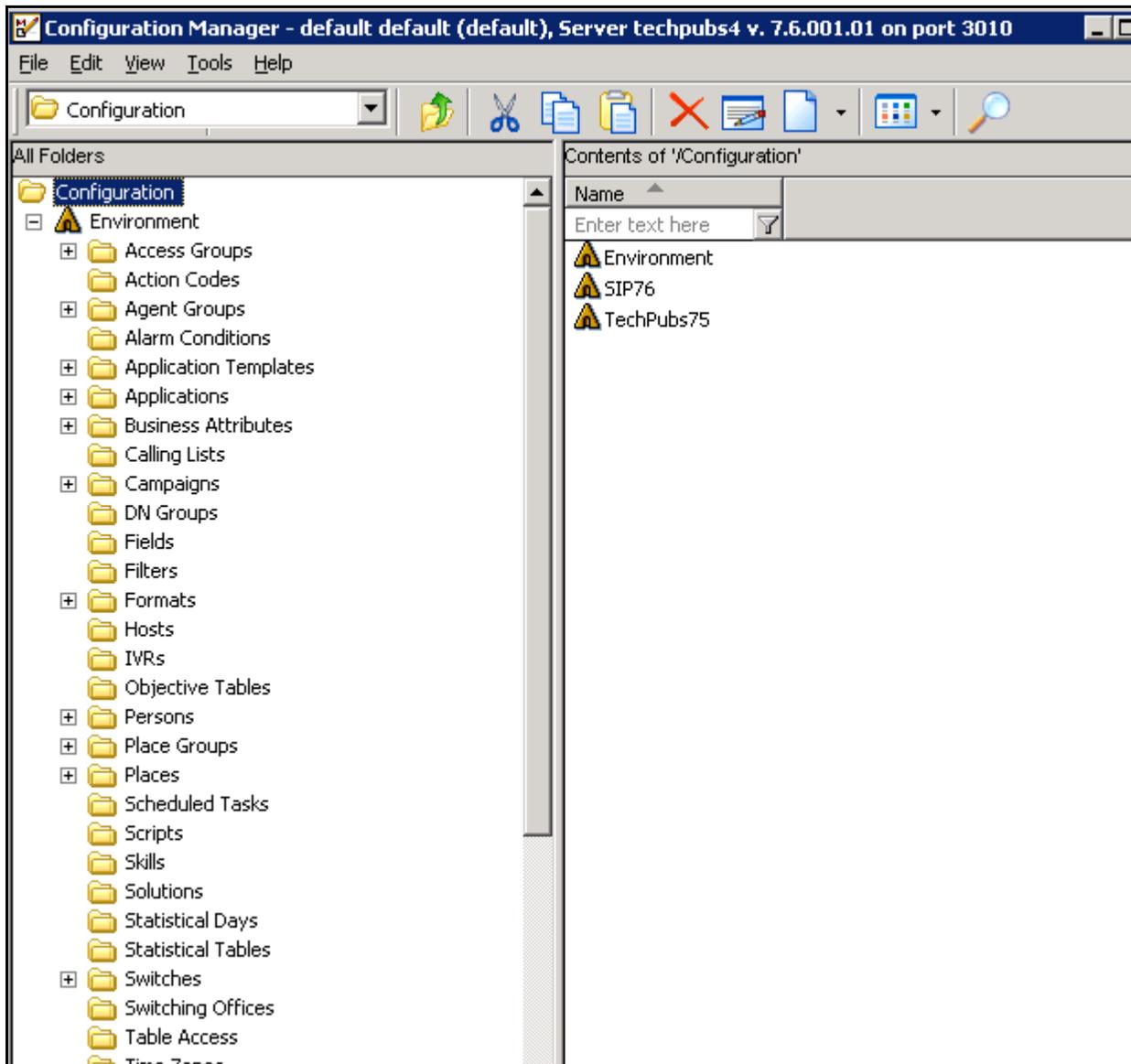


Figure 76: Configuration Manager

Note: If you wish to use the Management Layer and its Solution Control Interface (SCI as shown in Figure 125 on page 334) to stop and start applications, you must install Local Control Agent (LCA) as documented in the *Framework 8.0 Deployment Guide*.

After installing and configuring a solution, any changes made to the properties of an application or a configuration object may delay URS processes for a few seconds.

Manually Configuring Enterprise Routing

An Enterprise Routing solution requires that you configure and install the application components necessary for that solution based on the needs of your contact center.

The basic application components include Universal Routing Server (URS) and Interaction Routing Designer (IRD). See “Manually Configuring URS” on [page 194](#) for information on configuring these applications in Configuration Manager.

Configuration Order for eServices

To facilitate configuration when running with Genesys eServices, it is important to configure components in the following order:

1. Framework components
2. Stat Server (if planning to route based on agent-capacity rules, you will also need to install the Resource Capacity Wizard from the Real-Time Metric Engine CD.
3. Universal Routing components
4. eServices components

Note: The eServices Configuration Wizard assumes that Universal Routing configuration has been completed. To simplify the entire installation process, Genesys recommends configuring components in the order listed above. See “eServices Installation Considerations” on [page 144](#) for more information.

Enterprise Routing Objects Configuration Order

Configuring Enterprise Routing requires that you create objects in Configuration Manager (see [Figure 13 on page 37](#)).

The required objects and recommended order for creating them are listed below.

Under the `Tenant` or `Environment` (see [Figure 76 on page 185](#)):

1. Create `Switching Offices`. Use the types that correspond to the physical switches used.
2. Create `Hosts`. These must be only those computers on the data network where Genesys daemon processes run. Configure/install LCA on `Hosts` if you wish to use SCI. See [Chapter 13 on page 311](#) for more information.

3. Create `Application` objects based on the corresponding `Application Templates`; their composition and number depend on the design of the enterprise configuration. (For Enterprise Routing, the applications are: Universal Routing Server (see [page 194](#)), Interaction Routing Designer (see [page 210](#)), and Custom Server (see [page 289](#)).
4. Create `Switches`. These premise switches will be connected to corresponding premise T-Servers.

Note: When you have two `Switch` objects with the same name created respectively in `Environment` and in a `Tenant`, you will not be able to select virtual queues created for the `Environment`'s `Switch` while working with `Selection`, `Route Interaction`, and `Workbin` strategy objects in the `Tenant`. Both `Switches` with the same name will display in the corresponding drop down box, but each of them will display virtual queues created only for the `Tenant`'s `Switch`.

5. For voice routing, create `DNs` and `Agent Logins` for each premise switch. For each DN of type `ACD Queue`, `Virtual Queue`, or `Voice Treatment Port` that will be used in routing strategies, specify the `Alias` property. Also specify the `Alias` property for those DNs that you wish to use as routing targets. If you do not create an alias for an item, an alias is automatically generated when you start IRD. This alias will be the concatenation of the switch name and the DN number. IRD automatically generates aliases for `Virtual Queues`, `Routing Points`, `Virtual Routing Points`, `Voice Treatment Ports`, `Routing Queues`, `Network Destinations`, and `Service Numbers`.

Notes: When configuring the `Advanced` tab in the `New DN Properties` dialog box in `Configuration Manager`, there is a `Default Route Type` value available for selection. `Configuration Manager` allows you to insert a DN from another location here, but URS will use only the DN number and won't use the specified location. This is by design, because default routing is intended to be as simple as possible. Genesys recommends configuring a `Default DN`, which is on the same switch as the associated routing point.

As described in the *eServices (Multimedia) 8.0 Deployment Guide*, you will not need to create `Agent Logins` or `DNs` on the logical switch that exists for the purpose of loading non-voice routing strategies on virtual routing points. `Agent Logins` apply only to voice interaction capabilities associated with traditional telephony devices. Routing of e-mail, chat, and Open Media interactions does not use DNs and their assignment to agents. Instead, it uses `Agents (Persons)`, `Agent Groups`, `Places`, and `Place Groups`.

6. Create `Place` objects.
7. Add an extension or an extension and a position ID to each `Place`.
8. If you want to route based on the agent-capacity rules, or if you have a blended media-based environment, configure rules using the Resource Capacity Wizard (install from the Real-Time Metrics Engine CD). In this case, the `use_agent_capacity` option, as described in the *Universal Routing 8.0 Reference Manual*, must be set to `true`. If you do not configure any agent-capacity rules and `use_agent_capacity` is set to `true`, the default rules will affect routing decisions.

Notes: The Application Template for URS has the `use_agent_capacity` option set to `true` by default. If you do not want to route based on the agent-capacity model, change the option to `false`.

After installing the Resource Capacity Wizard, you can create and assign capacity rules in Configuration Manager. After selecting the Tenant or Environment, right-click the Scripts folder and select Wizard > New > Capacity Rule.

9. Create Access Groups. In order to use IRD, each `Person` object must have one or more Access Groups assigned. Any user without an assigned Access Group can open IRD, but can only view the IRD GUI.
10. Create a `Person` object to represent each agent. If you have not already done so during Framework configuration and plan to use skills-based routing, define `Skill` objects and assign the `Skill` objects to `Person` objects. When you configure eServices, the *eServices (Multimedia) 8.0 Deployment Guide* instructs to create at least two skills with the following literal names to handle the two types of media interactions you might handle: `email` and `chat`. You can do this now or wait until you configure eServices.
11. In the `Members` of tab of the `Person Properties` dialog box, assign an Access Group to each user that will be working in IRD.
12. Add an Agent Login and `Place` to each `Person`.
13. Assign agent capacity rules to each `Person` designated as an agent.
14. Add agents to the Agent Groups.
15. Review `Business Attributes` in Configuration Manager (see [page 37](#)) and/or Knowledge Manager if your site will use them for routing decisions and reporting. For general information on these attributes, see the section on `Business Attributes` in the *Universal Routing 7.6 (or later) Business Process User's Guide*. It provides important information about `Business Attributes` that are user-accessible as well as information on `Business Attributes` that are safe to modify.

Limitations

When developing strategies and other objects, familiarize yourself with the limitations listed below.

Warning! If you do not observe these limitations, the result could be routing to an incorrect destination, URS unexpectedly shutting down, or not performing a requested operation.

Configuration Manager Limitations

- While the Configuration Layer supports the full character set for use in object names, using certain characters can cause problems in the behavior of other Genesys applications.
The objects affected by this limitation that are used in Universal Routing include `Tenant`, `Field`, `Format`, `Script` (strategies and subroutines), `Table Access`, and `Transaction` (routing rules, interaction data, attributes, business rules, and custom statistics).
- The maximum length of an IRD object name is shorter than the maximum length of any Configuration Server object name, which is 254 bytes (characters). For example, the name of a strategy can incorporate only 194 characters.
- When configuring `Employee IDs`, `Login IDs`, `Place names`, `DN numbers`, and `Switch names` that will be used to define routing targets, the maximum length is 63 bytes.
- When a URS name is added to the name of a T-Server that is connecting through a Load Distribution Server, the sum of the bytes cannot exceed 126 bytes.
- When defining agent states on the `Annex` or `Options` tab of a URS `Application` object (for example when preparing to use the `UseAgentState` function), the maximum number of user-defined agent states is 32.
- The names of `Skills`, if they are used in a skill expression, cannot exceed 126 bytes.
- The `Priority` of an interaction cannot exceed 1,000,000,000.

IRD Limitations

When creating or modifying strategies in IRD:

- Certain symbols, listed in Figure 77 on [page 190](#), should not be used in object names. The names of all IRD reusable objects and data (strategies, routing rules, schedules, and so on) as well as `Interaction Design` window configuration objects (business processes, queues, and so on) are limited to

alphanumeric characters and cannot begin with a space, an underscore, or a hyphen. Please note that although some IRD objects allow you to enter names containing special characters, Genesys strongly recommends not using them.

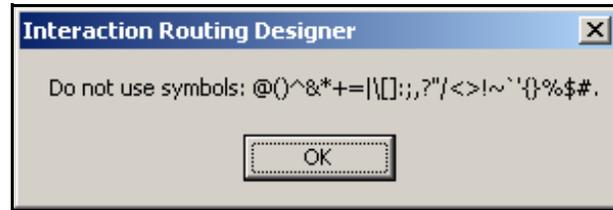


Figure 77: Invalid Symbols and Characters

- The underscore is the only special character that you can safely use in the middle of the name of a strategy or other reusable object, service, method, or method parameter name of any strategy object that results in executing an external or internal service or procedure.

Warning! Using commas and square brackets as data separators can make IRD incorrectly interpret the entered values if those symbols are used in Treatment, Data and Services, eServices, and Outbound objects.

- Skill expressions that are used by the Selection, Service Level, and Route Interaction objects cannot exceed 100 elements (skill names, numbers, comparisons, and logical operands). That is, a skill expression should have no more than 25 constructions, such as `English > 1`. It is especially important to observe this limitation in the case of Service Level routing rules, which use skills internally—that is, every skill criterion used in a Service Level routing rule generates 1–3 constructions in the form `<Skill> > <Number>`.
- The maximum size of an overall skill expression (as text) is 10239 bytes.
- The maximum length of a string using the format `placename@statservname.AP agentname@statservname.A` for any routing target is 253 bytes.
- The maximum length of a request to Custom Server or DB Server through XData (used for 5.x strategies) is 10239 bytes.
- When creating a key-value list that uses a string format (such as `aaa:5|gggg:123|...`), the maximum Key length is 1000 bytes and the maximum length of one key-value pair is 4096 bytes.
- When using the Database Wizard object to query a database, the maximum size of a SQL statement created by the DB Wizard is 1010 bytes.
- When using the IRD statistical adjustment functions, the name of an object plus the name of the statistic cannot exceed 250 bytes.

- The name of any single target (including a skill group) cannot exceed 256 bytes.
- Integer variables can have a maximum value of approximately four billion, with a range of -2 billion to +2 billion. If a value number exceeds the maximum capacity of an integer variable, the number wraps back around. If the number gets larger than largest allowed number, the integer variable wraps to a negative number. If the value number gets smaller than the smallest allowed number (around negative two billion), the integer variable wraps back into being positive. In any case, the returned number will be an error.
- If IRD's connection to Configuration Server is lost, it can appear as though the Interaction Design window is still operable. If the connection is lost, the Interaction Design window still allows you to continue working with a workflow, but you will not be able to save any result of your work.

Manually Configuring Network Routing

Network Routing configuration includes both an enterprise side and a network side. Components must be configured and installed for both sides to properly set up Network Routing.

Configuration Order for eServices

To facilitate configuration when running with Genesys eServices, it is important to configure components in the following order:

1. Framework components
2. Universal Routing components
3. eServices components

Note: The eServices Configuration Wizard assumes that Universal Routing configuration has been completed. To simplify the entire installation process, Genesys recommends configuring components in the order listed above. See “eServices Installation Considerations” on [page 144](#) for more information.

Enterprise Routing Objects of Network Routing

Network Routing requires that you create Enterprise Routing Configuration objects in Configuration Manager. The recommended configuration order for

setting up the enterprise-side Configuration objects of Network Routing is as follows:

1. Create `Switching Offices`. Use the types that correspond to the physical switches used.
2. Create `Hosts`. These must be only those computers on the data network where Genesys daemon processes run. Configure/install LCA on Hosts if you wish to use SCI. See Chapter 13 on [page 311](#) for more information.
3. Create `Application` objects based on the corresponding `Application Templates`; their composition and number depend on the design of the enterprise configuration (see [page 187](#)).
4. Create `Switches`. These premise Switches will be connected to corresponding premise T-Servers (see “Network Routing Component” on [page 56](#)).
5. For voice routing, create `DNs` and `Agent Logins` for each premise switch (see the Note on `Agent Logins` and `DNs` on [page 187](#)).

Use the `External Routing Point DN` type and other DN types according to your switch configuration.

If you need `External Routing Point DN`s (such as when an external `Route Type of Route` or `DNIS Pooling` is used), configure `External Routing Point DN`s in one-to-one correspondence with the configured `Network Destinations`.

Note: You specify the External routing type in the `Switch` object, `Access Codes` tab (see Figure 85 on [page 225](#)) by clicking `Add` to open a dialog box.

The `External Routing Points` corresponding to different `Network Destinations` may belong to different switches. Specify the `Network Ports DN Group` of the `Network Destination` corresponding to an `External Routing Point` in the `Group` field on the `Advanced` tab of the `External Routing Point DN Properties` dialog box.

If you plan to use an external `Route Type of direct-ANI`, configure the `DN`s that will receive interactions from the network in one-to-one correspondence with the configured `Network Destinations`. Specify the `Network Ports DN Group` of the `Network destination` corresponding to a `DN` that will receive interactions from the network in the `Group` field on the `Advanced` tab of the properties dialog box for that `DN`.

Note: External routing for T-Servers is now called `Inter Server Call Control (ISCC)`.

For each `DN` of type `ACD Queue` or `Virtual Queue` that will be used in routing strategies, specify the `Alias` property. Also specify the `Alias` property for those `Routing Points` and `Network Destinations` that you

wish to use as routing targets. If you do not create an alias for an item, an alias is automatically generated when you start IRD. This alias will be the concatenation of the switch name and the DN number.

6. Create `Places` including `Place` objects.
7. Add an extension or an extension and a position ID to each `Place`.
8. Configure agent-capacity rules using the Resource Capacity Wizard.
9. Create a `Person` for each agent. If you have not already done so during Framework configuration and plan to use skills-based routing, define `Skill` objects and assign the `Skill` objects to `Person` objects. When you configure eServices, the *eServices (Multimedia) 8.0 Deployment Guide* instructs to create at least two skills with the following literal names to handle the two types of media interactions you might handle: `email` and `chat`. You can do this now or wait until you configure eServices.
10. Add an `Agent Login` and `Place` to each `Person`.
11. Assign agent capacity rules to each `Person` designated as an agent.
12. Add agents to the `Agent Groups`.

Network Routing Objects Configuration Order

The recommended configuration order for setting up the network side Configuration objects is as follows:

1. Create a network switching office. Specify the switch office type. This switching office represents the network interface. Refer to the Framework documentation for instructions.
2. Create `Hosts`. These must only be those computers on the data network where Genesys daemon processes run.
3. Create `Applications` based on the corresponding `Application Templates`. The following `Applications` must be created:
 - Network T-Server
 - Universal Routing Server
 - Stat Server
 - Interaction Routing Designer
 - Message Server
 See “Manually Configuring URS” on [page 194](#).
4. Create a `Network Switch` that will be connected to the `Network T-Server`.
5. Create `DNs` under the `Network Switch` of the following types: `Service Number` and `Network Destination`.

Create a `Service Number` `DN` for every toll-free number from which interactions will be routed.

Create as many `Network Destination` `DNs` as specified by the network provider.

6. Create DN Groups of the type Network Ports, under the premise Tenant. Create one such DN Group for every Network Destination DN; the group must contain a shortcut to that DN only.

Note: For information on configuring a Network T-Server, refer to the documentation for your particular Network T-Server. This is located with other T-Server documents.

Manually Configuring URS

This section describes the following procedures:

- “[Creating/Configuring the URS Application object](#)”
- “[Viewing/changing the use_agent_capacity option](#)”
- “[Configuring URS to communicate with WFM](#)”

The URS Application object that you created in Configuration Manager must be configured as described below.

Procedure: Creating/Configuring the URS Application object

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Go to Application Template folder. Import a UR_Server_801 Application Template from the CD. If you need help with this procedure, consult the *Framework 8.0 Configuration Manager Help*.
3. Go to the Applications folder.

4. Create a new Application object for URS based on the imported Application Template as described in the *Framework 8.0 Configuration Manager Help*. [Figure 78](#) shows an example.

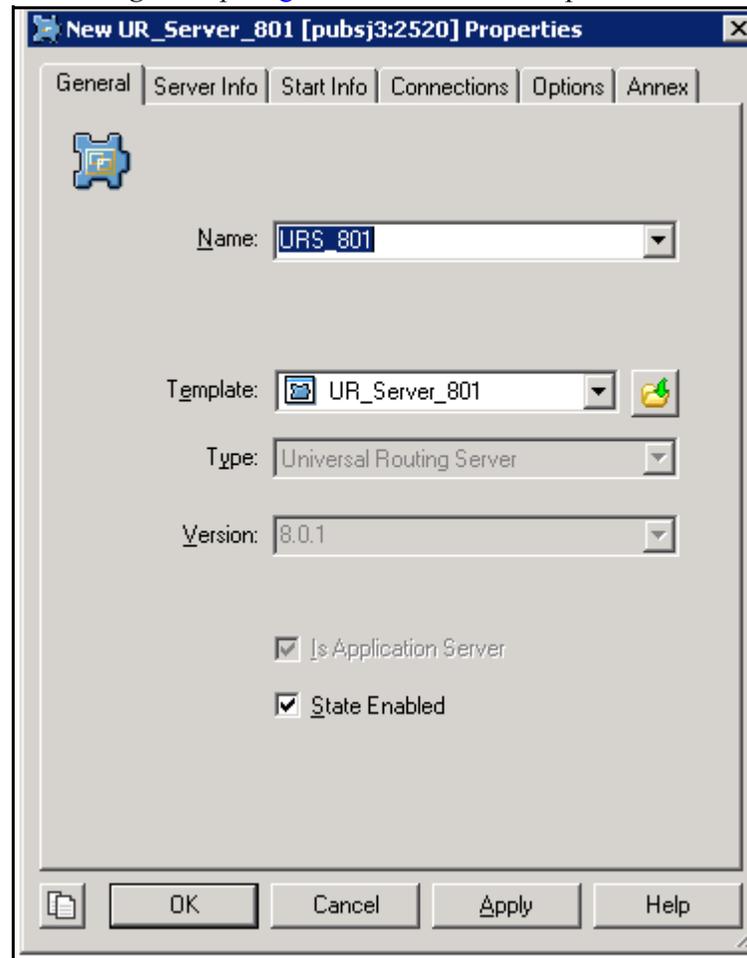


Figure 78: URS Application Object

5. On the General tab, enter the Application name.

Note: The Application name should not contain spaces.

6. Make sure that the State Enabled check box is selected.
7. In a multi-tenant environment, select the Tenants tab and set up the list of tenants that use URS.
8. Click the Server Info tab (see [Figure 25](#) on [page 80](#)) Select the following:
 - Host—the name of the host on which URS resides

- Port—the port through which communication with URS can be established. After you select a Host, a default port is provided for your convenience. You select the port and click **Edit Port** or you can configure a new port by clicking **Add Port**. Either action brings up the **New Port Info** dialog box (see Figure 79 on page 196).

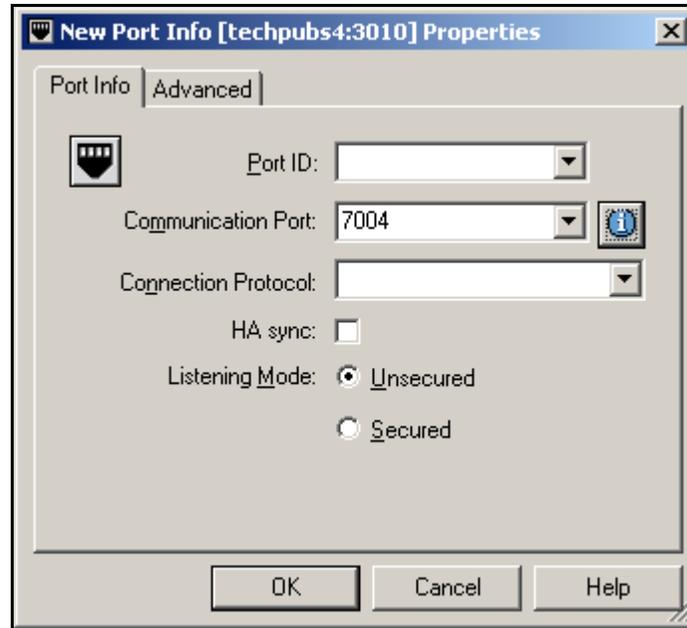


Figure 79: New Port Info Dialog Box

Note: For information on this dialog box, see the **Port Info Tab** topic in the *Framework 8.0 Configuration Manager Help*.

Continuing with the **Server Info** tab, select the following:

- **Backup Server**—If you plan to run URS in Warm or Hot Standby mode (the redundancy level of High Availability) or Warm Standby mode (the High Availability routing level), the primary URS must reference the backup URS.

Notes: Strategies loaded against the primary URS are not automatically loaded against the backup URS if the backup URS is configured after loading strategies on the primary URS. You must unload the loaded strategies in IRD and reload them once the backup URS is configured.

The base product offering includes a single primary server without a backup server. A primary and backup server in Warm Standby mode is available at the redundancy level of high availability. Hot Standby mode is available for High Availability Routing which requires the purchase of HA Proxy and Warm Standby (premise) T-Server. For more information, see Chapter 3 on [page 75](#).

- **Redundancy Type**—Select Warm Standby if you purchased the Redundancy level of high availability routing. Select Hot Standby if you purchased the High Availability Routing level. If you need more information, see “High Availability Routing” on [page 85](#).

Warm Standby refers to a redundant server that is running but not active. Hot standby refers to a redundant server that is running and active.

Note: To use the routing High Availability functionality, you need the High Availability licensing feature, `router_ha_option`, as described in *Genesys Licensing Guide*). High availability routing includes the ability to run with Hot Standby redundancy type as well as the ability to set the URS option `pickup_calls` to `true`. In case of the absence of a High Availability license, URS will downgrade the Hot Standby redundancy type to Warm Standby and force the `pickup_calls` option value to be `false`.

- **Reconnect Attempts**—Select number of attempts to reconnect to this URS server before trying to connect to the backup URS server. This value must be 1 or higher. This property makes sense only if you specify a backup URS for this server.
9. Select the **Start Info** tab and specify the following:
 - **Working Directory**—the Application location (example: `C:/GCTI/ur_server`)
 - **Command Line**—name of executable file (example: `ur_server.exe`)

Note: If there is a space in the URS Application name, then you must place quotation marks before and after the name of the URS Application.

- **Command Line Arguments**—list of arguments to start the Application (example: `-host <name of Configuration Server host> -port <name of Configuration Server port>-app <name of URS Application> -l <the full path and name of license file>`)

The license file path must also be enclosed in quotation marks.

Note: If you are using Configuration Server Proxy and do not use Management Layer, enter the name of Configuration Server proxy for host and the port of the Configuration Server Proxy. See Chapter 5 on [page 115](#) for information on distributed configuration environments and Configuration Server Proxy.

- **Startup time**—the time interval the server waits until restart if the server fails.
 - **Shutdown time**—the time interval the server takes to shut down.
 - **Auto-Restart setting**—selecting this option causes the server to restart automatically if the server fails.
 - **Primary setting**—selecting this option specifies the server as the primary routing server (unavailable).
10. Select the **Connections** tab and specify all the servers to which URS must connect:
- **Network T-Server** (Network Routing only).
 - **Premise T-Servers.** (For Network Routing 7.0, premise T-Servers are not necessary in the **Connections** tab of URS unless the ability to specify agent reservation with respect to a particular T-Server and URS is required. In Network Routing 7.0, Network T-Server (versions 6.1 and later) provides the Access Number to URS for the premise T-Server. However, including premise T-Servers in the **Connections** tab in addition to the Network T-Server for Network Routing will not cause any problems.)
 - **Stat Servers.**
 - **Database Access Points/DB Server.** Database Access Points configure access to databases through DB Server. For more information, refer to the *Framework 8.0 Configuration Manager Help*. Database Access Points are also used for customer database lookups. For more information on performing database lookups, refer to the *Universal Routing 8.0 Interaction Routing Designer Help*.
-

Note: To use Workforce Management (WFM) with URS, see [page 209](#).

- Custom Server, if using a non-SQL database for database lookups (supports Warm Standby mode only).
- Message Server(s) (optional)—see [page 256](#).
- Load Distribution Server (LDS) (optional) (Specify LDS if you previously configured and installed it. For more information on LDS and the levels of High Availability, see the *Framework 7.2 Load Distribution Server User's Guide*).
- Configuration Server Proxy (optional). Configuration Server Proxy is a component of a distributed configuration environment. See Chapter 3 on [page 75](#) for more information on a distributed configuration environment and Configuration Server Proxy.

Note: To support reconnecting to Configuration Server, you must still create or update the existing connection to Configuration Server in the URS Application object's Connections tab. Follow the standard procedure for configuring connections to other servers. For specific instructions associated with client-side port connections, see the *Genesys 8.0 Security Deployment Guide*.

11. Select the Options tab and specify URS options as described in the *Universal Routing 8.0 Reference Manual*. [Figure 80](#) shows the defaults.

abc agent_reservation	"false"
abc automatic_attach	"true"
abc call_tracking	"true"
abc compat_treatments	"true"
abc default_object	"agent"
abc extrouter_timeout	"0"
abc give_treatment	"false"
abc on_route_error	"delete"
abc on_router_activated	"default"
abc reg_delay	"2"
abc report_targets	"true"
abc request_timeout	"0"
abc route_consult_call	"false"
abc skip_targets	"empty"
abc targets_order	"random"
abc transfer_time	"15"
abc transfer_to_agent	"false"
abc transition_time	"15"
abc unloaded_cdn	"defaultignore"
abc use_agent_capacity	"true"
abc use_agentid	"false"
abc use_dn_type	"any"
abc use_extrouter	"false"
abc use_ivr_info	"true"
abc use_service_objective	"false"
abc use_translation	"false"
abc validate	"false"
abc verbose	"true"
abc verification_time	"0"

Figure 80: Default URS Option Settings

Note: When installing URS, new customers will probably use the URS Application Template, which includes some, but not all, of the options described in the *Universal Routing 8.0 Reference Manual*. Options not included in the template must be added manually.

12. When finished, click OK.
The URS Properties dialog box closes.

End of procedure

Note: After installing and configuring a solution, any changes made to the properties of an Application or a configuration object may delay URS processes for a few seconds.

Next Steps

- If you wish to use the Management Layer and its Solution Control Interface (SCI) to stop and start applications, you must install Local Control Agent (LCA). See Chapter 13 on [page 311](#) for more information.

Defining Client-Side Port Information

To increase security, you can define a fixed port for the connection between a URS/Custom server component and another server that is behind a firewall. The client-side port definition feature allows a server application to control the number of client connections, preventing the server from an excessive number of malicious requests to the same server-side port.

For configuration instructions, see the “Client-Side Port Definition” chapter of the *Genesys 8.0 Security Deployment Guide*. Table 1 on [page 52](#) identifies which Universal Routing-specific components support this type of configuration.

Note: When defining primary and backup URS `Application` objects, you can enter one set of client-side port data for the primary URS and a different set of client-side port data for the backup URS. However, if/when the backup server is assigned to be primary, any existing client-side port data previously defined for the backup will be replaced by connections defined for the primary. After that, you can open the properties dialog box for the backup URS, click the `Connections` tab, define the correct client-side port data for any valid `Application` defined in this tab, and save the object with updated parameters.

Setting the `use_agent_capacity` Option

Note: For a definition of routing based on agent capacity rules, see [page 41](#).

The URS `Application Template` sets the `use_agent_capacity` option to `true`. Genesys assumes that many new customers installing Universal Routing 8.0 are or will be routing both voice and non-voice interactions and will want to use agent-capacity rules (see “Routing Based on Agent Capacity” on [page 41](#)). Routing based on agent-capacity rules has benefits in voice scenarios where agents have multiple voice DN. Voice-only customers who do not define agent-capacity rules will not see behavioral differences from previous versions of Universal Routing. If you do not want to route based on the agent-capacity rules, set the `use_agent_capacity` option to `false`.

Procedure: Viewing/changing the use_agent_capacity option

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Select the Tenant or Environment.
3. Open the URS Application object.
4. Select the Options tab (see Figure 80 on [page 200](#)).
5. Select the default section.
6. Double-click use_agent_capacity.
7. In the resulting Edit Option dialog box, keep the default Option Value of true or change to false.
8. Click OK to save.

End of procedure

Notes

- If using Universal Routing in a pure voice environment and not routing based on agent-capacity rules, you are not required to set the use_agent_capacity option to true. You can set this option to true to take advantage of more efficient agent state messaging, but are not required to do so.
- If you are using SIP Server and routing chat/instant messaging interactions, you must set the use_agent_capacity option to true.
- URS is required to use agent capacity information supplied by Stat Server when routing non-voice interactions. Agent-capacity rules, set with the Resource Capacity Wizard, provide information about whether an agent is available for routing. When configured, URS takes agent-capacity rules into consideration when finding a complete set of available agents. Once this set is defined, URS applies the routing selection criteria specified in the routing strategy objects.
- If you wish to create capacity rules while creating a routing solution, first install the Resource Capacity Wizard, which is available on the Real-Time Metrics Engine CD. You can create and assign capacity rules to a newly created Tenant, Place, or Person.

Agent Capacity and Instant Messaging

A single agent can support several chat callers simultaneously (see [page 45](#)). You define the maximum number of simultaneous sessions for an agent by an agent capacity rule. Using an agent capacity rule, you can define the maximum

simultaneous instant messaging and maximum simultaneous voice interactions that can be handled by an agent; You can also specify if the agent can take a voice call when he is handling IM(s)

Note: For instant message routing, URS supports only the chat media type shown in Figure 13 on [page 37](#), selectable as the `TMediaChat` inside a routing strategy.

URS works according the following rules:

- When getting `CurrentTargetState` information from Stat Server, any DN not marked by Stat Server (URS does not check DN properties in the Configuration Database) as capable of handling multimedia is considered by URS as a voice DN. Such a DN must have a `WaitForNextCall` state in order to be considered by URS as ready.
- If a DN is a multimedia DN, then URS ignores the status of such a DN and instead use this DN's capacity vector. URS considers that a multimedia DN is ready for specific a media if the capacity vectors of both the agent and DN are opened for this media (= both have positive margin for this media).
- For compatibility, by default URS ignores multimedia DNs for interactions provided by Interaction Server and uses media channels for such interactions. For interactions provided by T-Server (meaning not Interaction Server) URS ignores media channels and considers multimedia DNs. Voice DNs are considered for all T-Servers (including Interaction Server).

Other Options That Affect URS

Configuration options affecting the operation of URS are not confined to those listed in the `URS Application` object (see Figure 80 on [page 200](#)). Some are located in the T-Server, DB Server, Data Access Point, Custom Server, and Interaction Server Applications, as well as in the Tenant, and DN objects. For information on these options, see the *Universal Routing 8.0 Reference Manual*. Also see the *Framework 8.0 DB Server User's Guide*, *Framework 8.0 Configuration Options Reference Manual*, *Framework 8.0 Configuration Manager Help File*, and the Universal Routing Server section in the *Combined Log Events Help File*.

Note: In order for URS to communicate with Interaction Server, the `service-timeout` option must be set. For information on options that affect URS/Interaction Server processing, including the `service-timeout` option, see the *Universal Routing 8.0 Reference Manual*.

Network Routing Options

Pay special attention to the following network routing options (Network Routing only):

`use_translation`—must be set to `true`.

`use_extrouter`—(1) Specify in the default section of URS application options, or (2) specify in the `Annex` tab or `Options` tab of the Network T-Server properties. For Network T-Server, under the `Annex` tab or `Options` tab, create a section with the name of the URS application. Under this Section, create the option `use_extrouter` with a value of `true`.

`extrouter_timeout`—(1) Specify in the default section of URS application options, or (2) specify in the `Annex` tab or `Options` tab of the premise T-Server properties. For Network T-Server, under the `Annex` tab or `Options` tab, create a section with the name of the URS application. Under this section, create the option `extrouter_timeout` with an appropriate value.

(Setting this option in the premise T-Server properties rather than the URS application properties, specifies the timeout that will be used between URS and that particular T-Server. If you set the timeout in URS application properties, the same value can be used for all T-Servers.) This option is explained in greater detail in the *Universal Routing 8.0 Reference Manual*.

Note: External routing for T-Servers is now called Inter Server Call Control (ISCC).

URS Web Service Options

You can interact with Web-based applications (Web Services) outside of Genesys applications.

- IRD provides a Web Service object for use in strategies. You specify request parameters, what type of data you expect to be returned, and what to do with the returned data. This object can be used in both voice and non-voice routing strategies.
- URS provides a tool, *HTTP Bridge*, for communicating with Web Services through SOAP/XML over HTTP/HTTPS protocols. You configure HTTP Bridge through various options in the URS `Application` object. There are HTTP Bridge options that you can use to control tracing and debugging when you use HTTP Bridge to access Web Services and to adjust HTTP Bridge performance characteristics (see next page).

For information on configuring the Web Service object and a sample strategy, see Appendix B of the *Universal Routing 8.0 Reference Manual*. The information below summarizes Web Services options.

Notes: The General, Log, and SOAP options are described in detail in the “Configuration Options” chapter of the *Universal Routing 8.0 Reference Manual*.

To use HTTP Bridge to access SAP RFC functions, you must also have installed the *Gplus* Adapter for mySAP Data Access Component.

General Options

Enter these options and values in the web section of the URS Application object. The General options are as follows:

Option	Description
http_port	Port on which HTTP Bridge is listening. Also see “Configuring URS Web Interface Functionality” on page 208 .
verbose	Level of log output. Valid values from 0 to 3. Level 0 produces no log messages. Levels from 1 to 3 produce log information with higher level of detail for higher log levels. Notes: Starting with 8.0: <ul style="list-style-type: none"> Logging functionality is simplified. All logging is controlled using verbose, http_log_file, http_log_size, log_buffering, log_remove_old_files and log_hide_private_data options. As a result, the following options were removed: xml_log_file, xml_log_size, parser_log_file, parser_log_size, wfm_xml_log_file, wfm_xml_log_size, wfm_parser_log_file, wfm_parser_log_size. The max_loading option is removed. HTTP Bridge can now process any number of simultaneous requests and does not require a thread to be allocated per every request. As result, option max_loading is removed.

Log Options

Enter these options and values in the web section of the URS Application object.

Note: If these options do not appear in the URS Application object, you must create them. See *Framework 8.0 Configuration Manager Help* for instructions.

Option	Description
http_log_file	Log file for HTTP Bridge error and trace messages. Starting with 8.0, all Web Service outputs from HTTP Bridge go into one log file as defined by the http_log_file option. What URS previously placed in three different logs (http_log_file, xml_log_file and parser_log_file) now goes into a single combined log file, http_log_file.
http_log_size	HTTP Bridge maximum log file size in bytes. Once the maximum size is reached, the oldest messages are deleted as new ones are written to the log. You must also configure the log_check_interval and http_log_file options to use this option.
log_buffering	Valid values are on/off, true/false, and 1/0. Set the value of this option to on, true, or 1 to turn on log buffering.
log_check_interval	Interval at which to check the HTTP Bridge log, in seconds. This option is used together with the http_log_size and http_log_file options.
log_remove_old_files	Valid values are on/off, true/false, and 1/0. Set the value of this option to on, true, or 1 to have old files deleted.

Note: When option `hide_private_data` is set to `true`: URS will not print in its logs parameters for Web Service requests and any result that a Web Service returns. Also HTTP Bridge will not print in its log parameters and results of Web Service requests and will not generate an xml-soap log.

SOAP Options

Enter values for these options in the user-created web section of the URS Application object.

Note: If these options do not appear in the URS Application object, you must create them. See *Framework 8.0 Configuration Manager Help* for instructions.

Option	Description
soap_conn_timeout	Timeout value for connect operation when working with Web Services, in seconds. If connection to a Web Service cannot be established within the time period specified, communication with the Web Service ends with failure. The value of 0 disables this timeout. Default: 5 seconds
soap_conn_idle_timeout	Time period HTTP Bridge will maintain persistent SOAP connections without activity, in seconds. When this period expires for a given connection, HTTP Bridge closes the connection. Default: 15 (seconds).
soap_port	This is the http port that clients need to use to access URS's service using SOAP protocol. Entered in section http when "Configuring URS Web Interface Functionality" on page 208 .
soap_recv_timeout	Timeout value for receive operation when working with Web Services, in seconds. If no data is received from a Web service within the time period specified, communication with the Web service ends in a failure. The value of 0 disables this timeout. The default value is 5.
soap_retry_attempts	Maximum number of attempts HTTP Bridge makes to communicate with a Web service. The default value is 2.
soap_retry_timeout	Time period HTTP Bridge waits between retry attempts, in seconds. The value of 0 (zero) forces HTTP Bridge to initiate the next communication attempt right away. The default value is 5.
soap_send_timeout	Timeout value for a send operation when working with Web Services, in seconds. If no data can be sent to a Web service within the time period specified, communication with the Web service ends in a failure. The value of 0 disables this timeout. The default value is 5.

URS Web Interface Options

Starting with 8.0, Universal Routing provides the URS Web Interface, which extends URS's routing functionality to an external Web Service/application server/third party applications.

To provide the URS Web Interface functionality, URS must implement Web Service functionality, which is facilitated with a URS executable component called *HTTP Interface* (in the same way as the HTTP Bridge component facilitates the web client functionality of URS). Web service functionality is configured inside the URS Application object in section http.

The URS Web Interface functionality provides a method for http requests or SOAP requests to be made directly to URS, in order to get statistics or visible targets from the target list.

Procedure: Configuring URS Web Interface Functionality

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. In Configuration Manager, right-click on the Universal Routing Server Application object and select Properties from the shortcut menu. The properties dialog box opens.
3. In the Properties dialog box, click the Options tab.
4. If you have not already done so, create the section, http.
5. Define the following options inside section http:
 - soap_port. This is the http port that clients need to use to access URS's service using SOAP protocol.
 - http_port. This is the http port that clients need to use in order to access URS's service through the HTTP GET/POST methods.

Warning! If both soap_port and http_port are not specified, URS does not start the HTTP Interface component

- verbose— Level of log output. Valid values from 0 to 3. Level 0 produces no log messages. Levels from 1 to 3 produce log information with corresponding higher levels of detail.
6. You have the option of setting log_file, log_size, log_buffering, and log_remove_old_files, which are similar to Genesys common log options debug, segment, buffering and expire, respectively (see [page 205](#)).

End of procedure

Note: Since HTTP interface is a TCP/IP client of URS, the URS Application itself must be configured as server (URS's port must be defined and valid).

Communicating with Workforce Management

You can enable URS to communicate with Genesys Workforce Management (WFM) components to route an interaction based on agent schedules and activities.

Procedure: Configuring URS to communicate with WFM

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Open the URS Application that will communicate with Workforce Management (WFM) 7.x.
3. Under the URS Options tab, create a new section called web.
4. Under this section create a new option called http_port (described in the *Universal Routing 8.0 Reference Manual*).
5. Under Value, specify a unique port on which to run HTTP Bridge used for asynchronous communication with WFM.
6. In the URS Connections tab, add the WFM Server (Type is CFGWFMServer).

End of procedure

Notes: When a routing strategy uses the Workforce Routing object with WFM, URS can misinterpret an agent schedule if the request for schedule information occurs during the agent's last working minute. It is likely that any significant usage of workforce routing will encounter this problem so please contact Technical Support for assistance.

To use Workforce Manager 6.x with Workforce routing rules, configure an additional Database Access Point. In the Annex tab for the Database Access Point, create a section with the name of the URS Application. In that section, create an option using with the value wfm.

Manually Configuring IRD

This section contains the following topics and/or procedures:

- [“Creating and configuring the IRD Application object”](#)
- [“Setting the inactivity-timeout option”](#)
- [“Displaying the Interaction Design shortcut bar”](#)
- [“Creating the ird_strategies table”](#)

Procedure: Creating and configuring the IRD Application object

Purpose: To create the Application object associated with the Interaction Routing Designer user interface used for strategy (see Figure 5 on page 25) and workflow (see Figure 12 on page 32) creation.

Start of procedure

1. Log into Configuration Manager as described on page 183.
2. Go to Application Template folder. Import an Application Template from the Universal Routing CD (Routing_Designer_801.apd) and create a new Application Template for IRD. For information about importing templates and creating Applications, refer to the *Framework 8.0 Configuration Manager Help*.
3. Go to Applications folder.
4. Create a new Application object for Interaction Routing Designer. The IRD Properties dialog box opens (see Figure 81).

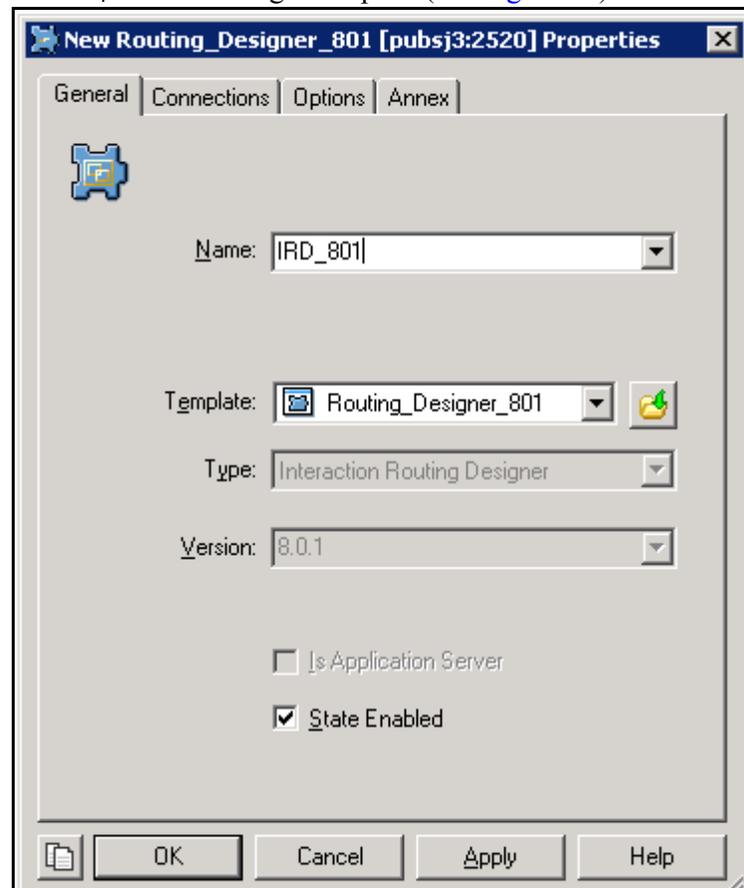


Figure 81: Example Interaction Routing Designer Application

5. Select the **General** tab and enter the **Application** object name.
6. Make sure that the **State Enabled** check box is selected.
7. If you require monitoring, add **Message Server** in the **Connections** tab.
8. If you wish to have the choice of saving strategies in the **Configuration Database**, add a **Database Access Point (DAP)** in the **Connections** tab.

Note: IRD can store the **.rbn** portion of a strategy in the **Configuration Server** database. In order to get this option in the save dialog, a **Database Access Point** must appear on the **Connections** tab of IRD.

Warning! Do not add more than one **DAP** to the **Connections** tab of IRD. If you add more than one, this can cause problems in the database when you save strategies. For information on configuring a **DAP**, see the *Framework 8.0 DB Server User's Guide*.

9. Click **OK**.
The **IRD Properties** dialog box closes.
10. Check whether your site uses the following **Configuration Manager** objects: **Agents**, **Places (desks)**, **Agent Groups**, **Place Groups**. If your site does not use these objects, you can disable them in IRD. If you do this, when IRD starts up, it does not need to read information about them from the **Configuration Database** and thereby improves startup performance.

End of procedure

Installing IRD on Desktops

After configuring the **IRD Application** object, you can install IRD on desktops and configure a security banner message (optional) as described in the *Genesys 8.0 Security Deployment Guide*.

The **Genesys Installation Wizard** will prevent you from installing more than one IRD per desktop. The **Maintenance** screen of **Genesys Installation Wizard** will require you to remove first the existing version of IRD in order to proceed with installation of new one.

Warning! Do not run more than one IRD application instance per desktop.

Note: The Event Log file is generated only for one instance of IRD. Running a second instance of IRD from the same location returns the following message: A sharing violation occurred while accessing <path to and name of log file>. Close this message by clicking OK button. IRD starts, but this event is not reflected in IRD's Event Log window.

Configuring the IRD Inactivity Timeout

In 7.6, IRD implemented an inactivity timeout, which is fully described in the *Genesys 8.0 Security Deployment Guide*. This feature requires re-authentication (logging in again) after a period of user inactivity. The default period of inactivity is 0 (zero) minutes. To change the default in the IRD Application object, configure a value for the `inactivity-timeout` option as described in the *Universal Routing 8.0 Reference Manual*.

Note: IRD interprets the absence of the `inactivity-timeout` option or the setting of its value to 0 (zero) as disabling the feature of application locking as a result of user inactivity.

Procedure: Setting the inactivity-timeout option

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Open the IRD Application object.
3. Select the Options tab (see Figure 80 on [page 200](#)).
4. Select the security section.
5. Double-click `inactivity-timeout`.
6. In the resulting Edit Option dialog box, enter a positive integer to specify a period of user inactivity in minutes. Users must log back into IRD after this period of time.
7. Click OK to save.

End of procedure

Important Information

- Upon expiration of the value set for the `inactivity-timeout` option, all opened IRD windows are hidden (minimized). You are presented with a re-login dialog box that, for authentication purposes, asks only for a user

password in order to resume the Application session. All other fields are grayed out since the connection of the IRD Application to Configuration Server remains active the entire time until re-login.

If you want to end the Application session by pressing **Cancel** button, on the re-login dialog box you will be warned that you are about to exit the Application. To exit the Application, press the **Yes** button on the **Do you want to close the application?** warning message. To return to the re-login dialog box, press the **No** button.

- If the administrator changes a user's password in Configuration Manager at any time during an IRD session before that user is presented with the re-login dialog box, the previous password will be required in order to resume the suspended session.
- In a case where the user changes a password in IRD before the expiration of the value set for the `inactivity-timeout` option, then only a new password will be accepted in the re-login dialog box in order to continue working with the Application. The re-login dialog box of IRD instances opened on all other machines that share settings of the same `inactivity-timeout` configuration option will still require the old password in such a scenario.
- The **Interaction Design** window behaves differently than the **Routing Design** window in the case where you want to end the Application session while you have unsaved changes. Unlike **Routing Design**, when you press the **Yes** button in order to close the Application, **Interaction Design** provides a message giving the possibility to save modifications made to a business process prior exiting.

IRD Access Permissions

Make sure that the **Person** object for each user that will access IRD is assigned the required **Access Group** for IRD. Beginning in 7.6, the **Environment** and **Tenant** objects have an option that defines the access permissions for a new **Person** object. The default is for a new **Person** object to have no access permissions unless you specify assign them in the **Security** tab of the **Person** properties dialog box.

Configuration Manager displays a warning message if you attempt to save a new **Person** object without assigning an **Access Group**. Any user without an assigned **Access Group** can open IRD, but can only view the IRD GUI. For more information on configuring Application level security, see the *Genesys 8.0 Security Deployment Guide*.

Interaction Design Shortcut Bar

When IRD starts up, it checks for an eServices solution installed by the eServices Configuration Wizard. If not found, the main IRD window or the window shown in Figure 11 on [page 31](#) does not display an **Interaction**

Design shortcut bar and you cannot open the Interaction Design window (shown in Figure 12 on [page 32](#)).

Use the following workaround if eServices components were manually configured in the Configuration Database (no eServices solution installed by the Configuration Wizard).

Procedure:

Displaying the Interaction Design shortcut bar

Purpose: To have access the Interaction Design window for creating workflows.

Start of procedure

1. Log into IRD as described on [page 268](#).
2. Click to Tools menu and select Routing Design Options.
3. Click the Views tab in the IRD Options dialog box.
4. Clear the Default check box.
5. Verify that the following are checked: Interaction Design and Business Processes.
6. Click OK to save your changes.

End of procedure

Samples

The IRD installation process places samples in the IRD installation directory. When you import this file into IRD, you get the sample strategies, subroutines, and other objects described in “Samples” on [page 267](#).

Creating Subfolders Under the Script Folder

You can save the script portion of a strategy, subroutine, or schedule as follows:

- Under the Scripts folder that represents the Script object in Configuration Manager or
- In any subfolder that you create under the Scripts folder. By default, the Scripts folder is displayed in IRD in the corresponding list pane.

You can create as many folders as you need of object type Script (apart from the Scripts folder in Configuration Manager) and store the Script portion of applicable IRD reusable objects.

Procedure: Creating a subfolder of object type Script

Purpose: To save strategy, subroutine, or schedule Script in a custom folder.

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Right-click the Tenant object and select New > Configuration Unit.
3. In the resulting Folder Properties dialog box, name the folder and click OK.
4. Right-click that folder and select New > Scripts Folder.
5. In the resulting Folder Properties dialog box, name the folder and click OK.

After you create a folder or subfolder, it is selectable in IRD in the Save and Save As dialog boxes.

To create a subfolder:

6. Right-click the Scripts folder (or an existing subfolder), and select New > Folder.
7. In the resulting Folder Properties window, name the folder and click OK.

End of procedure

Once you save a Script object in a subfolder, to change to a different folder, use the Save As function.

Note: Unlike the Routing Design window, the Interaction Design window does not support the functionality of saving business processes in a tree of customized folders/subfolders on its list pane (see [Figure 11 on page 31](#)). Business processes can be saved only in the predefined default Scripts folder.

Running the Configuration Database Update Script

Warning! The Database Update Script initializes the Configuration Database so that routing strategies can be stored there. Running the Configuration Database Update script twice on the same database could destroy the old content (if any) of these tables. For example, if you are upgrading to Universal Routing 8.0, running the script on the existing database will re-initialize the database and destroy the previous content.

You have the option of saving strategy *.rbn files to a network drive or to the Configuration Server database. If security is a consideration, you may wish to use the database storage method. For example, if you are a Service Provider, you may not want your subscribers to have access to your corporate servers. In this case, saving to the database is the preferred method of storage.

In order to have this storage option in the save dialogs, you must run a script which creates the table (ird_strategies) in the Configuration Server database, that will be used for storage.

The script is contained in a directory on the installation CD, configuration_database_upgrade_scripts. The exact subfolder where you find the script depends on your database type.

Note: The Configuration database update script for DB2 allows you to define a strategy table with a different format that allows storing more than 1 megabyte of raw data.

Procedure: Creating the ird_strategies table

Purpose: To run the Configuration Database Update script so the Save (for a new strategy), Save As, Import from File, or Create Copy command give the option of saving strategy *.rbn files to a database.

Follow the steps below to run the script that creates the ird_strategies table.

Start of procedure

1. On the installation CD, go to the configuration_database_upgrade_scripts directory.
2. Load and execute the initialization script that corresponds to your database type.
See [Table 9](#) for a list of database types and their corresponding initialization script names.

Table 9: Configuration Database Initialization Scripts

Database Type	Enterprise Script Name
DB2	SQL_SCRIPTS\ird_create_tables_db2.sql
Informix	SQL_SCRIPTS\ird_create_tables_ifx.sql
Microsoft SQL	SQL_SCRIPTS\ird_create_tables_sql.sql

Table 9: Configuration Database Initialization Scripts (Continued)

Database Type	Enterprise Script Name
Oracle	SQL_SCRIPTS\ird_create_tables_ora.sql
Sybase	SQL_SCRIPTS\ird_create_tables_syb.sql

3. Load and execute the script that loads the `ird_strategies` table into the Configuration Database, depending on your database type.

End of procedure**Next Steps**

- After the script runs, it creates the `ird_strategies` table. If you open the table, you will see that it contains two columns: `dbid` and `rbn`. IRD users now have the option of saving `*.rbn` files to the Configuration Server database.

Other Manual Configuration Operations

This section describes other manual configuration operations that you may wish to perform in Configuration Manager.

Creating Business Attributes

For background information, see “Support for Business Attributes” on [page 37](#). Some Business Attributes may already be defined:

- Any screening rule Business Attributes, defined in Knowledge Manager, will have carried over into Configuration Manager from the Universal Contact Server database.
- Any classification category Business Attributes, defined in Knowledge Manager, will have also carried over from the Universal Contact Server database. The names will be viewable in the Category Structure folder along with the name of the associated standard responses.

You may still need to use Configuration Manager to define the following Business Attributes:

- Service Type and Customer Segment used in the MultiAttach object or for a Cost-based Routing solution as described in the *Universal Routing 8.0 Routing Application Configuration Guide*.
- Disposition Code. Currently not used by any IRD object, but contained in Interactions table.

- **Language.** The primary motivation for creating is the necessity to use this business attribute in conjunction with some particular Tenant in Knowledge Manager. All objects in Knowledge Manager, including classification categories and screening rules, can only be created under specific combination of Language and Tenant. The following IRD objects use the Language business attribute: Screen, MultiScreen, and Screen Segmentation.
- **PIN.** Currently not used by any IRD object, but you may want to include in interactions.
- **Reason Code and Stop Processing Reason** when using the IRD Stop Interaction object.
- **E-mail Accounts** specifies an external e-mail address required when using the following IRD objects: Acknowledgement, Autoresponse, Chat Transcript, Forward E-Mail, Redirect E-Mail, Reply E-Mail From External Resource, and Send E-Mail.
- **Media Type.** When implementing the Genesys Open Media, another business attribute that you may want to create is a custom Media Type that will be added to the Genesys predefined set of Media Types. A service performed by some third-party server associated with a custom media type can be used in the External Service object.

Scheduling Permissions

A *schedule* instructs URS when to automatically load/release a voice routing strategy, including the switch and DNs to use. In order to create, view, and modify strategy schedules (as described in the *Universal Routing 8.0 Interaction Routing Designer Help*), you must first have the appropriate permissions in Configuration Manager. Set this up by following the steps below.

Procedure: Creating permissions to modify strategy scheduling

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Expand the **Switches** folder.
3. Expand the folder for the **Switch** object associated with the DN.
4. Expand the **DNs** folder.
5. Open the properties dialog box for the DNs associated with the schedule.
6. Click the **Security** tab.
7. Click the **Permission** button.

8. In the resulting `Object Permissions` dialog box, assign `FULL CONTROL` to `SYSTEM`.

End of procedure

Note: The `System` account includes the `URS Application` working with scheduling.

Voice Callback Server

Voice Callback, and its associated `Universal Callback Server` component, is a Genesys option that gives customers a choice when wait times are high to:

- Request a callback ASAP.
- Request a callback at a specific time.
- Continue to hold.

Voice Callback Server is used to:

- Process callback orders
- Keep list of ordered callbacks
- Create a virtual call
- Request URS for callback destination and get response.
- Deliver callback to agent
- Process callback failure treatments

For information on installing Genesys Voice Callback, see the *Genesys 7.1 Voice Callback Deployment Guide*. For information on designing routing strategies for voice callback, see Chapter 5: Design Considerations, IVR and Routing.

Enterprise Routing Solution Object

An Enterprise Routing solution can be started using `Solution Control Interface (SCI)`, a Management Layer application (see Figure 125 on [page 334](#)). `SCI` uses the `Enterprise Routing Solution` object to start and stop the server components for Enterprise Routing.

Procedure:

Configuring an Enterprise Routing Solution object

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Expand the `Environment` folder.

3. Select the Solutions folder.
4. Right-click in the Solutions folder and select Import Solution from the menu.
5. Import the Enterprise Routing solution template from the Universal Routing CD.
6. In the General tab, enter the Name field, select a Tenant and a Solution Control Server (see Figure 82).

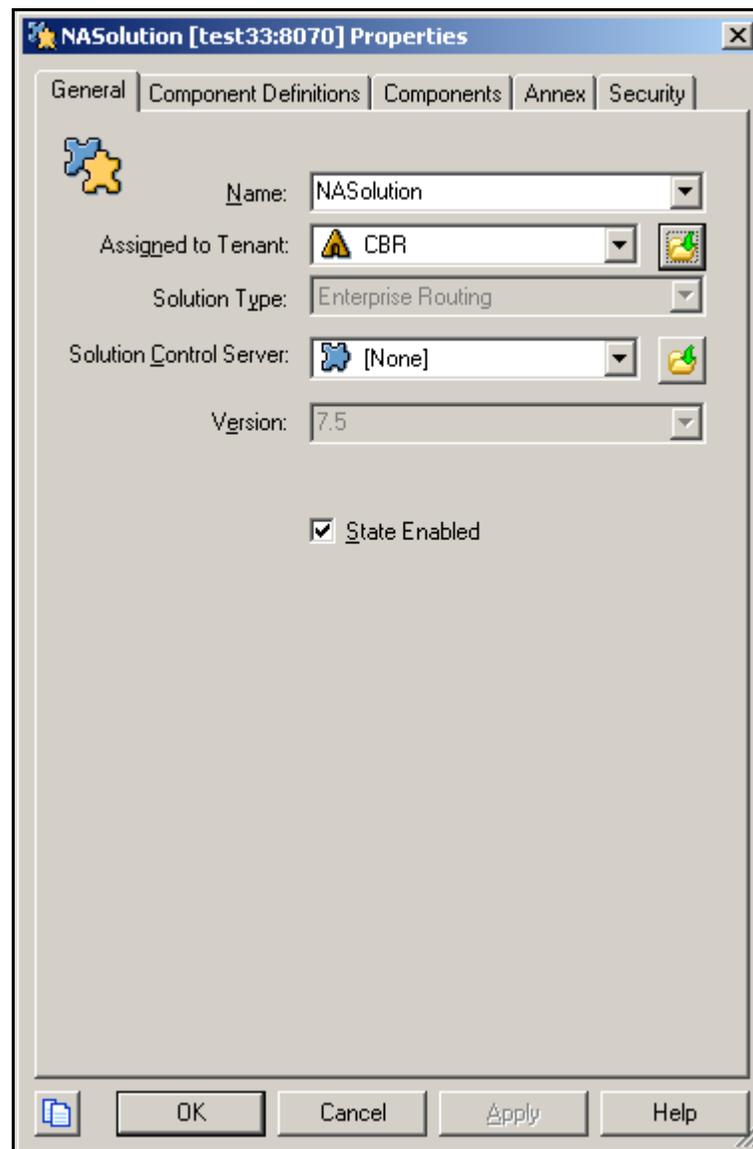


Figure 82: Example Enterprise Routing Solution Object

7. Select the Component Definitions tab to view all components that can be used in the solution. Figure 83 shows an example.

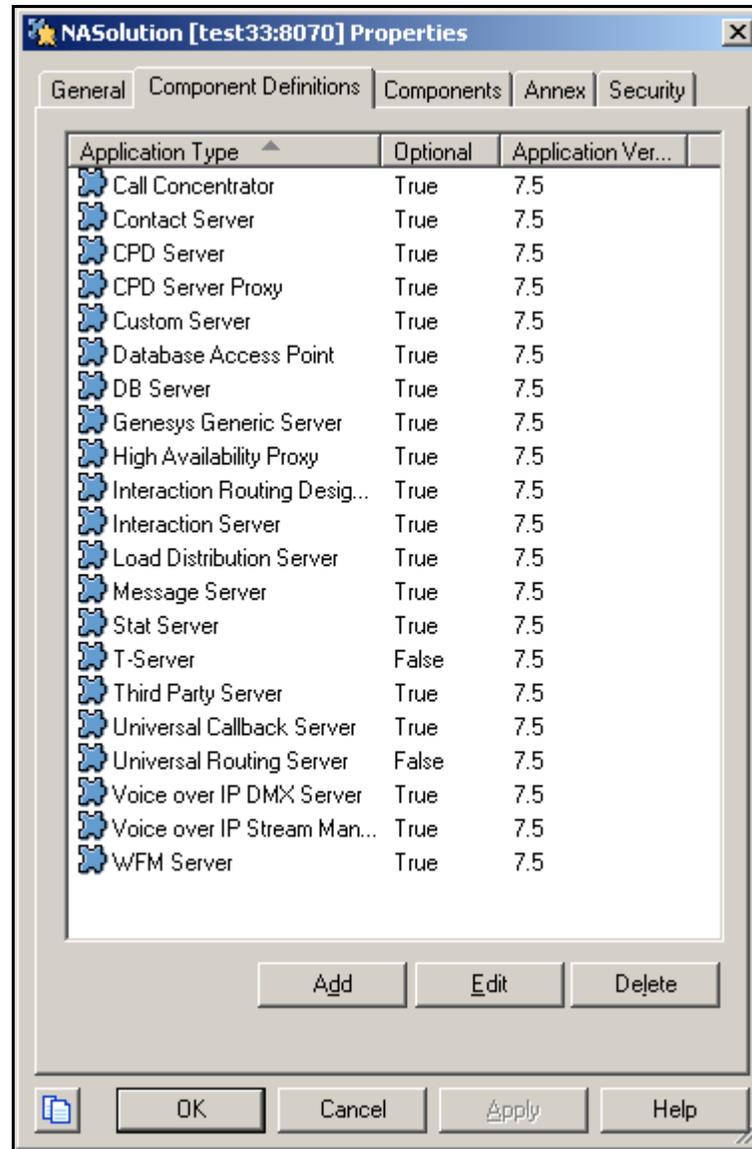


Figure 83: Example Components Tab for Enterprise Routing Solution Object

8. Select the Components tab.
9. Click Add. The New Solution Component dialog box opens (see [Figure 84](#)).

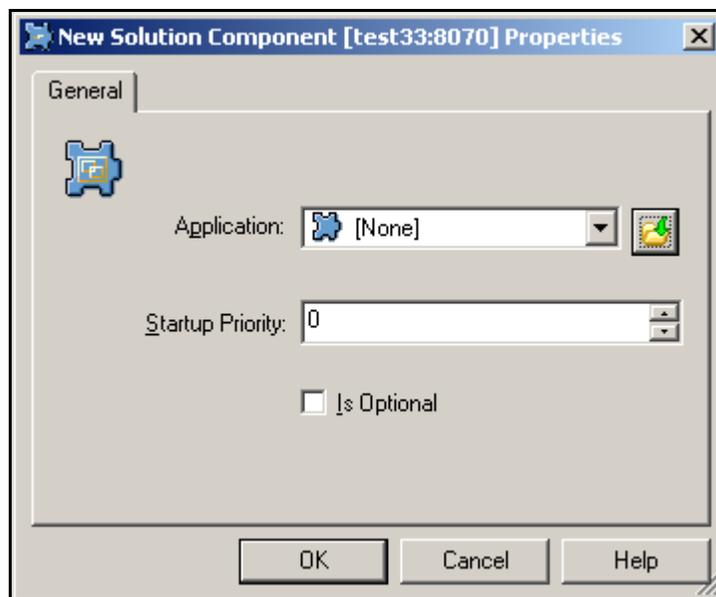


Figure 84: New Solution Component Dialog Box

10. Use the New Solution Component dialog box to add the servers that Enterprise Routing needs to run (see [Table 10](#)).
 - a. Application is the name of the Application object that Enterprise Routing needs to run.
 - b. Is Optional is the requirement status. False means that the application must run successfully for the Enterprise Routing solution to run. True means that the application is optional for the Enterprise Routing solution to run.
 - c. Startup Priority is the order in which to start the Application.

Table 10: Enterprise Routing Solution Object's Components Tab

Application	Optional	Startup Priority
Message Server(s)	True	1
T-Server	False	2
DB Server	True	3
Stat Server	True	4
Universal Routing Server	False	5
Custom Server (optional)	True	6

Note: You cannot start the Enterprise Routing Solution until you have installed and run LCA on each host where the components are located.

11. Click OK.

End of procedure

Solution States

The Enterprise Routing Solution object has three possible states that can appear in SCI under *Status* (see Figure 125 on [page 334](#)):

- Started (all mandatory applications are running)
- Stopped (one or more mandatory applications have stopped)
- Unknown (Management Layer cannot provide information about the solution status, but this does not mean that the solution components are unable to perform their functions).

An Enterprise Routing Solution object can only be controlled by one Solution Control Server (SCS).

Note: Most applications, unlike the solution object, have various states that can appear in SCI. For more information, see the *Framework 8.0 Management Layer User's Guide*.

Network Routing Solution object

A Network Routing solution can be started using Solution Control Interface (SCI), a Management Layer application (see Figure 125 on [page 334](#)). SCI uses the Network Routing solution object to start and stop the server components for Network Routing.

Procedure: Configuring a Network Routing Solution object

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Expand the `Environment` folder.
3. Select the `Solutions` folder.
4. Right-click in the `Solutions` folder and select `Import Solution` from the menu.

5. Import the Network Routing solution template from the Universal Routing CD. If the Network Routing Solution object is not in the list, import the Network Routing solution template from the Network Media Interfaces CD:
6. In the **General** tab, enter the **Name** field, select a **Tenant** and a **Solution Control Server**.
7. Select the **Component Definitions** tab to view all components that can be used in the solution.

Note: You cannot start the Network Routing Solution until you have installed and run LCA on each host where the components are located.

8. Select the **Components** tab.
9. Click **Add**. The **New Solution Component** dialog box opens (see Figure 84 on page 222).
10. Use the **New Solution Component** dialog box to add the servers that Network Routing needs to run (see Table 11).
 - a. **Application** is the name of the **Application** object that Network Routing needs to run.
 - b. **Is Optional** is the requirement status.
 - c. **False** means that the application must run successfully for the Network Routing solution to run. **True** means that the application is optional for the Network Routing solution to run.
 - d. **Startup Priority** is the order in which to start the application.

Table 11: Network Routing Solution Object's Components Tab

Application	Optional	Startup Priority
Network T-Server	False	1
Premise T-Servers	False	2
Universal Routing Server	False	3
DB Server	True	4
Stat Server	True	5
Message Server(s)	True	6
Custom Server (optional)	True	7

11. Click **OK**.

End of procedure

Switches for Network Routing

For a Network Routing solution, select one of the Network T-Servers and associated Switching Office objects listed in Table 7 on [page 178](#).

Procedure: Configuring a Network switch

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Select a Tenant or Environment.
3. Right-click the Switches folder and select New > Switch. The New Switch dialog box opens (see [Figure 85](#)).

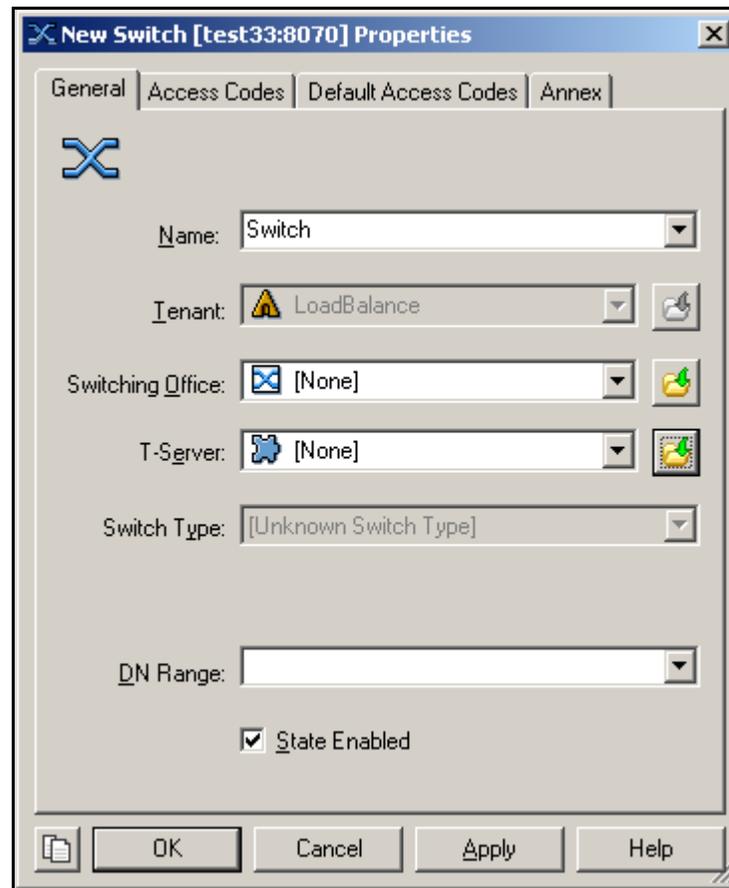


Figure 85: New Switch Properties Dialog Box

4. On the General tab of the Switch Properties dialog box:
 - a. Name the Switch object.
 - b. Specify the Network Switching Office that this Network Switch belongs to (see Table 7 on page 178).
5. On the Access Codes tab, click Add. The New Switch Access Code dialog box opens (see Figure 86).

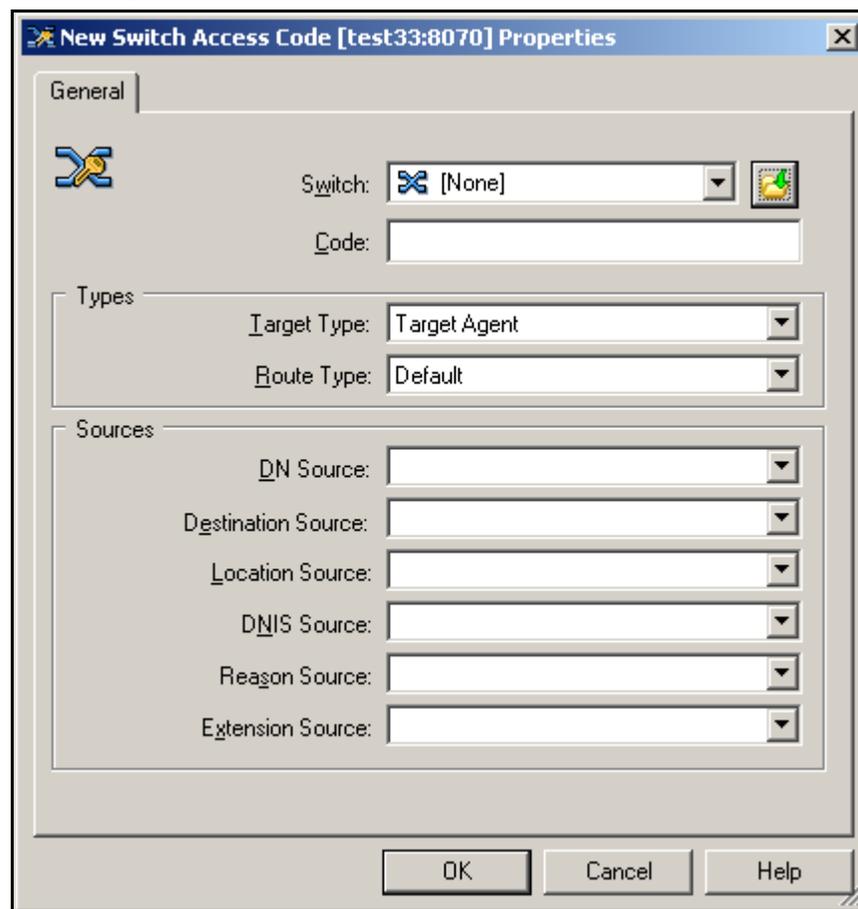


Figure 86: New Switch Access Code Dialog Box

6. In the New Switch Access Code dialog box, specify all access codes for each premise Switch that the Network Switch transfers interactions to. An access code must be configured for every premise T-Server and Target Type pair that is used in a strategy.
 - a. For the Target Type, select the type according to the strategy.
 - b. For the Route Type, select Label.
 - c. For the Destination Source, enter [DN.DL].
7. If external routing type Direct ANI will be used in routing to a premise Switch, create an access code to that switch with a target type of Target External Routing Point and a route type of Label. All other fields are ignored.

8. If external routing type `DNIS Pooling` will be used in routing to a premise Switch, create an access code to that switch with a target type of `Target External Routing Point` and a route type of `Overwrite DNIS`. All other fields are ignored.

Note: External routing for T-Servers is now called Inter Server Call Control (ISCC).

9. Add each DN of the type `Network Destination` to a `DN Group` of the type `Network Ports` under the premise `Tenant`. Each group may include several `Network Destination` DNs.

End of procedure

Manually Configuring Stat Server

Stat Server tracks the real-time status of the contact center (see [page 60](#)). The information below summarizes the process of configuring Stat Server. For more detailed information, see the *Framework 8.0 Stat Server User's Guide*.

Procedure: Configuring a Stat Server Application

Start of procedure

1. Open Configuration Manager (see Figure 13 on [page 37](#)).
2. Select a `Tenant` or `Environment`.
3. Open the `Stat Server Application` object.
4. On the `Server Info` tab of the properties dialog box, enter the host where your Stat Server application will run and the communication port that clients will connect to. To use another Stat Server as a backup in case the primary one fails, enter the `Backup Server` and adjust the `Reconnect Timeout`.
5. On the `Tenants` tab, add all `Tenant` objects that this Stat Server will work with.
6. On the `Connections` tab, add all premise T-Server Application objects that this Stat Server will work with.
7. On the `Options` tab, change the values of the configuration options to suit your Stat Server configuration.
8. For a description of the options, refer to the *Framework 8.0 Configuration Options Reference Manual*.

End of procedure

Premise T-Server for Network Routing

See “Network Routing Component” on [page 56](#).

Procedure: Configuring a premise T-Server Application for Network Routing

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Select Environment.
3. Open the T-Server Application object.
4. Click the Server Info tab of the properties dialog box (see [Figure 87](#)).

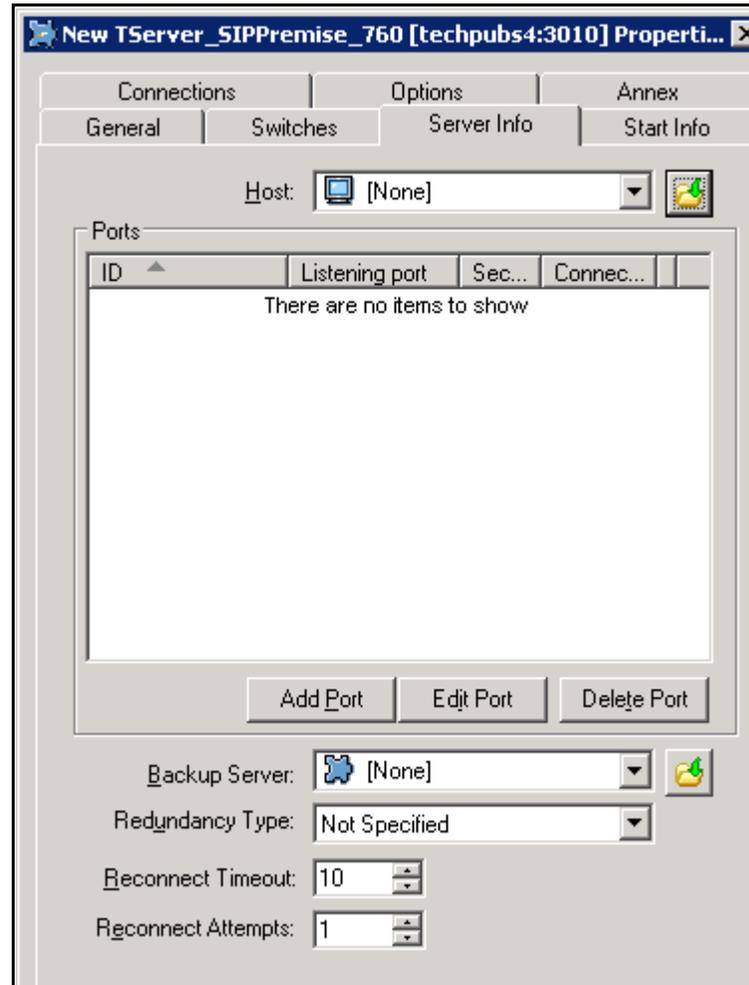


Figure 87: T-Server Application Object, Server Info Tab

5. Click the browse button and select the Host where the premise T-Server application will run and the communication port that clients will connect to.
To use another premise T-Server as a backup in case the primary one fails, enter the Backup Server and adjust the Reconnect Timeout.
6. On the Connections tab, add all T-Server Applications that this T-Server will communicate with in order to perform external routing operations.
7. On the Options tab, change the values of the configuration options to suit your T-Server and switch configuration. For a description of the several options, see the *Framework 8.0 Configuration Options Reference Manual*.
8. Repeat Steps 1 through 3 for each premise T-Server to be used.

End of procedure

Configuring Message Servers for Different Functions

Note: The `monitoring_time` option enables you to specify the time interval that URS uses for sending monitoring data to IRD, which can reduce the amount of data passed. In large environments, or with slow network connections or slow IRD host performance, this can prevent the situation of too many messages for Message Server to read.

By default, Message Servers are configured to handle both logging operations for servers and real-time monitoring of IRD clients. If there is a high demand for logging operations at the same time that many IRD clients are conducting real-time monitoring, the performance can degrade for not only Message Server but also URS and IRD. To prevent this degradation in performance, configure Message Servers to handle only one function, logging or IRD monitoring. Dedicate one Message Server to logging. Dedicate several others to IRD monitoring.

The ability to configure multiple Message Servers according to their function allows you to design your contact center architecture to better manage messaging and monitoring conditions.

Message Server for IRD monitoring should be added to the IRD Connections tab.

Note: When creating Message Servers for specific functions, be sure to name the Message Server in a way that helps identify its purpose. For example, a Message Server used for logging might be named `MessageLogging`. Message Server should be in connections for URS and IRD.

Procedure: Configuring a Message Server for logging

The steps below summarize the process of configuring Message Server for logging. For more detailed instructions on other options, see the *Framework 8.0 Configuration Options Reference Manual*.

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. In Configuration Manager, right-click on the Message Server Application object.
3. Select Properties from the menu. The properties dialog box opens for the Message Server. [Figure 88](#) shows an example dialog box.

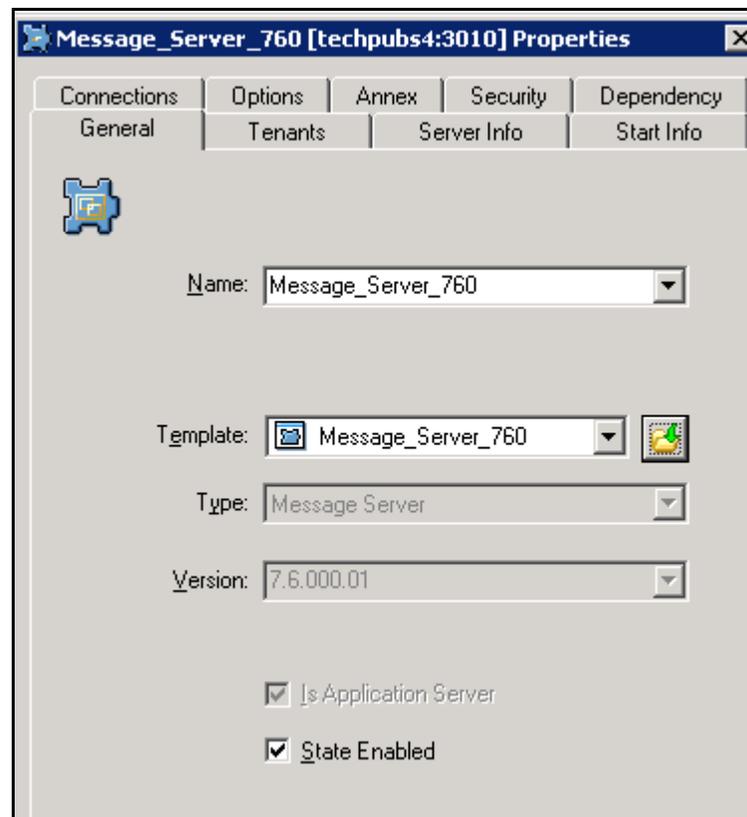


Figure 88: Message Server Application Object

4. In the Annex tab of the Message Server properties dialog box, create the section with the name MessageServer .
 - If the Annex tab is not displayed, choose View > Options, select Show Annex tab in object properties in the dialog box, and click OK.
5. In this section, create the option signature and set the value to log.
 - The names of the option and value are case-sensitive.

6. Click OK to save the changes and close the dialog box.

End of procedure

Procedure:

Configuring a Message Server for monitoring only

For general information on configuring a Message Server, see the *Framework 8.0 Deployment Guide*. The information below describes configuring a Message Server for IRD monitoring.

Note: The information passed between URS and IRD through Message Server for monitoring interactions, server status, and routing points does not appear in the Message Server log. That information is internal to Message Server and was not designed for readability.

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. In Configuration Manager, right-click on the Message Server Application object.
3. Select Properties from the menu.
4. In the Annex tab of the Message Server properties dialog box, create the section with the name MessageServer .
5. In this section, create the option signature and set the value to general. The names of the option and value are case sensitive.
6. Click OK to save the changes and close the dialog box.

End of procedure

Note: Also see “Message Server for Multiple URSs” on [page 264](#) and “Router Self-Awareness” on [page 255](#).

How URS Selects a Message Server

Universal Routing Server can only use one Message Server for logging. Which Message Server is used for logging is determined according to the following:

- URS ignores any Message Server with option using set to lds .
- URS then examines whether any of Message Servers listed in the URS Connections tab are configured for logging (if the value for the signature option is log). The *first* Message Server URS finds in the list with this setting is selected for logging.

Any other Message Server in this list with the `signature` option value set to `log` will not be used by URS.

- URS examines all Message Servers for working with IRD and considers which Message Server is appropriate for working under IRD that has the `signature` option with a value of `general` or a Message Server without a `signature` option at all.

The first Message Server URS finds that is not configured for IRD monitoring or that does not have the `signature` option is selected for logging.

Configuring DNs Manually

When configuring URS, you may need to change the configuration of individual DNs for two reasons:

1. To specify the DNs controlled by URS and URS's behavior in routing interactions coming to such DNs.
2. To create Aliases for DNs of type `Queue`, `Virtual Queue`, `Routing Point`, or `Network Destination` that will serve as names for such DNs in IRD.

DNs Controlled by URS

IRD can load strategies only for the DNs controlled by URS by default and route interactions from the DN according to the loaded strategy. URS can control any DN that can generate one of the following T-Library events:

- `EventRouteRequest`
- `EventRinging`
- `EventEstablished`
- `EventTreatmentRequired`

By default, URS controls all DNs of the following types:

- `Routing Point`
- `Virtual Routing Point`
- `Service Number`
- `Voice Treatment Port`
- `Routing Queue`

URS starts processing an interaction upon receiving `EventRouteRequest` from a `Routing Point`, a `Virtual Routing Point`, or a `Service Number` and `Routing Queue`, and upon receiving `EventEstablished` from a `Voice Treatment Port`.

If URS needs to control a DN of another type, not to control a DN of one of the types enumerated above, or to start processing an interaction upon receiving `EventTreatmentRequired` from a `Voice Treatment Port`, the URS option `event_arrive` must be used for the corresponding DN. This option must be placed in the `Annex` tab, in a folder bearing the name of the URS Application.

- If the option `event_arrive` is specified with a value of `none`, URS does not control the corresponding DN.
- If the option's value is `routerequest`, `ringing`, `established`, or `treatmentrequired`, URS controls the DN and starts processing an interaction upon receiving the corresponding event from the DN.

If the event specified in the option cannot be generated for the particular DN, URS does not control the DN. The only exceptions to this rule are DNs of type `Routing Point`, `Virtual Routing Point`, or `Service Number`, for which the values `EventEstablished`, `EventRinging`, and `EventTreatmentRequired` are ignored, and `EventRouteRequest` is expected for URS to start processing the interaction.

For more information about the `event_arrive` option, see Chapter 5, “Universal Routing Server Options” in the *Universal Routing 8.0 Reference Manual*.

Aliases of Queues, Virtual Queues, Routing Points, and Network Destinations

The name that IRD reads from Configuration Layer for a Queue, Virtual Queue, and Routing Point is configured in Configuration Layer as the Alias of the corresponding DN. Therefore, an Alias for the following DNs must be created on the `Advanced` tab of the properties dialog box:

- Every `Virtual Queue`, `Queue`, or `Routing Point` on which statistics will be collected in strategies.
- Every `Queue`, `Routing Point`, or `Network Destination` that will appear as a target in strategies.

`Routing Points` are read by IRD as actual DNs.

Note: If you do not create an Alias, one is automatically generated when you start IRD. This Alias is a concatenation of the switch name and the DN number.

Service Number to DN Alignment

If you want to have multiple service numbers distributed to the same non-configured (non-monitored) DN, you must align the SN to DN table, the network switch, and URS so that all calls that are sent to a non-configured DN are processed by the same URS.



Chapter

9

Load Balancing

This chapter discusses three types of load balancing:

1. Load balancing among multiple URSs with or without Load Distribution Server
2. Load balancing between routing targets (queues) using statistics.
3. IVR Server load balancing.

This chapter contains the following topics:

- [Load Balancing Among Multiple URSs, page 235](#)
- [Load Balancing Between Targets, page 242](#)
- [Other Ways to Load Balance, page 260](#)
- [IVR Server Load Balancing, page 262](#)

Load Balancing Among Multiple URSs

This section builds on the information contained in Chapter 2, “URS as a Client to LDS,” on [page 71](#). It describes how to configure load balancing to distribute requests among multiple URSs and combine their processing power to increase total throughput.

Task Summary: Configuring Load Balancing

Objective	Related Procedures and Actions
Configure load balancing with Load Distribution Server	Procedure: Configuring Load Balancing With LDS, on page 237
Configure load balancing without Load Distribution Server	Procedure: Configuring load balancing without LDS, on page 239

Task Summary: Configuring Load Balancing (Continued)

Objective	Related Procedures and Actions
Synchronize strategy loading	Procedure: Synchronizing strategy loading , on page 240
Configure Universal Routing Server for router self-awareness	Procedure: Configuring URS for Router Self-Awareness , on page 256

Using Load Distribution Server

As described in the *Framework 7.2 Load Distribution Server User's Guide*, using Load Distribution Server (LDS) is one way to distribute requests among multiple URSs and combine their processing power to increase total throughput. LDS includes the following features that affect routing:

- A Weighted Round Robin (WRR) distribution mode that enables you to vary load proportions per receiver or group of receivers. You can use this mode to isolate a URS instance from traffic to enable the introduction of new URS versions without stopping the system.
- The `loading-coefficient` option, which defines the relative loading coefficient for each specific Receiver for Weighted Round Robin (WRR) mode.
- The `group-id` option, which specifies whether this Receiver is part of a group. When you configure two or more Receivers with the same value for this option, LDS treats them as a single group and they receive the same event set.

Enterprise Routing Versus Network Routing Load Balancing

Although no redundancy (see [page 79](#)) is available for the base product offering, you can achieve a form of load balancing and redundancy by manually configuring a load balancing solution using multiple primary URSs. This form of load balancing is used in two scenarios:

- In a Network Routing solution where two or more instances of URS are running in parallel and the `event_arrive` and `unloaded_cdn` options are not used. The intelligent network automatically balances routing requests between two or more separate gateways with each URS processing a percentage of the traffic based on the number of URSs.

For example, if using two URSs, each URS will process one-half of the total traffic. However, each URS must be sized to handle full traffic when there is a system switch over due to failure on a gateway or any of the data links between SCP, the Routing Gateway, and the Network Routing Platform.

- In an Enterprise Routing solution where two or more primary URSs are used to achieve load balancing without LDS. Each primary URS registers only a subset of routing points from T-Server(s) using the option `event_arrive`. When the value of the `event_arrive` option is set to `none` at specific routing points (on the `Annex` tab in a section bearing the name of the URS) the corresponding URS does not register to this particular routing point and thus does not receive any routing requests on that routing point. For more information, see “Load Balancing Without LDS” on [page 238](#).

Load Balancing Between URSs Using LDS

You can configure load balancing between URSs with or without Load Distribution Server. The instructions below assume you have already imported the 8.0 Application Templates for URS and LDS, as described in the *Framework 8.0 Configuration Manager Help*.

Note: If LDS is used, you must enter the name of the Load Distribution Server in the `Connections` tab of all participating in load balancing URSs.

Procedure: Configuring Load Balancing With LDS

Start of procedure

1. Log into Configuration Manager as described on [page 183](#).
2. Expand the `Environment` folder.
3. Configure an LDS `Application` object as described in the *Framework 7.2 Load Distribution Server User's Guide*.
4. Configure the URS `Application` objects as described on [page 194](#) of this guide.
5. In the `Connections` tab of each URS `Application` object, enter the LDS `Application` object name. Figure 89 on [page 238](#) shows an example.

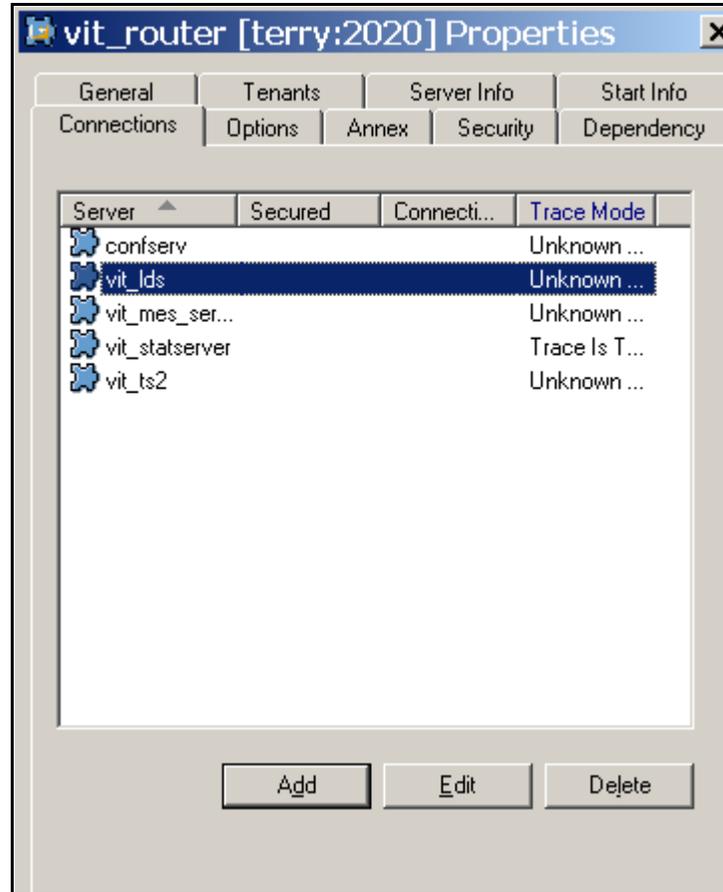


Figure 89: Example URS Application Object, LDS in Connections Tab

- Click the Apply button (not shown above).

End of procedure

Load Balancing Without LDS

To manually configure load balancing without LDS in an Enterprise Routing solution, configure routing points with the `event_arrive` option. The goal is to have only one URS in the group of load-balancing URSs register on a particular routing point. Backup URSs should be registered on the same routing points as the primary URSs.

In other words, for every routing point, you must decide which URS will register on it. After that, for all other URSs, set the `event_arrive` option for those routing points to none. Other than none, there are no special option values

for load balancing using `event_arrive`. You can use any of the values described in the *Universal Routing 8.0 Reference Manual*.

Note: This type of load balancing does not necessarily provide equal loading on participating URSs. The actual loading on each separate URS depends on rate of calls arriving on those routing points for which each URS is registered.

Procedure: Configuring load balancing without LDS

Start of procedure

1. Open Configuration Manager as described on the previous page.
2. Complete the login dialog and click **OK** to open Configuration Manager.
3. Expand the `Environment` folder.
4. Expand the `Switch` object associated with the routing points that you wish to configure.
5. Open `DNs` folder.
6. Complete the following steps for the subset of routing points that URS should register.
7. Open the properties dialog box for a DN that you want URS to monitor.
8. If it does not already exist, create a section of the same name as the URS `Application` object.
9. Create a new option in this section called `event_arrive`.
10. Set the value to one of the valid values described in the *Universal Routing 8.0 Reference Manual* (other than `none`).
For the other URSs in the load-balancing group, complete the following steps.
11. Open the properties dialog box for a DN that you do not want URS to monitor.
12. If it does not already exist, create a section of the same name as the URS `Application` object.
13. Create a new option in this section called `event_arrive`.
14. Set the value to `none` so that only one URS in the group of load-balancing URSs registers on this particular routing point. Figure 90 on [page 240](#) shows an example.

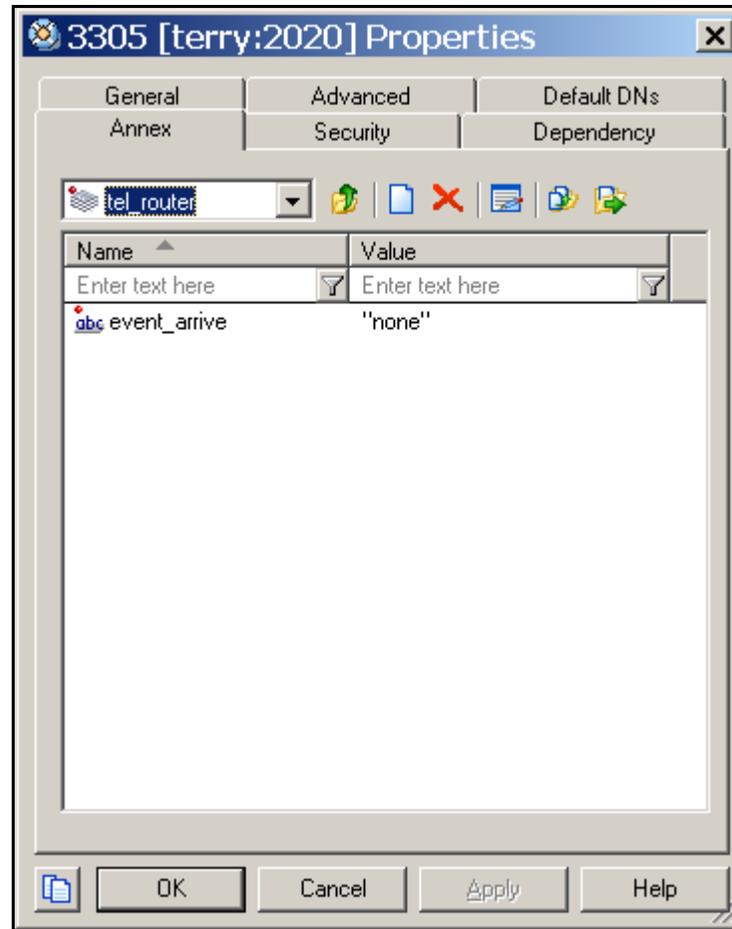


Figure 90: DN Properties Dialog Box

End of procedure

Strategy Loading In Load Sharing Mode

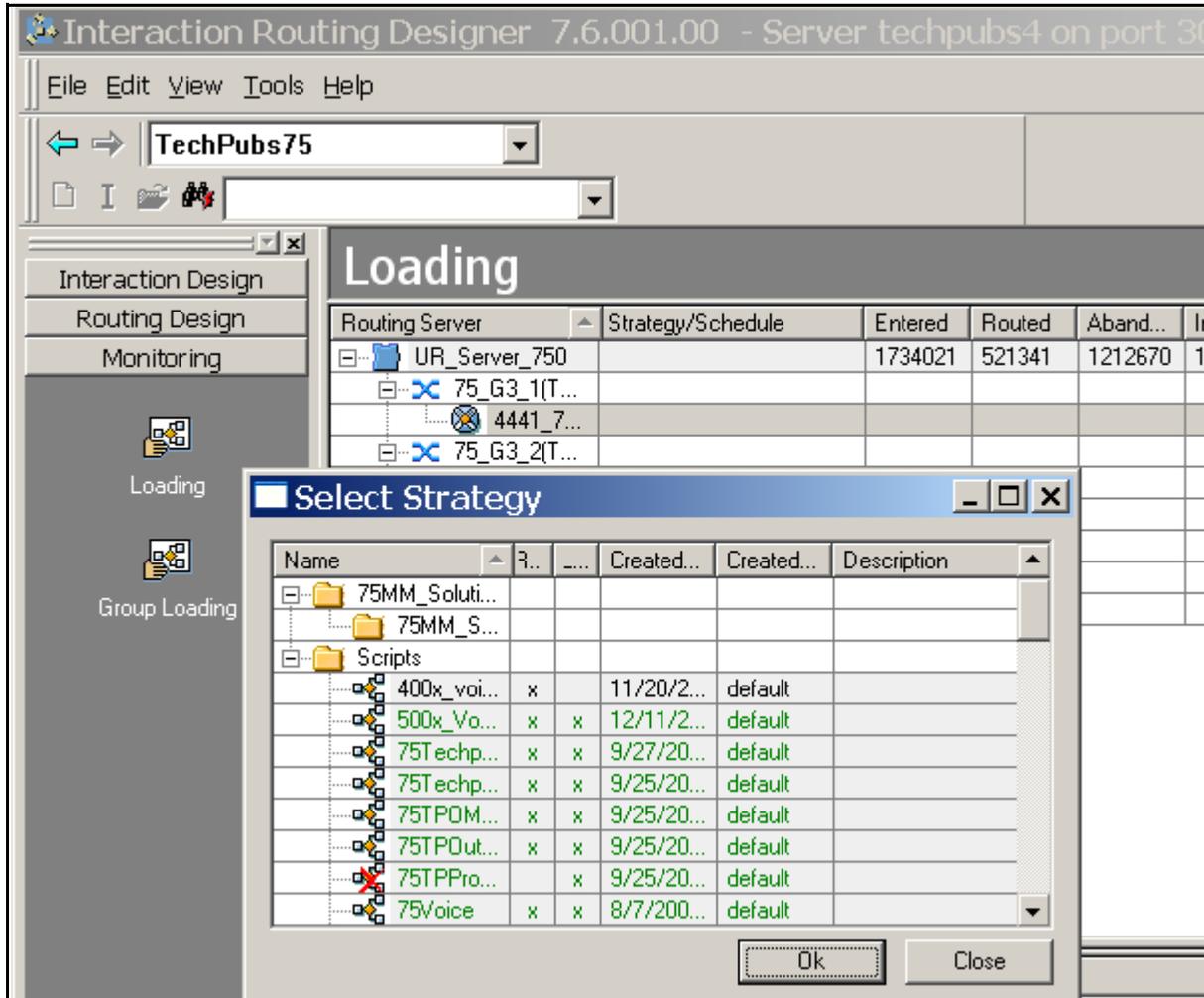
When two or more URS instances run in load sharing mode through LDS, IRD synchronizes strategy loading so you do not have to load the same strategy on every URS for each routing point in IRD's Monitoring view.

Procedure: Synchronizing strategy loading

Purpose: To avoid having to load the same strategy on multiple routing points when running in load sharing mode through LDS.

Start of procedure

1. Log into IRD as described on [page 268](#).
2. In the IRD main window, click the Monitoring shortcut bar.
3. Click the Loading icon.
4. Under Routing Server in IRD's Loading view, expand the URS/T-Server until you find the routing point where you want to load the strategy.
4. Right-click the routing point and select Load Strategy from the shortcut menu. The Select Strategy dialog box opens (see [Figure 91](#)):

**Figure 91: Select Strategy Dialog Box**

5. Select a strategy and click OK.

End of procedure

After you select a strategy, IRD determines whether the URS associated with the selected routing point is:

- Indirectly associated with T-Server, through LDS (LDS in the `Connections` list of the `URS Application` object). If LDS is involved, IRD scans your Configuration environment for all URSs that have the same LDS in their `Connections` list. You can see this information in the `Dependency` tab of LDS.

In this case, for every such URS, IRD performs same strategy loading operation on the selected Routing Point. There is no dialog box involved.

- If the URS associated with the selected routing point is directly associated with the T-Server (T-Server in the `Connections` list of the `URS Application` object), IRD loads the strategy on the selected routing point as usual.

Note: The strategy unloading operation is executed in a similar way.

Load Balancing Between Targets

Previously in Universal Routing, load balancing between targets assumed the use of the Load Balancing Routing object (described in *Universal Routing 8.0 Reference Manual*) in strategies. Internally, the Load Balancing Routing object uses the predefined `StatLoadBalance` statistic (see [page 246](#)), which results in a distribution of interactions to queues based on estimated waiting time or agent occupancy.

The `StatLoadBalance` statistic, however, does not take into account the additional complexities that can arise from load balancing among multiple URSs and from possible network delays. As a result, this statistic can provide an unsatisfactory level of load balancing between targets in the case of LDS and network delays.

As result of this possible issue, Universal Routing 8.0 extends the set of statistics oriented on load balancing between queues. Now, when you wish to perform load balancing between targets, you can select from the “[Load Balance Family of Statistics](#)” as described on [page 245](#). This section also discusses:

- How to use statistics to address the issue of calls in transition.
- How to set Router Self-Awareness, as described on [page 255](#)) so that URSs communicate with each other regarding selected targets and target statistics.
- How to use the `StrTargets` function (see [page 260](#)) and the utility subroutines ([page 260](#)) for load balancing.

Calls In Transition

When load balancing calls across multiple sites, you can more accurately determine the number of calls at each site by including calls that are in transition from the network to the contact center site (to ACDQueue, AgentGroup, or Agent targets) from all resources. Another way to think of the number of calls in transition is “the number of calls in the air.”

In general, calls are not expected to arrive at their destination immediately after a routing request, but after some (usually short) period of time has elapsed. As result, in any given moment there is a certain amount of calls in transition. The stars in [Figure 92](#) depict calls in transition that exist after the initial routing decision, that have not yet arrived at the contact center.

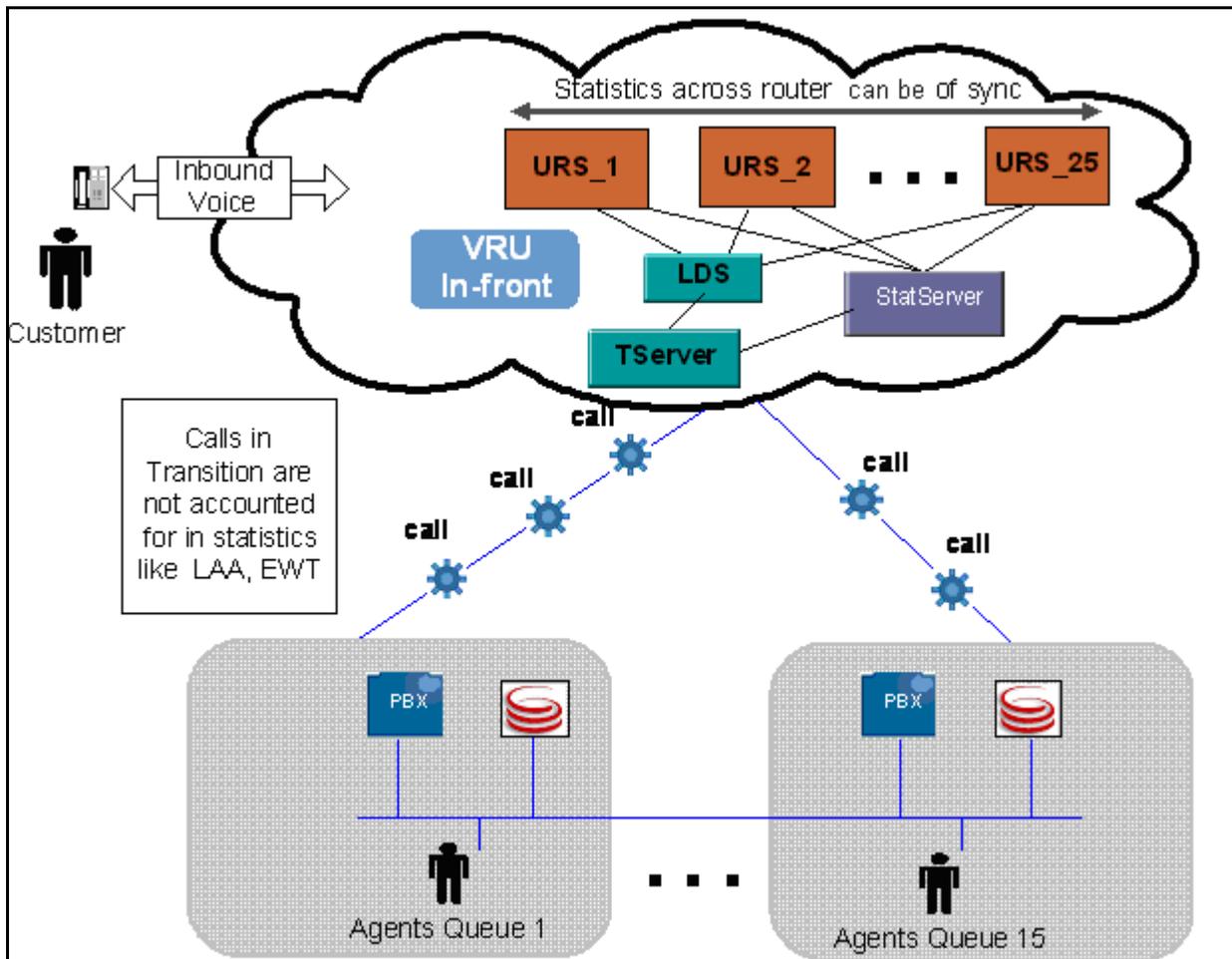


Figure 92: Calls In Transition

The important point is that calls in transition (if neglected) can sometimes cause a discrepancy in statistics used by URS for decision making. This, in turn, can affect load balancing or routing. The potential for this situation is amplified when there are multiple concurrently-deployed URS instances in load-sharing mode.

Number of Calls In Transition

URS calculates load balancing statistics based on either an *actual* number of calls in transition or an *expected* number of calls in transition.

- The **actual** number of calls in transition is the total current number of calls in the air (equal to the number of stars in Figure 92 on [page 243](#)) as it known to URS. Basically it is number of calls in a special calls-in-transition list maintained by URD in following way:
 - When URS issues the call routing request to T-Server, the call is placed in a calls-in-transition list.
 - When URS receives an event message from T-Server indicating that the call arrived at its destination, the call is removed from the calls-in-transition list.
 - A call also is removed from the list if some configurable timeout for the call has expired (by default, URS option `transfer_time`).
- The **expected** number of calls in transition is number of calls that URS expects will arrive at a destination during next short period of time (usually the time required for the currently considered call to reach the destination). URS makes an approximate estimate of this number based on assumptions it has to make about “streams of calls.” Specific proportions of calls distributed to some destination by multiple routers change slowly. The expected number can be bigger or less than the actual number of calls in transition depending on the amount of time required to deliver the currently considered call.

[Table 12](#) lists IRD predefined statistics that can be used in strategies for load balancing including those that address calls in transition.

Table 12: Load Balancing Statistics

Predefined Statistic	Description	New for 8.0?
StatLoadBalance (see page 246)	Calculated by Stat Server.	No
RStatCallsInQueue (page 247)	This URS-calculated statistic calculates the number of calls in queue. URS counts the number of calls in transition for this statistic by tracking every sent call. If Router Self-Awareness (see page 255) is activated, then calls in transition will include calls sent by all URSS participating in the same Self-Awareness group.	No

Table 12: Load Balancing Statistics (Continued)

Predefined Statistic	Description	New for 8.0?
RStatLoadBalance (page 247)	This URS-calculated statistic calculates a load balancing metric, which can be used for load balancing distribution based either on expected waiting time (if it is positive) or occupancy. URS counts the number of calls in transition for this statistic by tracking every sent call. If Router Self-Awareness is activated (see page 255), then calls in transition will include calls sent by all URSs participating in the same Self-Awareness group.	No
RStatExpectedLoadBalance (page 249)	This URS-calculated statistic calculates a load balancing metric, which can be used for load balancing distribution based on either expected waiting time (if it is positive) or occupancy. In this case, URS does not itself count the number of calls in transition for this statistic. Instead, it uses Stat Server to get the expected number of such calls.	No
RStatLBEWTLAA (page 250)	This new statistic calculates another load balancing metric based on either expected waiting time (if it is positive) or time in ready state for the longest available agent among all agents behind the queue. URS counts the number of calls in transition for this statistic by tracking every sent call. If Router Self-Awareness is activated (see page 255), then calls in transition will include calls sent by all URSs participating in the same Self-Awareness group.	No
RStatExpectedLBEWTLAA (see page 251)	Just like the previous statistic, this new statistic calculates the same load balancing metric based on either expected waiting time (if it is positive) or time in ready state for the longest available agent among all agents behind the queue. In this case, URS does not itself count the number of calls in transition for this statistic. Instead, it uses Stat Server to get the expected number of such calls.	No

Load Balance Family of Statistics

Load balancing between targets is essentially *equal* distribution of interactions among ACD queues. *Equal* can refer to the number of calls, the percentage of busy agents, the expected waiting time, or something else. There are many different criteria. What is equal for one customer will not be equal for another. The family of load balance statistics described below use some of more common criteria, but cannot cover all criteria. If the supplied statistics are not

appropriate, then you must configure a strategy to implement the criteria you prefer.

Note: Statistic names prefixed with “R” indicate a statistic calculated by URS (“Router”).

StatLoadBalance

Note: The statistic, `StatEstimatedWaitingTime`, used in versions 5.1.x and 6.0, was replaced by `StatLoadBalance` in the `LoadBalance` statistic category.

Information on the `StatLoadBalance` statistic calculated by Stat Server is provided here for the sake of completeness, and also because some of the other load balancing statistics are derived from it. In summary, the `StatLoadBalance` statistic:

- Works for “queue-like” targets (`ACDQueue`, `Routing Point`).
- Does not account for calls in transition (see [page 243](#)).
- Is calculated entirely by Stat Server according to formula shown below.

LB =

```
If (ALI= 0) /*no agents at all, wait forever*/ 10,000,000,000
Else if (AR <= CIQ) /*no available agents, use expected waiting
time*/AHT * (CIQ - AR +1)/ALI
Else /*enough agents, waiting time is 0, use minus occupancy*/ · (AR
- CIQ)/ALI
```

Where:

LB or LB(Q) = calculated `StatLoadBalance`

CIQ or CIQ(Q) = calls waiting in queue

AR or AR(Q) = number of ready agents logged into the queue

ALI or ALI(Q) = number of all agents logged into the queue

AHT or AHT(Q) = the average handling time for agents logged into the queue

Note: For details about how Stat Server calculates AHT as well as information about options that could possibly affect AHT calculation, refer to the *Framework 8.0 Stat Server Deployment Guide*.

Warning! The `StatLoadBalance` statistic does not fully address the additional complexities that can result from load balancing among URSs using LDS as described on [page 235](#). As a result, when LDS is used for load balancing, this statistic may provide an unsatisfactory level of load balancing between targets.

RStatCallsInQueue

This statistic:

- Works with “queue-like” targets (`ACDQueue` and `Routing Point`).
- Has the same meaning as `StatCallsInQueue`, but takes into account the number of calls in transition to the destination.

If URS cannot see the destination, `RStatCallsInQueue` becomes just the number of calls in transition. This can happen if the statistic is requested for a non-DN type of target or if you do not configure certain options. Use of this statistic requires registering URS on the destination DN. In addition, to properly configuring URS to calculate this statistic, you must:

- Configure the `call_monitoring` option to instruct URS to listen for call-termination events, if you want to use this statistic for non-configured DNs.
- URS must be registered on the appropriate objects and requested to count calls on them, if you want to use this statistic for ACD queues, routing points, and routing queues.
- If URS does not register on a requested destination by default, you can force it to register using the `transit_dn` option, specified on the desired destination.
- You can request URS to count calls on registered destinations using the `count_calls` option.

URS automatically delays the completion of a strategy after an interaction has been successfully routed to a destination if calls are being counted in order to allow time for the call to arrive to its destination. The delay is for the number of seconds set in the `transfer_time` option.

You can override the `transfer_time` setting by using the `Delay` function during postrouting to disable automatic call termination.

Previous to 7.6, the number of calls in transition could apply only to calls routed by the current URS (by the router that calculates this statistic). If Router Self-Awareness (see [page 255](#)) is activated, then calls in transition will include calls sent by all URSs participating in the same Self-Awareness group.

RStatLoadBalance

The `RStatLoadBalance` statistic:

- Is calculated by URS based its own data as well as utility statistics provided by Stat Server.
- Works for “queue-like” targets (ACDQueue and Routing Point).
- Takes into account the number of calls in transition to the destination.
- Uses one formula when a call is expected to wait at the destination (expected waiting time).
- Uses another formula (occupancy or percentage of busy agents) when a call isn’t expected to wait at the destination.

The formula for calculating `RStatLoadBalance` is the same as for `StatLoadBalance`, but calls waiting in queue from URS is used instead of calls waiting in queue from Stat Server. The accuracy of this statistic is based on the correctness of the load balancing statistics provided by Stat Server.

Note: Since this statistic reuses calls waiting in queue from URS, it requires the corresponding URS configuration, such as registering URS on the destination DN and setting up the `count_calls` option.

Average Handling Time

Since Stat Server does not have a readily available statistic to directly count Average Handling Time for agents logged into a queue (`AHT(Q)`), URS uses `TotalTime(Q)` and `TotalNumber(Q)` to calculate `AHT(Q)` for `RStatLoadBalance` and all the other load balancing statistics:

```
AHT(Q)=
  If TotalTime(Q) > 0 and TotalNumber(Q) > 0
    TotalTime(Q) / TotalNumber(Q)
  Else
    90 sec
```

`TotalTime(Q)` is hardcoded in URS as follows:

```
Category=TotalTime, MainMask=CallReleased, CallMissed,
ACWCompleted, ACWMissed, Subject=DNAction,
Interval=SlidingSelection, 50 calls.
```

`TotalNumber(Q)` is hardcoded in URS as follows:

```
Category=TotalNumber, MainMask=CallReleased, CallMissed,
Subject=DNAction, Interval=SlidingSelection, 50 calls.
```

If Router Self-Awareness (see [page 255](#)) is activated, then calls in transition will include calls sent by all URSs participating in the same Self-Awareness group.

RStatExpectedLoadBalance

Note: In order to use this statistic, both URS and Stat Server must be at least 7.6.

Introduced in 7.6, the `RStatExpectedLoadBalance` statistic:

- Works for “queue like” targets (`ACDQueue` and `Routing Point`).
- Is variation of the `StatLoadBalance` statistic described on [page 246](#).
- Takes into account the *expected* number of calls in transition to the destination (see [page 244](#)).
- Uses one formula when a call is expected to wait at the destination (expected waiting time) and another formula (occupancy or percentage of busy agents) when a call is not expected to wait at the destination.
- Is calculated by URS based on its own data and utility statistics provided by Stat Server.
- Has a predictive nature in that every URS tries to build a prediction about the loading of every target that will occur a few seconds ahead in time.
- Requires neither Router Self-Awareness (see [page 255](#)) nor any other special URS configuration (as opposed to cases when the `RStatCallsInQueue` statistic is involved).

The calculation formula for `RStatExpectedLoadBalance` is based on the formula for `StatLoadBalance`. Instead of calls waiting in queue, it uses the expected number of calls in queue.

Expected Waiting Time Versus Occupancy

The three load balance statistics just discussed (`StatLoadBalance`, `RStatLoadBalance`, and `RStatExpectedLoadBalance`):

- Use one formula when a call is expected to wait at the destination (expected waiting time).
- Use another formula (Occupancy or percentage of busy agents) when a call isn't expected to wait at the destination. Occupancy is discussed below.

Occupancy Rate Statistics

Occupancy rate is the ratio between the time the agent has been busy since last login relative to the agent's total login time. Statistics that use Occupancy enable URS to evaluate multiple available agents and select the least occupied agent so that the workload among available agents is balanced.

Longest Available Agent Statistics

The next two statistics, `RStatLBWTLAA` and `RStatExpectedLBWTLAA`, use an alternative formula for achieving load balance.

- When a call is expected to wait, the statistics still use expected waiting time.
- When a call is not expected to wait, the statistics use the waiting time of the Longest Available Agent (LAA) behind the destination queue.

LAA statistics are basically “agents level statistics” and work well in cases where target selection is done at the agent level. Although applied to the `ACDQueue/Routing Point` target types, the LAA statistics work at the agent level – using them allows URS to find a queue or routing point that has the best agent.

LAA Statistic Use Case

Assume that a call must be sent to an agent (for example, to an agent among all agents that meet some skill expression or to an agent among all agents for all sites) based on the `StatTimeInReadyState` statistic (a statistic that is very commonly used to select between agents). Assume also that the call should not be routed directly to the selected agent, but through some queue.

Previous to 7.6 and the LAA statistics, URS solved this task with regular agent type targets (groups of agents/skill expressions) using a subsequent number translation that substitutes an agent DN with some queue DN. However, the configuration required for this number translation is significant (for every agent DN, a destination queue must be defined).

The LAA statistics do the same thing in more simple way—by directly targeting queues where calls need to be routed (no number translation is needed in this case).

RStatLBEWTLAA

Introduced in 7.6, the `RStatLBEWTLAA` statistic:

- Works for “queue-like” targets (`ACDQueue` and `Routing Point`).
- Takes into account number of calls in transition to the destination.
- Uses expected waiting time when a call is expected to wait. However, when a call is not expected to wait, the statistic uses the waiting time of the Longest Available Agent behind the destination queue.
- Is calculated by URS based on its own data and utility statistics provided by Stat Server.
- Reuses calls waiting in queue from URS just like the `RStatLoadBalance` formula. As a result, if Router Self-Awareness is not activated (see [page 255](#)), the number of calls in transition includes only calls routed by the current URS (by the URS that calculates this statistic).

If Router Self-Awareness is activated, then `RStatLBEWTLAA` includes all calls in transition (from all participating URSs).

The main difference from the previously described load balancing statistics is the criteria used when there are no available agents, which is Longest

Available Agent (longest available time among all agents behind the queue after first excluding those from `RStatCallsInQueue`).

Calculating Longest Available Agent

To calculate the value of LAA, URS must have a list of all agents behind the queue. This works in following way:

Having the queue, URS looks for an Agent Group with the same name as the queue Alias. If there is no such group, then URS uses the first Agent Group it finds that has this queue as the origination DN.

- If no Agent Group is found, the value of LAA is 0 (zero).
- If an Agent Group is found (ideally, this group must a virtual one comprised of all agents logged into this queue), then URS:
 - Goes through the list of all agents from the group.
 - Arranges them according to longest available time (time in Ready state).
 - Ignores top `RStatCallsInQueue` number of them and returns the longest available time for the next agent.

Note: Since this statistic reuses calls waiting in queue from URS, it requires corresponding URS configuration, such as registering URS on the destination DN, setting up the `count_calls` option, and so on.

RStatExpectedLBEWTLAA

Introduced in 7.6, the `RStatExpectedLBEWTLAA` statistic:

- Works for “queue-like” targets (`ACDQueue` and `Routing Point`).
- Uses expected waiting time when a call is expected to wait. However, when a call is not expected to wait, the statistic uses the waiting time of the LAA behind the destination queue.

The main difference from `RStatLBEWTLAA` is that `RStatExpectedLBEWTLAA` uses the *expected* number of calls in transition instead of the actual number of calls in transition (see [page 244](#)).

Reporting on Load Balancing

To support load balancing reporting, and to allow for both real-time and historical reporting of calls in transition, use the `report_statistics` option. Setting this option to `true` instructs URS to attach additional reporting data to calls. If option `report_statistics` is set to `true`, URS extends the existing reporting data with load balancing statistical information.

The data attached to a call when this option is set to `true` depends on the load balancing statistic you select in the Routing Selection object properties dialog box. For every target evaluated, URS attaches separate pieces of User Data

(AttributeUserData) with key LBR_DEST. For a description of the data attached for each load balancing statistic, see the `report_statistics` option description in the *Universal Routing 8.0 Reference Manual*.

Deciding Which Statistic to Use

Table 13 compares the load balancing statistics:

Table 13: Load Balancing Statistic Comparison

Name	Uses calls in transition? *	Requires Router Self-Awareness & registration on destin. DNSs?	No available agents cases	There are available agents	Calculated by
StatLoadBalance	No	No	EWT**	Occupancy	Stat Server
RStatLoadBalance	Currently available info.	Yes	EWT**	Occupancy	URS+SS
RStatExpectedLoadBalance	Expected number	No	EWT**	Occupancy	URS+SS
RStatLBEWTLAA	Currently available info.	Yes	EWT**	Longest Available Agent***	URS+SS
RStatExpectedLBEWTLAA	Expected number	No	EWT**	Longest Available Agent***	URS+SS

* See [page 244](#) for a comparison of calls in transition and expected number of calls in transition.

** EWT = Expected Waiting Time. This is not StatExpectedWaitingTime. Stat Server counts StatExpectedWaitingTime a little differently than StatLoadBalance (which also returns expected waiting time when there are calls in queues). For the statistics prefixed with RStat above, URS calculates Expected Waiting time by a formula similar to that used by Stat Server for StatLoadBalance. StatLoadBalance replaced StatEstimatedWaitingTime used in 5.1.x and 6.0.

*** See [page 251](#) for a definition of Longest Available Agent.

Recommendations

While there are no clearly defined criteria regarding which load balancing statistic to use, some general recommendations are given below.

- The statistics shown in [Table 13](#) are all queue-based (not agent-based). The Occupancy rate for a queue is the percentage of busy agents. For example, if 10 agents are logged into a queue and 7 of them are busy, then occupancy is 70% or 0.7.
- Whether Occupancy or Longest Available Agent is better is completely up to you as the customer. The Genesys point of view is that Occupancy is better; however some customers will prefer Longest Available Agent.
If you prefer to use Occupancy (see [page 249](#)), then you must use `StatLoadBalance`, `RStatLoadBalance`, or `RtatExpectedLoadBalance`.
If you prefer longest available agent, then you must use `RStatLBWTLAA` or `RStatExpectedLBWTLA`.
- If you do not wish to use any of the above predefined statistics, then you must implement your own version of “correct” load balancing in a strategy.
- `StatLoadBalance` (see [page 246](#)) may be the best statistic for customers that can use Occupancy and have no delays in routing; for example, customers that use local premise routing.
- The statistics in [Table 13](#) that use expected number of calls in transition (see [page 244](#)) attempt to predict the future. If those statistics correctly predict the future, then using those statistics is preferable to using the statistics based on currently available information for calls in transition.
Unfortunately, it is difficult to say in advance whether or not the statistics that use the expected number of calls in transition will be successful at your particular contact center. Since their prediction is based on extrapolating current behavior into the future, these statistics usually work well when calls flow is stable (without sudden peaks and valleys). You may need to experiment to determine this.
Genesys recommends that you first try the statistics that use expected number of calls in transition. If they do not work well, then use the alternative statistics based on currently available calls in transition information.
- Other than `StatLoadBalance`, the load balancing family of statistics take into account the possibility that URSs may work in load balance mode with LDS. These statistics will work equally well with or without LDS.

RStatCallsInTransition

Starting with 7.6, URS provides a statistic that cannot be used for load balancing distribution, but can be used in a strategy to adjust other statistics received from Stat Server (similar to `RStatCallsInQueue`). The `RStatCallsInTransition` statistic:

- Works for “queue like” targets (`ACDQueue`, `Routing Point`) as well as for `Agent`, `Place`, and `Agent/Place Group`.

- Returns the number of calls that URS believes are on the way to corresponding targets, but not yet arrived.
- Also includes calls distributed by all participating URSs when Router Self-Awareness (see [page 255](#)) is activated for this statistic.
- Is counted by URS based on available information from T-Servers.
 - Every time URS selects a call to be routed to a target, the corresponding counter for this target is incremented by 1.
 - Every time URS detects that a call has arrived at the target, the counter is decremented by 1.
 - When URS deletes the call from its memory, the counter is also decremented.

URS Deletion of Calls From Memory

URS's calculation of calls in transition is based upon the number of calls that URS keeps in its memory (only these calls are counted). As result, it is crucial for URS to keep calls in memory for a long enough time period to permit a correct counting of calls in transition.

By default., URS deletes calls from its memory upon receiving `EventRouteUsed`. In cases where this event does not indicate a call arriving at a destination, you must suspend call deleting in the strategy. You can achieve this by using function `Delay` in post routing for the time that is expected to be the delivering time.

URS automatically suspends call deleting if a call is routed to a destination queue with URS option `count_calls` is set to `true`.

Additionally if Router Self Awareness is active, then every URS receives the corresponding values of this statistic from all participating URSs (and updates all other URSs about any changes in this statistic). `RStatCallsInTransition` returns the sum of these counters (URS's internal counter and all counters reported by external URSs).

RStatCallsInTransitionEx

Since the `RStatCallsInTransition` statistic (see [page 253](#)) does not take into account calls with issued agent reservation requests, this statistic is not always suitable for use as part of target threshold expressions. To overcome this limitation, a new built-in statistic called `RStatCallsInTransitionEx` is provided. This statistic behaves like the existing `RStatCallsInTransition` statistic, but also counts those calls which—while not yet sent to the target—already have an active reservation request issued to this target or to one of the target's members (if the target is a group of agents/places).

Router Self-Awareness

Previous to 7.6, the number of calls in transition could apply only to calls routed by the current URS (by the router that calculates this statistic). Starting with 7.6, you can configure URSs, such as those in a load balancing scenario, to share routing information between themselves by setting a mode called *Router Self-Awareness*. If Router Self-Awareness is activated for the participating URSs, then the number of calls in transition sent to this target will include calls in transition (see Figure 92 on [page 243](#)) sent by all URSs participating in the same Router Self-Awareness group.

Using Router Self-Awareness, URSs deployed in a load sharing mode can communicate with each other regarding selected targets and target statistics. This addresses potential load balancing issues across multiple URSs. It also addresses certain race (timing) conditions that can occur in agent-based routing.

When Router Self-Awareness mode is configured, URSs can exchange internal data in order to have more real time information on their working environment. In addition to information about any URS decision to send a call to some destination, information in the communication channel between URSs can also be used for:

- Calculating some of the “Load Balance Family of Statistics” on [page 245](#).

Note: Router Self-Awareness affects statistics `RStatCallsInTransition`, `RStatLoadBalance`, `RStatCallsInQueue`, and `RStatLBewTLAA`. These statistics will return different values depending it is on or off. They will include calls handled either by all URSs participating in the Self-Awareness group) or, if not set, will only include calls for one particular URS.

- Agent blocking. If Router Self-Awareness is on, then every URS will block the agent for routing as soon it receive notification that an agent is selected by some other URS. The assumption here is that other URS’s notification can arrive much sooner than the agent will be reported as busy by Stat Server. This can save URS from the necessity to doing a reserving request that will unconditionally fail.
- Preventing other URSs from selecting the same targets (Agents, Places) during the early phase of routing before the agent reservation mechanism detects the call.

Router Self-Awareness and Agent Reservation

Agent reservation flow is as follows (T = a moment in time):

1. An interaction is queued internally by URS.
2. URS requests agent reservation from T-Server (T1).

3. T-Server reserves the agent and notifies URS (T2).
4. URS requests T-Server to route the call (`RequestRouteCall`).
5. T-Server communicates to Stat Server that the call is ringing (T3).
6. Stat Server communicates to URS that the agent is busy (T4).

Previous to Router Self-Awareness, any other URS would have been able to start a `RequestRouteCall` or reserve the agent between the T2 and T4 times (being unaware that the routing attempt will fail), which could possibly lead to a race condition and a waste of call processing time.

When Router Self Awareness is set, the URS that does the agent reservation communicates this information to all the other URSs at T2. All other URSs then disregard the agent based on this event and do not attempt to route to the agent.

In summary, before Router Self-Awareness, all URSs knew that an agent became busy for routing at the T4 moment (when notified by Stat Server). With Router Self-Awareness, all other URSs know that an agent becomes busy for routing at the T2 moment (when the information is shared between participating URSs).

For more information on agent reservation, see the `agent_reservation` option in the *Universal Routing 8.0 Reference Manual*.

Implementation

Router Self-Awareness uses a dedicated Message Server. In order to become part of a Router Self-Awareness group (to provide and receive data from other URSs from this group), you must dedicate a Message Server to the sole task of exchanging routing data between multiple URSs.

Procedure:

Configuring URS for Router Self-Awareness

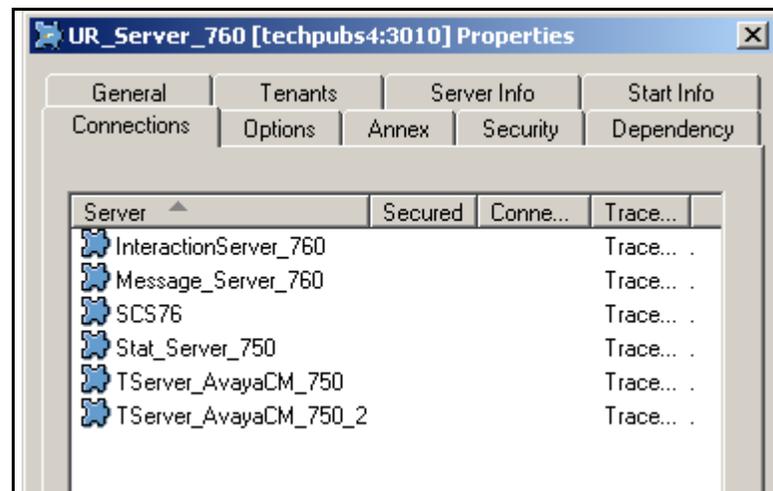
Purpose: To enable URSs to participate in a Router Self-Awareness group.

Summary:

- Each URS must have the common Message Server in their `Connections` list (`Connections` tab of the `URSAplication` object).
- The `MessageServerApplication` object must have the `URS` option `using` set to a value of `lds` in its `Annex` tab (`__ROUTER__` section or section `<URSAplication Name>`).
- This Message Server must also have the `URS` option `lds` set to `ciq` (calls in queue). The name of this option, which specifies the type of information that the URSs will exchange, is the value of option `using`.

Start of procedure

- Connections List**
1. Log into Configuration Manager as described on [page 183](#).
 2. Select the Tenant or Environment.
 3. Open the URS Application object.
 4. Select the Connections tab (see [Figure 78](#) on [page 195](#)).
 5. Click Add.
 6. In the resulting New Connection Info dialog box opposite Server, click the browse button.
 7. Select the common Message Server.
 8. Complete the remaining fields in the New Connection Info dialog box.
 9. Click OK in the Connections tab. [Figure 93](#) shows an example completed Connections list tab.

**Figure 93: URS Application Object, Example Connections List**

- Option Using in Message Server**
10. Open the common Message Server Application object.
 11. Click the Annex tab.

Note: If the Annex tab is not displayed, choose View > Options, select Show Annex tab in object properties in the dialog box, and click OK.

12. Click the button to create a new section/option (see [Figure 94](#)).

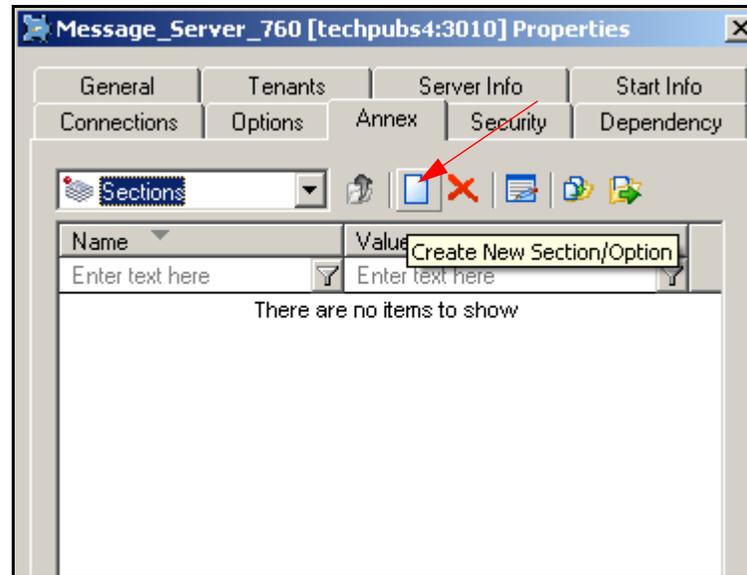


Figure 94: Message Server Application Object, Annex Tab

13. In the Add Section dialog box, name the section `__ROUTER__` or `<URS Application Name>`.
 14. Click OK in the Add Section dialog box.
 15. Double-click the new section to select it.
 16. Click the button to create a new section/option (see [Figure 94](#)).
 17. In the resulting Edit Option dialog box, enter `using` for the Name, `lds` for the Value, and click OK.
- Option Ids in Message Server**
18. Still in the Annex tab and in the `__ROUTER__` section, click the button to create a new section/option.
 19. In the resulting Edit Option dialog box, enter `lds` for the Name, `ciq` for the Value, and click OK.

Note: If necessary, a single Message Server can be used to pass information for both IVR Load Balancing (as described on [page 262](#)) and calls in transition (see [page 244](#)). In this case, option `lds` must have both `ciq` and `ar` as its value (comma-separated). [Figure 95](#) shows an example.

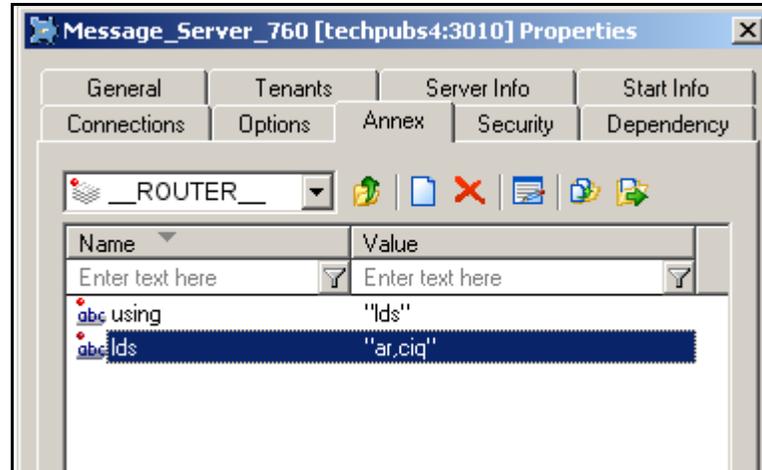


Figure 95: Message Server Application Object, Router Self-Awareness

- Click OK to save the option descriptions in the Message Server Application object.

End of procedure

Notes: It is not necessary for this Message Server to run with a Load Distribution Server.

A dedicated Message Server is also used for IVR load balancing as described on “IVR Server Load Balancing” on [page 262](#)).

Effect of Router Self-Awareness

Router Self-Awareness mode affects the following:

- Statistics `RStatCallsInTransition`, `RStatLoadBalance`, `RStatCallsInQueue`, and `RStatLBWTLAA` return different values depending on whether Router Self-Awareness is on or off.

If Router Self-Awareness is on, the statistics will include calls handled by all URSs that have the common Message Server in their `Connections` list. If Router Self-Awareness is off, the statistics will include only those calls handled by the current URS.
- Agent blocking. When Router Self-Awareness is turned on, every URS will block an agent for routing as soon it receive notification that this agent is selected by some other URS. The assumption is that the notification from the other URS arrives before the time that Stat Server reports the agent as busy. This can save URS from the necessity of making a reserving request, which will unconditionally fail.

Router Self-Awareness means that when URS selects an agent based on any regular agent statistic (`StatTimeInReadyState`, for example), it will

automatically take into account all calls in transition. There is no need for ANY additional configuration (other than configuring Router Self-Awareness).

Mixing Load Balancing Types

In addition to using Router Self-Awareness for load balancing between targets, you can also use Router Self-Awareness when mixing together:

- Load Balancing Among Multiple URSs (see [page 235](#)) and
- Load Balancing Between Targets (see [page 242](#))

For example, you can have multiple URSs running with LDS, and have those URSs executing strategies that use the load balancing statistics (performing load balancing among routing targets).

Router Self-Awareness is one of tools that facilitates acceptable load balancing between targets when URSs run in an LDS environment.

Note: Router Self-Awareness is required for cases when there is more than one URS load balancing between the same targets.

Other Ways to Load Balance

You can perform “customized” load balancing by implementing a load balancing algorithm in your strategies using a Genesys-supplied function and subroutines.

StrTargets Function

Instead of using atomic statistics, this function lets you implement your own selection and apply it to list of targets for the best one to be selected. The StrTargets function is a pure string formatting function. It facilitates the creation of a comma-separated list of targets for use as input parameters for the Genesys-provided subroutines (see “Utility Subroutines” on [page 261](#)), which may return the optimal target from the provided list.

Note: Prior to 7.6, you could use function Cat[], which required you to manually type in the names of all targets. Starting with 7.6, the StrTargets strategy function makes this task easier.

When you use this function in a strategy, for every function parameter, IRD provides a TARGET dialog box. [Figure 96](#) shows the TARGET dialog box after selecting the Agent Group target type.

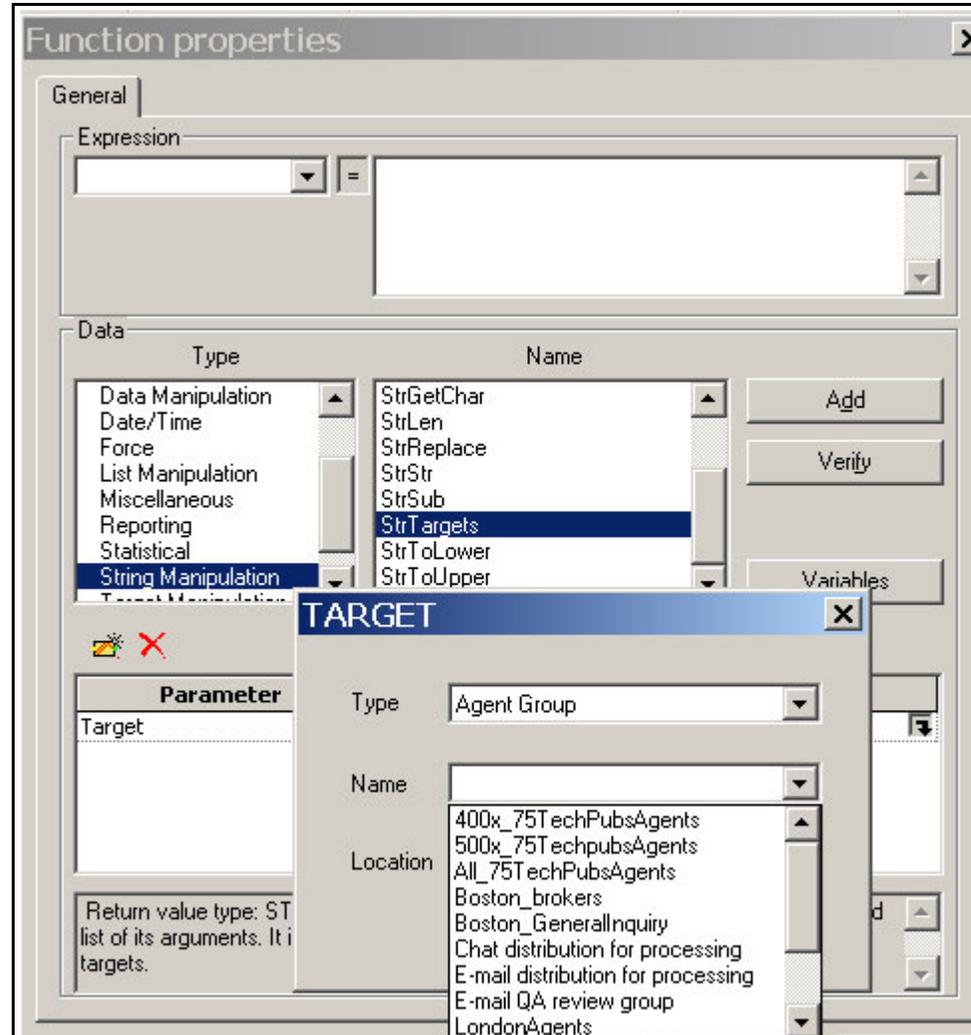


Figure 96: Function StrTargets and TARGET Dialog Box

The dropdown menus in the dialog box free you from the need to manually type in target names. For example, as shown in Figure 96 on page 261, after selecting the Agent Group target type, the Name dropdown lists all Agent Group target types in your Configuration Database so you don't have to manually enter them. This function also marks all used targets as taken by including these targets in the Check Integrity mechanism.

Utility Subroutines

The Universal Routing 8.0 installation package places an `RLU.zcf` file in the IRD installation directory. When imported into IRD, this file contains several utility subroutines along with a description of the stored procedure used by each. Figure 97 shows the subroutines after you import the `*.zcf` file into IRD.

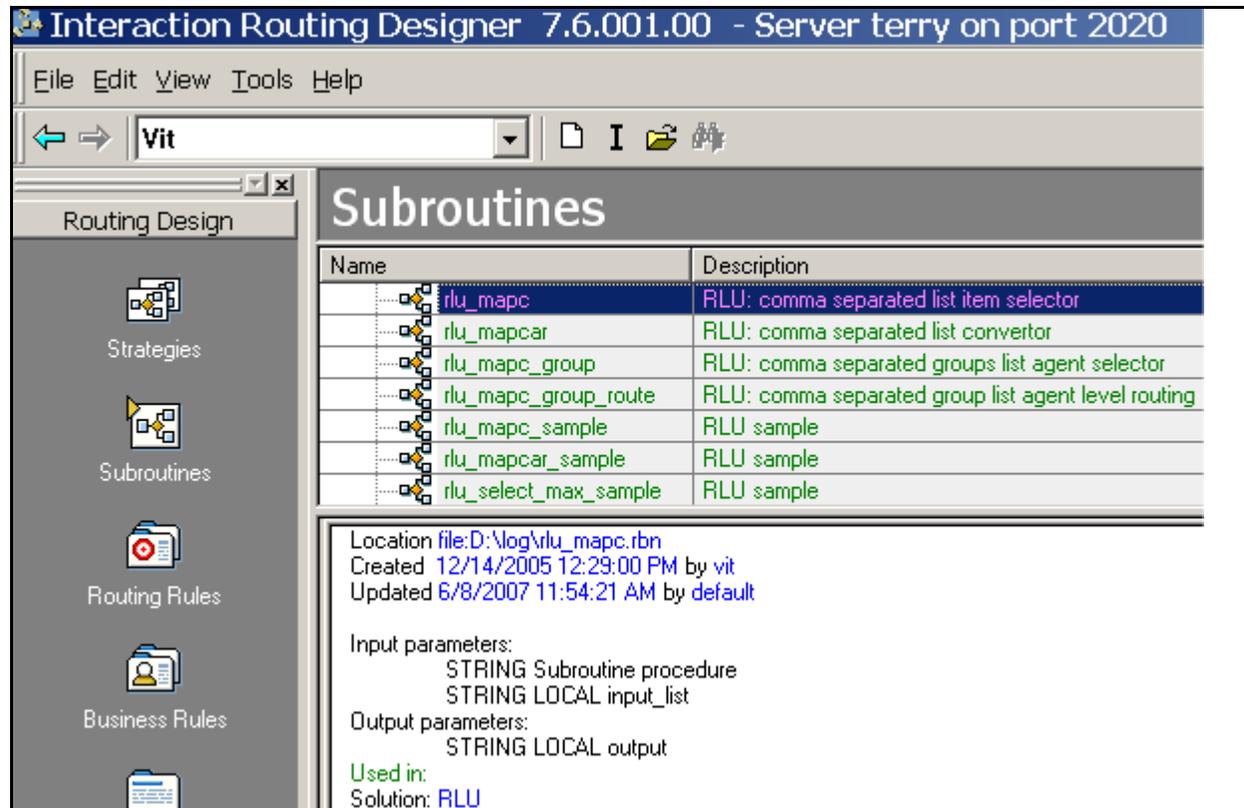


Figure 97: Sample Utility Subroutines

Each subroutine accepts a comma-separated list of targets and a corresponding “selecting” procedure and then returns the optimal target. For more information on these subroutines, see “Files in RLU.zcf” on [page 274](#).

IVR Server Load Balancing

URS supports IVR Server Load Balancing deployed in IVR In-Front mode (as described in the *IVR Interface Option 7.5, IVR Server System Administrator's Guide*).

As described in the above guide, when a vendor-provided IVR is connected directly to the Public Switched Telephone Network (PSTN), without a premise switch, the configuration is called *IVR-In-Front*.

As described below, URS controls the DNIS pool, but it doesn't resolve access numbers. Instead, URS instructs a “source” T-Server to route a call to the destination (DN@switch) and provides the DNIS number to use. The source T-Server's job then is then to negotiate with the destination T-Servers.

Access Resources

In a scenario with IVR Server Load Balancing operating in IVR In-Front mode, URS selects an access resource (DN) and provides it to the source T-Server for each interaction. For every pair of Switches (where Switch A is the source switch and Switch B is the destination switch), URS maintains a list of references of suitable access resources. URS also keeps a counter of free access resources for each pair of Switches.

To route a call from Switch A to Switch B, URS scans the list of suitable resources for the next free access resource. When found, URS marks the access resource as used and decrements the counter of available resources for this pair of Switches. If the selected access resource is shared by several source Switches, then URS decrements all other counters respectively. Upon startup, URS considers every access resource as free.

When it selects an access resource for routing a call, URS marks the resource as occupied. URS clears the occupation flag upon receiving `EventRouteUsed` (or `EventError`) after trying to route the call. URS then considers the access resource available for another call.

An access resource is any DN having:

- `type Access Resource` and `Resource Type (LoginID)` equal to `dnis` or
- `type External Routing Point` and its `Annex option dnis` equal to 1, which is set in the `TServer` section of the `Annex` tab

Access resources for communicating with Switch B from Switch A are specified on Switch B (destination Switch). Some access resources can be used only by one source Switch (Switch A); other access resources are shared (can be used by more than one source Switch). For information on when Switch A can use a Switch B resource, consult the applicable Genesys T-Server deployment guide.

Option Settings

Working with this type of IVR Server deployment means you need special option settings as follows:

- `use_extrouter = false` (meaning that URS delegates external routing functionality to T-Server)
- `use_extrouting_type = dnis` (meaning that although external routing is performed by T-Server, URS is still required to provide T-Server with access numbers that will be reserved on remote T-Servers)

Because URS has no exact knowledge of what access resources are actually available on a destination Switch (since the destination T-Server performs the actual resource allocation), URS considers all access resources to be available. For this reason, there is always a chance that a routing attempt can fail. In this case, use the `reroute` or `try_other` values for URS option `on_route_error`.

Message Server for Multiple URSs

If multiple URSs can route a call to some destination, there is a possibility (even if every URS has exact knowledge of the available resources at a destination) that more than one URS will select the same access resource, which can result in a failure to route the call.

Notification Approach

To prevent congestion, Genesys recommends coordinating access resource selection through a *notification* approach where URSs provide each other (through a common Message Server) with information about selected access resources. For this notification approach to work, you must use a Message Server dedicated to exchanging routing data between multiple URSs.

To specify the type of information the URSs must exchange, this Message Server must be in the `Connections` list of each URS and have following options in section `__ROUTER__` or section `<URS Application Name>`:

- `using = lds`
- `lds = ar`

For an example, see Figure 95 on [page 259](#). Finding such a Message Server in its `Connections` list forces any URS to distribute a short message each time an access resource is allocated or freed. View the messages in the URS log with verbose level 6 (for debugging).

This notification approach doesn't always eliminate congestion completely, but it does allow URSs to work even in cases of temporary unavailability of Message Server.

Also see “Configuring Message Servers for Different Functions” on [page 229](#).



Chapter

10

Orchestration Support

Starting with release 8.0, Universal Routing takes a more open approach to routing strategies. In addition to its ability to execute routing strategies that are created by using the Genesys Interaction Routing Language (IRL), routing strategies written in SCXML (State Chart EXtensible Markup Language) can be executed by the new Orchestration Server (ORS) component. ORS includes SCXML capabilities—enabling it to interpret SCXML code.

Note: Refer to the *Orchestration Server 8.0 Deployment Guide* for more information on SCXML strategy support and Orchestration Server.

This chapter contains the following topic:

- [Configuring URS to Work with ORS, page 265](#)

Configuring URS to Work with ORS

The general recommendations to configure the Universal Routing Server (URS) application to work in tandem with ORS are the following:

- The URS configuration option `strategy` *must* be set to `ORS`. This ensures that URS is prepared to process interactions according to requests that are received from ORS.
- The `event_arrive` option should be set to `ringing`. This allows URS to “see” the same pool of interactions that the ORS “sees.” If resource allocation for any interaction will be needed only when the interaction is on a routing point, setting this option to `ringing` is not necessary.
- The `use_ivr_info` option should be set to `false`. This reduces the amount of CPU that URS uses when interactions are moved from DN to DN.
- URS must be connected to the same T-Servers as ORS.
- URS must have the same list of tenants as ORS.

- Strategies on DNs and Interaction Queues that are controlled by ORS can be loaded by IRD or by Orchestration, but not a mix of both. The same method must be used for loading all strategies on DNs and Interaction Queues that are controlled by ORS. Refer to the *Orchestration Server 8.0 Deployment Guide* for more information.

For more information about URS configuration options, refer to the *Universal Routing 8.0 Reference Manual*.



Chapter

11

Samples

The Interaction Routing Designer (IRD) installation process places two *.zcf files in the IRD installation directory. After you import the *.zcf files into IRD as described below, you get sample strategy, subroutine, and list objects. If you install eServices (called Multimedia in 8.0.0 and earlier), with its Interaction Workflow Samples component, you also get sample strategies and subroutines, as well as other objects. This chapter familiarizes you with the samples.

The samples are not designed for use in a Production environment. Instead, use them to get started configuring your own strategies, subroutines, and list objects. Consider them as guides when developing your own objects adjusted to your company's specific business needs.

Note: *Universal Routing 8.0 Strategy Samples* provides examples of various types of voice and multimedia routing strategies. The information includes strategy flows and the properties of the various strategy-building objects. If you need an example of how to use an IRD strategy-building object, start with this guide. Also see *Universal Routing 7.6 (or later) Business Process User's Guide*, which provides example multimedia routing strategies for use in interaction workflows.

This chapter includes the following topics:

- [Importing the *.zcf Files, page 268](#)
- [Files in Samples.zcf, page 272](#)
- [Files in RLU.zcf, page 274](#)
- [Interaction Workflow Samples, page 279](#)

Task Summary: Importing Samples

Objective	Related Procedures and Activities
Log into Interaction Routing Designer	Procedure: Logging into Interaction Routing Designer , on page 268
Import sample strategies and other objects	Procedure: Importing the sample strategies and other objects , on page 270
Open a business process	Procedure: Opening a business process in IRD's Interaction Design window , on page 279
View the path for the .RBN file	Procedure: Viewing the path for the .RBN file , on page 287

Importing the *.zcf Files

The IRD installation process places `Samples.zcf` and `RLU.zcf` files in the IRD installation directory. Follow the steps below to import these files into IRD.

Procedure: Logging into Interaction Routing Designer

Start of procedure

1. Click the desktop shortcut if a shortcut is present (see [Figure 98](#)).



Figure 98: Interaction Routing Designer Desktop Icon

As an alternative, click the Start button on your computer desktop and select Programs > Genesys Solutions > Routing > Interaction Routing Designer > Start Interaction Routing Designer.

Either action brings up the Interaction Routing Designer login dialog. [Figure 99 on page 269](#) shows dialog box with example entries after clicking Details.

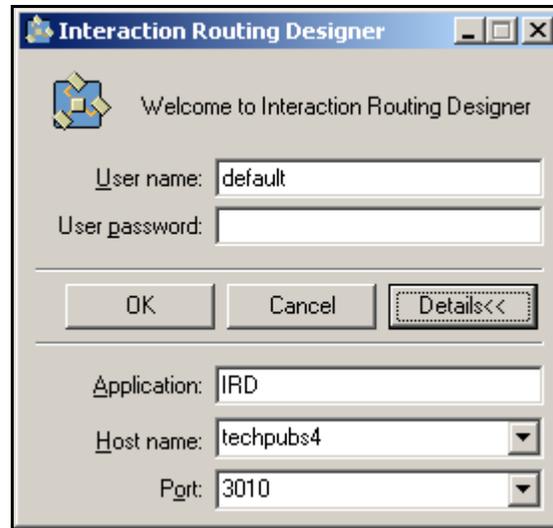


Figure 99: Interaction Routing Designer Login Dialog Box

Note: The first time the login dialog box opens, all fields are empty. The next time you log in, IRD “remembers” previous entries for the following fields: User name, Application, Host name, and Port.

2. Use the information in [Table 14](#) to complete the login dialog box.

Table 14: IRD Login Dialog Box

Field	Description
User name:	Name of Person object defined in Configuration Manager.
User password:	Password of Person object defined in Configuration Manager.
Application:	Enter the name of an IRD Application defined in Configuration Manager.
Host name:	Name of machine where Configuration Server is running.
Port:	Port number used by Configuration Server.

End of procedure

After you complete the login fields and click OK, the Strategies list pane opens (see [Figure 100](#) on [page 270](#) for an example).

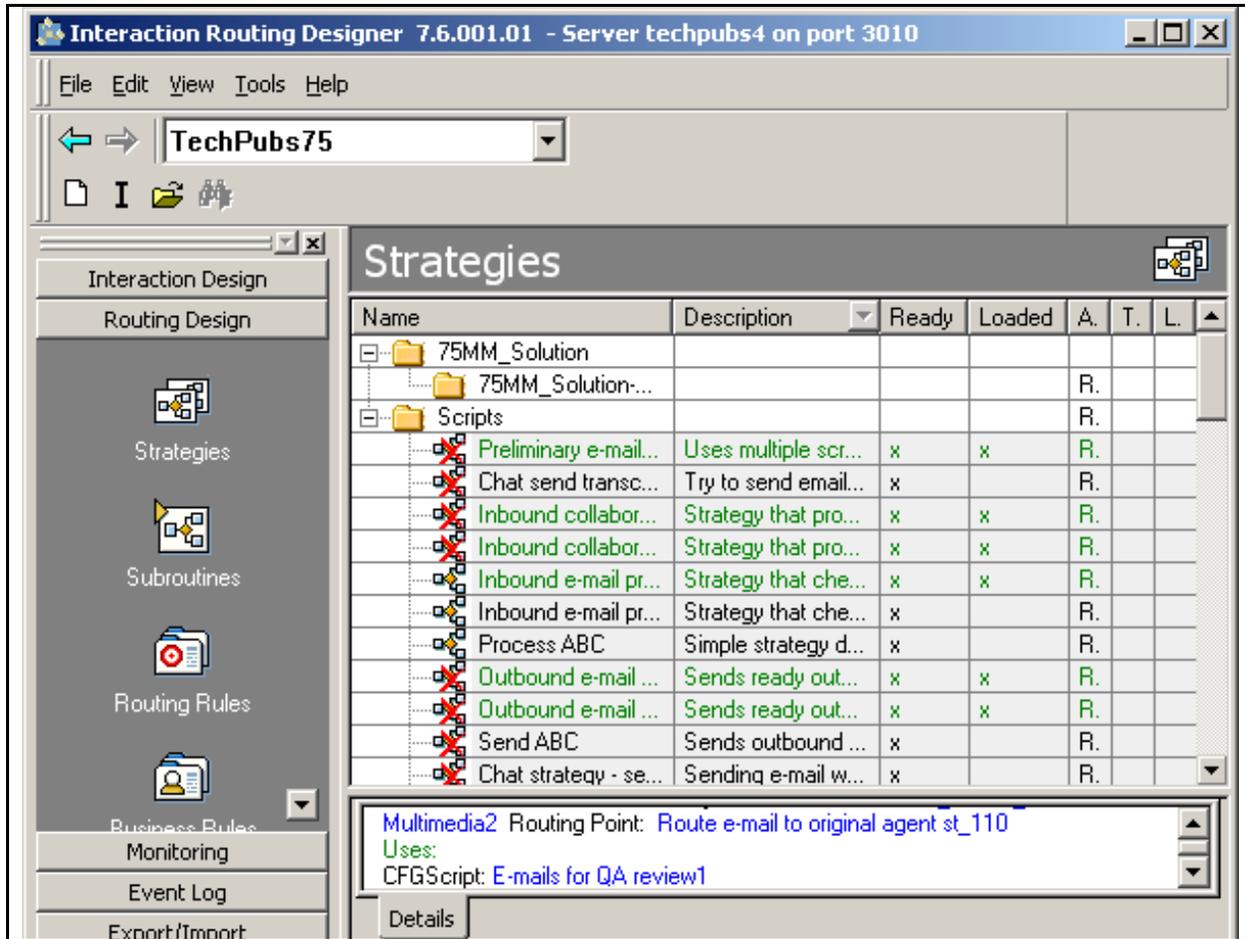


Figure 100: Strategies List Pane

Procedure: Importing the sample strategies and other objects

Follow the instructions below to import either `Samples.zcf` or `RLU.zcf` into IRD:

Start of procedure

1. On the left side IRD main window (see [Figure 100](#)), click the Strategies icon. Clicking this icon enables `Import From File` on the File Menu.
2. Click the File menu and select `Import From File`.
3. In the resulting `Import of strategy` dialog box, select either `Samples.zcf` or `RLU.zcf` and click OK. The `Import To` dialog box opens (see [Figure 101](#) on [page 271](#)).

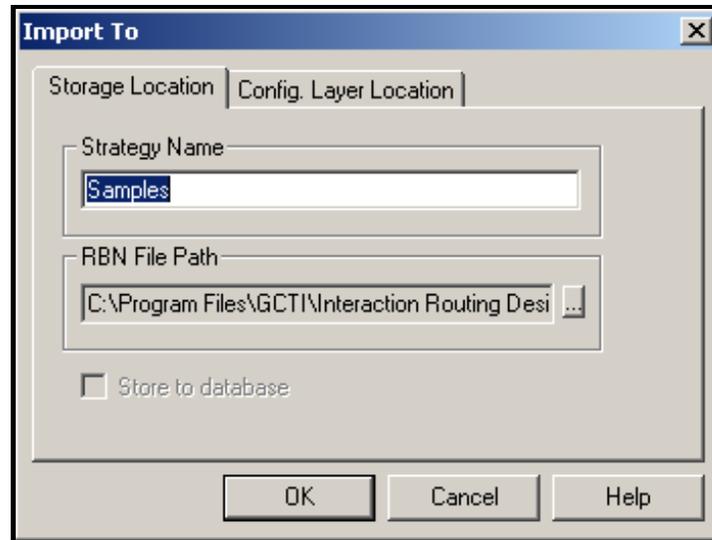


Figure 101: Import To Dialog Box, Storage Location

4. Do not change the entry in Strategy Name. It will contain either Samples or RLU.
5. In the RBN File Path field, keep the default selection or click the browse button (...) to change the location where the graphical (*.rbn) portion of the strategies/subroutines will be stored. The graphical portion of the strategy (used for display in IRD) requires more storage space than the strategy Script file, which is stored in the Scripts folder of Configuration Manager (Config Layer Location tab).
6. Click the Config Layer Location tab. [Figure 102](#) shows the dialog box.

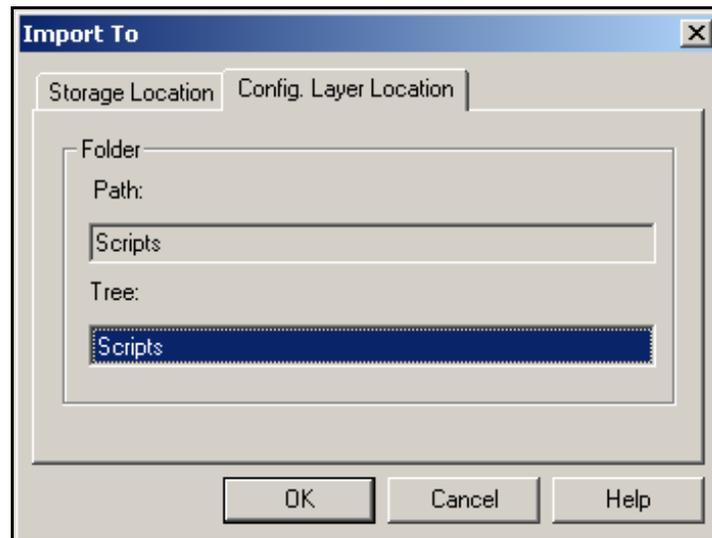


Figure 102: Import To Dialog Box, Config Layer Location

7. You have the option of changing the default storage folder from the `Scripts` folder to another folder under `Scripts`, which must already exist. Double-click to select the desired folder or keep the default.
8. When finished in the `Import To` dialog box, click `OK`. The `Importing operation for script` dialog box asks if you wish to proceed.
9. Click `OK` to proceed with the import operation.

End of procedure

Files in Samples.zcf

When you import `Samples.zcf` into IRD, you get the sample voice strategies, subroutines, and other objects listed in [Table 15](#).

Table 15: Sample Strategies in Samples.zcf

Name	Description
Strategies	
<code>cbr_enabled_sample</code>	Demonstrates one method for activating cost-based routing. Shows a simple way to cause URS to consider cost as additional target selection criteria without the need to modify existing strategies. For more information on cost-based routing, see the <i>Universal Routing 8.0 Routing Application Configuration Guide</i> .
<code>costbasedrouting_sample</code>	Demonstrates another way to activate a cost-based routing solution, as described in the <i>Universal Routing 8.0 Routing Application Configuration Guide</i> .
<code>getcity_sample</code>	Demonstrates how to use IRD's Web Service object in strategies to communicate with Web Services outside of Genesys. Documented in Appendix B of <i>Universal Routing 8.0 Reference Manual</i> .
<code>outbound_sample</code>	This type of strategy is used when running an Outbound campaign, such as when you have the Genesys Outbound Contact product installed. It demonstrates routing outbound related calls, specifically the ones that have a <code>CampaignGroup</code> target type.
<code>sla1_sample</code>	Demonstrates how to use <code>SetTargetsThreshold</code> function for Share Agent by Service Level Agreement routing. Documented in <i>Universal Routing 8.0 Routing Application Configuration Guide</i> .
<code>sla2_sample</code>	Demonstrates how to use <code>Routing Selection</code> object to build threshold expression for Share Agent by Service Level Agreement Routing. Documented in <i>Universal Routing 8.0 Routing Application Configuration Guide</i> .

Table 15: Sample Strategies in Samples.zcf (Continued)

Name	Description
OutboundMultiCampaign.000	<p>To support multi-Campaign agent management, the installation package includes pre-written strategy bytecode designed to route outbound calls/interactions to Campaign Groups. If you are a pure Genesys Outbound Contact customer, you do not have the rights to edit strategies. In this case, to activate automatic outbound routing:</p> <ul style="list-style-type: none"> -Load OutboundMultiCampaign.000 file into URS memory. Achieved by URS command line extension: -b default (see “Starting” on page 334). -Instruct URS to run OutboundMultiCampaign.000 strategy for every unloaded routing point by setting the URS option strategy to value: OutboundMultiCampaign. This option is described in the <i>Universal Routing 8.0 Reference Manual</i>. For information on the Campaign Group target in the Routing Selection and Route Interaction objects, see the section on Statistical Objects (Target Types) in the Interaction Routing Designer Objects chapter of the <i>Universal Routing 8.0 Reference Manual</i>.
Subroutines	
servicelevelagreement_sample	Demonstrates how to write a subroutine for Share Agent by Service Level Agreement Routing. Documented in <i>Universal Routing 8.0 Routing Application Configuration Guide</i> .
List objects	
businesslines_sample	Shows how a list object can be used for processing business lines. For information on how this type of list object is used, see “List Objects in SLA Routing” in the <i>Universal Routing 8.0 Routing Application Configuration Guide</i> .
CreditCardsSample	Shows a how a list object can be used for credit card processing. For information on how this type of list object is used, see “List Objects in SLA Routing” in the <i>Universal Routing 8.0 Routing Application Configuration Guide</i> .

Files in RLU.zcf

When you import RLU.zcf, you get the sample utility subroutines in [Table 16](#).

Table 16: Sample Subroutines in RLU.zcf

Name	Description
<p>rlu_mapc rlu_mapc_sample (selecting) May be used for (but is not limited to) load balancing</p>	<p>Accepts a customer-provided selecting procedure (CPSP) and a comma-separated list of arguments (targets). Applies the CPSP to every argument starting from the beginning of list and returns one of them or an empty string. The CPSP is a subroutine with three input parameters and one output parameter. It selects which argument (target), if any, will be returned (selects the best target). During every call of the CPSP:</p> <ul style="list-style-type: none"> • The first input parameter is 1 for initialization step and 0 for working step. • The second is set to the currently evaluated argument (target). • The third is set to the currently selected argument (current best target) or empty if there is none. <p>CPSP returns nothing on initialization (possible internal variables). After initialization, returns:</p> <ul style="list-style-type: none"> • Either the second or third input parameter (the best target) in the output parameter or an empty string if both are wrong. <p>Use Rlu_mapc to implement an alternative to using single statistics as a way to select targets. Recommended for cases when it is not possible to define selection criteria as a single statistic. Provided sample of CPSP (rlu_mapc_sample) selects the last target from the list. The sample always returns the currently evaluated target as the best target.</p>
<p>rlu_mapcar rlu_mapcar_sample (selecting)</p>	<p>This subroutine accepts a customer-provided replacing procedure (CPRP) and comma-separated list of arguments (targets). It applies the CPRP to every argument starting from the beginning of the list and returns comma separated list of results.</p> <p>The CPRP provides a replacement value for every argument (target). It is a subroutine with two input parameter and one output parameter: PROC[init, target]. During every call of the CPRP:</p> <ul style="list-style-type: none"> • The first input parameter is 1 for initialization step and 0 for working step. • The second is set to current argument. <p>CPRP returns nothing on the initialization step (possible internal variables). After initialization, returns:</p> <ul style="list-style-type: none"> • Replacement value in the output parameter. <p>Use to implement a different reformatting of a target list. CPRP (rlu_mapccar_sample) converts the target list into an increasing sequence of numbers starting from 1. For every target, it returns the sequential number of this target in the list.</p>

Table 16: Sample Subroutines in RLU.zcf (Continued)

Name	Description
rlu_mapc_group	<p>The subroutine is special case of the more general rlu_mapc subroutine described above. It accepts a customer-provided selecting procedure (CPSP) and a comma-separated list of Agent or Place Groups.</p> <p>The subroutine expands every Group into a list of Agents or Places. It then applies the CPSP to every Agent/Place (in the same way as rlu_mapc does) and returns one of them as a result of searching for the best Agent/Place through the set of Agent or Place Groups.</p> <p>The CPSP is a subroutine with three input parameters and one output parameter. It selects which argument (target) will be returned (selects the best target).</p> <p>During every call of CPSP:</p> <ul style="list-style-type: none"> • The first input parameter is 1 for initialization step and 0 for working step • The second input parameter is set to the currently evaluated argument (target). • The third input parameter is set to the currently selected argument (current best target) or empty if there is none. <p>The CPP must return:</p> <ul style="list-style-type: none"> • Nothing on initialization step (used to initialize possible internal variables). • Either the second or third input parameter (the best target) in the output parameter. <p>Rlu_mapc_group is just a shortcut to the use of rlu_mapc in the case where Agent/Place Groups are input, but the output need to be on the Agent/Place level. Using it allows you to skip the expand group step required in the case of rlu_mapc.</p>

Table 16: Sample Subroutines in RLU.zcf (Continued)

Name	Description
rlu_mapc_group_route	<p>This subroutine is a special case of the more general rlu_mapc_group subroutine above. Just like rlu_mapc_group, it accepts a customer-provided selecting procedure (CPSP) and a comma-separated list of Agent or Place Groups. It expands every Group into list of Agents or Places, applies the CPSP to every Agent/Place, and selects one of them (it searches for the best Agent/Place through the set of Agent or Place Groups).</p> <ul style="list-style-type: none"> • If selection is successful, the subroutine tries to send the call to the selected target. • If routing is successful, the subroutine returns the selected agent and continues through the green port; otherwise it continues through red port. <p>The CPSP is a subroutine with three input parameters and one output parameter. It select which argument (target) will be returned (selects the best target).</p> <p>During every call of CPSP:</p> <ul style="list-style-type: none"> • The first input parameter is 1 for the initialization step and 0 for the working step. • The second input parameter is set to the currently evaluated argument (target). • The third input parameter is set to the currently selected argument (current best target) or empty if there is none. <p>The CPSP must return:</p> <ul style="list-style-type: none"> • Nothing on the initialization step (used to initialize possible internal variables). • Either the second or third input parameter (the best target) in the output parameter. <p>Rlu_mapc_group_route is just a concatenation of rlu_mapc_group followed by routing on the selected target.</p>

Table 16: Sample Subroutines in RLU.zcf (Continued)

Name	Description
<p>rlu_select_max rlu_select_max_sample (selecting) May be used for (but is not limited to) load balancing</p>	<p>This subroutine extends a statistic-based selection of targets by replacing a custom statistic with a customer-provided selecting procedure (CPSP). It accepts a CPSP and a comma-separated list of arguments (targets).</p> <p>The subroutine applies the CPSP to every argument and returns the one for which the CPSP returns the largest value.</p> <p>The CPSP is a subroutine with one input parameters and one output parameter. It is responsible for providing a numeric value (“statistic”) that will be used to find the best target (see below).</p> <ul style="list-style-type: none"> • During every call of CPSP, its input parameter is set to the currently evaluated argument (target). • The CPSP must return a numeric value that will be used to find the best target in the output parameter. <p>Use Rlu_select_max for the same purposes as the rlu_mapc subroutine. Recommended for cases when target selection criteria can be expressed by a single number.</p> <p>The provided sample of CPSP (Rlu_select_max_sample) is equivalent to selecting a target by the StatTimeInReadyState statistic. For every target, the sample returns the value of this statistic.</p>

Table 16: Sample Subroutines in RLU.zcf (Continued)

Name	Description
<p>rlu_select_vq_route May be used for (but is not limited to) load balancing</p>	<p>The subroutine provides virtual queue-based target selection and routing. It accepts a comma-separated list of virtual queues, a customer-provided selecting procedure (CPSP), and a wait time. The subroutine applies the CPSP to select the best virtual queue for the current call.</p> <ul style="list-style-type: none"> • If selection is successful, the subroutine targets the call for all Agent Groups associated with the selected virtual queue through the origination DN property. Wait time defines how long to wait for agents from the selected virtual queue to become ready. • If routing is successful, the subroutine returns the selected virtual queue and continues through the green port. Otherwise, it continues through red port, clearing targets for the selected virtual queue. <p>The CPSP is a subroutine with three input parameters and one output parameter: PROC[init, next vq, currently best vq]. It is responsible for selecting best virtual queue.</p> <p>With every call, the CPSP selects the best one of two virtual queues provided in the parameters and return it as an output parameter (see below).</p> <p>During every call of CPSP:</p> <ul style="list-style-type: none"> • The first input parameter is 1 for an initialization step and 0 for a working step. • The second input parameter is set to the currently evaluated argument (virtual queue). • The third input parameter is set to the currently selected argument (virtual queue) or empty if there is none. <p>The CPSP must return</p> <ul style="list-style-type: none"> • Nothing on an initialization step (used to initialize possible internal variables). • Either the second or third input parameter (the best target) in the output parameter or an empty string if neither virtual queue is acceptable. <p>Use rlu_select_vq_route for group-based routing in cases when the selecting criteria is calculated not for the groups themselves, but for related virtual queues, i.e., group-based routing combined with queue-based target (group) selection.</p>

Interaction Workflow Samples

If you install eServices, with its Interaction Workflow Samples component, you also get sample strategies, subroutines, as well as other objects.

- In order to work with these objects, you must be able to open IRD's Interaction Design window.
- In order to open this window, the IRD main window must contain an Interaction Design shortcut bar as shown in Figure 100 on [page 270](#).

By default, an Interaction Design shortcut bar only appears if eServices components were installed and configured in your environment by the eServices Configuration Wizard. As a result of this installation, an eServices solution appears in Configuration Manager in the `Solutions` folder under `Environment`.

For information on displaying the shortcut bar, see [page 214](#).

Procedure:

Opening a business process in IRD's Interaction Design window

IRD's Interaction Design window is where you can view/edit the business processes, strategies, subroutines, and other objects included as part of the Interaction Workflow Samples component.

Start of procedure

1. In the IRD main window, click the Interaction Design shortcut bar.
2. Click the Business Processes icon. Existing business processes display in the list pane (see Figure 11 on [page 31](#)). The name of each business process describes its functionality (see "Interaction Workflow Samples Functionality" on [page 282](#) for names).
3. To view/edit an existing business process, double-click it. The business process opens in the Interaction Design window.
4. Expand the Strategies and Subroutines folders to view their names and icons in the left pane (see Figure 103 on [page 280](#)).

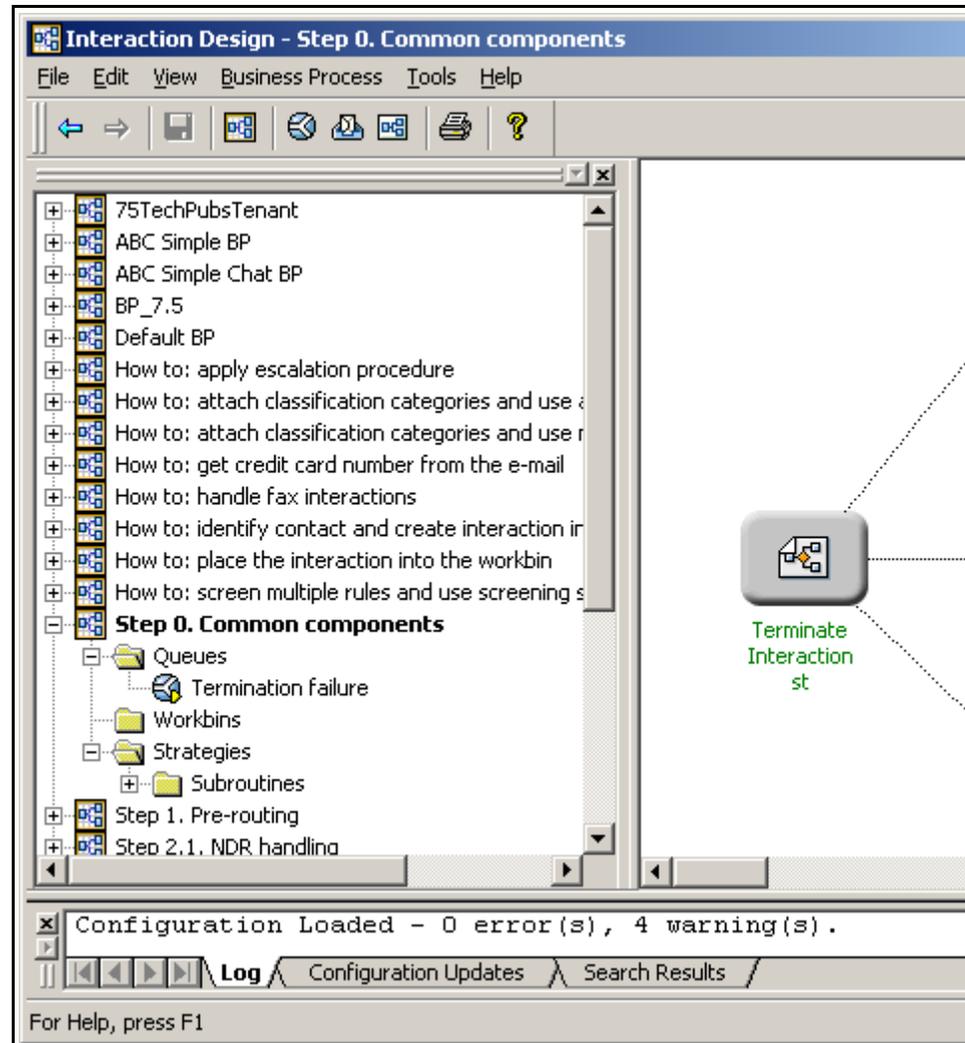


Figure 103: Interaction Design Window

- To open a strategy, right-click it in the right pane and select Edit/View Strategy from the shortcut menu.

Note: In order to view the strategy if you have access to its graphical portion (its *.rbn file). For more information, see “Graphical Portion of a Strategy” on [page 287](#).

In [Figure 103](#), E-mail service failure analysis st and Terminate Interaction st are both strategies. Termination failure is a queue. [Figure 104](#) on [page 281](#) shows the Terminate Interaction st strategy opened in the Routing Design window.

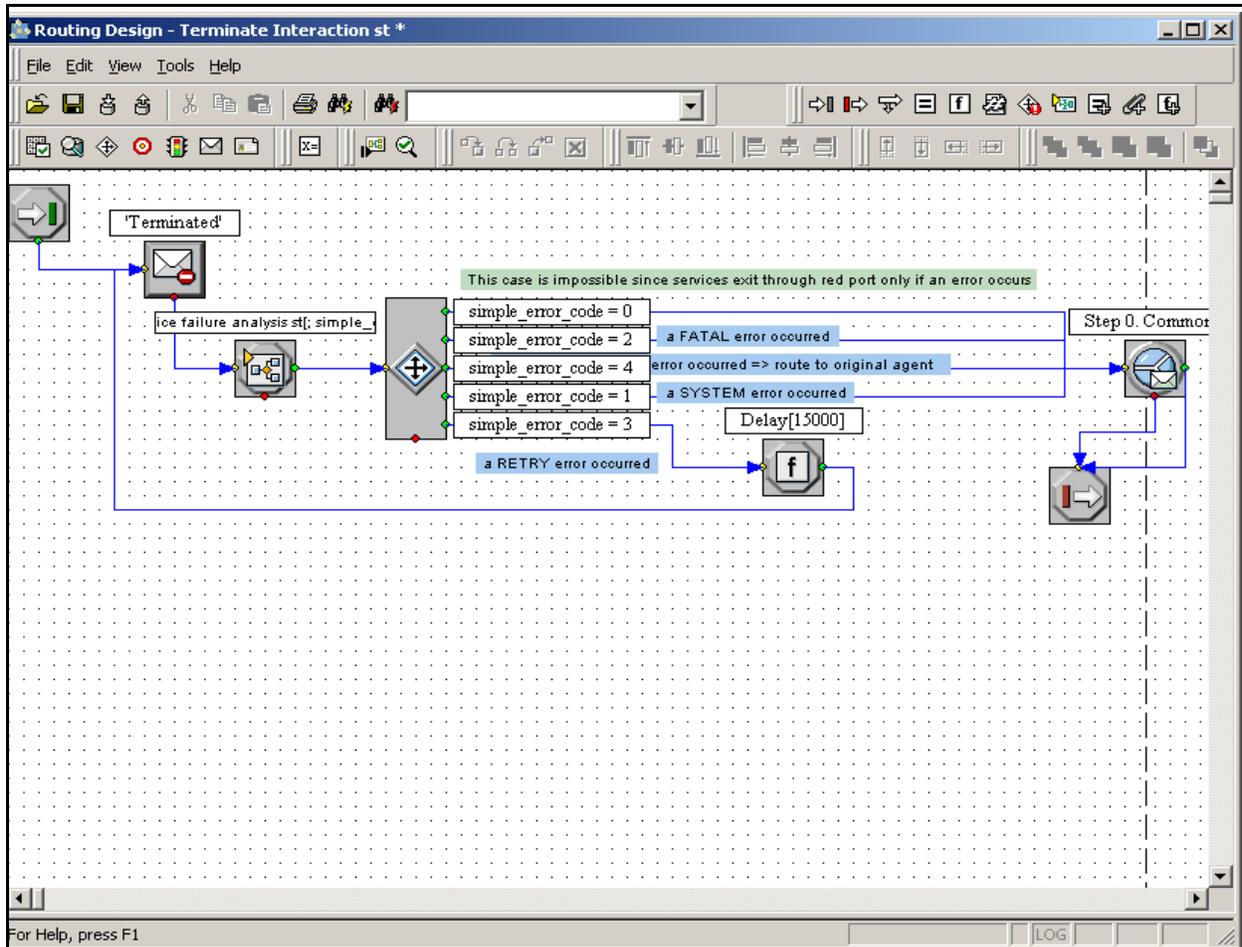


Figure 104: Sample Strategy in Routing Design Window

- Double-click an object to open its properties dialog box. For example, if you click the third object in Figure 104, the Call Subroutine properties dialog box opens (see Figure 105 on page 282).

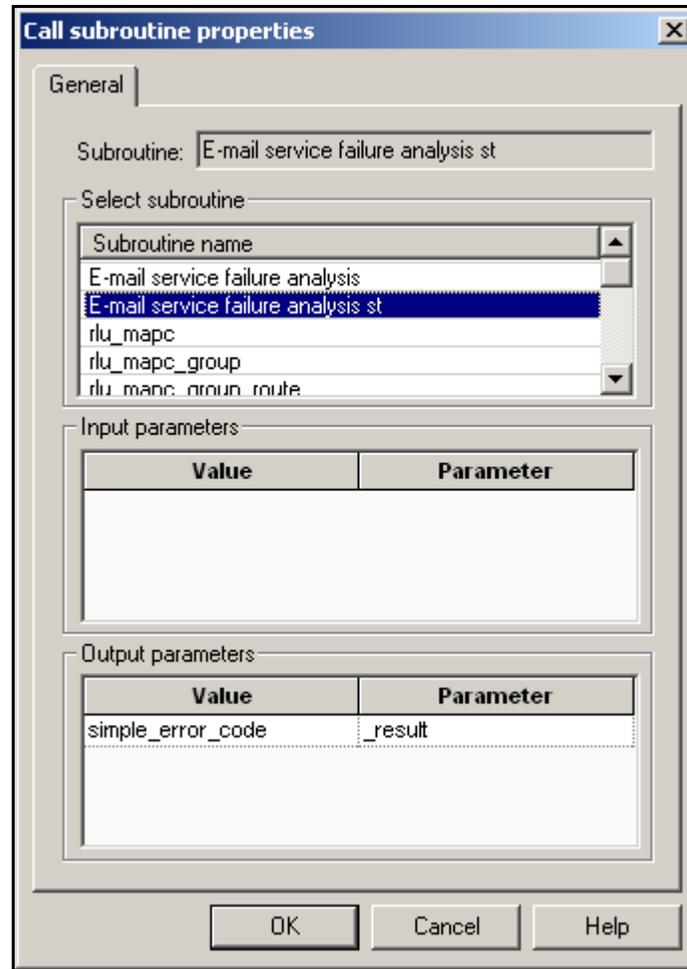


Figure 105: Sample Object Properties Dialog Box

7. To change to a different business process, right-click the business process and select `Open the Process` from the menu.

End of procedure

Interaction Workflow Samples Functionality

The Interaction Workflow Samples component supplies the following functionality:

- Pre-routing based on interaction sub-type
- Routing interactions to the original agent
- Screening of inbound interactions
- Attaching classification Categories
- Processing of attached data
- Redirecting interactions

- Forwarding interactions
- Collaboration reply sending
- Automatic treatment with an acknowledgement e-mail
- Autoresponse e-mail when applicable
- Placing interactions in workbins
- Escalating overdue interactions to supervisor workbins
- Routing to agents
- Assigning failure codes to interactions
- Promoting an interaction that failed pre-routing to the next process
- Routing interactions for QA review
- Skill-based review of agent response
- Re-processing interactions that failed QA review
- Quality control for outbound interactions based on screening
- Re-processing interactions that failed quality control
- Sending e-mail responses to customers
- Re-processing interactions that failed sending
- Stopping an interaction with a reason code
- Handling fax interactions
- Identify customer contacts and create interaction records
- Get credit card information from an interaction

Note: For information on each business process in the Interaction Workflow Samples component, see the *Universal Routing 7.6 (or later) Business Process User's Guide*.

Step-Numbered Business Processes

The smaller step-numbered business processes (connected via queues) isolate the various functionality found in a more complex business process called Default BP. Table 17 on [page 284](#) summarizes the functionality of each step-numbered business process.

Table 17: Step-Numbered Business Processes

BP and Functional Area(s)	Strategies and Objects Used	Interaction Queue(s)
<p>Step 0: Common Components</p> <p>These components are shared by the remaining step-numbered business processes</p>	<p>Terminate Interaction st strategy uses IRD objects Call Subroutine, Stop Interaction, Generic Segmentation, and Queue Interaction</p> <p>E-mail service failure analysis strategy uses IRD objects Multi-Assign, Assign, Multi-Attach, Function, Generic Segmentation, and If</p>	<p>Termination failure</p>
<p>Step 1: Pre-Routing</p> <p>Pre-routing based on interaction sub-type</p> <p>Promoting an e-mail that failed pre-routing to the next process</p>	<p>Inbound e-mail preprocessing str strategy uses IRD objects Generic Segmentation, Multi-Assign, and Queue Interaction</p> <p>Uses Interaction Subtype Business Attributes</p>	<p>Inbound e-mails, Collaboration reply e-mails, E-mails to route to original agent, Inbound e-mail failure, Chat inbound queue, Inbound e-mail postprocessing</p>
<p>Step 2.1. NDR Handling</p> <p>Route Interactions to original agent</p>	<p>Route Interactions to original agent st strategy uses IRD objects Multi-Assign, Assign, If, Queue Interaction, and Route Interaction</p>	<p>E-mails route to original agent, E-mails for QA review</p>
<p>Step 2.2. Inbound Collaboration Reply</p> <p>Agent collaboration</p> <p>Collaboration reply sending</p>	<p>Inbound collaboration reply processing st strategy uses IRD objects Reply From External Resource, Stop Interaction, Call Subroutine, Generic Segmentation, and Queue Interaction</p> <p>Terminate Interaction st strategy (see Step 0) and E-mail service failure analysis st (see Step 0)</p>	<p>E-mails for QA review, Termination failure, Collaboration reply e-mails, Collaboration reply failure</p>
<p>Step 2.3 New Inbound E-mails Handling</p> <p>Screening of inbound e-mails</p> <p>Automatic treatment with an acknowledgement e-mail</p> <p>Autoresponse when applicable</p> <p>Assigning failure codes</p>	<p>Preliminary e-mail screening st strategy uses IRD objects Screen, Call Subroutine, Generic Segmentation, Function, If, Queue Interaction, Stop, Autoresponse, and Acknowledgement</p> <p>Terminate Interaction st strategy (see Step 0) and E-mail service failure analysis st (see Step 0)</p>	<p>Inbound e-mail postprocessing, Redirect e-mail, Forward e-mails, Preprocessing failure, Termination failure, E-mails for processing by agents, Outbound e-mails</p>

Table 17: Step-Numbered Business Processes (Continued)

BP and Functional Area(s)	Strategies and Objects Used	Interaction Queue(s)
Step 3.1. Processing by Agents Routing to agents Processing by agents	E-mail distribution for processing st strategy contains IRD object Queue Interaction	E-mails for processing by agents, E-mails for QA review, Forward e-mails
Step 3.2: Routing E-mails for QA Review Processing of attached data Skill-based review of agent response Re-processing e-mails that failed quality control Re-processing e-mails that failed sending	Outbound e-mail 65x QA st strategy uses IRD objects Multi-Assign, Assign, Generic Segmentation, Route Interaction, Queue E-mail, If, and Function	E-mails for QA review, Quality Control, E-mails failed QA, Outbound e-mails
Step 3.3. Forwarding Stopping an e-mail with a reason code	Forward e-mail processing st strategy uses IRD objects Forward E-mail, Stop Interaction, Call Subroutine, Generic Segmentation, and Function Terminate Interaction st strategy (see Step 0) and E-mail service failure analysis st (see Step 0)	Forward e-mails, Termination failure, Forward e-mail failure, Outbound e-mails
Step 3.4. Redirecting Redirecting an e-mail	Redirect e-mail processing st strategy uses IRD objects Redirect E-mail, Stop Interaction, Function, Queue Interaction, Call Subroutine, and Generic Segmentation Terminate Interaction st strategy (see Step 0) and E-mail service failure analysis st (see Step 0)	Redirect e-mail, Termination failure, Redirect e-mail failure, Outbound e-mails
Step 4. Outbound Sending Quality control for outbound e-mails based on screening Sending e-mail responses to customers	Quality Control st strategy uses IRD objects Screen, If, Queue E-mail, Call Strategy, and Function Outbound e-mail sending st strategy uses IRD objects Send E-mail, Stop Interaction, Call Subroutine, Generic Segmentation, Function, Queue Interaction Terminate Interaction st strategy (see Step 0) and E-mail service failure analysis st (see Step 0)	Quality Control, Outbound e-mails, E-mail failure analysis, Termination failure, E-mails send error

How To: Business Processes

In addition to the step-numbered business processes just described, the Interaction Workflow Samples also supply various “How to” business processes (see Figure 11 on [page 31](#)). [Table 18](#) lists and describes the “How to” business processes.

Table 18: How To: Business Process Functionality

Functional Area	Strategies and Subroutines Used	Output Interaction Queues/Workbins
Apply escalation procedure (move interactions overdue for processing from an agent workbin to a supervisor workbin)	Move overdue interactions	Workbins: Supervisors workbin, Workbin for Original Agent
Attach classification Categories and use Attach Categories object	Classify customer inquiry strategy att-cc Subroutines: E-mail service failure analysis, Terminate Interaction	Queues: New inbound interactions, E-mails for QA Review, Preprocessing failure Workbin: E-mail distribution
Attach classification Categories and use Multi-Screen object	Classify customer inquiry Subroutines: E-mail service failure analysis, Terminate Interaction	Queues: New inbound interactions, E-mails for QA Review, Preprocessing failure Workbin: E-mail distribution
Get credit card numbers	Screen e-mail for credit card numbers	Inbound e-mail queue, Non-credit card payment, Paid with credit card
Handle fax interactions	Preliminary fax screening	Processing by agents, Forward interaction, Redirect interaction, Queue for responses
Identify whether a contact is new or existing and create an interaction record for a new contact	Identify contact and create interaction	Interactions with new contacts Interactions with existing contacts

Table 18: How To: Business Process Functionality (Continued)

Functional Area	Strategies and Subroutines Used	Output Interaction Queues/Workbins
Place an interaction into a workbin	Place the interaction into the workbin	E-mails route to original agent, E-mails for QA review Workbin: Workbin for original agent
Screen multiple rules	Preliminary e-mail screening ms Subroutines: E-mail service failure analysis, Terminate interaction	E-mails for processing by agents, Forward e-mails, Inbound e-mail postprocessing, Redirect e-mails, Preprocessing failure, Outbound e-mails, Termination failure

Graphical Portion of a Strategy

Use the information below if you try to open a strategy but get the message:
`<strategy_name> .rbn is not accessible.`

The script portion of a strategy and the graphical portion (.rbn file) may be stored in different locations. For example, while the script file is always stored in the Configuration Database, the graphical portion (which takes up more space) may be stored:

- In the Configuration Database in the `ird_strategies` table if an initialization script supplied by Genesys created the table.
- On a user's local drive.
- On a network drive.

Procedure:

Viewing the path for the .RBN file

Start of procedure

1. Double-click the strategy in the `Scripts` folder,
2. In the resulting properties dialog box, select the Annex tab. (If the tab does not appear, select `View > Options` from the Configuration Manager menu and select `Show annex tab in object properties.`)
3. Select the strategy section.
4. Double-click the path option (see [Figure 106](#)).

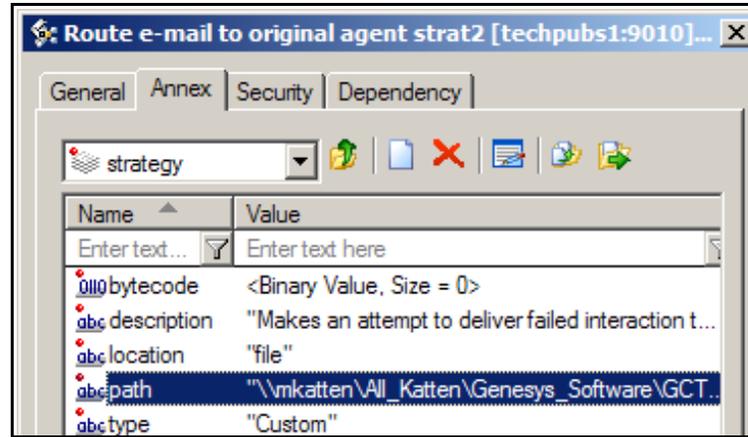


Figure 106: Strategy RBN File Path

End of procedure

Note: If you do not have access to the .rbn file, you will not be able to view a strategy in the Routing Design window.



Chapter

12 Configuring Custom Server

Note: While Custom Server can perform many types of customer-specific actions, this chapter concentrates on using Custom Server to retrieve data from non-SQL databases.

For systems that do not use standard SQL databases, Custom Server enables you to make database queries to send and receive data. Custom Server acts as a translator of a sort. When Universal Routing Server (URS) sends a request for database information, Custom Server translates the request for non-SQL databases and translates values returned by the database into a format that URS can read.

Note: URS 8.0 supports 6.1, 6.5, 7.x, and 8.x versions of Custom Server. Custom Server 8.x, 7.x, and 6.5.x can be configured either using the Custom Server Wizard or manually. Custom Server 6.1 can only be configured manually.

This chapter includes the following topics:

- [Configuring Custom Server, page 290](#)
- [Sample Procedure File, page 293](#)
- [Logging, page 305](#)
- [Custom Server FAQs, page 307](#)

Configuring Custom Server

Custom Server can be configured using the Custom Server Wizard or manually. Like Universal Routing Server, configuring Custom Server creates a Custom Server object in Configuration Layer.

Task Summary: Configuring Custom Server

Objective	Related Procedures and Activities
Configure the Custom Server Application object	Procedure: Creating and configuring the Custom Server Application object, on page 290
Configure a backup Custom Server	See “Backup Custom Server” on page 292
Modify the sample procedure file	Procedure: Modifying the sample procedure file on Windows, on page 293 or Procedure: Modifying the sample procedure file on UNIX, on page 294

Configuring Custom Server

Procedure: Creating and configuring the Custom Server Application object

Start of procedure

1. Open Configuration Manager.
2. Go to the Applications Template folder. Import the Custom_Server_801.apd Application Template from the Universal Routing CD.

Note: For information about importing Templates and creating Applications, refer to *Framework 8.0 Configuration Manager Help*.

3. Go to the Applications folder.
4. Create a new Application for Custom Server based on the imported Application Template. The Custom Server Properties dialog box opens.

5. Select the **General** tab and enter the **Application name**.
6. Make sure that the **State Enabled** check box is selected.
7. In a multi-Tenant environment, select the **Tenants** tab and set up the list of **Tenants** that use URS.
8. Select the **Server Info** tab, and select the following:
 - **Host**—the name of the host on which the Custom Server resides
 - **Port**—the port through which communication with Custom Server can be established. After you select a **Host**, a default port is provided for your convenience. You select the port and click **Edit Port** or you can configure a new port by clicking **Add Port**. Either action brings up the **New Port Info** dialog box (see Figure 79 on [page 196](#)).

Note: For information on using the **New Port Info** dialog box, see the **Port Info Tab** topic in the *Framework 8.0 Configuration Manager Help*.

- **Backup Server**—the backup Custom Server that should be used if the primary shuts down, if a backup server is configured
 - **Redundancy Type**—the standby mode
 - **Reconnect Timeout**—the time in seconds that Custom Server clients (URS) wait between reconnect attempts after their connection to the Custom Server fails
 - **Reconnect Attempts**—the number of attempts to connect to the server before trying to connect to the backup server (if configured)
9. Select the **Start Info** tab and specify the following:
 - **Working Directory**—the application location (example: `C:/GCTI/custom_server`)
 - **Command Line**—name of executable file (example: `custom_server.exe`)
 - **Command-Line Argument**—list of arguments to start the application (example: `-host <name of Configuration Server host> -port <name of Configuration Server port>-app <name of Custom Server Application>`)

If there is a space in the **Custom Server Application name**, then quotation marks are required before and after the name of the **Custom Server Application** object.

 - **Startup time**—the time interval the server waits until restart if the server fails.
 - **Shutdown time**—the time interval the server takes to shut down.
 - **Auto-Restart setting**—selecting this option causes the server to restart automatically if the server fails.
 - **Primary setting**—selecting this option specifies the server as the primary Custom Server.

10. Select the **Connections** tab. Specify all the servers to which Custom Server must connect as a client and (optionally) client-side port data.

Note: To support reconnecting to Configuration Server, you must still create or update the existing connection to Configuration Server in the Custom Server Application object's **Connections** tab. Follow the standard procedure for configuring connections to other servers. For specific instructions associated with client-side port connections, see the *Genesys 8.0 Security Deployment Guide*.

11. Select the **Options** tab and specify options. See the *Universal Routing 8.0 Reference Manual*, Chapter 5, Configuration Options, for Custom Server option descriptions.
12. When finished, click **OK**.
13. The **Custom Server Properties** dialog box closes.
14. If you wish to use the Management Layer and its Solution Control Interface (SCI) to stop and start applications such as Custom Server, you must install Local Control Agent (LCA). See the *Framework 8.0 Deployment Guide* for more information.

End of procedure

Backup Custom Server

Note: If you have purchased either the Redundancy Level package (see [page 79](#)) or the High Availability Routing package (see [page 85](#)), you can configure a backup Custom Server.

Configure a backup Custom Server as you would the primary Custom Server. A backup Custom Server can be configured in the **Not Specified** standby mode and used in case the primary Custom Server unexpectedly shuts down. A backup Custom Server accepts no requests and connections from URS. Only when the primary Custom Server shuts down and the backup is started does the backup accept requests and connections.

Solution Control Server switches the backup Custom Server to the primary Custom Server when the original primary Custom Server shuts down only if the backup Custom Server is started.

If the primary Custom Server shuts down, any Custom Server requests being processed at the time may be lost, resulting in the control of interactions being sent to the red port of the Database object in the IRD strategy. To avoid default routing in this situation, attach objects to the red port that will instruct URS how to route interactions affected by the shutdown.

Sample Procedure File

Custom Server acts as a translator between URS and the Router Custom Procedure (RCP) library you create from the `rcp.c` file in the Custom Server's directory. Set up a Custom Server according to the platform, Windows or UNIX.

Note: The `rcp.c` and `rcp.h` files are sample source files for customer RCP module creation, but are not a source of specified options

Windows Sample Procedure File

Follow the procedure below to modify the sample `rcp.c` and `rcp.h` files in a Windows operating system environment.

Procedure: Modifying the sample procedure file on Windows

Start of procedure

1. Navigate to the directory where the Custom Server executable resides. The `rcp.c` and some other files are located in the same directory. This directory is created by installation of Custom Server as an executable program, rather than as a configuration object.
2. Modify the sample `rcp.c` procedure file and the `rcp.h` file as necessary to fit your needs. See “About `rcp.c` (Router Custom Procedure)” on [page 295](#) for the contents of the `rcp.c` file. The `rcp.h` file is located in the directory created for Custom Server. See “Sample `rcp.h` File” on [page 296](#) for the contents of the `rcp.h` file.
3. Create a DLL file.
4. Copy the DLL file to where Custom Server is located.
5. Modify the configuration options in Custom Server's `Application` object.
6. Start URS.
7. Start Custom Server.

End of procedure

Note: The procedure name that needs to be specified is entirely dependent on the program the customer writes. The sample DLL, `RCP.dll`, does not care what is specified for a procedure name. The name can be anything, `edummi` for example. You can create a DLL that does or does not expect a specific procedure name to be used or not, just as you can specify or not specify that certain arguments be used.

UNIX Sample Procedure File

Follow the procedure below to modify the sample `rcp.c` and `rcp.h` files in a UNIX operating system environment.

Procedure: Modifying the sample procedure file on UNIX

Start of procedure

1. Modify the sample `rcp.c` procedure file and the `rcp.h` file as necessary to fit your needs. The `rcp.c` file is located in the directory created for Custom Server. See “About `rcp.c` (Router Custom Procedure)” on [page 295](#) for the contents of the `rcp.c` file. The `rcp.h` file is located in the directory created for Custom Server. See “Sample `rcp.h` File” on [page 296](#) for the contents of the `rcp.h` file.
2. Create a shared object from `rcp.c` and `rcp.h`. The shared object must have the name `libRCP.so` or `libRCP.sl` depending on your platform.

Note: You can have different Custom Servers (each one with their own library) on different computers or even on same computer but located in different folders.

- If you use GNU software, you may use the utility `make` and the provided file `makefile_unix_rcpso`. In the first line of `makefile_unix_rcpso`, replace

```
$(shell /release/bin/config.guess.short)
```

with the name of the particular platform as it appears in the later sections of the file. Then use the command line

```
make -f makefile_unix_rcpso so_lib
```

- If you create the shared object otherwise than in a), use the compilation and linking parameters listed in the section of `makefile_unix_rcpso` corresponding to your platform.

3. Copy the shared object to where Custom Server is located.
4. If necessary, modify the configuration options of the Custom Server Application.
5. Start URS.
6. Start Custom Server.

End of procedure

Next Steps

After you have started URS, you may check the log file for an event confirming that a connection has been established.

Note: Custom Server communicates either synchronously or asynchronously with created shared objects depending on the `async` option in the Custom Server Application.

Although Custom Server returns a single string to URS (the variable `buffer` in `rcp.c`), you may encode multiple key-value pairs in the buffer by using the following format:

`Key1:Value1|...|KeyN:ValueN`

Then, you can use this string in the Interaction Routing Designer functions `GetIntegerKey` and `GetStringKey` on this string to parse the values belonging to a given key.

For example:

`DD:99|AA:77|BB:88`

Refer to the line `strcpy (buffer, "DD:99|AA:77|BB:88");` in the `rcp.c` file.

About `rcp.c` (`R_CUSTOMER_PROCEDURE`)

Custom Server acts as a translator between URS and the `R_Customer_Procedure` (RCP) library you create from the `rcp.c` file. Parameters are passed to the `rcp.c` file as arguments using the Database Wizard object in IRD (see Figure 107 on [page 296](#)).

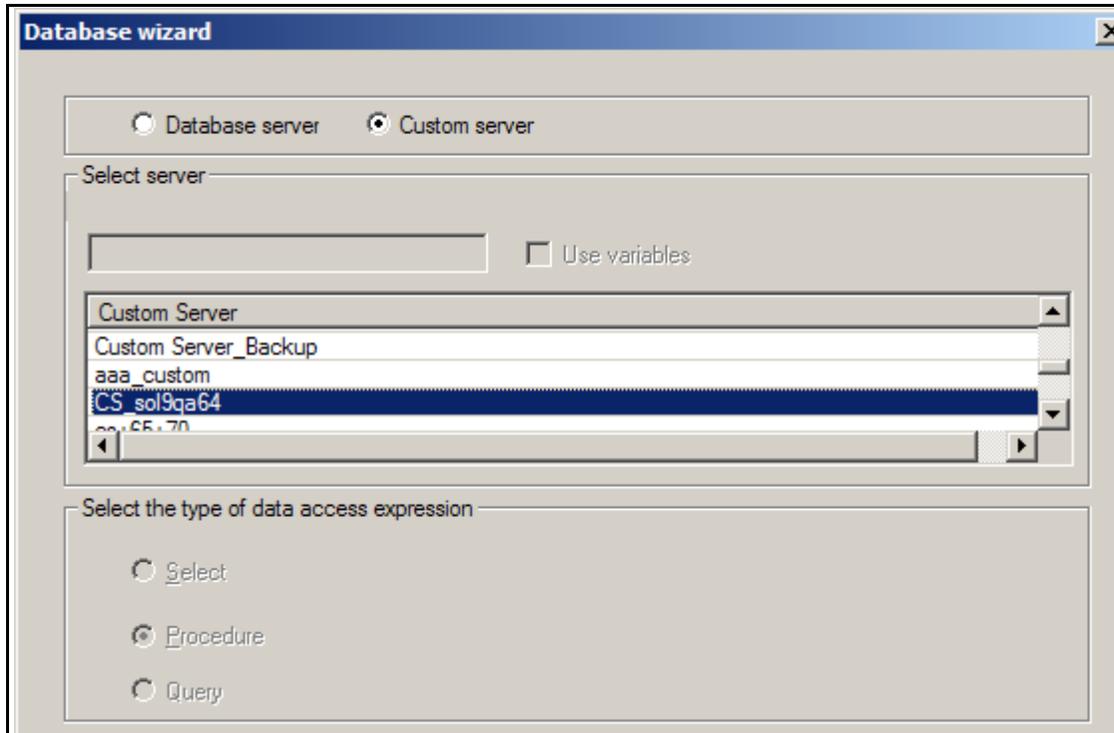


Figure 107: IRD Database Object, Configuring a Custom Procedure

Sample rcp.h File

The following is a sample of a complete rcp.h definition file:

```

/*
*****
THIS FILE AND RELATED RCP MODULE ARE PROVIDED AS A SAMPLE ONLY. THEY INTENTIONALLY WERE MADE VERY
SIMPLE TO BE EASY UNDERSTANDABLE AND TO DEMONSTRATE SOME BASIC PRINCIPLES OF RCP DLL/SO/SL CREATION
AND FUNCTIONING.
*****
*/

#ifndef _RCP_H_ /*{*/
#define _RCP_H_

#define RC_LOGINIT "LOGINIT"
#define RC_START "START"
#define RC_OPEN "OPEN"
#define RC_CLOSE "CLOSE"
#define RC_EXEC "EXECUTE"
#define RC_REPEAT "REPEAT"
#define RC_DISMISSED "DISMISSED"
#define RC_QUIT "QUIT"
#define RC_OPTIONS "OPTIONS"

#ifdef WIN32 /*{*/
#define RCP_API _declspec(dllexport)

```

```
#else /*}{*/
#define RCP_API extern
#endif /*}{*/

/**{ EXTERNAL FUNCTIONS *****/
RCP_API int R_CUSTOMER_PROCEDURE( int arg_c, char * arg_v[], char buffer[], int refID );
/**} EXTERNAL FUNCTIONS *****/

/**{ COMMON LOG FACILITIES *****/
/*
The Custom Server log subsystem is guaranteed to be fully initialized on RC_START call. However the
provided common log function can be used on earlier steps, i.e. on RC_LOGINIT and subsequent calls.
```

To be able to control details level of customer output with the Custom Server log options, four messages with different types (DEBUG, INTERACTION, TRACE, and STANDARD) have been provided. Please do not change the predefined log messages IDs. Only those messages IDs are allowed to be used. In case of an ID mismatch the CS_RCPLLOG_STANDARD identifier will be in effect.

A presence of the CustomServer.lms file is a required condition for successful of the common log function.

Because of uncertainty of customer output very common message format ("%s") is used. So a message string should be composed first before logging with the provided function.

The common log function is thread-safe.

```
*/
typedef void (* P_COMMON_LOG_FUNC)( int iLogMesID, char * pMesStr );
extern P_COMMON_LOG_FUNC pCommonLogFunc;
#define COMMON_LOG_FUNC( iLogMesID, pMesStr ) {if( pCommonLogFunc ) (*pCommonLogFunc)( iLogMesID),
(pMesStr) );}
/*Predefined log messages IDs (please, do not change them)*/
#define CS_RCPLLOG_DEBUG                21997 /* %s */
#define CS_RCPLLOG_INTERACTION          21998 /* %s */
#define CS_RCPLLOG_TRACE                21999 /* %s */
#define CS_RCPLLOG_STANDARD             22000 /* %s */
/**} COMMON LOG FACILITIES *****/

#endif /*}{*/
```

Sample rcp.c File

The following is a sample of a complete **rcp.c** custom procedure file:

```
/*
*****
THIS FILE AND RELATED RCP MODULE ARE PROVIDED AS A SAMPLE ONLY.
THEY INTENTIONALLY WERE MADE VERY SIMPLE TO BE EASY UNDERSTANDABLE AND TO DEMONSTRATE SOME BASIC
PRINCIPLES OF RCP DLL/SO/SL CREATION AND FUNCTIONING.
*****
*/
```

```

#include <stdio.h>
#include <string.h>
#include <time.h>

#include "RCP.h"

#ifdef WIN32 /*{*/
#undef RCP_API
#define RCP_API
#endif /*}*/

/**{ COMMON LOG FACILITIES *****/
/* Please, see comments in the RCP.h file */
P_COMMON_LOG_FUNC pCommonLogFunc = NULL;
/**} COMMON LOG FACILITIES *****/

/*{ FOR TEST PURPOSES *****/
void DumpToFile( char * pStr, int nFlush );
void PrintOptions( int arg_c, char * arg_v[] );
/*} FOR TEST PURPOSES *****/

/*****/
RCP_API int R_CUSTOMER_PROCEDURE( int arg_c, char * arg_v[], char buffer[], int refID )
{
  /*****/
  static int async      = 0;
  static int lifetime   = 0;
  static int frequency  = 0;
  static int buffersize = 0;
  /*****/

  /*****/
  if( !strcmp( arg_v[0], RC_LOGINIT ) )
  {
    pCommonLogFunc = (P_COMMON_LOG_FUNC) arg_v[1];
    COMMON_LOG_FUNC( CS_RCPCLOG_DEBUG,      "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
    COMMON_LOG_FUNC( CS_RCPCLOG_INTERACTION, "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
    COMMON_LOG_FUNC( CS_RCPCLOG_TRACE,      "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
    COMMON_LOG_FUNC( CS_RCPCLOG_STANDARD,   "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
    return( 1 );
  }
  /*****/
  if( !strcmp( arg_v[0], RC_START ) )
  {
    async = (strcmp( arg_v[1], "async=1" ) == 0)? 1: 0;
    sscanf( arg_v[2], "lifetime=%d", &lifetime );
    sscanf( arg_v[3], "frequency=%d", &frequency );

    /*for compatibility with old versions of custom_server*/
    if( arg_c > 4 )
      sscanf( arg_v[4], "buffersize=%d", &buffersize );
  }
}

```

```

else
    buffersize = BUFSIZ;

fprintf( stderr, "\n*** RCP SAMPLE: START REACTION ***\n" );
DumpToFile( "*** RCP SAMPLE: START REACTION ***", 0 );
PrintOptions( arg_c, arg_v );
COMMON_LOG_FUNC( CS_RCPCLOG_DEBUG,      "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
COMMON_LOG_FUNC( CS_RCPCLOG_INTERACTION, "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
COMMON_LOG_FUNC( CS_RCPCLOG_TRACE,      "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
COMMON_LOG_FUNC( CS_RCPCLOG_STANDARD,    "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
return( 1 );
}
/****options have been changed*****/
if( !strcmp( arg_v[0], RC_OPTIONS ) )
{
    async = (strcmp( arg_v[1], "async=1" ) == 0)? 1: 0;
    sscanf( arg_v[2], "lifetime=%d", &lifetime );
    sscanf( arg_v[3], "frequency=%d", &frequency );
    sscanf( arg_v[4], "buffersize=%d", &buffersize );

    fprintf( stderr, "\n*** RCP SAMPLE: OPTIONS REACTION ***\n" );
    DumpToFile( "*** RCP SAMPLE: OPTIONS REACTION ***", 0 );
    PrintOptions( arg_c, arg_v );
    COMMON_LOG_FUNC( CS_RCPCLOG_DEBUG,      "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
    COMMON_LOG_FUNC( CS_RCPCLOG_INTERACTION, "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
    COMMON_LOG_FUNC( CS_RCPCLOG_TRACE,      "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
    COMMON_LOG_FUNC( CS_RCPCLOG_STANDARD,    "*** RCP SAMPLE: OUTPUT TO COMMON LOG ***" );
    return( 1 );
}
/******/
if( !strcmp( arg_v[0], RC_QUIT ) )
{
    fprintf( stderr, "\n*** RCP SAMPLE: QUIT REACTION ***\n" );
    DumpToFile( "*** RCP SAMPLE: QUIT REACTION ***", 1 );
    return( 1 );
}
/******/
if( !strcmp( arg_v[0], RC_OPEN ) )
{
    fprintf( stderr, "\n*** RCP SAMPLE: OPEN REACTION ***\n" );
    DumpToFile( "*** RCP SAMPLE: OPEN REACTION ***", 0 );
    return( 1 );
}
/******/
if( !strcmp( arg_v[0], RC_CLOSE ) )
{
    fprintf( stderr, "\n*** RCP SAMPLE: CLOSE REACTION ***\n" );
    DumpToFile( "*** RCP SAMPLE: CLOSE REACTION ***", 0 );
    return( 1 );
}
}

```

```

/*****/
if( !strcmp( arg_v[0], RC_DISMISSED ) )
{
    fprintf( stderr, "\n*** RCP SAMPLE: DISMISSED REACTION ***\n" );
    return( 1 );
}
buffer[0] = 0;

/*****/
if( !strcmp( arg_v[0], RC_EXEC ) )
{
    if( async )
    {
        fprintf( stderr, "\n"
"*** RCP SAMPLE: EXECUTE(async.mode) REACTION (refID <%d>) ***\n"
"*** request with refID <%d> will be waiting during <%d> msec\n"
"*** but no longer than <%d> sec ... \n",
refID, refID, frequency, lifetime );
        return( 0 );
    }
    else
    {
        strncpy( buffer, "agent@stat.A", buffersize );
        fprintf( stderr,
"\n*** RCP SAMPLE: EXECUTE(sync.mode) REACTION (refID <%d>) ***\n",
refID );
        return( 1 );
    }
}

/*****/
if( !strcmp( arg_v[0], RC_REPEAT ) )
{
    strncpy( buffer, "agent@stat.A", buffersize );
    fprintf( stderr,
"\n*** RCP SAMPLE: REPEAT(async.mode) REACTION (refID <%d>) ***\n",
refID );
    return( 1 );
}
return( 1 );
} /*end of R_CUSTOMER_PROCEDURE*/

/*{ FOR TEST PURPOSES *****/

/*
This function is provided for testing of CustomServer working as a service that is not allowed to
interact with desktop (>> CustomServerLogFile.log).
*/
void DumpToFile( char * pStr, int nFlush )
{
/*****/

```

```

FILE * stream = NULL;
/*****/

if( (stream = open( "CustomServerLogFile.log", "a+" )) )
{
    time_t aclock;
    struct tm * newtime;
    time( &aclock );
    newtime = localtime( &aclock );
    fprintf( stream, "\n%s%s\n", asctime( newtime ), pStr );
    if( nFlush )
        fflush( stream );

    fclose( stream );
}

return;
} /*end of DumpToFile*/

void PrintOptions( int arg_c, char * arg_v[] )
{
/*****/
int i;
/*****/

fprintf( stderr,
        "Native CustomServer options (with default options)\n"
        "and alien options defined in CustomServer section:\n"
        "-----\n" );
for( i = 1; i < arg_c; i++ )
    fprintf( stderr, " %s\n", arg_v[i] );

fprintf( stderr,
        "*****\n" );
return;
} /*end of PrintOptions*/

/*} FOR TEST PURPOSES *****/

```

Arguments in the rcp.c File

The arguments for the `R_CUSTOMER_PROCEDURE` function in the `rcp.c` file are explained in [Table 19](#). The function is `int R_CUSTOMER_PROCEDURE(argc, argv, buffer, refID)`

Table 19: rcp.c File Arguments

Format	Argument	Explanation
int argc	argc	The number of arguments passed.
char* argv[]	argv[0]	The type of request. Values: OPEN = initialization CLOSE = finishing work EXECUTE = execute query REPEAT = check if data is ready DISMISSED = request is no longer valid LOGINIT = initialize custom logging functionality
char* argv[]	argv[1]	Values: The procedure name as specified in the Database object function in the strategy. The address of the print function if the request is LOGINIT . In this case, argv[1] is an address, not a string and should not be handled as a string. In other words, do not attempt to print the content or perform any other string-specific actions.
char* argv[]	argv[2]... argv[n]	Parameters key=value in a format as specified in Database object function.
char* buffer[1024]	buffer	The buffer for the data passed from the custom procedure to the Custom Server and URS.
int refID	refID	The request reference ID generated by the Custom Server.

Synchronous Versus Asynchronous Mode

Since Custom Server is a one-thread application, it is unable to make more than one DLL call at a time. For example, two calls hit the route point simultaneously. URS will subsequently make two requests to Custom Server. Custom Server will retrieve the first request from the socket and make the DLL call. After DLL function returns a value (not necessarily the result), Custom Server will retrieve the next request from the socket and make the next DLL call, and so on.

If Custom Server works in synchronous mode, then it can process only one data request at a time. All other pending requests will be queued on its socket until their time comes.

If Custom Server works in asynchronous mode, it is not necessarily to return data on the first DLL call. Instead, you can create a new thread on `RC_EXEC` (see “Sample `rcp.h` File” on [page 296](#)), that would retrieve data and store it in a table or a list accessible from the main thread. Custom Server will call `RC_EXEC`, that will spawn a thread, and return. Now it can process new requests.

Periodically, it will be calling `RC_REPEAT` and that can check whether the result for the passed `refID` is available yet. If the result is ready, it will return it, otherwise it will be called again after time specified in the `frequency` option expires.

If the `life_time` expires and the result is not available yet, `RC_DISMISSED` is called and the thread must be terminated.

All these actions, such as creating a new thread and checking for the result in the table, do not take much time, so Custom Server can process a fairly large amount of data in asynchronous mode.

Calling a Custom Procedure in Synchronous Mode

When URS is connected to Custom Server, a custom procedure with `argv[0] = "OPEN"` is called. When URS has disconnected from Custom Server, custom procedure with `argv[0] = "CLOSE"` is called.

When URS requests data from Custom Server, a custom procedure with `argv[0] = "EXECUTE"` is called. If the custom procedure returns 1, the contents of the buffer are sent back to URS. If the custom procedure returns 0, no data is sent back. [Table 20](#) contains the return values in synchronous mode.

Table 20: Return Values in Synchronous Mode

argv[0] Value	Return Value	Notes
"OPEN"	0	None
"CLOSE"	0	None
"EXECUTE"	1	If data should be passed back to URS
"EXECUTE"	0	If data should not be passed back to URS

Calling a Custom Procedure in Asynchronous Mode

When URS is connected to Custom Server, a custom procedure with `argv[0] = "OPEN"` is called. When URS has disconnected from Custom Server, a custom procedure with `argv[0] = "CLOSE"` is called.

When URS requests data from Custom Server, a custom procedure with `argv[0] = "EXECUTE"` is called and a unique request ID is passed to the custom procedure in the `refID` parameter. If the custom procedure returns 1, then the contents of the buffer are immediately sent back to URS. If a custom procedure returns 0, the Custom Server calls the custom procedure again after the timeout specified in the `frequency` option expires until the custom procedure returns 1 or until the timeout specified in the option `life_time` expires. Every time URS requests data, a valid request ID is sent to the Custom Server. If the `life_time` option has expired, the custom procedure with `argv[0] = "DISMISSED"` is called, indicating that the Custom Server will no longer wait for data. [Table 21](#) contains the return values in asynchronous mode.

Table 21: Return Values in Asynchronous Mode

argv[0] Value	Return Value	Notes
"OPEN"	0	None
"CLOSE"	0	None
"EXECUTE"	1	If data is ready to be passed back to URS
"EXECUTE"	0	If Custom Server should wait until data is available
"REPEAT"	1	If data is ready to be passed back to URS
"REPEAT"	0	If Custom Server should wait until data is available
"DISMISSED"	0	None

Output Formats

Although Custom Server returns a single string (the variable `buffer` in `rcp.c`) to URS, you may encode multiple key-value pairs in the buffer by using the following format:

```
Key1:Value1|...|KeyN:ValueN
```

Use this string in the IRD functions `GetIntegerKey` and `GetStringKey` to parse the values belonging to a given key.

Logging

Custom Server implements the standard logging options that other Genesys applications use. The following function calls between URS and Custom Server are logged by Custom Server using this standard logging method:

- CMDStart
- CMDOpen
- CMDClose
- CMDExecute
- CMDRepeat
- CMDDismissed
- CMDQuit
- CMDUnknown

The events logged for these functions in the Custom Server log include passed parameters and timestamps.

Custom Server Automatically Generated Log

The log automatically generated by Custom Server records all calls of the RCP library. All invocations of `R_CUSTOMER_PROCEDURE` function from the `rcp.c` file are recorded into the Custom Server log. That includes the `LOGINIT` (starting from 7.5), `START`, `OPEN`, `CLOSE`, `EXECUTE`, `REPEAT`, `DISMISSED`, `QUIT` and `OPTIONS` commands. Logged information includes timestamps and passed parameters (where applicable) so there is no necessity to additionally log this data with User-Implemented Logging as described below.

User-Implemented Logging

Note: Starting with 7.5, Genesys recommends use of “[Custom Server Automatically Generated Log](#)” for user implemented logging. Consider any private logging mechanisms that you implement as a backup to this recommended way of logging.

Logging plays an important part in troubleshooting. Without a complete beginning-to-end picture of the call your ability to find problems is severely limited.

Note: Logging functionality can also be implemented by the end user (which Custom Server allows). The reason for this is that Custom Server is a very customer-specific application; exactly how it behaves is very much determined by the end user (customer). Because of this, it is not clear in advance what information will be important for searching out problems in the customer's code. So the customer must implement what to log and how.

The following points are important to know regarding user-implemented logging:

In essence, Custom Server functions as a media layer between URS and the user-developed library. Custom Server's responsibility is just to:

- Answer to URS requests.
- Transfer URS requests to the user-developed library.
- Transfer data from a user-defined function back to URS.

As in previous releases, it is possible to implement your own logging mechanism in your user-developed library where all aspects of logging can be controlled:

- Creating and opening a log file during the user-developed library initialization process
- Printing all information that is considered useful into this log file
- Controlling log file parameters (size, etc.)

Note: Custom Server can print initialization information as well as information about all abnormal situations (example: URS disconnections) into standard C++ streams (`stdout`, `stderr`), which can be redirected into a disk file by OS commands.

User-Defined Strings in Custom Server Log

Alternatively, you can use the logging functionality supported by Genesys Framework. To make this possible Custom Server functionality was extended in the 7.5 release to allow user-defined strings to be written to the Custom Server log. The command, `LOGINIT`, was introduced for the `R_CUSTOMER_PROCEDURE` for this purpose. This command is defined as `RC_LOGINIT` in the `rcp.h` file. The `argv[1]` parameter of this call contains the address of the log function. In order to use this functionality, the address must be stored for future use. Users who are not interested in this functionality may simply ignore this `R_CUSTOMER_PROCEDURE` call. In order to have messages

printed in the Custom Server log, you need to prepare text (string format) for output and call the function by this address.

Refer to the section “Sample rcp.h File” on [page 296](#) for more information.

The common log function is defined in `rcp.h` as:

```
typedef void (* P_COMMON_LOG_FUNC)(int iLogMesID, char * pMesStr);
```

`LogMesID` controls the level of detail output to the log file with the Custom Server log options. Four messages with different types (DEBUG, INTERACTION, TRACE, and STANDARD) are provided in `CustomServer.lms`. Their corresponding log messages IDs (`CS_RCPLLOG_DEBUG`, `CS_RCPLLOG_INTERACTION`, `CS_RCPLLOG_TRACE`, `CS_RCPLLOG_STANDARD`) are provided in `rcp.h`.

Important Notes

- You can only use the above message IDs. Any other ID will be replaced with `CS_RCPLLOG_STANDARD`.
- The presence of the `CustomServer.lms` file is a required condition for successful work of the common log function.
- The common log function is unaffected by the setting of Custom Server option `hide_private_data`.

Custom Server FAQs

This section presents some Frequently Asked Questions (FAQs) on deploying Custom Server and answers to them.

Can URS call methods contained in `rcp.dll`?

Is it possible to have differently named methods in the `rcp.dll` and be able to call them from URS? I have declared several new methods in my `rpc.c` source code and it successfully compiles. These methods are declared as `RCP_API` methods and have, aside from the method name, the same parameters as the `R_CUSTOMER_PROCEDURE` method.

However, when I try to call this from my routing strategy, it does not work. Instead, it seems to only call the `R_CUSTOMER_PROCEDURE` method.

I could use the `R_CUSTOMER_PROCEDURE` as my entry point and use parameters to decide which method to call, but I would still like to have this question answered. I assume that the `R_CUSTOMER_PROCEDURE` method has to be there anyway, as removing it means that Custom Server no longer starts.

Answer

`R_CUSTOMER_PROCEDURE` is purely a virtual method and is the only method that the Custom Server stub will call. You can declare as many other functions as

you like with the RCP_API return signature, but they won't get called as there is no way to set up a mapping for the custom server stub code.

One approach is to have a single R_CUSTOMER_PROCEDURE and then pass across a KVP with a function key to determine which procedure to finally call. This raises the possibility of strategy misconfiguration and so you will need to make sure that you have a sensible default behavior and error recovery.

The other approach is to create multiple Custom Server application instances, each with their own supported functionality. That way you can give each one its own behavior and configuration and can put the decision about which function to call back into the strategy and the Custom Server call itself. You can define each one by name and expose it in the strategy. From a maintenance perspective, this approach also has some advantages.

Can the DLL be multi-threaded?

I was considering making the DLL file (see “Modifying the sample procedure file on Windows” on [page 293](#)) multi-threaded, as the incoming call load is potentially quite high. The DLL needs to make a call through a C++ interface to a Java back-end system (using JINI). We will work on the interface to try and optimize the connection/request/disconnection process of any client, but will a high volume of calls put too big of a load on the system?

Answer

If the dynamics of the system are that you will have a high volume of calls and that you need to decouple the JINI back end and minimize connects/disconnects, then consider building a Custom Server that operates in asynchronous mode and internally queues the requests into a call queue. Then have a dispatcher thread that read items from the request queue and spawns threads for each interaction: they could submit the request and wait synchronously for the response. The interaction threads could write the final response of the call to the call queue which would, in turn, respond to status requests/cancellations to return the interaction result to the Custom Server call (which polls automatically in async mode). The goal is to manage as many requests as possible from URS and to decouple the response of the external calls from the throughput of the system.

How do I get the number of idle and used trunk lines?

Answer

Genesys can't directly monitor the trunks. However you can query the switch to tell you the status of a trunk group. On the G3, you can perform a query to get the number of used and idle trunks in a trunk group. To do this, perform an address query specifying 14 as the address infotype and specify the trunk

access code as the `thisDN`. You will receive the trunk information in the Attribute Extensions. In order to allow URS to use this, you will probably need to create a Custom Server.

Does Load Distribution Server Support Custom Server?

By doing a migration, we wish to replace our current routing engine with Genesys ERS and Universal Routing Server. In order to avoid complicating the migration (12 sites and 3000 agents), the first step consists in implementing all the routing logic defined in our current product in a Custom Server procedure called by ERS in a strategy.

We know that ERS supports LDS but what about Custom Server? Would it be possible to use several Custom Server procedures with the intention to split the load? Is there an LDS mechanism for Custom Server or are there other solutions to split the load for Custom Server.

Answer

Genesys does not provide automatic support of LDS with Custom Server. Possible alternative solutions include using multiple URSs, each with its own Custom Server, encoding load distribution in Custom Server itself, or explicitly control Custom Server loading sharing via the routing strategy (add two Custom Servers to the URS Application Connection List in Configuration Manager and use a `SCRIPT` variable to control which Custom Server to access).

Can I run more than one Custom Server on the same host?

Is it possible to run more than one instance of Custom Server on the same server (host)? If it is possible, will it impact production?

Answer

You would need to configure each Custom Server with its appropriate host/port combination and assign the appropriate Application name to each one. Once this is done, you should be able to reference each Custom Server by name. The important thing is that the ports must be unique if they are on the same host.

Will it impact production? If Custom Server fails due to outside causes (network failure, etc.), there could be an impact the rest of the live URS installation. In theory, you should just lose your current interaction, but you probably would not want to risk production and test servers connected to my live URS.

As a safer bet, you could run a second test URS. This would be safer even if a true test environment is not deployed. At least this way the active components are independent of live routing altogether.

Can Custom Server trigger an external program?

Could the program name be passed as a parameter?

Answer

There is no such function in IRD/URS. However this can be easily done using Custom Server. Simply modify the sample procedure code to pass the name of the executable as a parameter and start it from there.

Note: Genesys does not offer support on coding within Custom Server.

Can I send a page (or e-mail) as part of a strategy?

Our routing rules send calls to voice mail under certain conditions. We want to page the supervisor of the group responsible for that queue when calls go to voice mail.

Answer

Custom Server can easily do this. An alternative would be to create an extended stored procedure and have the strategy call the procedure (the stored procedure can create the e-mail).



Chapter

13

Installing a Routing Solution

Before you install Universal Routing, you must:

- Install and configure the required Framework components: T-Server, Stat Server, DB Server, Network T-Server (Network Routing only), and the Configuration Layer and Management Layer components.
- Configure Enterprise Routing or Network Routing manually (as described in Chapter 8 on [page 181](#)).

When you have completed the preceding tasks, you are ready to install the configured routing solution as described in this chapter.

This chapter includes the following topics:

- [Installation Package Location, page 312](#)
- [Installing on Windows Operating Systems, page 313](#)
- [Installing on UNIX, page 328](#)

Note: For a list of supported operating systems and databases, see *Genesys Supported Operating Environment Reference Manual*, which is available on the Genesys Technical Support website at <http://genesyslab.com/support>.

Warning! Genesys does not recommend installation of its components via a Microsoft Remote Desktop connection. The installation should be performed locally.

Task Summary: Installing a Routing Solution

Objective	Related Procedures and Activities
Locate the installation package	See “Installation Package Location” on page 312
Install components on Windows	<ol style="list-style-type: none"> 1. Procedure: Installing URS on Windows using the Installation Wizard, on page 313 2. Procedure: Installing IRD on Windows using the Installation Wizard, on page 321 3. Procedure: Installing Custom Server on Windows using the Installation Wizard, on page 322
Install components on UNIX	<ol style="list-style-type: none"> 1. Procedure: Installing Universal Routing Server on a UNIX platform, on page 328 2. Procedure: Installing Custom Server on UNIX, on page 330

Installation Package Location

The installation package, whether on CD or from an FTP site, contains three setup folders: one for URS, one for IRD, and one for Custom Server.

On the Universal Routing CD, the setup file is located as follows:

- For URS, find the file in the folder named for your operating system:
 - \\solution_specific\universal_routing_server\aix
 - \\solution_specific\universal_routing_server\hp-ux
 - \\solution_specific\universal_routing_server\linux
 - \\solution_specific\universal_routing_server\solaris
 - \\solution_specific\universal_routing_server\tru64UNIX
 - \\solution_specific\universal_routing_server\windows.
- For IRD, the file is in the folder:
 - solution_specific\interaction_routing_designer\windows

- For Custom Server, the file is in the folder:
 - \\solution_specific\custom_server\aix
 - \\solution_specific\custom_server\hp-ux
 - \\solution_specific\custom_server\linux
 - \\solution_specific\custom_server\solaris
 - \\solution_specific\custom_server\tru64UNIX
 - \\solution_specific\custom_server\windows.

If you downloaded Enterprise Routing or Network Routing from an FTP site, all setup files are located in the windows or other operating system folder of the zip archive.

Note: If you install several instances of a Universal Routing component (for example, URS) on the same computer, separate shortcuts are created for each based on the Application name stored in Configuration Layer.

Installing on Windows Operating Systems

The installation process does not present the option of installing a server component as a Service. By default, starting with 7.5, all server components are installed as Services in Automatic startup mode.

If you want to use Management Layer and SCI, you must also install LCA on the URS host computer, as documented in the *Framework 8.0 Deployment Guide*.

Procedure: Installing URS on Windows using the Installation Wizard

Note: The URS Application object must already be configured before you begin the installation.

The installation process does not present the option of installing a server component as a Service. By default, all 7.5 and later server components are installed as Services in Automatic startup mode.

Start of procedure

1. Double-click the setup.exe file.
 - On the Universal Routing CD, this file is located in the following folder: \\solution_specific\universal_routing_server\windows. If URS was downloaded from an FTP site, the file is located in the download directory.

Install Shield opens the Welcome screen (see [Figure 108](#)).

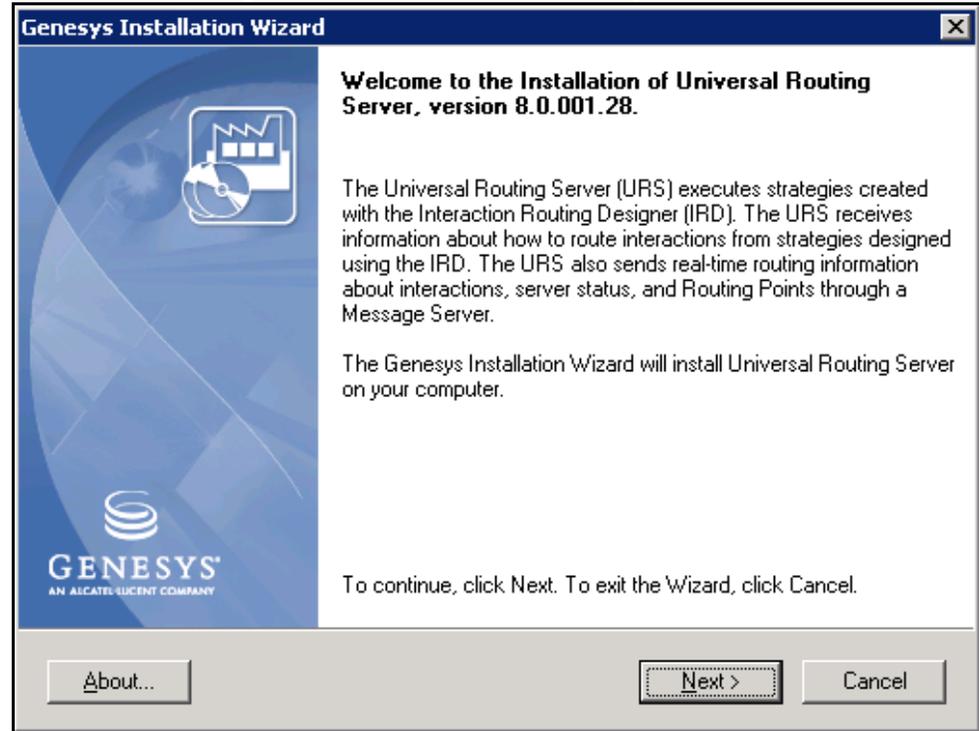
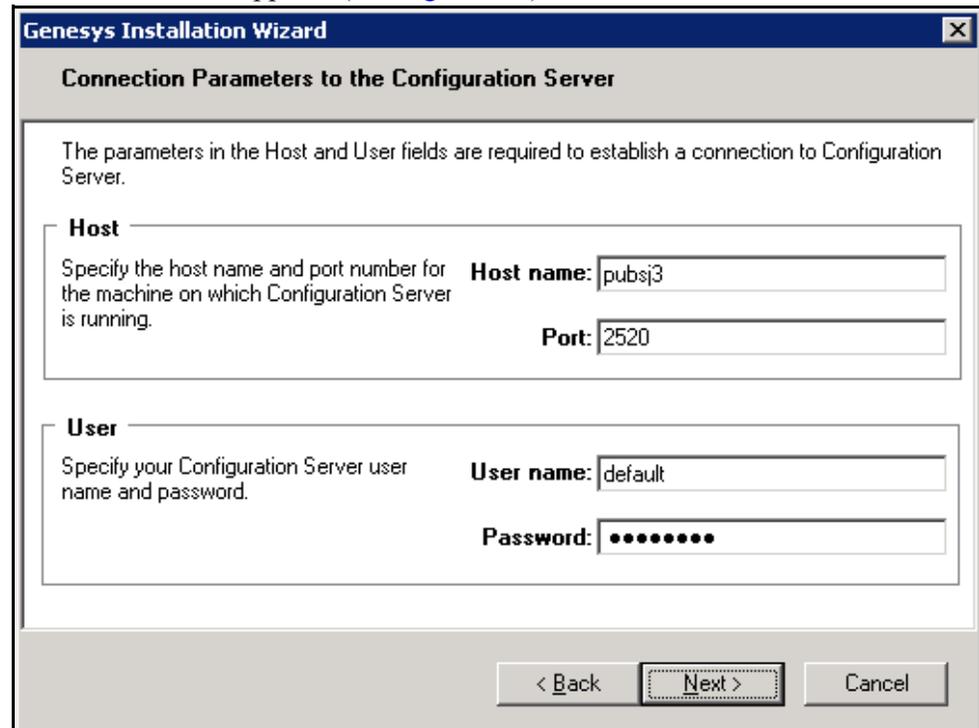


Figure 108: Welcome Screen, URS Installation Wizard

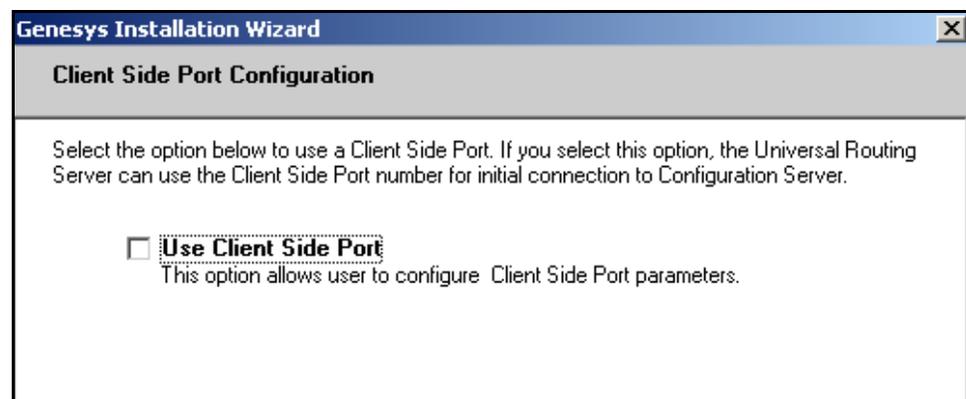
2. Click Next. The Connection Parameters to the Genesys Configuration Server screen appears (see [Figure 109](#)).



The screenshot shows a dialog box titled "Genesys Installation Wizard" with a sub-header "Connection Parameters to the Configuration Server". Below the sub-header is a text box stating: "The parameters in the Host and User fields are required to establish a connection to Configuration Server." There are two main sections: "Host" and "User". The "Host" section includes a text box for "Host name" containing "pubsj3" and a text box for "Port" containing "2520". The "User" section includes a text box for "User name" containing "default" and a text box for "Password" containing seven dots. At the bottom of the dialog are three buttons: "< Back", "Next >" (which is highlighted with a dashed border), and "Cancel".

Figure 109: Connection Parameters to the Genesys Configuration Server

3. Under Host, specify the host name and port number for the computer on which Configuration Server is running. This is the main “listening” port that is entered in the Server Info tab for Configuration Server, which is used for authentication in the Configuration Manager login dialog box.
4. Under User, enter the user name and password that you use for logging in to Configuration Server.
5. Click Next to open the Client Side Port Configuration screen. (see [Figure 110](#)).



The screenshot shows a dialog box titled "Genesys Installation Wizard" with a sub-header "Client Side Port Configuration". Below the sub-header is a text box stating: "Select the option below to use a Client Side Port. If you select this option, the Universal Routing Server can use the Client Side Port number for initial connection to Configuration Server." There is a single checkbox labeled "Use Client Side Port" which is currently unchecked. Below the checkbox is a text box stating: "This option allows user to configure Client Side Port parameters."

Figure 110: Client-Side Port Configuration

6. If you are setting up client-side port configuration for the initial connection to Configuration Server as described in the *Genesys 8.0 Security Deployment Guide*, select the `Use Client Side Port` check box to reveal additional fields (see [Figure 111](#)).

Genesys Installation Wizard

Client Side Port Configuration

Select the option below to use a Client Side Port. If you select this option, the Universal Routing Server can use the Client Side Port number for initial connection to Configuration Server.

Use Client Side Port
This option allows user to configure Client Side Port parameters.

Configuration Options

Specify Client Side Port Number. **Port:**

Specify Client Side IP Address. **IP Address:**

< Back Next > Cancel

Figure 111: Client-Side Port Configuration Screen, Configuration Options

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

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

8. The Select Application screen appears. Figure 112 on [page 317](#) shows an example.

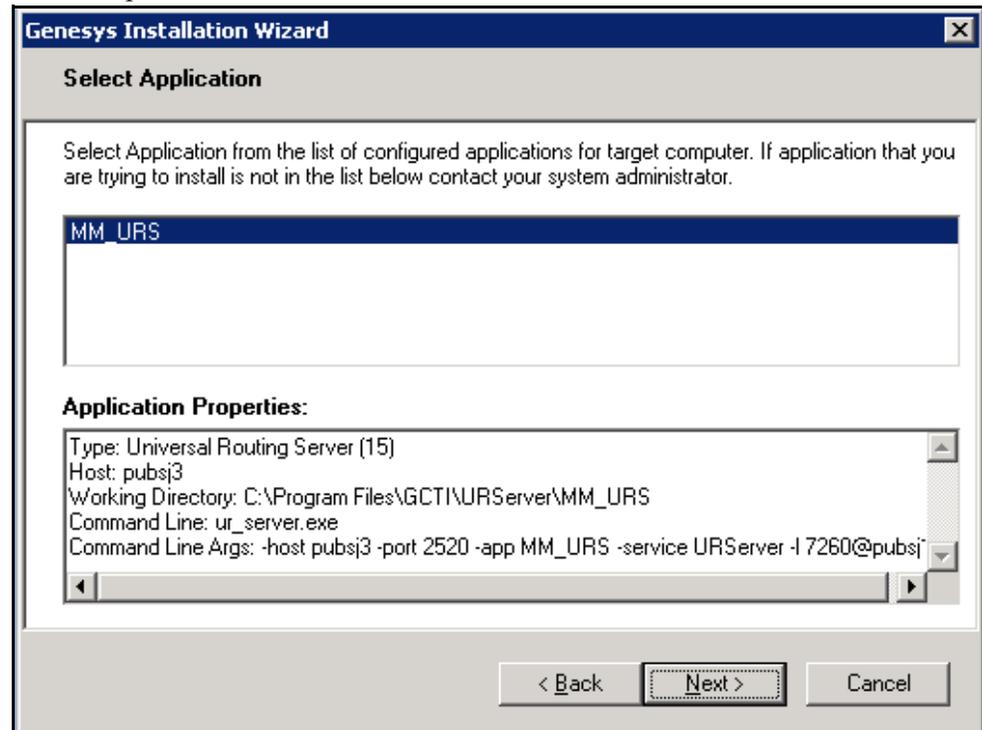


Figure 112: Select Application

9. On the Select Application screen, select the URS Application that you are installing. The Application Properties area shows the Type, Host, Working Directory, Command Line executable, and Command Line Arguments information that was previously entered in the Server Info and Start Info tabs of the selected URS Application object.
10. Click Next. The Access to License screen appears. Figure 113 on [page 318](#) shows the screen, with License Manager selected.

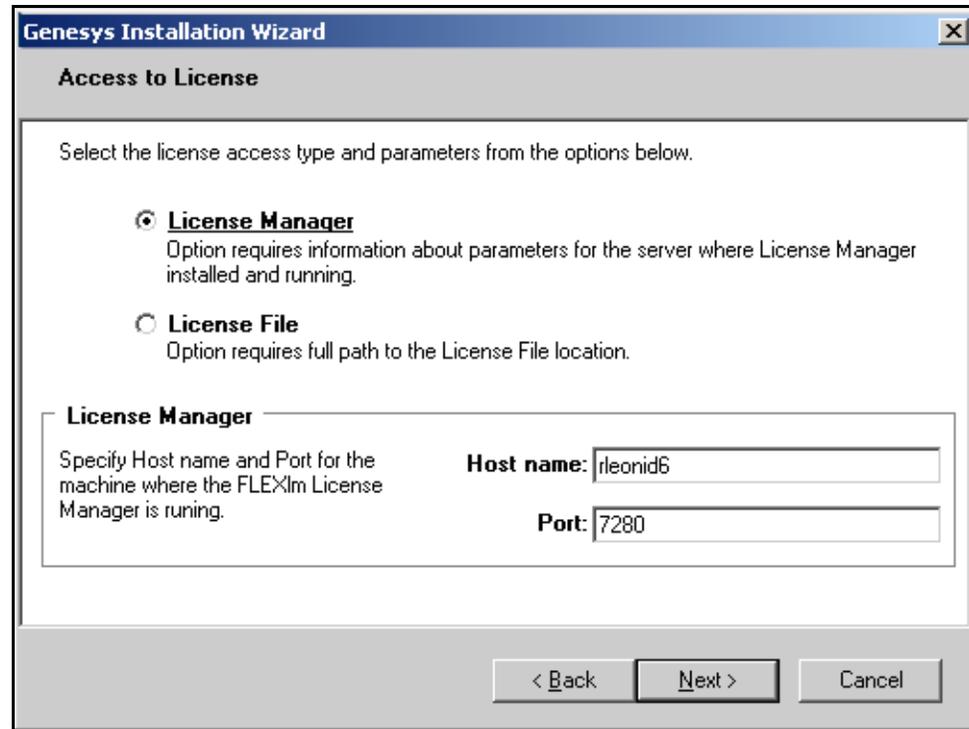


Figure 113: Access to License

11. Select one of the following options:

- **License Manager.** Requires the host name and port information for the server on which License Manager is installed and running as shown in Figure 113 on [page 318](#).
- **License file.** Requires the full path to the license file location.

These instructions assume that you select the License Manager radio button.

12. Click Next. The Choose Destination Location screen appears (see [Figure 114](#)).

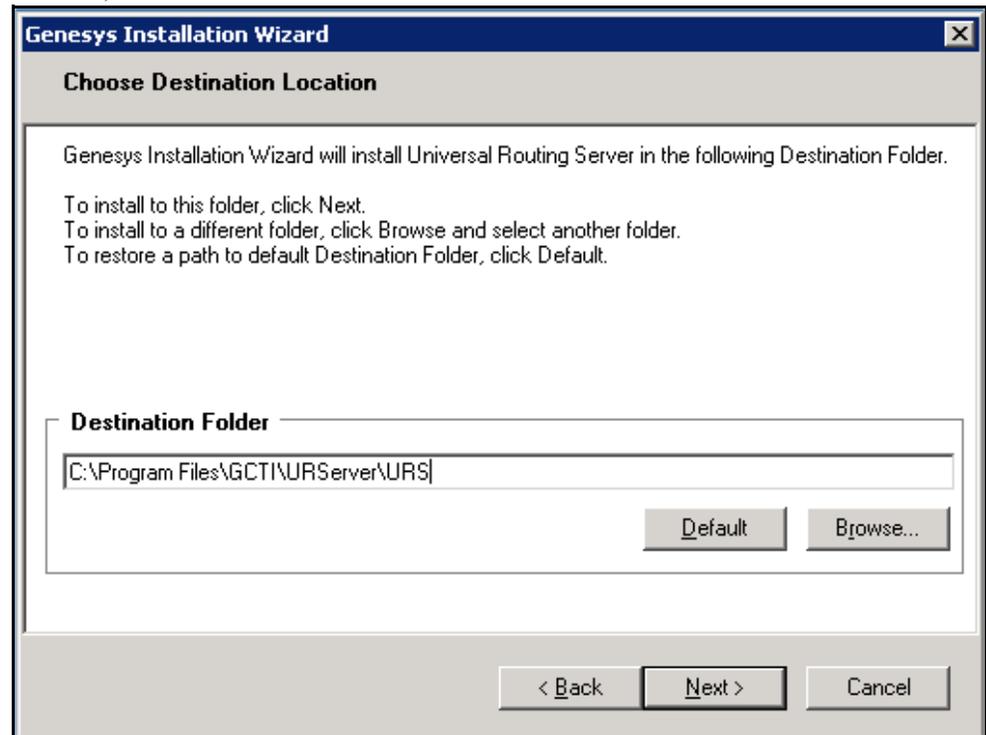


Figure 114: Choose Destination Location

13. Under Destination Folder, keep the default location, or browse for the installation location for URS.
14. Click Next. The Ready to Install screen appears.
15. Click Next on the Ready to Install screen. The Genesys Installation Wizard indicates that it is performing the requested operation for URS. When it has finished, the Installation Complete screen appears (see [Figure 115](#) on [page 320](#)).

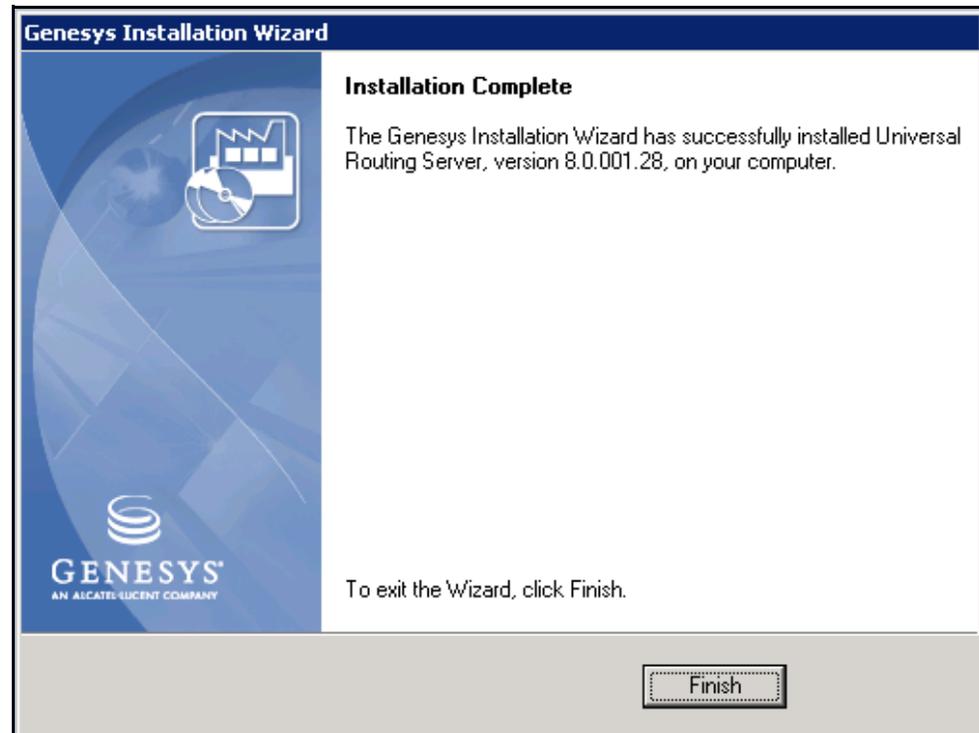


Figure 115: Installation Complete

16. Click **Finish** on the **Installation Complete** screen.

If you open the Registry Editor, you can see the configuration information that you previously entered (see Figure 111 on page 316). Figure 116 shows an example.

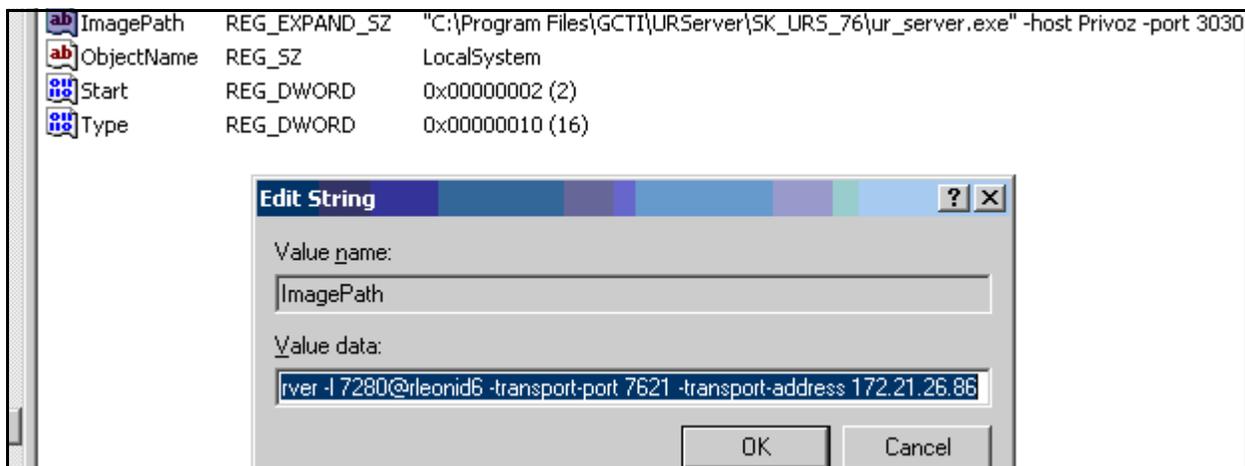


Figure 116: Registry Editor

End of procedure

Procedure: Installing IRD on Windows using the Installation Wizard

Start of procedure

1. Double-click the `setup.exe` file.
 - On the Universal Routing CD, this file is located in the following folder: `\\solution_specific\interaction_routing_designer\windows`. If IRD was downloaded from an FTP site, the file is located in the download directory.

InstallShield opens.

2. On the Welcome screen, click Next when you are prompted to continue with the setup program.
3. After you click Next, a Security Banner Configuration screen appears. Use this screen to choose whether you want to configure a security banner for the current application (IRD).

Note: Refer to the *Genesys 8.0 Security Deployment Guide* for detailed information about configuring the security banner.

4. Do one of the following:
 - If you do not want to configure a security banner:
 - a. Clear the Enable Security Banner check box if it is selected and then click Next.
 - b. Follow the prompts to continue with the setup program.
 - If you want to configure a security banner:
 - a. Select Enable Security Banner.
 - b. Follow the instructions in the procedure “Installing and Configuring the Security Banner” in the *Genesys 8.0 Security Deployment Guide*.
 - c. When you are finished with that procedure, return here and follow the wizard prompts to continue with the setup program.

End of procedure

Next Steps

If you want to store Interaction Routing Designer *.rbn files (the GUI portion of routing strategies) in the Configuration Database, run the script that creates the tables in the database as described on “Running the Configuration Database Update Script” on [page 215](#).

Procedure: Installing Custom Server on Windows using the Installation Wizard

Note: The Custom Server Application object must already be configured before you begin the installation.

The installation process does not present the option of installing a server component as a Service. By default, all 7.5 and later server components are installed as Services in Automatic startup mode.

Start of procedure

1. Double-click the setup.exe file.

On the Universal Routing CD, this file is located in the following folder: \\solution_specific\custom_server\windows. If Custom Server was downloaded from an FTP site, the file is located in the download directory.

Install Shield opens the Welcome screen (see [Figure 117](#)).

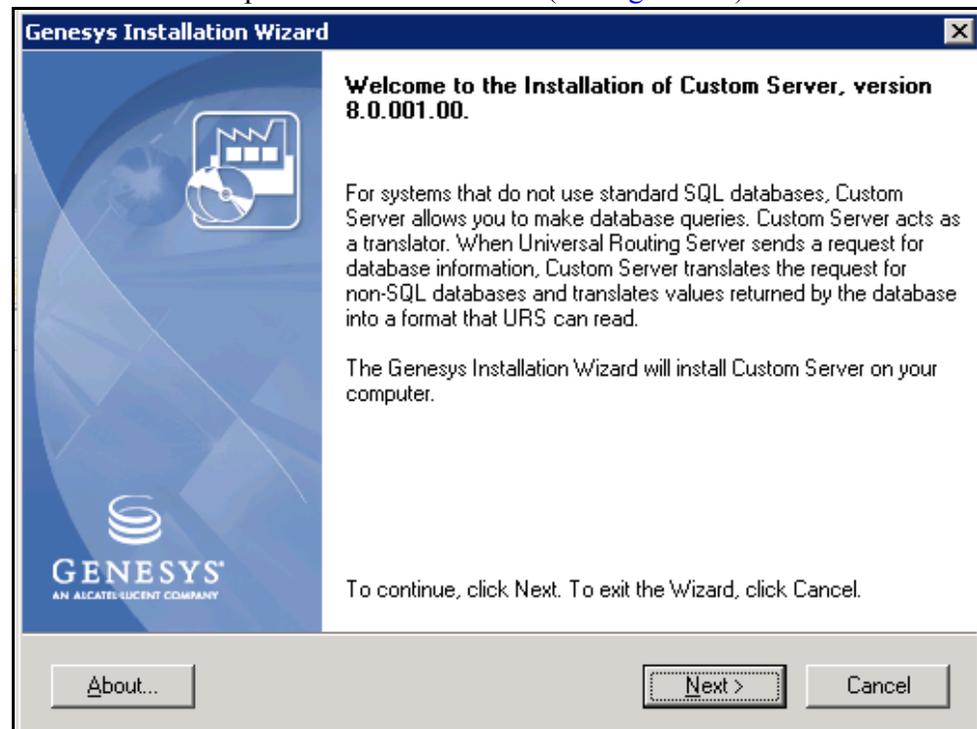


Figure 117: Welcome Screen, URS Installation Wizard

2. Click Next. The Connection Parameters to the Genesys Configuration Server screen appears. See [Figure 118](#) on [page 323](#).

Genesys Installation Wizard

Connection Parameters to the Genesys Configuration Server

The parameters in the Host and User fields are required to establish a connection to Genesys Configuration Server.

Host

Specify the host name and port number for the machine on which Genesys Configuration Server is running.

Host name: Privoz

Port: 3030

User

Specify your Genesys Configuration Server user name and password.

User name: default

Password: ●●●●●●●●

< Back Next > Cancel

Figure 118: Connection Parameters to the Genesys Configuration Server

3. Under Host, specify the host name and port number for the computer on which Configuration Server is running. This is the main “listening” port that is entered in the Server Info tab for Configuration Server, which is used for authentication in the Configuration Manager login dialog box.
4. Under User, enter the user name and password that you use for logging in to Configuration Server.
5. Click Next to open the Client Side Port Configuration screen (see [Figure 119](#)).

Genesys Installation Wizard

Client Side Port Configuration

Select the option below to use a Client Side Port. If you select this option, the Custom Server can use the Client Side Port number for initial connection to Configuration Server.

Use Client Side Port
This option allows user to configure Client Side Port parameters.

Figure 119: Client-Side Port Configuration

6. If you are setting up client-side port configuration for the initial connection to Configuration Server as described in the *Genesys 8.0 Security Deployment Guide*, select the Use Client Side Port check box to reveal additional fields (see [Figure 120](#)).

Genesys Installation Wizard

Client Side Port Configuration

Select the option below to use a Client Side Port. If you select this option, the Custom Server can use the Client Side Port number for initial connection to Configuration Server.

Use Client Side Port
This option allows user to configure Client Side Port parameters.

Configuration Options

Specify Client Side Port Number. **Port:**

Specify Client Side IP Address. **IP Address:**

< Back Next > Cancel

Figure 120: Client-Side Port Configuration Screen, Configuration Options

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

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

8. Click Next. The Select Application screen appears. [Figure 121 on page 325](#) shows the screen with example data.

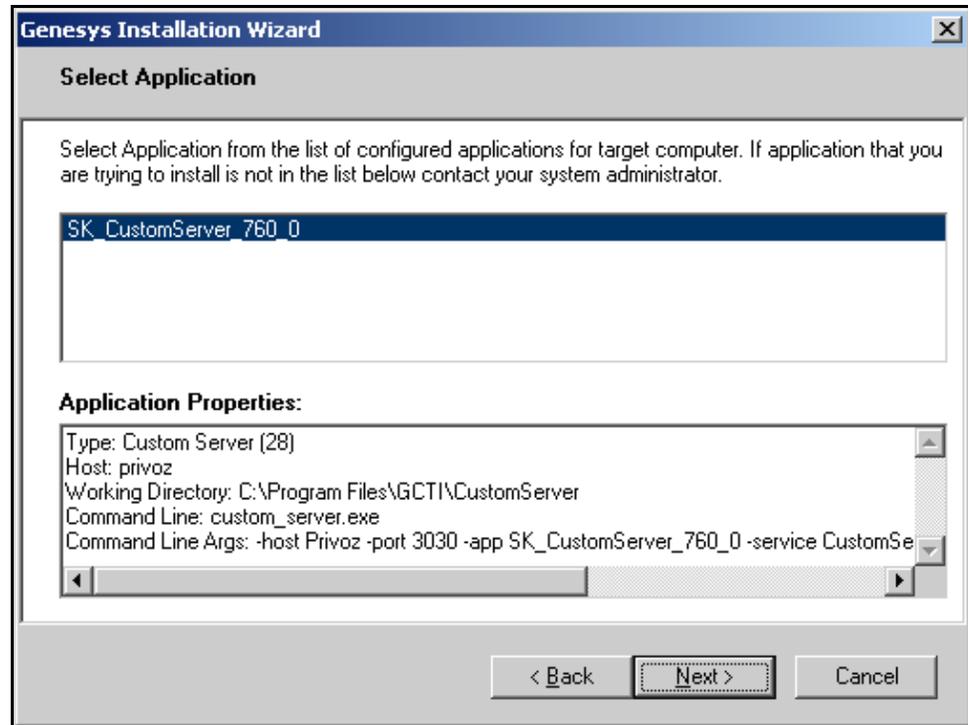


Figure 121: Select Application

9. On the Select Application screen, select the Custom Server Application that you are installing. The Application Properties area shows the Type, Host, Working Directory, Command Line executable, and Command Line arguments information previously entered in the Server Info and Start Info tabs of the selected Custom Server Application object.
10. Click Next. The Choose Destination Location screen appears (see Figure 122 on [page 326](#)).

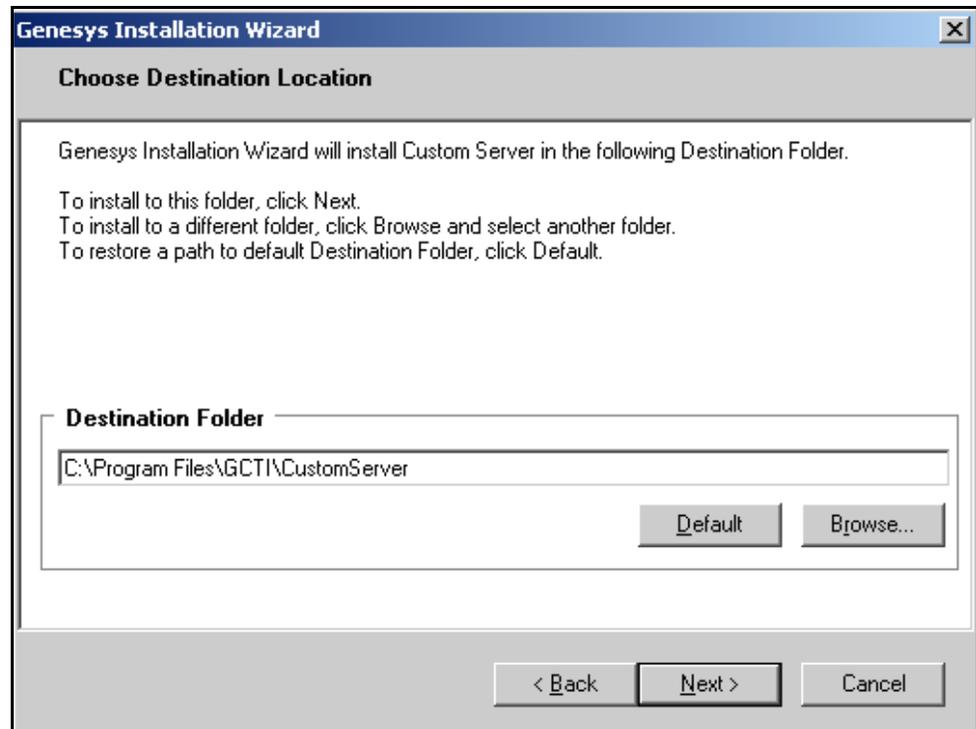


Figure 122: Choose Destination Location

11. Under **Destination Folder**, keep the default location, or browse for the installation location for Custom Server and then click **Next**. The **Ready to Install** screen appears.
12. Click **Next** on the **Ready to Install** screen. The Genesys Installation Wizard indicates that it is performing the requested operation. When it has finished, the **Installation Complete** screen appears (see Figure 123 on [page 327](#)).

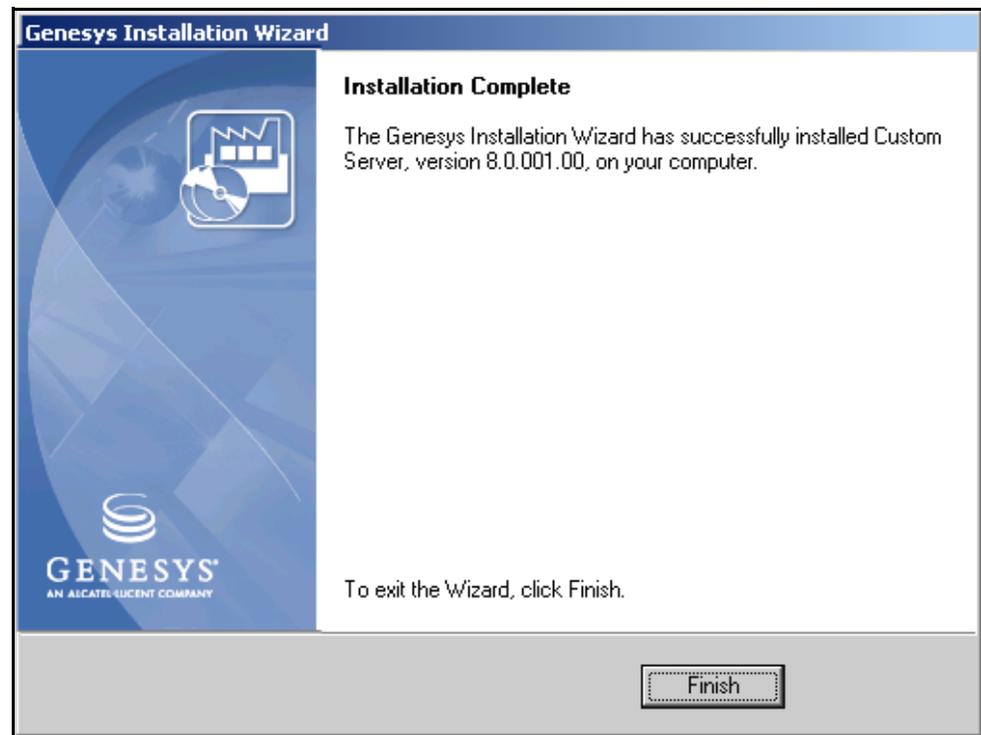


Figure 123: Installation Complete

13. Click **Finish** on the **Installation Complete** screen. If you open the Registry Editor, you can see the configuration information that you previously entered (see Figure 111 on page 316). Figure 124 shows an example.

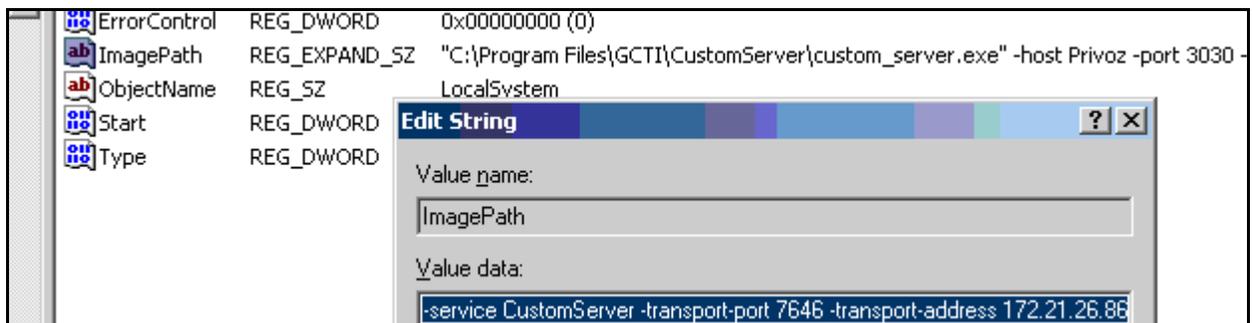


Figure 124: Registry Editor

End of procedure

Warning! If you plan to upgrade later, these files will be overwritten during the upgrade. Be sure to save the original files in another location if you want to retain the original files.

Installing on UNIX

Warning! Before starting the installation, make sure that all instances of URS and Custom Server already installed on your computer are shut down. If you do not do this, you will not be able to back up your files in case you want to use the same installation directory for another version of those components.

Installing URS on UNIX-Based Operating Systems

Enterprise Routing and Network Routing are distributed on a CD. Complete the following directions to install URS on a UNIX platform.

Procedure: Installing Universal Routing Server on a UNIX platform

Start of procedure

1. Go to the directory in which the installation is created.
2. Copy all files to a temporary directory.

Note: Files that are included in the installation package require permission to execute.

3. Run the installation script by entering `./install.sh` (see “Installation Package Location” on [page 312](#)).
4. When prompted, enter the host name of the computer on which URS will be installed or press the `Enter` key for the supplied entry.
5. When prompted, enter the following information about your Configuration Server:
 - Configuration Server Hostname
 - Network port
 - User name
 - Password
6. Prompts appear regarding securing connections between URS and Configuration Server.
`Client Side Port Configuration`

Select the option below to use a Client Side Port. If you select this option, the application can use Client Side Port number for initial connection to Configuration Server.

Do you want to use Client Side Port option (y/n)

7. When prompted, type either Y for yes or N for no. The instructions that follow assume that you typed Y.

8. Enter an IP address or press Enter for the supplied entry after the following prompt:

Client Side IP Address (optional), the following values can be used:

9. Choose the URS Application to install after this prompt (which may list several URSs):

Please choose which application to install:

1 : <URS_application>

10. Enter the destination directory for the installation after this prompt:

Please enter full path of the destination directory for installation:

After you enter the destination directory, the installation continues. A message appears that starts with extracting tarfile: d

11. When this instruction appears:

There are two versions of this product available: 32-bit and 64-bit. Please enter 32 or 64 to select which version to use, enter 32 or 64 according the UNIX make that you use.

12. Respond to this prompt:

Please select the format for your license location. Enter the number 1 or 2.

1) Full path to the license file

2) License Manager port and host

End of procedure

As soon as the installation process is finished, a message appears announcing that installation was successful. The process created a directory, with the name specified during the installation, containing Universal Routing Server.

Note: If you want to use Management Layer and SCI, you must also install LCA on the URS host computer as documented in the *Framework 8.0 Deployment Guide*.

Installing Custom Server on UNIX-Based Operating Systems

Enterprise Routing and Network Routing are distributed on a CD. Complete the following directions to install Custom Server on a UNIX platform.

Procedure: Installing Custom Server on UNIX

Start of procedure

1. Go to the directory in which the installation is created.
2. Copy all files to a temporary directory.
3. In the temporary directory, locate a shell script called CSERVER-INSTALL.SH.

Note: Files included in the installation package require permission to execute.

4. Run this script by entering `./install.sh`

The following message appears:

```
* Welcome to the Genesys 8.0 Installation Script *
*****
Installing Custom Server, version 8.0.001.00
Please enter the hostname or press enter for "<host_name>" =>
```

5. Specify the host name of the computer on which the Custom Server will be installed.

Note: The following message appears and can be ignored when an installation is done not from the wizard: Unable to find configuration information. Either you have not used configuration wizards and the GCTISetup.ini file was not created or the file is corrupted.

6. The next prompt asks for your Configuration Server:

Please enter the following information about your Configuration Server:

Enter the requested information for Configuration Server.

```
Configuration Server Hostname =>
Network port =>
User name =>
Password =>
```

7. The installation script then asks if you want to use the client-side port option.

```
Client Side Port Configuration
```

```
Select the option below to use a Client Side Port. If you select
this option, the application can use Client Side Port number for
initial connection to Configuration Server.
```

```
Do you want to use Client Side Port option (y/n)?
```

8. Assuming you type *y*, prompts appears for client-side port data. For information on responding to these prompts, refer to the *Genesys 8.0 Security Deployment Guide*.
9. After the client-side port prompts, the installation script asks which Application to install. A list appears of Applications of the Custom Server type configured for this host. Select a Custom Server to install. Example entries are shown below.

```
Please choose which application to install:
```

```
1 : SK_CSS_0SF51_80
```

```
=>1
```

10. Specify the destination directory into which Custom Server will be installed. An example entry is shown below.

```
Press ENTER to confirm /home/akoret/tru64/cs_osf51 as the
destination directory or enter a new one =>
```

```
If the directory already exists and is not empty, the following message
appears:
```

```
The target install directory name_of_target_directory has files in
it. Please select an action to perform:
```

1. Back up all files in the directory
2. Overwrite only the files contained in this package
3. Wipe the directory clean

```
1, 2, or 3 =>
```

```
After you select 1, 2, or 3, installation continues according to the chosen
option. Selecting 1 backs up all existing files in the directory. Selecting 2
overwrites only the files that are contained in the package. If you select 3,
after an additional confirmation question, all content of the destination
directory is erased.
```

Note: If you are running under the HP-UX operating system, do not attempt to install another instance of Custom Server in a directory in which an instance of Custom Server is already running. If you do, after selecting the Back up all files in the directory option, the following message is returned: `mv: libRCP.sl: cannot move text busy shared Library: Text file busy Couldn't back up.`

End of procedure

As soon as the installation process is finished, a message appears that announces that installation was successful. The process created a directory (with the name that was specified during the installation) that contains Custom Server.

Note: If you want to use Management Layer and SCI, you must also install LCA on the Custom Server host computer as documented in the *Framework 8.0 Deployment Guide*.



Chapter

14

Starting and Stopping Procedures

This chapter provides instructions for starting and stopping solutions with the Solution Control Interface (SCI) and for manually starting and stopping Universal Routing Server (URS), Network T-Server (Network Routing only), and Custom Server.

This chapter includes the following topics:

- [Prestart Information, page 333](#)
- [Starting, page 334](#)
- [Stopping, page 340](#)
- [Non-Stop Operation, page 341](#)
- [Version Identification, page 342](#)

Prestart Information

Before starting the solution, it must be configured and installed. For more information, see Chapter 6 on [page 143](#) and Chapter 13 on [page 311](#).

Before starting Solution Control Interface (SCI), start the Configuration DB Server (example: `cfg_dbserver`), Configuration Server, and Solution Control Server. Make sure that Local Control Agent (LCA) is running.

Note: SCI, Solution Control Server, and LCA are a part of the Management Layer. See the *Framework 8.0 Management Layer User's Guide* for information about these components.

Starting

Task Summary: Starting Universal Routing components

Objective	Related Procedures and Actions
Start a solution with Solution Control Interface	Procedure: Starting a solution with Solution Control Interface , on page 335
Start Universal Routing Server on UNIX	Procedure: Starting URS on UNIX , on page 338
Start Custom Server 6.5-7.x on Windows	Procedure: Starting Custom Server 6.5–8.x On Windows , on page 339
Start Custom Server 6.1	Procedure: Starting Custom Server 6.1 , on page 340

Start Universal Routing solutions from the Solution Control Interface (SCI). [Figure 125](#) shows an example solution that could include URS as well as other server applications.

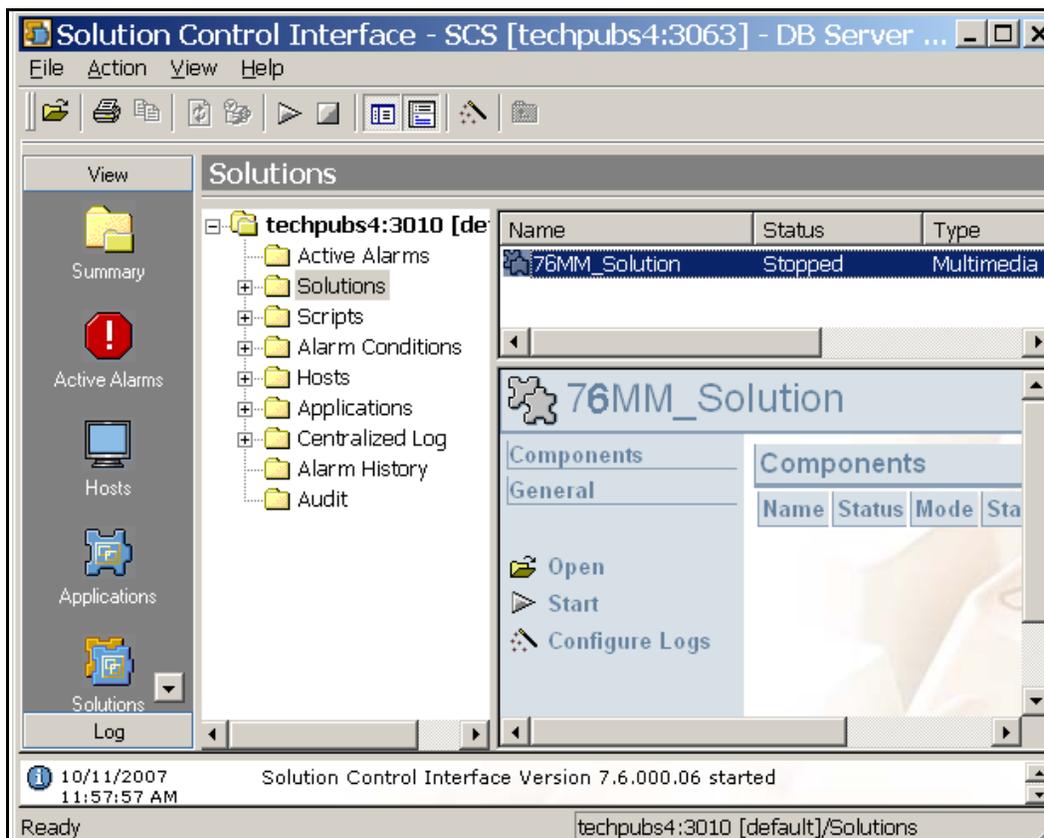


Figure 125: Solution Control Interface

SCI can start the following servers:

- DB Server(s) (for Database Access Points used in routing, not configuration DB Server)
- Universal Routing Server(s)
- Message Server(s)
- Stat Server(s)
- Load Distribution Server(s)
- Custom Server
- premise T-Server(s)
- Network T-Server(s) (for Network Routing only)

Since IRD is a GUI application and not a server application, you must start IRD manually. See the section “Starting Interaction Routing Designer” on [page 338](#).

Note: Message Server(s) can also be started after SCI.

Procedure:

Starting a solution with Solution Control Interface

Start of procedure

1. Start the Solution Control Interface.
2. Go to the `Solutions` view.
3. Right-click on the desired solution and select `Start` from the shortcut menu.
-or-
Select the desired solution and choose `Action > Start` on the menu bar.

End of procedure

The command to start Enterprise Routing or Network Routing is sent to Solution Control Server, which uses Local Control Agents to activate the solution components in the solution configuration object.

Enterprise Routing

Enterprise Routing consists of optional and required components.

The required components are:

- Universal Routing Server
- T-Server

The optional components are:

- Message Server
- DB Server
- Database Access Point
- Stat Server
- Custom Server
- Interaction Routing Designer

If both T-Server and Universal Routing Server have reported Started status within the configured timeout, SCI reports a successful start of Enterprise Routing.

Upon a successful start of the solution, the solution status changes from Stopped to Started.

Network Routing

Network Routing consists of both optional and required components.

Required components:

- Universal Routing Server
- Network T-Server
- T-Server

Optional components:

- Message Server
- DB Server
- Database Access Point
- Stat Server
- Custom Server
- Interaction Routing Designer

If Network T-Server, T-Server, and Universal Routing Server have reported Started status within the configured timeout, SCI reports a successful start of the solution.

Upon a successful start, solution status changes from Stopped to Started.

Note: Because many components are shared by a number of solutions, some components can have status Running before the solution is started.

- For more information, see *Framework 8.0 Solution Control Interface Help*.
- See the *Framework 8.0 Management Layer User's Guide* for information about Message Server, Solution Control Server, Solution Control Interface, Local Control Agent, and the Log Database.

SCI, Redundancy, and LCA

In redundant configurations, both primary and backup routing components start simultaneously for components in Hot or Warm Standby mode. They are assigned primary or backup roles according to their configuration. If a component configured as primary cannot be started, the backup component assumes the primary role.

If an application shuts down, LCA, which monitors whether an application exists and is running, sends a message to Solution Control Server to conduct appropriate recovery action depending on the system configuration. If a backup is configured and started, Management Layer automatically switches from the primary application that failed to the backup application.

Starting Universal Routing Server Manually

This section describes how to manually start Universal Routing Server (URS). For information on manually starting Framework components necessary to use Universal Routing, see the *Framework 8.0 Deployment Guide*.

Select Start > Programs > Genesys Solutions > Routing > Universal Routing Server > Start Universal Routing Server

Note: The above path is the default location. If you installed the software at a different location, navigate to the appropriate location to start URS.

You can also start from the command line or the Start Info tab for URS in Solution Control Interface. [Figure 126](#) shows an example.

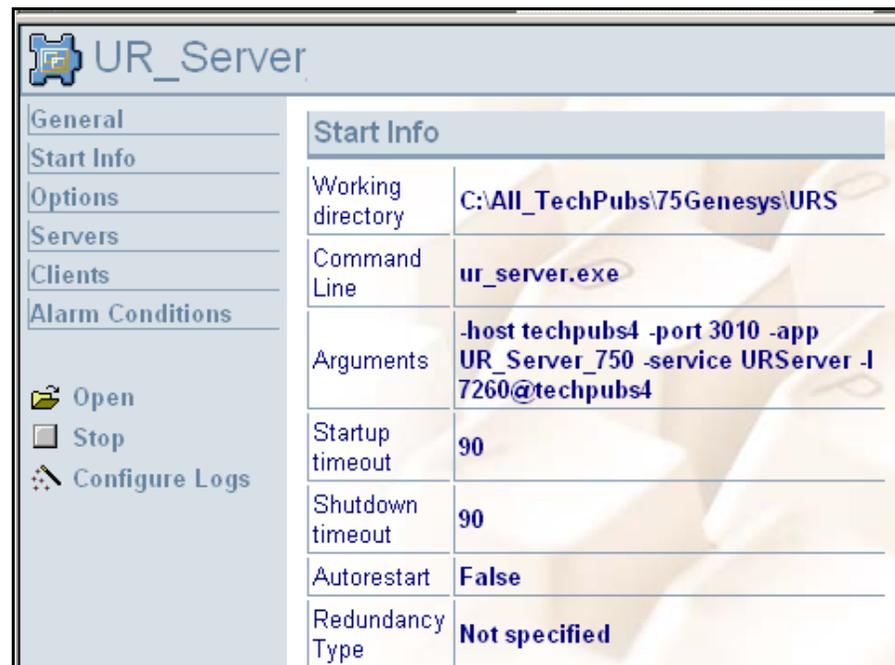


Figure 126: Example Start Info Tab for URS in SCI

Procedure: Starting URS on UNIX

Installation of URS creates a `run.sh` file. You can start Universal Routing Server on UNIX by just running this file which contains:

```
ur_server -host <name of Configuration Server host> -port <name of  
Configuration Server port>-app <name of URS Application> -l <the  
full path and name of license file>
```

Start of procedure

1. Open a terminal window.
2. Log in.
3. Choose the appropriate directory.
4. Run the `run.sh` file.

End of procedure

Note: The text in angle brackets (<text>) above indicates the variables you enter that are unique to your environment and are required. Your information should replace the text and the brackets. See the example below for clarification.

An example `run.sh` file is as follows:

```
ur_server -host Daemon -port 5010 -app Router_Server -l \FLEXlm  
\license.dat
```

If there is a space in the URS application name, then quotation marks are required before and after the name of the URS application. The license file path must also be enclosed in quotation marks.

When URS is started, a window opens and messages are sent regarding its status. URS also establishes connections to all servers listed in the `Connections` tab of the Universal Routing Server Application object.

Starting Interaction Routing Designer

See “Logging into Interaction Routing Designer” on [page 268](#).

Note: Universal Routing Server does not need to be running to use Interaction Routing Designer (IRD).

Starting Network T-Server from the Command Line

Applies to Network Routing only. You can start Network T-Server from the command line. The command line options include the following:

- host—The name of the host where Configuration Server is running.
- port—The Configuration Server port
- app—The name of the application as configured in the Configuration Database
- l—The name of the license data file with the full path to the file

Windows example:

```
tserver.exe -host cs-host -port cs-port -app application-name -l
c:\FLEXlm\license.dat
```

UNIX example:

```
mci800_server -host cs-host -port cs-port -app application-name -l
\license.dat
```

Starting Custom Server Manually

Universal Routing 8.0 supports Custom Server 6.1, 6.5, 7.x and 8.x. Because Custom Server 6.1 is configured differently from Custom Server 7.x and 8.x, it also started differently from Custom Server 7.x and 8.x.

Chapter 12 on [page 289](#) for information on configuring Custom Server.

Procedure:

Starting Custom Server 6.5–8.x On Windows

Start of procedure

- Select Start > Programs > Genesys Solutions > Routing > Custom Server > Start Custom Server

End of procedure

Note: This is the default location. If you installed the software at a different location, navigate to the appropriate location to start Custom Server.

Procedure: Starting Custom Server 6.1

If Custom Server 6.1 is part of your solution, you must manually start it.

Start of procedure

1. Locate and open the directory where Custom Server was installed.
2. For Windows, enter `custom_server -c <configuration_file_name>`
For UNIX, enter `custom_server -c cserver.conf`

End of procedure

Stopping

This section describes the procedures for stopping solution components using the Solution Control Interface and stopping Interaction Routing Designer.

Task Summary: Stopping Universal Routing Components

Objective	Related Procedures and Activities
Stop a solution using Solution Control Interface	Procedure: Stopping solutions using Solution Control Interface, on page 340
Stop Interaction Routing Designer	Procedure: Stopping Interaction Routing Designer, on page 341

Procedure: Stopping solutions using Solution Control Interface

Enterprise Routing and Network Routing solutions should be stopped using the Solution Control Interface (SCI).

Start of procedure

1. Start the Solution Control Interface.
2. Go to the `Solutions` view.
3. Right-click on the desired solution and select `Stop` from the shortcut menu.
-or-
Select the desired solution and choose `Action > Stop` on the menu bar.

End of procedure

The command to stop the solution is sent to Solution Control Server, which uses Local Control Agent to terminate the solution components in the reverse order of the component startup. (The component startup order is defined in the solution configuration object.)

SCI reports a successful stop of the solution after all of the solution components have stopped within the configured timeout. When stopped, routing solution's status changes from Started to Stopped.

Important Information

- Because many components are shared by a number of solutions, some routing solution components can continue to have the status Started after the solution is stopped.
- In redundant configurations, both primary and backup routing components stop simultaneously.
- Starting and stopping procedures using SCI do not affect 6.1 versions of Custom Server. Custom Server 6.1 must be stopped manually.

Procedure: Stopping Interaction Routing Designer

Purpose: To stop Interaction Routing Designer.

Start of procedure

1. Save your work.
2. From the File menu, select Exit.

End of procedure

Non-Stop Operation

The *non-stop operation* (NSO) feature enables URS to continue to run even if it encounters problems. NSO prevents a shutdown in the event of failures. This works by allowing URS to operate on two levels designated by the command-line parameters described below.

Built-in NSO provides the option of running URS in non-stop operation mode (NSO).

Note: When URS is started, by default non-stop operation is disabled.

The command-line parameter `-nco` is used to control non-stop operation. URS built with NSO support runs in NSO only if one of the following arguments is specified in the command line:

<code>-nco xcount/xthreshold</code>	where <code>xcount</code> (exception counts) is the number of faults allowed during a specified interval before the application exits and <code>xthreshold</code> (exception threshold) is the time interval in seconds. The values must be separated by a slash.
<code>-nco</code>	start NCO with default parameters (six faults in 10 seconds)

Examples:

```
ur_server -host ra -port 2000 -app router -nco
```

```
ur_server -host ra -port 2000 -app router -nco 100/1
```

See the Framework documentation on T-Servers for more information about faults.

Version Identification

To print the URS version number to the log, use `-v`, `-version`, or `-V` in the command line. This option does not actually start URS. It simply prints the version number to the log and then exits.

To print the Custom Server version number to the log, you can also use `-v`, `-version`, or `-V` in the command line.



Chapter

15

Uninstalling a Routing Solution

This chapter describes how to uninstall an Enterprise Routing or Network Routing solution. It contains the following topics:

- [Removing Components with Genesys Administrator, page 344](#)
- [Removing Components Manually, page 344](#)
- [Deleting Tables, page 346](#)

Task Summary: Uninstalling a Routing Solution

Objective	Related Procedures and Activities
Uninstall routing components in Genesys Administrator	Procedure: Removing Universal Routing components with Genesys Administrator, on page 344
Uninstall routing components manually	Procedure: Manually removing Universal Routing components on Windows, on page 345 Procedure: Manually removing Universal Routing components on UNIX-based operating systems, on page 346
Delete tables	See “Deleting Tables” on page 346

Removing Components with Genesys Administrator

This section describes how to remove Universal Routing components by using Genesys Administrator.

Procedure: Removing Universal Routing components with Genesys Administrator

Purpose: If you are using Genesys Administrator in your environment, you can uninstall Universal Routing components directly from the Genesys Administrator interface.

Note: Interaction Routing Designer (IRD) cannot be uninstalled from Genesys Administrator. To uninstall IRD, see [Procedure: Manually removing Universal Routing components on Windows](#).

Start of procedure

1. Login to Genesys Administrator.
2. Locate the Universal Routing component that you want to remove.
3. Click `Uninstall`.

Note: For more information about working with Genesys Administrator, refer to the *Framework 8.0 Genesys Administrator Deployment Guide*.

End of procedure

Removing Components Manually

This section describes how to remove the Universal Routing components manually.

Procedure: Manually removing Universal Routing components on Windows

Start of procedure

Perform the following steps on each computer that is hosting Universal Routing components:

1. From the Windows Start menu, open the Control Panel (Settings > Control Panel) and click Add/Remove Programs.
2. In the Add/Remove Programs dialog box, select one of these programs:
 - Genesys Interaction Routing Designer 8.0
 - Genesys Universal Routing Server 8.0
 - Genesys Custom Server 8.0 (if it is installed)
3. Click Add/Remove. Repeat this step for each component that is listed in Step 2 and that you want to uninstall.

Note: Close IRD before trying to uninstall it. If you try to uninstall IRD with both IRD and Configuration Manager running, an error message informs you that you must close both IRD and Configuration Manager before proceeding. In fact, you do not have to close Configuration Manager to uninstall IRD. Click Ignore in the Install Shield message box. As soon as you close IRD, the uninstallation of IRD completes successfully without the need to close Configuration Manager.

4. Using Windows Explorer, browse to the GCTI main directory and delete the complete Enterprise Routing Solution subdirectory (including all subfolders).

End of procedure

Procedure:
Manually removing Universal Routing components on UNIX-based operating systems**Start of procedure**

- For the directory in which each Universal Routing component is installed, run the following command:

```
rm -R
```

Note: If an instance of the component is running, the command will fail.

End of procedure

Deleting Tables

The `Genesys_ERS` database stores information that is used by Universal Routing Server for routing interactions. Genesys recommends that you not delete the tables in this database, because they are used for purposes other than routing.

The `Genesys_Log` database contains the logs (Standard and Trace level) that are generated by applications, such as Universal Routing Server and Custom Server, which you can safely delete.

If you are storing strategy `.rbn` files in a database, the Configuration Database Update script (see “Running the Configuration Database Update Script” on [page 215](#)) created an `ird_strategies` table. Do not delete this table if you plan to import strategies into a future routing solution.

Consult with your database administrator(s) about having them delete the database files themselves.



Supplements

Related Documentation Resources

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

Universal Routing

- *Universal Routing 8.0 Reference Manual*, which describes and defines routing strategies, IRD objects, Universal Routing Server and other server functions and options, number translation, pegs, and statistics that are used for routing.
- *Universal Routing 8.0 Business Process User's Guide*, which contains step-by-step instructions for creating interaction workflows (business processes), that direct incoming customer interactions through various processing objects. The goal is to generate an appropriate response for the customer.
- *Universal Routing 8.0 Strategy Samples*, which simplifies strategy configuration for first-time users of the strategy development tool, Interaction Routing Designer. To achieve this goal, this document supplies examples of simple voice and e-mail routing strategies that can be used as general guides during the design stage.
- *Universal Routing 8.0 Routing Application Configuration Guide* (previously *Universal Routing 7.0 Routing Solutions Guide*), which contains information on the various types of routing that can be implemented, including skills-based routing, business-priority routing, and share agent by service level agreement routing. It also summarizes cost-based routing, proactive routing, and a SIP/instant message solution.
- *Universal Routing 7.6 (or later) Cost-Based Routing Configuration Guide*, which documents how to configure Universal Routing Server to use the cost of routing to a target, consisting of Infrastructure cost and/or Resource cost, as addition selection criteria when choosing the right target.

- *Universal Routing 8.0 Interaction Routing Designer Help*, which describes how to use Interaction Routing Designer to create routing strategies. It also describes Interaction Design view, in which you create business processes that route incoming interactions through various processing objects with the goal of generating an appropriate response for the customer.

eServices and Other

- *eServices (Multimedia) 8.0 Deployment Guide*, which includes a high-level overview of features and functions of Genesys Multimedia together with architecture information and deployment-planning materials. It also introduces you to some of the basic concepts and terminology that are used in this product.
- *eServices (Multimedia) 8.0 User's Guide*, which provides overall information and recommendations on the use and operation of Genesys Multimedia.
- *eServices (Multimedia) 8.0 Open Media Interaction Models Reference Manual*, which presents a set of basic interaction models—showing the components that are involved and the messaging (requests, events) among them.
- “Universal Routing and eServices (Multimedia) Log Events” in the *Framework 8.0 Combined Log Events Help*, which is a comprehensive list and description of all events that may be recorded in Management Layer logs.

Genesys

- *Genesys 7.6 Proactive Routing Solution Guide*, which documents a solution that enables you to route outbound_preview interactions to Genesys Agent Desktop proactively, as well as to process Calling List and Do Not Call List records completely, solely from the logic of a routing strategy without agent intervention.
- *Genesys 7 Events and Models Reference Manual*, which provides information on most of the published Genesys events and their attributes, and an extensive collection of models that describe core interaction processing in Genesys environments.
- *Genesys Technical Publications Glossary*, which ships on the Genesys Documentation Library DVD and provides a comprehensive list of the Genesys and computer-telephony integration (CTI) terminology and acronyms that are used in this document.
- *Genesys Migration Guide*, which ships on the Genesys Documentation Library DVD and provides documented migration strategies for Genesys product releases. Contact Genesys Technical Support for more information.

- The Release Notes and Product Advisories for this product, which are available on the Genesys Technical Support website at <http://genesyslab.com/support>.

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

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

Consult these additional resources, as necessary:

- *Genesys Hardware Sizing Guide*, which provides information about Genesys hardware sizing guidelines.
- *Genesys Interoperability Guide*, which provides information on the compatibility of Genesys products with various Configuration Layer Environments; Interoperability of Reporting Templates and Solutions; and Gplus Adapters Interoperability.
- *Genesys Licensing Guide*, which introduces you to the concepts, terminology, and procedures that are relevant to the Genesys licensing system.
- *Genesys Database Sizing Estimator 7.6 Worksheets*, which provides a range of expected database sizes for various Genesys products.

For additional system-wide planning tools and information, see the release-specific listings of System Level Documents on the Genesys Technical Support website, which is accessible from the [system level documents by release](#) tab in the Knowledge Base Browse Documents Section.

Genesys product documentation is available on the:

- Genesys Technical Support website at <http://genesyslab.com/support>.
- Genesys Documentation Library DVD, which you can order by e-mail from Genesys Order Management at orderman@genesyslab.com.

Document Conventions

This document uses certain stylistic and typographical conventions—introduced here—that serve as shorthands for particular kinds of information.

Document Version Number

A version number appears at the bottom of the inside front cover of this document. Version numbers change as new information is added to this document. Here is a sample version number:

```
80fr_ref_06-2008_v8.0.001.00
```

You will need this number when you are talking with Genesys Technical Support about this product.

Screen Captures Used in This Document

Screen captures from the product graphical user interface (GUI), as used in this document, may sometimes contain minor spelling, capitalization, or grammatical errors. The text accompanying and explaining the screen captures corrects such errors *except* when such a correction would prevent you from installing, configuring, or successfully using the product. For example, if the name of an option contains a usage error, the name would be presented exactly as it appears in the product GUI; the error would not be corrected in any accompanying text.

Type Styles

[Table 22](#) describes and illustrates the type conventions that are used in this document.

Table 22: Type Styles

Type Style	Used For	Examples
Italic	<ul style="list-style-type: none"> Document titles Emphasis Definitions of (or first references to) unfamiliar terms Mathematical variables <p>Also used to indicate placeholder text within code samples or commands, in the special case where angle brackets are a required part of the syntax (see the note about angle brackets on page 351).</p>	<p>Please consult the <i>Genesys Migration Guide</i> for more information.</p> <p>Do <i>not</i> use this value for this option.</p> <p>A <i>customary and usual</i> practice is one that is widely accepted and used within a particular industry or profession.</p> <p>The formula, $x + 1 = 7$ where x stands for . . .</p>

Table 22: Type Styles (Continued)

Type Style	Used For	Examples
Monospace font (Looks like teletype or typewriter text)	All programming identifiers and GUI elements. This convention includes: <ul style="list-style-type: none"> • The <i>names</i> of directories, files, folders, configuration objects, paths, scripts, dialog boxes, options, fields, text and list boxes, operational modes, all buttons (including radio buttons), check boxes, commands, tabs, CTI events, and error messages. • The values of options. • Logical arguments and command syntax. • Code samples. Also used for any text that users must manually enter during a configuration or installation procedure, or on a command line.	Select the Show variables on screen check box. In the Operand text box, enter your formula. Click OK to exit the Properties dialog box. T-Server distributes the error messages in EventError events. If you select true for the inbound-bsns-calls option, all established inbound calls on a local agent are considered business calls. Enter exit on the command line.
Square brackets ([])	A particular parameter or value that is optional within a logical argument, a command, or some programming syntax. That is, the presence of the parameter or value is not required to resolve the argument, command, or block of code. The user decides whether to include this optional information.	<code>smcp_server -host [/flags]</code>
Angle brackets (<>)	A placeholder for a value that the user must specify. This might be a DN or a port number specific to your enterprise. Note: In some cases, angle brackets are required characters in code syntax (for example, in XML schemas). In these cases, italic text is used for placeholder values.	<code>smcp_server -host <confighost></code>



Index

Symbols

[] (square brackets)	351
< > (angle brackets)	351

Numerics

3rd Party Media interaction channel	112
-------------------------------------	-----

A

Access Codes tab	
Switch Properties	226
access permissions	213
ACD queue	187, 192
ADDP	76, 79
advanced disconnect detection protocol	79
Advanced tab	
DN	187
External routing point	192
IRD	233
Network Routing Solution	192
agent	
capacity rules	147
defined as Person object	188
profile	34
Agent Capacity	41
agent capacity rules	149, 188
Agent Reservation options	98
agent_reservation	99, 255
reduced	99
reservation_pulling_time	99
transition time	99
treatment_delay_time	99
agent transition time	121, 133
agent-based routing	120, 123, 128
agent-level routing	34
Alias	187
alias	
DN Types	233

Network Destination	232, 233
queue	232, 233
routing point	232, 233
virtual queue	232, 233
angle brackets	351
Annex tab	
Message Server	230, 231, 257
Network T-Server	204
URS	232
Annex tab in Config Mgr	287
Application	
IRD login dialog box	184, 269
application changes and URS	152, 185, 200
application failure management	78
ADDP	78
exception handling	78
application template	187, 192
Apply escalation procedure	286
architecture planning	116
Attach classification Categories sample	286
audience, for document	16
auto-restart	
Custom Server	291
URS	198
Average Handling Time	248

B

backup Custom Server	292
backup routing points	92
backup server	75, 197, 291
bandwidth	117
base product	76, 77
brackets	
angle	351
square	351
business attributes	188, 217
business process	107
businesslines_sample List object	273
business-priority routing	347

Busy

StatAgentOccupancy 249

C

cache (Interaction Server database) . . . 106, 110

calling a custom procedure
 in asynchronous mode 303
 in synchronous mode 303

Calling Lists 44

calls in transition 243

Campaign Groups 273

Capacity Wizard 43

case conversion 120

cbr_enabled_sample 272

CC Analyzer 136

CCPulse+ 136

changes to applications 152, 185, 200

chat mode 111

Chat Server 105

chat/instant messaging 45

CIM (Customer Interaction Management)

Platform 14, 15

Classification Server 107

client side port definition 51, 168, 169, 201, 329

Custom Server 323

URS 201, 315

Command Line

Custom Server 291

URS 197, 198

Command-Line Argument

Custom Server 291

commenting on this document 17

components

Configuration Manager 57, 148, 150

Configuration Server 57, 148, 149

Custom Server 148, 150

DB Server 148, 149, 311

Framework 57

Interaction Routing

Designer 56, 148, 150, 186, 193

License Manager 148, 150

Management Layer 57

Message Server 148, 150, 193, 229

Network T-Server 56, 150, 193

Reporting 147

Stat Server 148, 150, 193, 311

T-Server 148, 150, 311

Universal Routing

Server 56, 148, 150, 186, 193

Components tab

ERS 220, 221, 224

configuration database 130, 131, 134

Configuration Layer 57, 233, 311

Configuration Manager 57, 148, 150

Configuration Manager limitations 189

configuration object changes and

URS 152, 185, 200

Configuration Server 57, 148, 149

master 84, 117

port for IRD 184, 269

proxies 84, 117

Configuration Server Proxy 198

configuring

DN 232

DNs 232

Interaction Routing Designer 210

network switch 225

Network T-Server 194

premise T-Server 228

solution object 219, 223

Stat Server 227

Universal Routing Server 194

Configuring URS to work with ORS 265

connection

Custom Server 121, 133

DAP 121, 133

Interaction Server 121

Message Server 121, 133

Stat Server 121

T-Server 120, 121, 129

Connections tab

Custom Server 292

IRD 211

premise T-Server 229

Stat Server 227

URS 198, 231

constraint 139

consult calls 121, 133

conventions

in document 350

type styles 350

cost-based routing 44, 347

costbasedrouting_sample 272

CTI link messaging 61

Custom Server 34, 56, 136, 138, 148, 150, 152

arguments in rcp.c file 302

auto-restart 291

backup 292

Command Line 291

Command-Line Argument 291

output format 304

planning 125

Primary setting 291

redundancy 291

Router Custom Procedure 295, 296

setting up version 6.1 293

shutdown time 291

startup time 291

UNIX installation 330

wizard installation 322

Working Directory	291
Customer Database	124

D

DAP connection	121, 133
Data Access Point	203
data type	139
database	131, 139
case conversion	120, 125
configuration	130, 131
customer	124
customer lookup	134
DB2	60
mainframe	34
management systems	34
MS SQL	60
name	120
nonrelational	34
Oracle	60
properties	139
relational	34
Sybase	60
Universal Contact Server database	105, 109
<i>See also</i> cache	
Database Access Point	198
database lookups	
routing	120, 123
database properties	139
database-driven routing	33
DB Server	60, 120, 148, 149, 311
for DB lookup	134
planning	124
DB2 database	60
DBMS	
name	120, 124, 131
type	120, 121, 124, 130, 131
Debug	29
Debug level	120, 129
Default Route Type value	187
direct-ani	192
directory number	187, 192, 232
DN	232
ACD queue	187, 192
event_arrive	232
Service Number	193
types	126, 136
types controlled by URS	232
virtual queue	187, 192
DN Groups	
Network Ports	136
Service Numbers	136
DN types	135
premise switch	126
dnis-pooling	192, 227
DNs	

non-configured	41
Do Not Call lists	44
document	
audience	16
change history	18
conventions	350
errors, commenting on	17
version number	350
documentation	
eServices (Multimedia) Deployment Guide	348
Events and Models Reference Manual	348
IRD Help	348
Log Events Help	348
Multimedia User's Guide	348
Proactive Routing Solution Guide	348
Universal Routing Business Process User's Guide	347
Universal Routing Cost-Based Routing Configuration Guide	347
Universal Routing Reference Manual	347
Universal Routing Routing Application Configuration Guide	347
Universal Routing Strategy Samples	347

E

education	17
E-mail Accounts	218
E-mail Server	105, 144
escalation procedure	286
eServices	15
eServices definition	103
Event messages	61
event_arrive	98
event_arrive option	232, 265
EventEstablished	232, 233
EventRinging	232, 233
EventRouteRequest	232, 233
EventRouteRequest message	63
EventTreatmentRequired	232, 233
exception handling	78
exporting strategies	268
external routing	192, 204, 227
external routing type	
direct-ani	192
route	192
External Service object	218

F

failover	81, 90, 96
failure management	78
field	
data	139

- description 139
- name 139
- font styles
 - italic 350
 - monospace 351
- Framework 57, 115
 - Configuration Manager 57
 - Configuration Server 57
 - Management Layer 57

G

- General tab
 - ERS 220, 224
 - IRD 211
 - Switch Properties 226
 - URS 195
- Genesys Administrator 344
- Genesys Content Analyzer 108
- Genesys Outbound Contact 44, 273
- Genesys University 17
- Get credit card numbers 286
- getcity_sample 272
- graphical portion of strategy 287

H

- Handle fax interactions 286
- help file
 - Interaction Routing Designer Help 348
- High Availability 39, 76, 85, 197, 199
- historical reporting 136
- Host name
 - IRD login dialog box 184, 269
- Hot Standby 75, 76, 77, 85, 86, 87, 88, 89
- How To
 - Business Processes 286
- HTTP Bridge 53
- HTTP bridge options 204
- HTTP Interface 207, 208
- http options 206
- http_bridge 209
- http_port 209

I

- Identify contact and create interaction 286
- IM sessions 45
- importing strategies 268
- Install Shield Silent 328
- installation package directories 312
- installation scripts 328
- installing Custom Server
 - Unix server 330

- installing Interaction Routing Designer 321
- installing URS
 - overview 311
 - UNIX 328
 - Windows 313
- instant messaging 45
- intended audience 16
- Inter Server Call Control 192, 204, 227
- interaction attributes 217
- Interaction Design Shortcut Bar 213
- Interaction Routing Designer 56, 148, 150, 193
 - configuring 210
 - installing 321
 - login 268
- Interaction Server 106, 144
 - URS Connections 119
- interactions 90, 105
- Interactive Voice Response (IVR) Interface 173
- Interactive Voice Response Option 126, 137
- interface for T-Serverless routing 46
- IRD
 - access permission 213
 - aliases 233
 - login 268
 - monitoring (Message Server) 231
 - objects used in business processes 284
- ISCC 192, 204, 227
- italics 350
- IVR Interface 63, 137
- IVR Server load balancing 262
- IVR-In-Front 262

K

- Kannel 105

L

- Language Business Attribute 218
- LBR_DEST key in User Data 252
- LCA 57, 58, 79, 81, 86, 337
- license file 198
- License Manager 148, 150
- limitations
 - Configuration Manager 189
 - IRD 189
- list object samples 273
- load balancing
 - IVR Server 262
 - multiple URSs 235, 236
 - Routing object 242
 - statistics 244
- load distribution 71
- Load Distribution Server 59, 82, 236
- loading strategies 29

Local Control Agent 57, 337
 location of rbn file 287
 log configuration
 DB Server 121, 131
 Message Server 121, 132
 Universal Routing Server 121, 132
 logging (Message Server) 230
 login
 Interaction Design via IRD 268
 IRD 268
 logs
 Management Layer 348
 URS Configuration Wizard 164
 Longest Available Agent 249, 251

M

Management Layer 57, 58, 311
 Local Control Agent 57
 Solution Control Interface 58
 Solution Control Server 58
 Management Layer logging 348
 media channels 15
 Media Layer 57
 Media Type 37, 218
 Message Server 58, 116, 124, 148, 150, 193, 229
 IRD monitoring 231
 IVR load balancing 264
 logging 230
 planning 132
 Router self-awareness 256
 Microsoft Live Communication Server 45
 Microsoft SQL database 60
 model 108
 monitoring strategies 29
 monospace font 351
 multi-Campaign agent management 273
 Multimedia 15
 Multimedia definition 103
 multi-site routing 36
 mySAP Data Access Component 205

N

network
 switch 177, 193, 225
 T-Server 123, 129, 178
 Network Destination 232, 233
 Network Routing 89
 Network Routing Platform 89
 Network Routing reporting 147
 Network T-Server 56, 57, 150, 193, 194, 311, 339
 Non-Configured DNS 41
 nonrelational databases 34
 non-routable 90

non-SQL database 56, 152
 Non-Stop Operation 76, 78, 341
 number of processors 117

O

object
 objects used in business processes 284
 solution 219, 223
 object browser 32
 Occupancy rate 249
 on_router_activated 82, 96
 Open Media concept 218
 Open Media interaction channel 113
 opening a strategy 287
 operational data 105
 options 96
 Options tab
 Custom Server 292
 Network T-Server 204
 premise T-Server 229
 Stat Server 227
 URS 199
 Oracle database 60
 Orchestration Server (ORS) 265
 ORS (Orchestration Server) 265
 Outbound Contact 44, 273
 Outbound objects 44
 outbound_sample sample strategy 272
 OutboundMultiCampaign.ooo 273

P

paging mode 111
 password
 IRD login dialog 184, 269
 path to strategy 287
 PBX 61
 permission for access 213
 pickup_calls 96
 PIN Business Attribute 218
 Place the interaction into the workbin 287
 Platform SDK 105
 Port
 IRD login dialog box 184, 269
 port info for URS 195
 premise
 switch 125, 135, 187, 192
 switch DN types 126, 135
 switching office 125, 135
 T-Server 125, 135, 228
 pre-start information 333
 Primary setting
 Custom Server 291
 private branch exchange 118

proactive customer contact 44
 processors 117
 prohibited symbols 190

Q

queue 106, 232, 233
 queue-based routing 120, 123, 128

R

RAID subsystem 117
 rbn file 287
 RCP library 293
 rcp.c
 Custom Server 293
 rcp.h
 Custom Server 293
 Real-Time Metrics Engine 147
 real-time reporting 136
 Reason Code 218
 recovering 90
 redundancy 197
 Custom Server 291
 Redundancy level 76, 79
 redundancy type
 Hot Standby 85
 Warm Standby 79
 redundant configurations 337
 reject-subsequent-request T-Server option 100
 reporting 147
 CD 147
 configuring in Wizard 173
 e-mail and chat interactions 147
 historical 136
 real-time 136
 Reporting Options
 historical 126
 real-time 126
 Request messages 61
 request-collection-time T-Server option 100
 reservation-time T-Server option 100
 reserved symbols 190
 Resource Capacity Planning 43
 Resource Capacity Wizard 149, 152
 Ring-No-Answer 43
 rlu_mapc 274
 rlu_mapc_group 275
 rlu_mapc_group_route 276
 rlu_mapcar 274
 rlu_select_max 277
 rlu_select_vq_route 278
 Router Connector License 50
 Router openness 46
 Router Self-Awareness 248

router_ha_option 197
 routing
 agent profile 34
 agent-level 34
 database-driven 33
 multi-site 36
 service level 35
 skill-based 34
 statistical 36
 strategy samples 347
 routing features 123
 agent-based 120, 123, 128
 database lookups 120, 123
 queue-based 120, 123, 128
 skill-based 120, 123, 128
 voice treatment 120, 123, 129
 routing point 232, 233
 routing strategy 107
 graphical portion 287
 routing strategy, cannot view 287
 RStatCallsInQueue 247
 RStatExpectedLBEWTLAA 251
 RStatExpectedLoadBalance 249
 RStatLBEWTLAA 250
 RStatLoadBalance 247

S

sample subroutines 273, 274
 Samples 144, 214, 347
 strategy
 samples 268
 Samples.zcf file 268
 SAP RFC functions 205
 Screen multiple rules 287
 screening rule 107
 Scripts object subfolders 214
 SCS 223
 SCXML 46, 49
 SCXML strategy support 265
 Secure port configuration 195
 security banner message 139
 Server Info tab
 premise T-Server 228
 Stat Server 227
 URS 195
 server speed 117
 service control point 116
 service interruption 90
 service level routing 35
 servicelevelagreement_sample 273
 Services Layer 60
 session (chat) mode 111
 Session Initiation Protocol 45
 Share Agents by Service Level

- Agreement 43, 347
 - sharing strategies 39
 - Shutdown time
 - Custom Server 291
 - silent installation 328
 - SIP Server 45
 - skill expression 35
 - limitation 190
 - skill-based routing 34, 120, 123, 128
 - sla1_sample 272
 - sla2_sample 272
 - SMPP 105
 - SMS Server 105
 - SOAP options 206
 - SOAP/XML 204
 - Solution Control Interface 58, 219, 223, 335
 - planning 126
 - Solution Control Server 58, 223
 - solution object 219, 223
 - square brackets 351
 - sServices/Multimedia Open Media Interaction
 - Models 348
 - standard response 105, 107
 - Start Info tab
 - Custom Server 291
 - URS 197
 - starting
 - Custom Server 339
 - Interaction Routing Designer
 - Windows 338
 - Network T-Server 339
 - Solution Control Interface 335
 - Universal Routing Server 337
 - Universal Routing Server in Unix 338
 - Universal Routing Server using SCI 335
 - startup priority 222
 - Startup time
 - Custom Server 291
 - Stat Server 60, 124, 148, 150, 227, 311
 - Real-Time Metrics Engine 147, 152
 - State Enabled
 - IRD 211
 - StatEstimatedWaitingTime 246
 - statistic reconciliation 95
 - statistical routing 36
 - statistics
 - loading balancing 244
 - StatLoadBalance 246
 - Stop Processing Reason 218
 - stopping
 - Enterprise Routing solution 340
 - Interaction Routing Designer 341
 - storing rbn file 287
 - strategy
 - agent-level routing 34
 - can't access rbn file 287
 - database-driven routing 33
 - definition 23
 - graphical portion 287
 - List view 269
 - monitoring 29
 - objects used in business processes 284
 - samples 347
 - SCXML-based 265
 - sharing 39
 - strategy export and import 268
 - strategy in IRD 268
 - strategy option 265
 - strategy viewer 32
 - subfolders in CME 214
 - subroutine samples 273, 274
 - switch
 - DN types
 - premise 135
 - network 177, 193
 - premise 125, 135, 187, 192
 - switching office 186
 - premise 125, 135
 - switching offices 192
 - Sybase database 60
 - symbols 190
 - system recovery 95
- T**
- table 139
 - field 139
 - format 136
 - ird_strategies 287
 - template, application 187, 192
 - Tenants tab
 - Stat Server 227
 - URS 195
 - Third Party Server 218
 - timeout feature 212
 - T-Library 61
 - T-Library events 232
 - trace on/off 29
 - training 17
 - Training Server 107
 - T-Server 148, 150, 311
 - events 61
 - Media Layer 57
 - network 123, 129, 150, 178, 194
 - premise 125, 135, 228
 - T-Server connection 120, 121, 129
 - type styles
 - conventions 350
 - italic 350
 - monospace 351
 - typographical styles 350

U

unlogin.dll	139
uninstall	
manually on UNIX-based operating systems	346
manually on Windows	345
with Genesys Administrator	344
Universal Callback Server	219
Universal Contact Server	105
Universal Contact Server Manager	105
Universal Routing Server	56, 148, 150, 193, 194
auto-restart	198
backup	80
Configuration Server Proxy	198
configuring	194
planning	124
primary	132
shutdown time	198
silent installation	328
startup time	198
Windows installation	313
wizard	163
Unix	
Custom Server	293, 294
Universal Routing Server	328
URS delays	152, 185, 200
URS options	84, 96
Agent Reservation	96
event_arrive	96
failover options	96
on_router_activated	84
pickup_calls	84, 96
URS Web Interface	46
use_ivr_info option	265
User Data	
LBR_DEST key	252
user interface	
Configuration Manager	57
Interaction Routing Designer	56
Solution Control Interface	58
User name	
IRD login dialog box	184, 269
User password	
IRD login dialog box	184, 269

V

verbose option	205
version identification	342
version numbering, document	350
video boards	117
viewing a strategy	287
virtual queue	232, 233
virtual queues	147
Voice	137

Voice Callback Server	56
Voice Treatment Option	173
Voice Treatment Port	187
voice treatment routing	120, 123, 129
VoIP	137
VoIP Option	
Gateway	126
Voice Terminal Endpoints	126

W

Warm Standby	79, 197, 199
Web API Server	105, 110
Web Service object	204
Weighted Round Robin distribution mode	82
Windows	321
Interaction Routing Designer	338
Universal Routing Server installation	313
Wizard	
Custom Server installation	322
ERS	152
IRD	172
Manager	153, 175
Network T-Server	178
NRS	175
Switch	178
Universal Routing Server	163
URS installation	314
Wizard Manager	120
workbin	34
sample	286
workflow design	107
Workforce Database	124, 131
Workforce Management	73
workforce routing	34
Working Directory	
Custom Server	291
URS	197

X

XML	46, 49, 265
-----	-------------