CISCO

ICM Scripting and Media Routing Guide
Cisco ICM/IPCC Enterprise & Hosted Editions
Release 7.0(0)

February 2009

Table of Contents

List of Figures

Preface

Purpose

This manual describes how to use the Cisco Intelligent Contact Manager/IP Contact Center software Script Editor tool to create and maintain routing and administrative scripts.

Audience

This document is intended for CIsco ICM/IPCC system managers. A system manager must have a general understanding of contact center operations and management, and specific information about the contact centers and carrier networks connected to the Cisco ICM/IPCC system.

Organization

Chapter	Description
Chapter 1, Common Tasks (page 9)	Contains information about common tasks you perform in Script Editor. This chapter does not contain information on every possible task you can perform. For more information on Script Editor, see the online help.
Chapter 2, Call Types, Contact Data, and Scripting (page 27)	Discusses how to use CallTypes and Contact Data when writing ICM scripts.
Chapter 3, Categorizing Contacts (page 35)	Contains information relating to categorizing contacts based on Call Types, Call Type qualifiers, Time and Date, Branching, External Data, and External Applications.
Chapter 4, Selecting Routing Targets (page 57)	Contains information on how to determine the destination for contacts using Script Editor nodes to specify how a contact is routed to a target.

Related Documentation

Obtaining Documentation

Cisco documentation and additional literature are available on Cisco.com. Cisco also provides several ways to obtain technical assistance and other technical resources. These sections explain how to obtain technical information from Cisco Systems.

Cisco.com

You can access the most current Cisco documentation at this URL:

http://www.cisco.com/techsupport

You can access the Cisco website at this URL:

http://www.cisco.com

You can access international Cisco websites at this URL:

http://www.cisco.com/public/countries_languages.shtml

Product Documentation DVD

Cisco documentation and additional literature are available in the Product Documentation DVD package, which may have shipped with your product. The Product Documentation DVD is updated regularly and may be more current than printed documentation.

The Product Documentation DVD is a comprehensive library of technical product documentation on portable media. The DVD enables you to access multiple versions of hardware and software installation, configuration, and command guides for Cisco products and to view technical documentation in HTML. With the DVD, you have access to the same documentation that is found on the Cisco website without being connected to the Internet. Certain products also have .pdf versions of the documentation available.

The Product Documentation DVD is available as a single unit or as a subscription. Registered Cisco.com users (Cisco direct customers) can order a Product Documentation DVD from the Ordering tool or Cisco Marketplace.

Cisco Ordering Tool:

http://www.cisco.com/en/US/partner/ordering/

Cisco Marketplace:

http://www.cisco.com/go/marketplace/

Ordering Documentation

Beginning June 30, 2005, registered Cisco.com users may order Cisco documentation at the Product Documentation Store in the Cisco Marketplace at this URL::

http://www.cisco.com/go/marketplace/

Cisco will continue to support documentation orders using the Ordering tool:

- Registered Cisco.com users (Cisco direct customers) can order Cisco product documentation from the Ordering tool:

  http://www.cisco.com/en/US/partner/ordering/

- Instructions for ordering documentation using the Ordering tool are at this URL:

  http://www.cisco.com/univercd/cc/td/doc/es_inpck/pdi.htm

- Nonregistered Cisco.com users can order documentation through a local account representative by calling Cisco Systems Corporate Headquarters (California, USA) at 408 526-7208 or, elsewhere in North America, by calling 1 800 553-NETS (6387).

Documentation Feedback

You can rate and provide feedback about Cisco technical documents by completing the online feedback form that appears with the technical documents on Cisco.com.

You can rate and provide feedback about Cisco technical documents by completing the online feedback form that appears with the technical documents on Cisco.com.

You can submit comments by using the response card (if present) behind the front cover of your document or by writing to the following address:

Cisco Systems Attn: Customer Document Ordering 170 West Tasman Drive San Jose, CA 95134-9883

We appreciate your comments.

Cisco Product Security Overview

Cisco provides a free online Security Vulnerability Policy portal at this URL: **http://www.cisco.com/en/US/products/products_security_vulnerability_policy.html**

From this site, you can perform these tasks:

- Report security vulnerabilities in Cisco products.

- Obtain assistance with security incidents that involve Cisco products.

- Register to receive security information from Cisco.

A current list of security advisories and notices for Cisco products is available at this URL: **http://www.cisco.com/go/psirt**

If you prefer to see advisories and notices as they are updated in real time, you can access a Product Security Incident Response Team Really Simple Syndication (PSIRT RSS) feed from this URL: **http://www.cisco.com/en/US/products/products_psirt_rss_feed.html**

Reporting Security Problems in Cisco Products

Cisco is committed to delivering secure products. We test our products internally before we release them, and we strive to correct all vulnerabilities quickly. If you think that you might have identified a vulnerability in a Cisco product, contact PSIRT:

- Emergencies - security-alert@cisco.com

An emergency is either a condition in which a system is under active attack or a condition for which a severe and urgent security vulnerability should be reported. All other conditions are considered nonemergencies.

- Nonemergencies - psirt@cisco.com

In an emergency, you can also reach PSIRT by telephone:

- 1 877 228-7302

- 1 408 525-6532

Note: We encourage you to use Pretty Good Privacy (PGP) or a compatible product to encrypt any sensitive information that you send to Cisco. PSIRT can work from encrypted information that is compatible with PGP versions 2.x through 8.x. Never use a revoked or an expired encryption key. The correct public key to use in your correspondence with PSIRT is the one that has the most recent creation date in this public key server list: **http://pgp.mit.edu:11371/ pks/lookup?search=psirt%40cisco.com&op=index&exact=on**

The link on this page has the current PGP key ID in use.

Obtaining Technical Assistance

Cisco Technical Support provides 24-hour-a-day award-winning technical assistance. The Cisco Technical Support & Documentation website on Cisco.com features extensive online support resources. In addition, if you have a valid Cisco service contract, Cisco Technical Assistance Center (TAC) engineers provide telephone support. If you do not have a valid Cisco service contract, contact your reseller.

Cisco Technical Support & Documentation Website

The Cisco Technical Support & Documentation website provides online documents and tools for troubleshooting and resolving technical issues with Cisco products and technologies. The website is available 24 hours a day, at this URL:

http://www.cisco.com/techsupport

Access to all tools on the Cisco Technical Support & Documentation website requires a Cisco.com user ID and password. If you have a valid service contract but do not have a user ID or password, you can register at this URL:

http://tools.cisco.com/RPF/register/register.do

Note: Use the Cisco Product Identification (CPI) tool to locate your product serial number before submitting a web or phone request for service. You can access the CPI tool from the Cisco Technical Support Website by clicking the **Tools & Resources** Tools. Choose **Cisco Product Identification Tool** from the Alphabetical Index drop-down list, or click the **Cisco Product Identification Tool** RMAs. The CPI tool offers three search options: by product ID

or model name; by tree view; or for certain products, by copying and pasting **show** command output. Search results show an illustration of your product with the serial number label location highlighted. Locate the serial number label on your product and record the information before placing a service call.

Submitting a Service Request

Using the online TAC Service Request Tool is the fastest way to open S3 and S4 service requests. (S3 and S4 service requests are those in which your network is minimally impaired or for which you require product information.) After you describe your situation, the TAC Service Request Tool provides recommended solutions. If your issue is not resolved using the recommended resources, your service request is assigned to a Cisco TAC engineer. The TAC Service Request Tool is located at this URL:

http://www.cisco.com/techsupport/servicerequest

For S1 or S2 service requests or if you do not have Internet access, contact the Cisco TAC by telephone. (S1 or S2 service requests are those in which your production network is down or severely degraded.) Cisco TAC engineers are assigned immediately to S1 and S2 service requests to help keep your business operations running smoothly

To open a service request by telephone, use one of the following numbers:

- Asia-Pacific: +61 2 8446 7411 (Australia: 1 800 805 227)

- EMEA: +32 2 704 55 55

- USA: 1 800 553-2447

For a complete list of Cisco TAC contacts, go to this URL:

http://www.cisco.com/techsupport/contacts

Definitions of Service Request Severity

To ensure that all service requests are reported in a standard format, Cisco has established severity definitions.

Severity 1 (S1) -- Your network is down, or there is a critical impact to your business operations. You and Cisco will commit all necessary resources around the clock to resolve the situation.

Severity 2 (S2) -- Operation of an existing network is severely degraded, or significant aspects of your business operation are negatively affected by inadequate performance of Cisco products. You and Cisco will commit full-time resources during normal business hours to resolve the situation.

Severity 3 (S3) -- Operational performance of your network is impaired, but most business operations remain functional. You and Cisco will commit resources during normal business hours to restore service to satisfactory levels.

Severity 4 (S4) -- You require information or assistance with Cisco product capabilities, installation, or configuration. There is little or no effect on your business operations.

Obtaining Additional Publications and Information

Information about Cisco products, technologies, and network solutions is available from various online and printed sources.

- Cisco Marketplace provides a variety of Cisco books, reference guides, and logo merchandise. Visit Cisco Marketplace, the company store, at this URL:

  http://www.cisco.com/go/marketplace/

- Cisco Press publishes a wide range of general networking, training and certification titles. Both new and experienced users will benefit from these publications. For current Cisco Press titles and other information, go to Cisco Press at this URL:

  http://www.ciscopress.com

- *Packet* magazine is the Cisco Systems technical user magazine for maximizing Internet and networking investments. Each quarter, Packet delivers coverage of the latest industry trends, technology breakthroughs, and Cisco products and solutions, as well as network deployment and troubleshooting tips, configuration examples, customer case studies, certification and training information, and links to scores of in-depth online resources. You can access Packet magazine at this URL:

  http://www.cisco.com/packet

- *iQ Magazine* is the quarterly publication from Cisco Systems designed to help growing companies learn how they can use technology to increase revenue, streamline their business, and expand services. The publication identifies the challenges facing these companies and the technologies to help solve them, using real-world case studies and business strategies to help readers make sound technology investment decisions. You can access iQ Magazine at this URL:

  http://www.cisco.com/go/iqmagazine

- *Internet Protocol Journal* is a quarterly journal published by Cisco Systems for engineering professionals involved in designing, developing, and operating public and private internets and intranets. You can access the Internet Protocol Journal at this URL:

  http://www.cisco.com/ipj

- World-class networking training is available from Cisco. You can view current offerings at this URL:

  http://www.cisco.com/en/US/learning/index.html

Obtaining Additional Publications and Information

Chapter 1

Common Tasks

This topic contains information about common tasks you perform in Script Editor. This topic does not contain information on every possible task you can perform. For more information on Script Editor, see the online help.

This chapter contains the following topics:

How to Use Toolbars

Viewing Toolbars

You control which toolbars appear in the application window by using the **View** menu:

Step 1   From the **View** menu, select **Toolbar**.

The submenu opens, listing the four toolbars. Toolbars that are currently open are checked.

Step 2   To open a toolbar, from the submenu, select the unchecked toolbar.

Step 3   To close a toolbar, from the submenu, select the checked toolbar.

You can also control the size of the toolbar icons from the Toolbar submenu:

- To display large toolbar icons, with text, select Large Icons when it is not checked.

- To display small toolbar icons, without text, select Large Icons when it is checked.

Note: Toolbar icons in this document are always shown as large icons with text.

The Palette

You display the Palette by clicking in the Main toolbar or by selecting **Palette** from the **View** menu.

The Palette contains the icons representing the nodes you use in scripts.

The available nodes are divided into four tabs:

- General (page 11)

- Routing (page 12)

- Targets (page 13)

- Queue (page 14)

Using the Palette to Build a Script

You use the Palette to build a script by moving icons representing different nodes into the script workspace.

You do this by:

- Placing cursor over object and pressing the left mouse button, then dragging the object into the workspace and releasing the mouse button.

- Placing cursor over object and clicking the left mouse button, then moving the cursor into the workspace and clicking the mouse button again. To insert more of the same node, reposition cursor and click again.

General Tab

Following is the General tab of the Palette:

Figure 1: General Tab of the Palette

The General tab contains icons for the following nodes:

- Comment (page 206)

- Day of Week (page 43)

- DB Lookup (page 51)

- End (page 85)

- Gateway (page 54)

- Go To Script (page 46)

- If (page 46)

- Line Connector (page 207)

- Percent Allocation (page 46)

- Set Variable (page 133)

- Skill Group (page 60)

- Start (page 205)

- Switch (page 46)

- Time (page 43)

Routing Tab

Following is the Routing tab of the Palette:

Figure 2: Routing Tab of the Palette
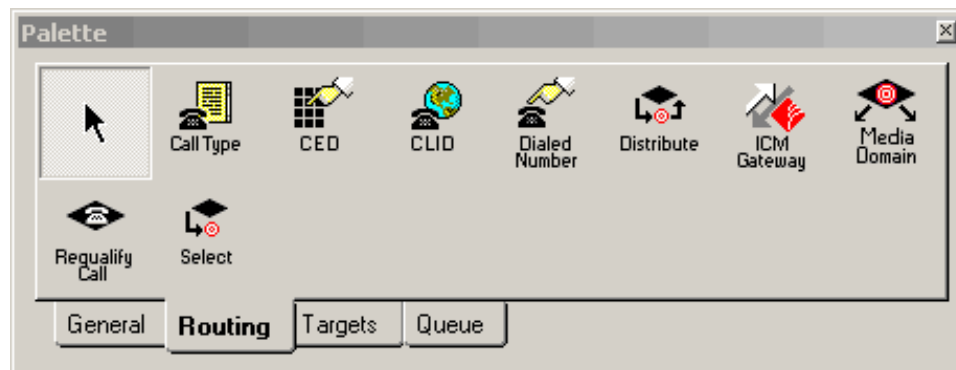
The Routing tab contains icons for the following nodes:

- Call Type (page 35)

- Caller-Entered Digits (page 37)

- Calling Line ID (page 37)

- Dialed Number (page 37)

- Distribute (page 78)

- ICM Gateway (page 84)

- Media Routing Domain (page 125)

- Requalify Call (page 35)

- Select (page 73)

Targets Tab

Following is the Targets tab of the Palette:

Figure 3: Targets Tab of the Palette

The Targets tab contains icons for the following nodes:

- Agent (page 59)

- Agent to Agent (page 82)

- Announcement (page 66)

- Busy (page 69)

- Divert Label (page 71)

- Enterprise Service (page 65)

- Enterprise Skill Group (page 63)

- Label (page 70)

- Release Call (page 85)

- Ring (page 69)

- Route Select (page 79)

- Scheduled Select (page 68)

- Service (page 62)

- Skill Group (page 60)

- Termination (page 85)

Queue Tab

Following is the Queue tab of the Palette:

Figure 4: Queue Tab of the Palette
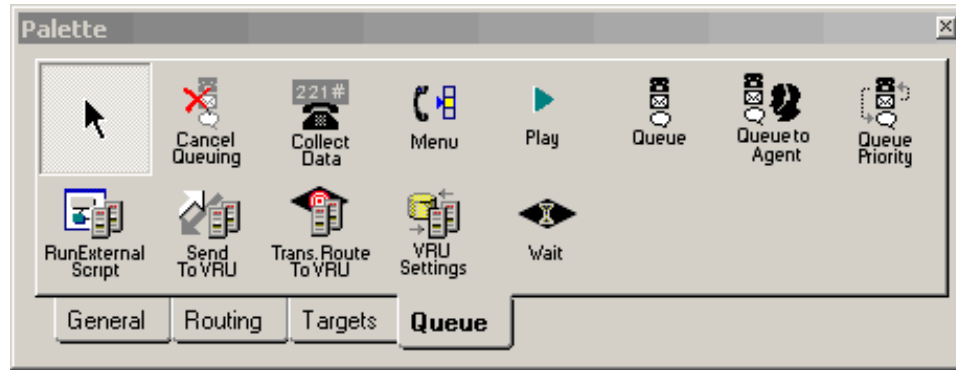
The Queue tab contains icons for the following nodes:

- Cancel Queuing (page 109)

- Collect Data (page 110)

- Menu (page 113)

- Play (page 116)

- Queue (page 105)

- Queue Priority (page 108)

- Queue to Agent (page 127)

- Run External Script (page 102)

- Send To VRU (page 97)

- Translation Route to VRU (page 98)

- VRU Settings (page 119)

- Wait (page 120)

How to Create a Routing Script

Step 1   In Script Editor, Select **File > New** or click **New**. You are prompted to select to create a Routing Script or Administrative Script:

Figure 5: File > New Dialog Box
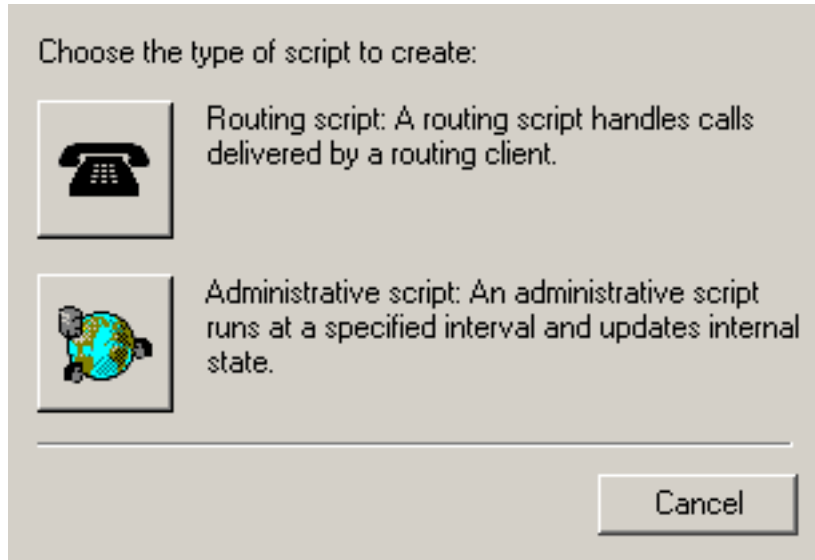
Step 2   Click

The new script opens in the Edit window, with a Start node (page 205).

Step 3   Build the script.

Step 4   To save the script, select **File > Save** or click **Save**. You are prompted for a script name.

How to Add Comments to a Node

Most nodes have a Comment tab:

Figure 6: Comment Tab

Step 1   You can add a comment in the **Enter text** field.

Step 2   You can also select the location in the node you want the comment to appear in by selecting a radio button in the **Text justification** area.

How to Specify Connection Label Location for a Node

Most nodes have a **Connection Labels** tab:

Figure 7: Connection tab

Step 1   You can specify the location of connection labels when viewing a script in monitor mode by moving the slider in the Label position area. Move the slider to:

a.   **Origin**, for the connection label to display close to the node you are editing.

b.   **Destination**, for the connection label to display close to the targeted node.

c.   **Center**, for the connection label to display between the nodes.

Step 2   You can also select no to show the connection label by clearing the Display monitor labels checkbox.

How to Validate Scripts

Step 1   To validate a single script, with the script open in the active window, select **Script > Validate** or on the toolbar click the Validate Icon.

Step 2   To validate multiple scripts, select **Script > Validate All** or on the toolbar, click the Validate All Icon.

You are prompted to choose between validating active versions of all scripts or all the opened scripts:

Figure 8: Validate All Query Dialog
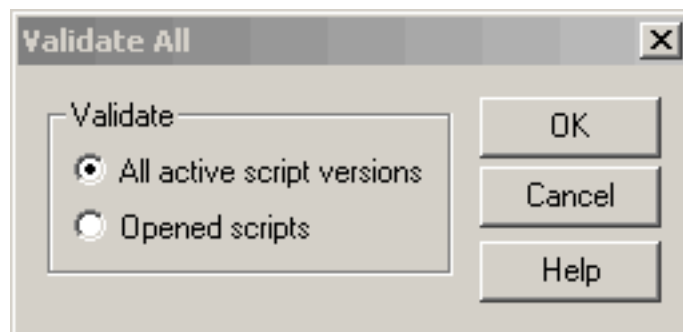
Step 3   Make the appropriate selection and click **OK** in the window.

a.   If a script is valid, a dialog box states it is valid.

b.   If the script is not valid, the Validate Script dialog box opens, with a list of the errors. When you select an error, the node where the error occurs is highlighted in the Edit window.

How to Open Script Explorer

Step 1   In Script Editor, select **File > Script Explorer** or on the toolbar, click the Explorer Icon

The Script Explorer dialog box opens, listing scripts by customer and business entity:
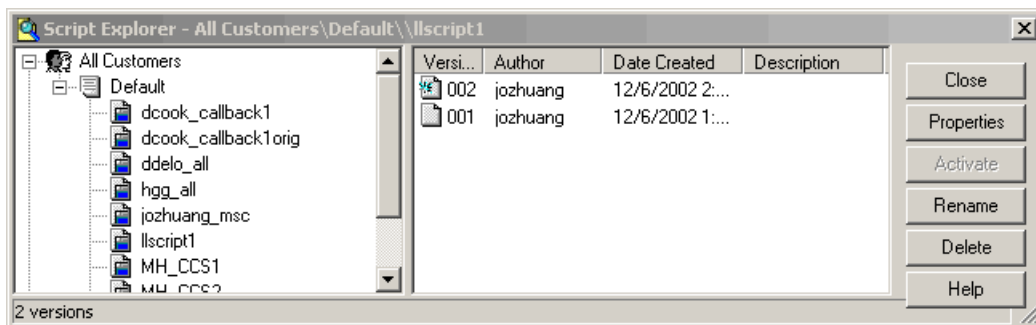
You can then set the active version of the script, view its properties, rename it, or delete it. See the online help for more information.

How to Schedule a Routing Script

You schedule a script by associating it with a as follows:

Step 1   Select **Script > Call Type Manager**. The Call Type Manager dialog box opens.

Figure 9: Call Type Manager Dialog Box - Call Directory Tab
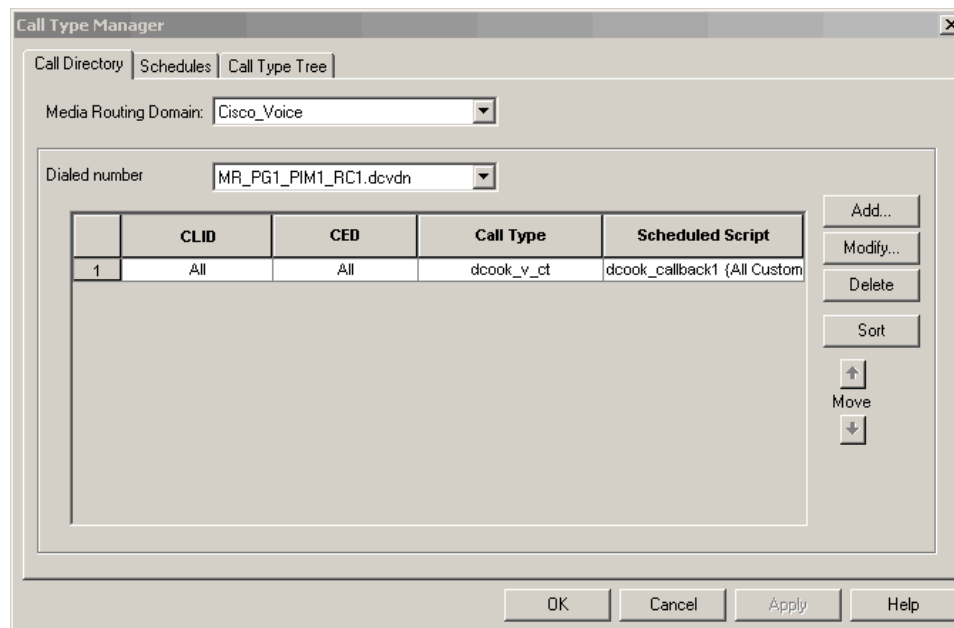
Step 2   Select the Schedules Tab:

Figure 10: Call Type Manager Dialog Box - Schedules Tab
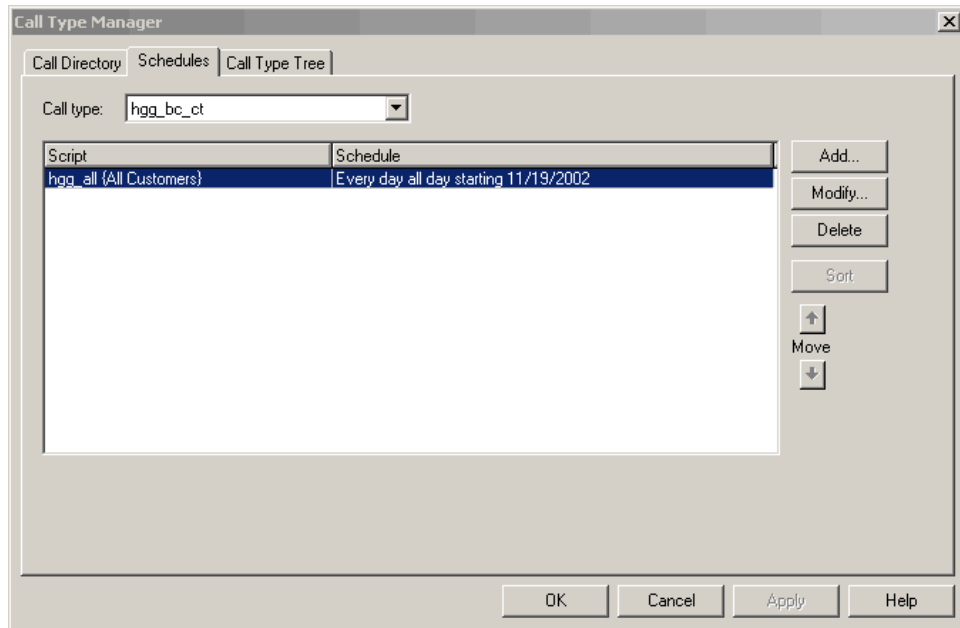
Step 3   Select the Call Type (page 35) to associate with the script.

Step 4   Click **Add**. The Add Call Type Schedule dialog box opens.

Step 5   In the Script tab, select the script to schedule:

Figure 11: Add Call Type Dialog Box - Script Tab
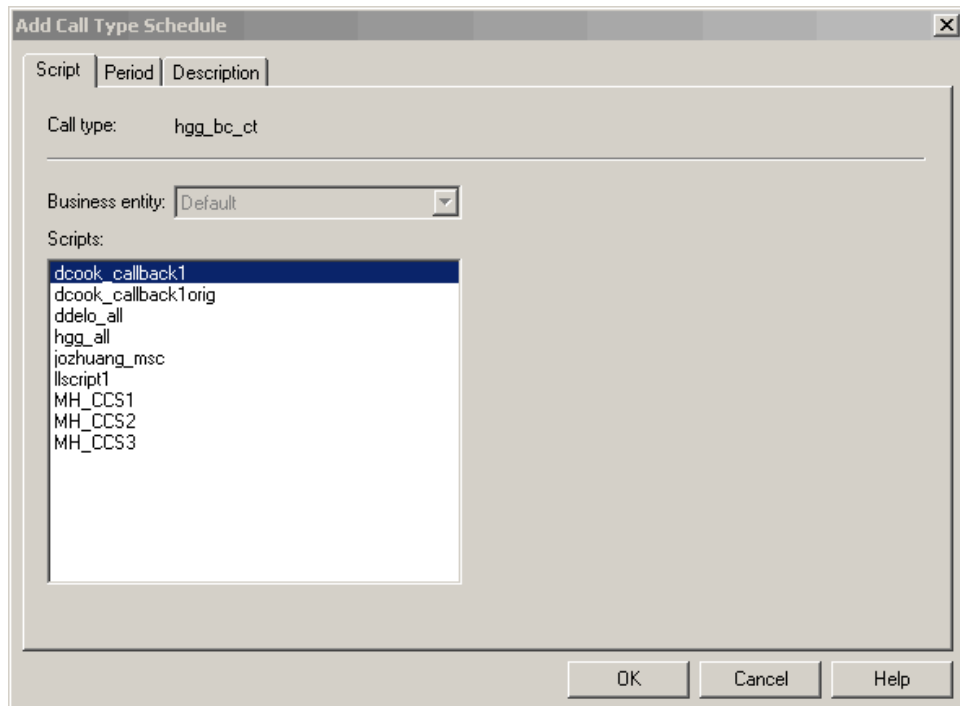
Step 6   In the **Period** tab, enter information to define the period for which the schedule will be in effect:

Figure 12: Add Call Type Schedule Dialog Box - Period Tab
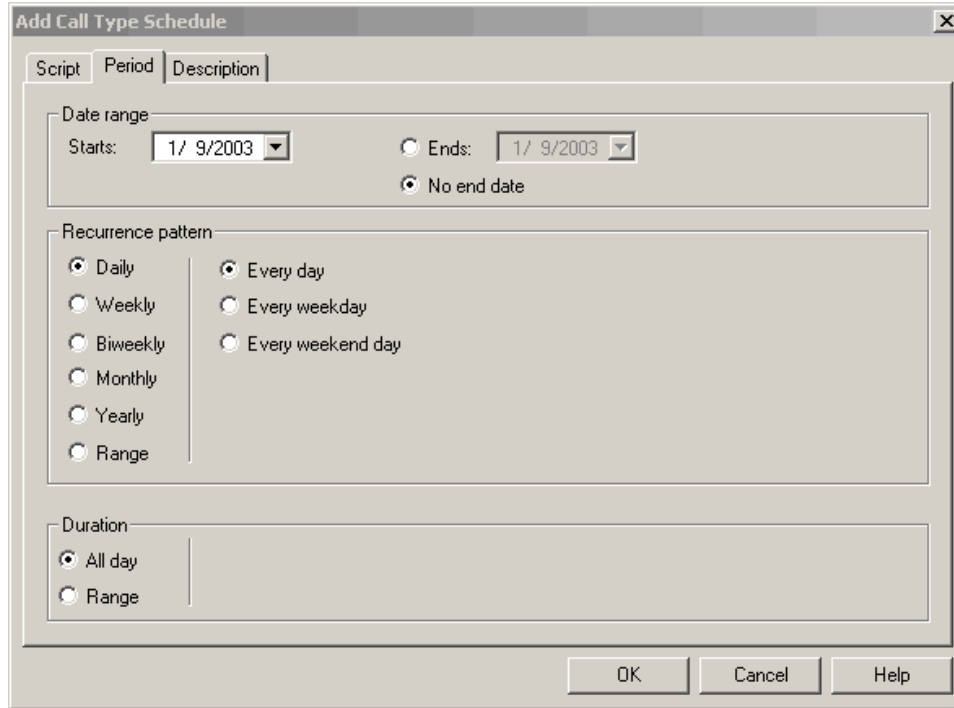
Step 7   Optionally, in the **Description** tab, enter a description of the schedule.

Step 8   Click **OK** in the Add Call Type Schedule dialog box.

Step 9   Click **OK** in the Call Type Manager dialog box.

Note: The schedule is not saved until you click OK in the Call Type Manager dialog box.

Viewing Modes

You can view a script in four different modes:

- **Browse** - Allows you to view the script, but not make any changes.

- **Edit** - Allows you to edit the script.

- **Monitor** - Allows you to monitor the script

- **Quick Edit** - Allows you to make certain modifications to a script, with the following guidelines:

- In Quick Edit mode:

    – You cannot add or delete a node

    – You can adjust most of the properties of the script nodes selected in the Node Control table of your assigned feature control set. However, any properties of the selected nodes

that change the structure of a script or that reset previous reporting data cannot be edited in Quick Edit mode.

- As a Quick Edit Only User:

    – You can only edit scripts through Quick Edit mode.

    – You cannot create or delete a script

    – You can access the Properties of any script node in any mode by either right clicking on the node and selecting Properties, or by double-clicking on the node.

    – You cannot edit the Call Type Manager dialog (Script > Call Type Manager).

    – You cannot edit the Administrative Manager dialog (Script > Administrative Manager).

    – You cannot edit the Custom Functions dialog (Script > Custom Functions).

    – You can select the viewing mode from the Scripting toolbar (page 10), or from the Script menu.

Using the Find Nodes Option

ICM Script Editor is a tool used to create, modify, and schedule routing scripts. A script consists of a series of nodes. When managing large, complex scripts, the Find Nodes option makes it easier to identify/access the node you want to view.

Use the Find Nodes option to find specific script node(s) by:

- **Node ID**

- **Object**

- **Node Type**

- **String**

Use the Find Nodes option to find script nodes in:

- the **Current script**

- **All active script versions**

- **All open scripts**

Find Nodes and Nodes Found Dialog Boxes

Most properties of the "Find Nodes" dialog box are common and appear regardless of the "Find Nodes By" selection. However, as each of the options in the "Find Nodes By" section is selected, the appearance of the "Find Nodes" dialog box changes. The "Nodes Found" dialog box displays the results of your "Find" operation. See the following for the "Find Node" dialog box and "Nodes Found" dialog box property descriptions.

Note:

- Keyboard equivalents are indicated in parenthesis after each field or button having one

- Use **Tab/Shift**+**Tab** in the "Find Nodes" dialog box to change focus and navigate to the sections, buttons, and input fields. Each field or button within a section/group (Find Nodes By/In) is reached using the *Up/Down* keys.

- Use **Tab/Shift**+**Tab** in the "Nodes Found" dialog box to change focus. Each node is reached using the **Up/Down** keys.

Common Find Nodes Dialog Box Properties

Name:	Description:	Default Value:
Find	Click to find nodes based on the search criteria set in the "Find Nodes" dialog box. The "Nodes Found" dialog appears displaying a list of the nodes found that matched the search criteria selected.	N/A
Close (Alt+F4)	Click to close the "Find Nodes" dialog box.	N/A
Help	Click to access the "Find Nodes" online help.	N/A

Find Nodes By section

Name:	Description:	Default Value:
Node ID (Alt+N)	The default selection. Selects a node based on its node ID. Disabled if no script is open or the Current script option is not selected. When selected a "Please enter node ID:" field appears. Displays the node in the Current script based on the node ID.	N/A
Object (Alt+O)	Selects node(s) based on references to an object. When selected, the "Please select object type:" and "Please select object:" drop-down lists appear. Select the object type first, then the object. Displays a list of nodes in the Current script, All active script versions, or All opened scripts having a reference to the selected object.	N/A
Node Type (Alt+T)	Selects node(s) based on the node type. When selected, the "Please select node type" drop-down list appears. Displays a list of nodes of the selected type in the Current script, All active script versions, or All opened scripts.	N/A

Name:	Description:	Default Value:
String (Alt+S)	Selects node(s) based on a string that is entered. This string is case insensitive. When selected, the "Please enter string" field appears. Displays a list of nodes containing the substring of the input string in the Current script, All active script versions, or All opened scripts. Note: By default, this search criteria does not search the "Comment" field in each node. Check the "Find Nodes" Search comment field option (Alt+F) to search on the "Comment" field as well.	N/A

Find Nodes In section

Name:	Description:	Default Value:
Current script (Alt+C)	Select to find nodes in the current script based on any of the "Find Nodes By" properties. Disabled if no script is open.	N/A
All active script versions (Alt+A)	Select to find nodes in all active script versions based on the Object, Node Type, or String "Find Nodes By" properties. The only "Find Nodes In" option enabled if no script is open.	N/A
All opened scripts (Alt+P)	Select to find nodes in all active or open scripts based on the Object, Node Type, or String "Find Nodes By" properties. Disabled if no script is open.	N/A

Common Nodes Found Dialog Box Properties

Name:	Description:	Default Value:
List field	A listing of the node(s) found as a result of the Find.	N/A
Close (Alt+F4)	Click to close the "Nodes Found" dialog box.	N/A
Help	Click to access the "Nodes Found" online help.	N/A
Next Node (Alt+N)	Click to select the next node in the list.	N/A
Status bar	Displays "Finding nodes in script." as the search proceeds, then displays the number of nodes found matching the search criteria.	N/A

How to find nodes by Node ID

Step 1   Open the Script Editor tool.

Step 2   Select **Edit > Find Node (Ctrl+F)**. The Find Nodes dialog box appears.

Step 3   In the "Find Nodes By" section, select **Node ID**. The "Please enter node ID:" field is displayed.

Step 4   Enter the ID of the node you wish to view into this field.

Step 5	Click **Find**. The node with the ID matching your input is highlighted.
	Note: If there is no node with a matching ID, a message appears stating: "Cannot find node with ID <xxx>." (where <xxx> is the node ID).
Step 6	Select a node in the list and that node is highlighted in the script.

How to find nodes by Object

Step 1	Open the Script Editor tool.
Step 2	Select **Edit > Find Node (Ctrl+F)**. The "Find Nodes" dialog box appears.
Step 3	In the "Find Nodes By" section, select **Object**. Two drop-down list boxes appear, one providing a list of object types ("Please select object type:"), the other providing a list of objects ("Please select object:").
Step 4	Select the desired object type.
Step 5	Select the desired object.
Step 6	Click **Find**. A "Nodes Found" dialog box appears displaying a list of nodes referencing the selected object.
	Note: If no node has a reference to the selected object, a message appears stating: "Cannot find any node with reference to <xxx> object: <yyy>." (where <xxx> is the object type selected and <yyy> is the object selected).
Step 7	Select a node in the list and that node is highlighted in the script.

How to find nodes by Node Type

Step 1	Open the Script Editor tool.
Step 2	Select **Edit > Find Node (Ctrl+F)**. The "Find Nodes" dialog box appears.
Step 3	In the "Find Nodes By" section, select Node Type. A drop-down list appears providing a list of node types ("Please select node type:").
Step 4	Select the desired node type.
Step 5	Click **Find**. A "Nodes Found" dialog box appears displaying a list of nodes of the selected type.
	Note: If no node of the selected type is found, a message appears stating: "Cannot find any node of <xxx> type." (where <xxx> is the node type).
Step 6	Select a node in the list and that node is highlighted in the script.

How to find nodes by String

Step 1   Open the Script Editor tool.

Step 2   Select **Edit > Find Node (Ctrl+F)**. The "Find Nodes" dialog box appears.

Step 3   In the "Find Nodes By" section, select **String**. The "Please enter string:" field is displayed.

Note: The Find By String "Please enter string:" field entry is case insensitive.

Step 4   Enter the appropriate string into this field.

Step 5   Click **Find**. A "Nodes Found" dialog box appears displaying a list of nodes containing a substring of the input string.

Note:

• By default, this **Find** does not search the "Comment" field in each node. Check the "Find Nodes" **Search comment field** option **(Alt+F)** to search on the "Comment" field as well.

• If no node has reference to the input string, a message appears stating: "Cannot find any node with reference to string <xxx>." (where <xxx> is the string selected).

Step 6   Select a node in the list and that node is highlighted in the script.

Using the Find Nodes Option

Chapter 2

Call Types, Contact Data, and Scripting

As one writing scripts to route contacts, you must understand Call Types and contact data. Call Types are typically created during ICM software configuration, through the ICM Configuration Manager, and are therefore discussed in greater detail in the Cisco ICM Software Configuration Guide and the Cisco ICM Software IP Contact Center Installation and Configuration Guide.

This chapter contains the following topics:

What is a Call Type?

A Call Type is the first-level category of a contact and is determined by data associated with the contact. You associate a script with a Call Type. When a contact of a certain Call Type is received, the associated script runs on that contact.

Note: You create Call Types through Configuration Manager before writing routing scripts. For more information, see the Cisco ICM Software Configuration Guide.

What is a Default Call Type?

A default Call Type is the Call Type used when a contact does not map to a defined Call Type.

You define a default Call Type for each routing client through ICM software's Configuration Manager. You also define a general default Call Type that is not specific to a routing client. For more information, see the *Cisco ICM Software Configuration Guide*.

How Call Types and Scripts are Related

Scripts are scheduled by **Call Type**. In other words, when ICM software receives a request to route a contact, it determines the Call Type of that contact, then runs the associated script.

Call Types provide the first level of **categorization of contacts**, enabling you to write scripts to route contacts differently depending on their call type. While other types of categorization take place within a script, Call Types enable you to provide contacts with different treatment by running different scripts to begin with. Call Types enable categorization before a script begins to execute.

Call Type Qualifiers

The Call Type is determined by the following data, which are referred to as Call Type qualifiers:

- Dialed Number

- Calling Line ID (CLID)

- Caller-Entered Digits

The Call Type qualifiers described in this section apply to contacts from all media. The terminology used is applicable to voice contacts; where the terminology differs for other media, the differences are explained in this section, as well as the following sections in this topic:

- Contact Data for Single Session Chat, Multi-Session Chat, and Blended Collaboration

- Contact Data for E-Mail

Note: You can also use the call type qualifiers for categorization within a script.

Dialed Number (DN)

A Dialed Number (DN) is a string that represents the telephone number dialed by the caller, preceded by the name of the routing client and a period. For example, "NICClient.18005551212" might be a Dialed Number.

Typically, a Dialed Number is associated with one or more Call Types. The **Calling Line ID** and **Caller Entered Digits** are used to further categorize the call and determine the Call Type.

Note: The Dialed Number is referred to as the Script Selector for media other than voice.

Calling Line ID (CLID)

The Calling Line ID (CLID) is a string that represents the telephone number from where the call originated. The CLID is sometimes referred to as the ANI (Automatic Number Identification).

Typically, you would not use a CLID to define a Call Type. Rather, you would use a CLID prefix or CLID region, as described below.

For Web requests, the CLID corresponds to the ApplicationString1 parameter set in the in.map.properties file on the Collaboration Server.

The CLID is not used by e-mail requests.

Using a CLID Prefix

You may want to define a Call Type based on the area code from where the call originated. To do so, you can use a CLID prefix.

For example, if you want to define a Call Type for all calls from the 508 area code, you specify 508 as the CLID prefix. You can further refine the Call Type to calls from a certain exchange within the area code. For example, you can specify 508486 as the CLID prefix.

A routing script can then process the call based on the area code or local exchange of the caller.

Using a CLID Region

You may want to define a Call Type that encompasses multiple CLID prefixes. For example, a useful Call Type could be defined as all calls from New York, which includes several area codes. To accomplish this, you must first define geographical regions through ICM software's Configuration Manager. For more information, see the Cisco ICM Software Configuration Guide.

A routing script can then process the call based on the region of the caller.

Caller-entered Digits (CED)

Caller-entered Digits (CED) are numbers entered by the caller in response to prompts. For example, a caller may enter a number to indicate the type of service needed.

The caller may enter digits through the carrier network or the call center system. Regardless, the Caller-entered Digits can be used in defining the call's Call Type.

A routing script can then process the call based on data entered by the caller.

Note: You can differentiate between the case where the caller is not prompted for digits ("None Required") and the case where the caller is prompted but does not respond ("None Entered"). You can also choose "None" as the caller-entered digits.

For **Web requests**, the CED corresponds to the ApplicationString2 parameter set in the in.map.properties file on Collaboration Server.

The CED is not used by **e-mail requests**.

Data for Web Requests

Cisco Collaboration Server sends the following data to ICM software when requesting that a single session chat, multi-session chat, or blended collaboration contact be routed. These variables are mapped in the in.map.properties file to variables in the call form. For information on setting up Cisco Collaboration Server to send route requests to ICM software, see the Cisco Collaboration Server Administration Guide.

- dialednumber - A string that determines which script to run on ICM software. The value of the scriptselector variable maps to the value of the Script Selector (which maps to the Dialed Number for voice contacts) in the Call Type created through ICM Configuration Manager. Therefore, you must ensure that a Script Selector with the value used by the Collaboration Server is configured in ICM software.

- applicationstring1 - An optional string that can be used to select an ICM routing script. The value of the ApplicationString1 variable corresponds to the Calling Line ID created through ICM Configuration Manager.

- applicationstring2 - An optional string that can be used to select an ICM routing script. The value of the ApplicationString2 variable corresponds to the Caller-Entered Digits created through ICM Configuration Manager.

- callvar1-10 - ICM call variables, up to 10. These are optional fields that can be used to pass any application-specific information to ICM.

- eccvar1-2 - Expanded call context variables, used to pass additional information to the ICM. ECC variables must be configured on the ICM before they can be used by the Collaboration Server. For information on creating ECC variables, see the Cisco ICM Software Configuration Guide.

Data for E-Mail Requests

Cisco E-Mail Manager sends the following data to ICM software when requesting that an e-mail message be routed:

- Instance and skill group name - A string that determines which script to run on ICM software. The value is the name of the E-Mail Manager instance and the ICM Routing skill group the message was assigned to, separated by a period; for example, "SupportInstance.techSupport". The string maps to the Script Selector value. Therefore, you must ensure that Script Selectors with the value of the E-Mail Manager instance name and each ICM Routing skill group are configured in ICM software.

- cisco.cem.Priority - The priority of the message. The value of the Priority variable, which is "0" through "3"

  - "0" for Normal

    Note: The priority if set through the E-Mail Manager rules. The value of the variable can be used to categorize the contact in a script.

  - "1" for High

  - "2" for Very High

  - "3" for Urgent

- cisco.cem.Category - The categories of the message. The categories variable is an array containing up to 10 category values. Category values are configured by the E-Mail Manager administrator. The value of the variable can be used to categorize the contact in a script.

- cisco.cem.MessageKey - The unique identifier of the message that E-Mail Manager is requesting ICM software to route. While typically the Message Key would not be used to categorize a contact in a script, it may be useful when ICM software is integrated with a CRM application; the Message Key could be recorded in the CRM database for future reference to e-mail correspondence with a customer.

For information on Expanded Call Variables, see the Cisco ICM Software Configuration Guide.

For information on setting up Cisco E-Mail Manager to send route requests to ICM software, see the Cisco E-Mail Manager Administration Guide.

How ICM Software Associates Contacts with Call Types

Following is the general process of how ICM software attempts to associate a contact with a Call Type.

1. If the Dialed Number, Calling Line ID, and Caller-Entered Digits of the contact map to a defined Call Type, ICM software uses that Call Type.

2. If no Call Type matches the contact, ICM software uses the default Call Type for the routing client.

3. If no default Call Type is defined for the routing client, ICM software uses the general default Call Type.

4. If no general default Call Type is defined, ICM software uses the default label defined for the dialed number.

5. If no default label is defined for the dialed number, ICM software returns an error to the routing client.

Example - How ICM Software Determines the Call Type for a Voice Contact

The following basic example demonstrates how ICM Software Determines the Call Type for a voice contact and runs the appropriate script:

1. When configuring ICM software, you create a Call Type called "MASSACHUSETTS_SALES". This Call Type is defined as:

   – Having a Dialed Number of "NICClient1.8005551234".

   – Being from Massachusetts. This is determined by using a Calling Line ID Region which consists of CLID Prefixes for all area codes in Massachusetts: 617, 508, 978, and 413.

   – Being a Sales call. This is determined by a Caller Entered Digits value of "1", which is the number in the voice menu to indicate that the caller needs sales help.

2. You create a script called "MASSACHUSETTS_SALES_SCRIPT" which finds the longest available agent in the "NORTHEAST_SALES" skill group.

3. You schedule the script to run for the "MASSACHUSETSS_SALES" Call Type.

4. A caller dials 1-800-555-1234, from the phone number 508-663-4958.

5. When prompted by a menu, the caller enters 1 to request sales help.

6. A route request is sent to ICM software.

7. ICM software examines the dialed number, which equals "18005551234".

8. ICM software evaluates the CLID value and determines that the CLID prefix is "508", which is an area code in Massachusetts.

9. ICM software examines the CED value, which is "1", which indicates that it is a Sales call.

10. ICM software determines that the Call Type is "MASSACHUSETTS_SALES" and executes the "MASSACHUSETTS_SALES_SCRIPT" script.

11. ICM software assigns the task to a particular agent.

Example - How ICM Software Determines the Call Type for a Web Request

The following basic example demonstrates how ICM Software Determines the Call Type for a single session chat Web request:

1. When configuring ICM software, you create a Call Type called "SSC_CT". This Call Type is defined as having a Script Selector (Dialed Number) of "SSC_DN".

2. When configuring Collaboration Server, you set the value of the scriptselector variable in the call form to "SSC_DN".

3. When configuring Collaboration Server, you set the dialednumber variable in the input map to equal the scriptselector variable in the call form.

4. You create a script called "SSC_SCRIPT" which finds the longest available agent in the "COLLABORATION_SALES" skill group.

5. You schedule the script to run for the "SSC_CT" Call Type.

6. A Web user requests a chat session.

7. A route request is sent to ICM software.

8. ICM software determines that the Call Type is "SSC_CT" and executes the "SSC_SCRIPT" script.

9. ICM software instructs the Collaboration Server to assign the task to a particular agent.

For information on setting up Cisco Collaboration Server to send route requests to ICM software, see the Cisco Collaboration Server Administration Guide.

Example - How ICM Software Determines the Call Type for an E-Mail Contact

The following example demonstrates how ICM Software Determines the Call Type for an E-Mail contact:

1. When configuring ICM software, you create a Call Type called "EMAIL_CT". This Call Type is defined as having a Script Selector (Dialed Number) of "Instance1.SalesQueue".

2. When configuring E-Mail Manager, you set the system rules to assign some messages to the ICM Routing queue "SalesQueue", set their priority to "Urgent", and associated them with the category "Sales".

3. You create a script called "EMAIL_SCRIPT". For messages with a priority of "Urgent" (with the cisco.cem.Priority Expanded Call Variable value of "3") and a category of "Sales" (with the cisco.cem.Category Expanded Call Variable value of "Sales"), the script assigns the message to the E-Mail Manager local skill group "UrgentSales".

4. You schedule the script to run for the "EMAIL_CT" Call Type.

5. In E-Mail Manager, an urgent "Sales" message is assigned to the "SalesQueue" queue.

6. E-Mail Manager sends a route request to ICM software.

7. ICM software determines that the Call Type is "EMAIL_CT" and executes the ""EMAIL_SCRIPT" script.

8. ICM software instructs E-Mail Manager to assign the task to a particular agent.

Chapter 3

Categorizing Contacts

When you create a routing script, you typically use the nodes available in Script Editor to define how the script is to categorize contacts. By categorizing contacts, a script can provide unique treatments for different customer needs.

This chapter contains the following topics:

Categorization and Call Types

Categorization is the process of classifying a contact based on certain data associated with the contact. Through categorization, a script can determine how to best process a contact.

Categorization through Scheduling Scripts by Call Type

Call Types provide the first level of categorization for routing scripts. You schedule scripts by call type; therefore the Call Type of a contact determines which script is executed, enabling you to create different scripts for different types of contacts.

How to Change the Call Type and Continue Script Processing

You can change the Call Type of a contact from within a script by using the Call Type node (in the Routing tab of the Palette).

Figure 13: The Call Type Icon
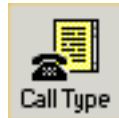
Following is the Properties dialog box of the Call Type node:

Figure 14: Call Type Properties Dialog Box - Call Type Node
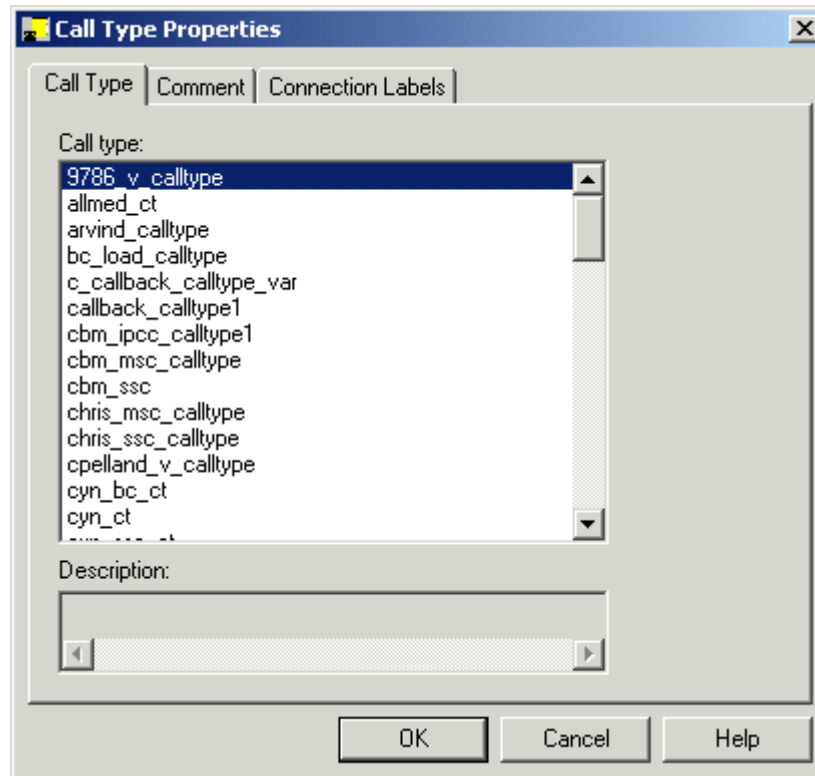
Define Call Type node properties as follows:

1. In the Call Type tab, select the **Call type** to assign to the contact.

2. Optionally, add comments and connection labels.

Warning: The Call Type node changes the call type and continues the current script execution. The Requalify Call node terminates the current script execution and executes a new script associated with that call type.

How to Change the Call Type and Execute a New Script

You can change the Call Type of a contact from within a script and execute a new script associated with the call type by using the Requalify Call node (in the Routing tab of the Palette).

Figure 15: The Requalify Call Icon

Following is the Properties dialog box of the Requalify Call node:

Figure 16: The Requalify Call Properties - Requalify Call Tab
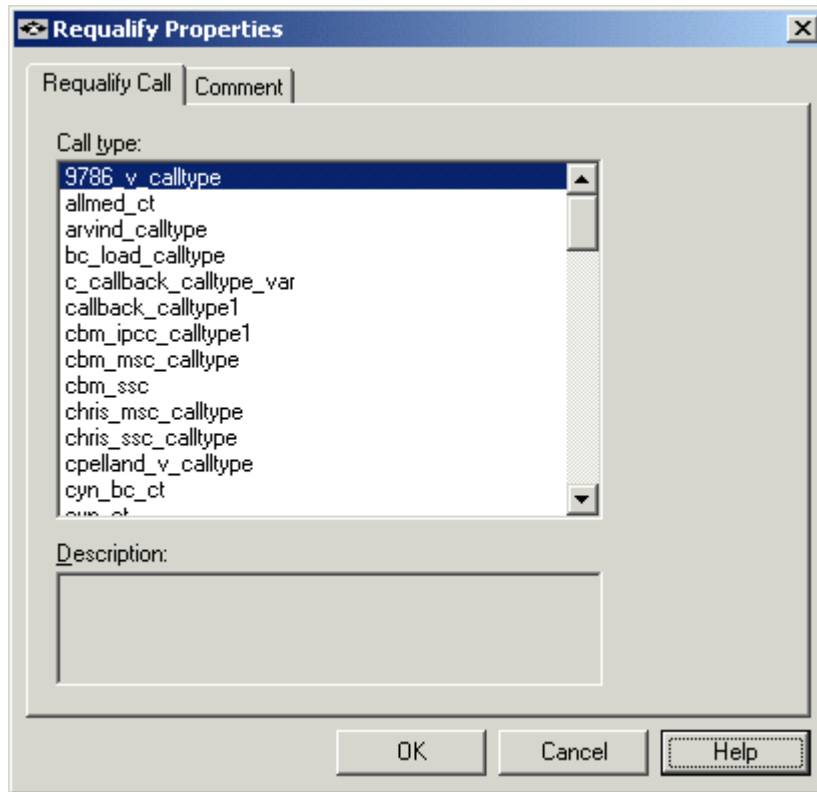
Define Requalify node properties as follows:

1. In the Requalify Call tab, select the Call type to assign to the contact.

2. Optionally, add comments.

Warning: The Call Type node changes the call type and continues the current script execution. The Requalify Call node terminates the current script execution and executes a new script associated with that call type.

Categorizing by Call Type Qualifiers

As described in the topic Call Types, Contact Data, and Scripting, a contact's Call Type is determined by three Call Type Qualifiers:

- Dialed Number

- Calling Line ID (CLID)

- Caller-Entered Digits

When ICM software determines a contact's Call Type based on these qualifiers, it executes the associated script.

However, after the script executes, you can further categorize the contact based on the values of the Call Type qualifiers.

For example, a Call Type may be defined as having a Calling Line ID that includes calls from all area codes in New England states. When the script associated with that Call Type executes, it can examine the specific area code in the Calling Line ID and branch differently depending on the value, thus treating calls from different area codes in New England differently.

You can categorize a contact based on its Call Type Qualifiers by using one or more of the following nodes:

- Dialed Number (DN) Node

- Calling Line ID (CLID) Node

- Caller Entered Digits (CED) Node

How to Categorize a Contact by the Dialed Number

You can categorize a contact based on its Dialed Number by using the Dialed Number (DN) node (in the Routing tab of the Palette).

Figure 17: The Dialed Number Icon

Following is the Properties dialog box of the Dialed Number node:

Figure 18: Dialed Number Properties - Dialed Number Tab
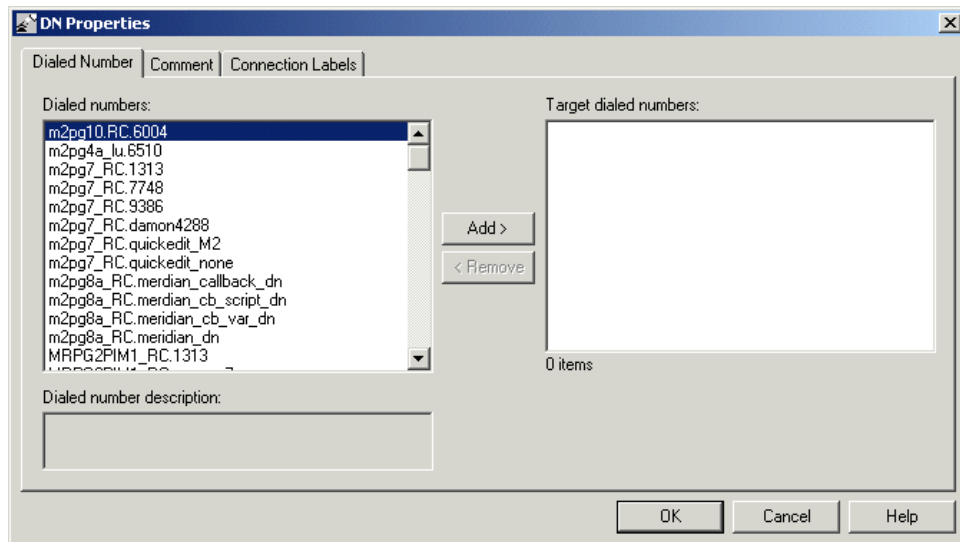
Define Dialed Number node properties as follows:

1. Select one or more dialed numbers or Script Selectors from the Dialed numbers list and click **Add>** to move them to the Target dialed numbers list. If the current contact matches one of the selections in the Target dialed numbers list, processing continues on the node's success branch; otherwise, processing continues on the failure branch.

2. Optionally, add comments and connection labels.

How to Categorize a Contact by the Calling Line ID

You can categorize a contact based on its Calling Line ID by using the CLID node (in the Routing tab of the Palette).

Figure 19: The CLID Icon

The CLID node tests the billing telephone number of the contact to see if it matches:

- A specific (CLID) that you provide (for example, a specific region, area code, or an area code plus a local exchange).

- A variable expression.

By default, ICM software compares the list of values against the calling line ID of the contact. However, you can specify another value or expression to be used instead of the CLID.

Note: For Web Collaboration requests, the CLID maps to the applicationstring1 variable.

Before defining CLID node properties, you must insert one or more targets and connections from the CLID node.

Then define CLID node properties as follows:

1. In the Calling Line ID tab:

   a. *Figure 20: CLID Properties - Calling Line ID*
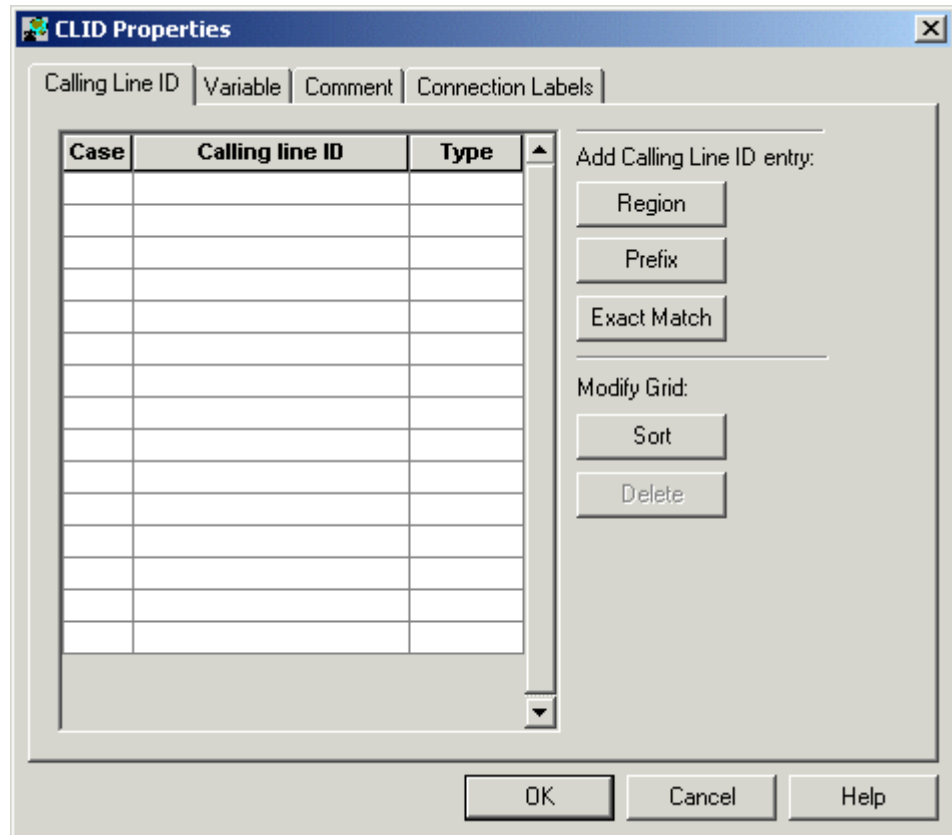
   For each branch to a different target, select the Case (the number displayed on each success connection branch); for example, 1, 2, and 3.

   b. For each Case, select the Type of match. Click Region, Prefix, or Exact Match.

   c. For each Case, enter the Calling line ID to match. If you selected Region, select a region from the drop-down list; for example, California. If you selected Prefix, enter the Prefix number. If you selected Exact Match, enter the Calling line ID to match.

   d. To sort the list by case, click Sort.

   e. To delete a row, select the row and click Delete.

2. In the Variable tab:

a. *Figure 21: CLID Properties - Variable*
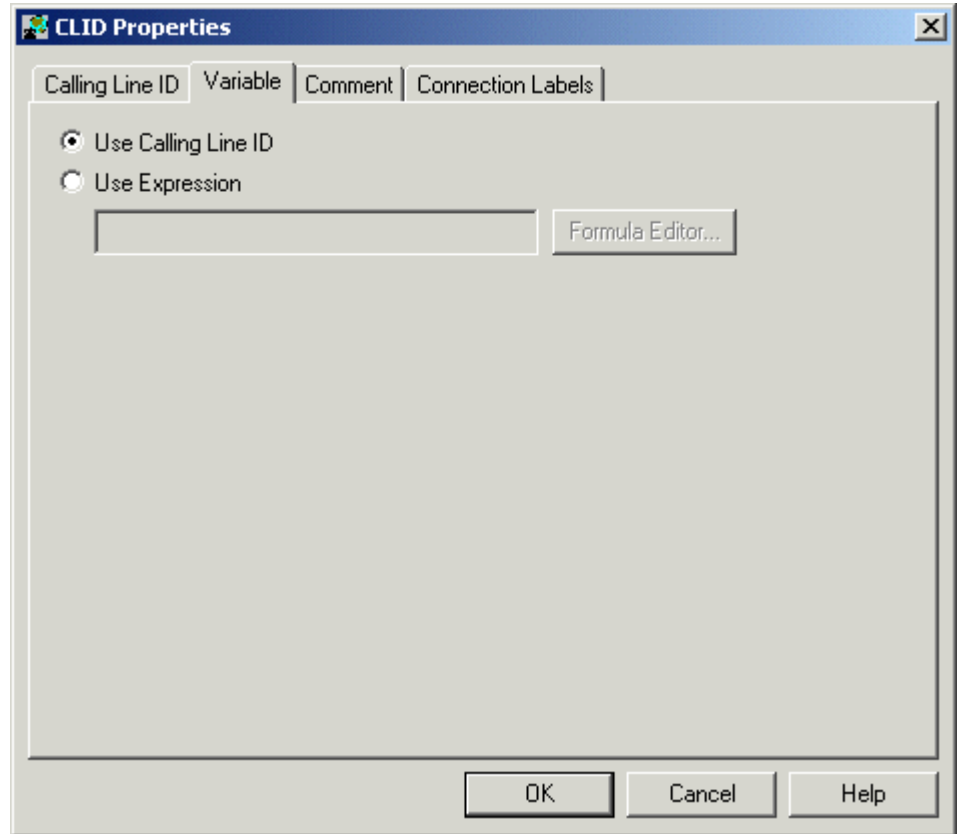
By default, Use Calling Line ID is selected, to have ICM software compare the list of values you define in the Calling Line ID tab against the calling line ID of the contact.

b. To have ICM software use the value of an expression instead of the Calling Line ID, select Use Expression and enter the expression directly or click Formula Editor to use a formula to define the expression.

3. Optionally, add comments and connection labels.

Note: If you delete a connection associated with a Case, the Case information you specified in the Properties dialog box is also deleted.

How to Categorize the Contact by the Caller Entered Digits (CED)

You can categorize a contact based on the caller-entered digits by using the CED Node (in the Routing tab of the Palette).

Figure 22: CED Properties Icon

For example, you can define the CED node to send contacts to:

- SkillGroupA if the caller-entered digits match a specific string

- SkillGroupB if there are no caller-entered digits

Note: For Web Collaboration requests, the CED maps to the applicationstring2 variable.

Following is the Properties dialog box of the CED node:

Figure 23: CED Properties
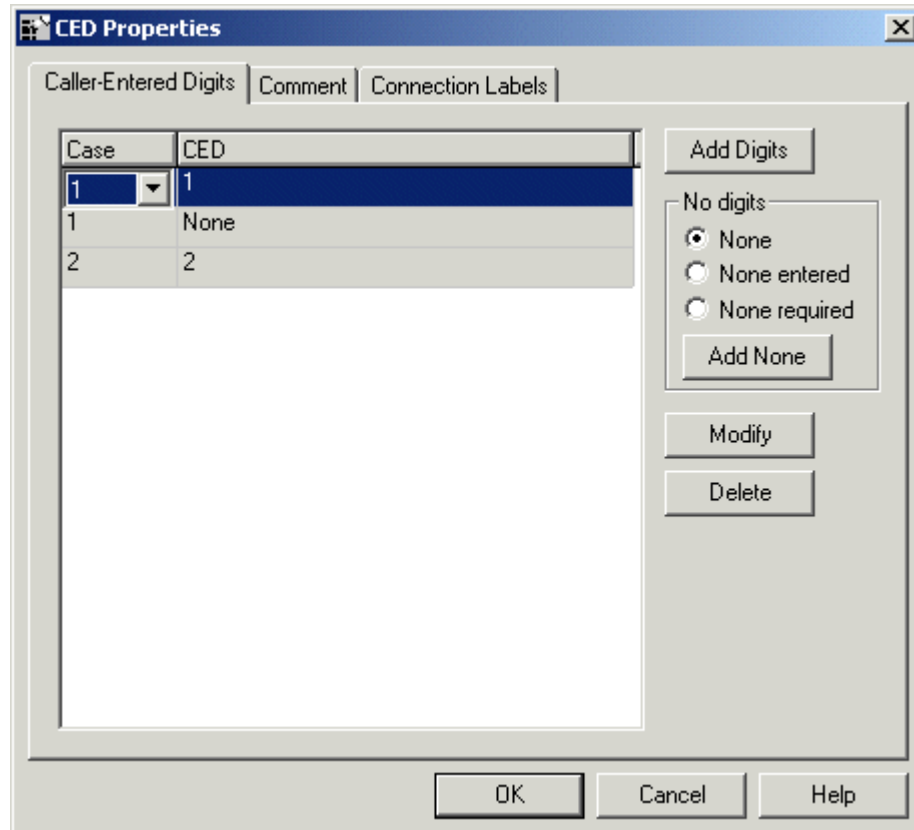
You must insert targets and connections from the Caller Entered Digits Node before you can define the node's properties.

Then define CLID node properties as follows:

1. Click Add Digits to add a new CED value for a branch. Then in the new row add a CED value and select the branch number. You can associate one or more CED values with each connection. Valid characters are the digits 0 through 9, the asterisk (*), and the number sign (#).

2. Click Add None to add the value of None to a connection, specifying that it matches only those cases where no digits are entered or where no digits are required. When specifying None, you can select:

   – None, to include both None Entered and None Required situations.

- None Entered, to apply when the caller was prompted for digits but did not enter any.

- None Required, to apply when the caller was not prompted for digits.

Note: If you delete a connection associated with a Case, the Case information you specified in the Properties dialog box is also deleted.

Categorizing by Time and Date

You schedule a script by associating it with a Call Type. When a contact of a certain Call Type is received, the associated script runs for that contact.

However, after the script executes, you can further categorize the contact based on the time and day of week; in effect, this refines the schedule.

Note: The time and day of week are determined by the settings on the computer running the ICM Central Controller.

For example, a Call Type named "SSC_CT" may be defined to include all Single Session Chat Web requests. A script named "SSC_SCRIPT" executes every time a contact with the Call Type "SSC_CT" is received. Typically, this script instructs Collaboration Server to assign the request to the longest available agent in the "SSC" skill group. However, the contact center is staffed differently over the weekend and the supervisor wants reports to better reflect weekend activity. Therefore, for Single Session Chat Web requests received on Saturday or Sunday, the script branches differently and instructs Collaboration Server to assign the request to the longest available agent in the "WKEND_SUPPORT" skill group.

As another example, for a contact center where no phone support is available during night hours or weekends, you may choose to design a script that during those off hours routes a phone call to an announcement instead of an agent.

How to Categorize a Contact by the Time

You use the Time node (in the General tab of the Palette) to choose from among several paths within the script based on the current time at the ICM Central Controller. Following is the Properties dialog box of the Time node

Figure 24: Time Icon

Following is the Properties dialog box of the Time node:

Figure 25: Time Properties
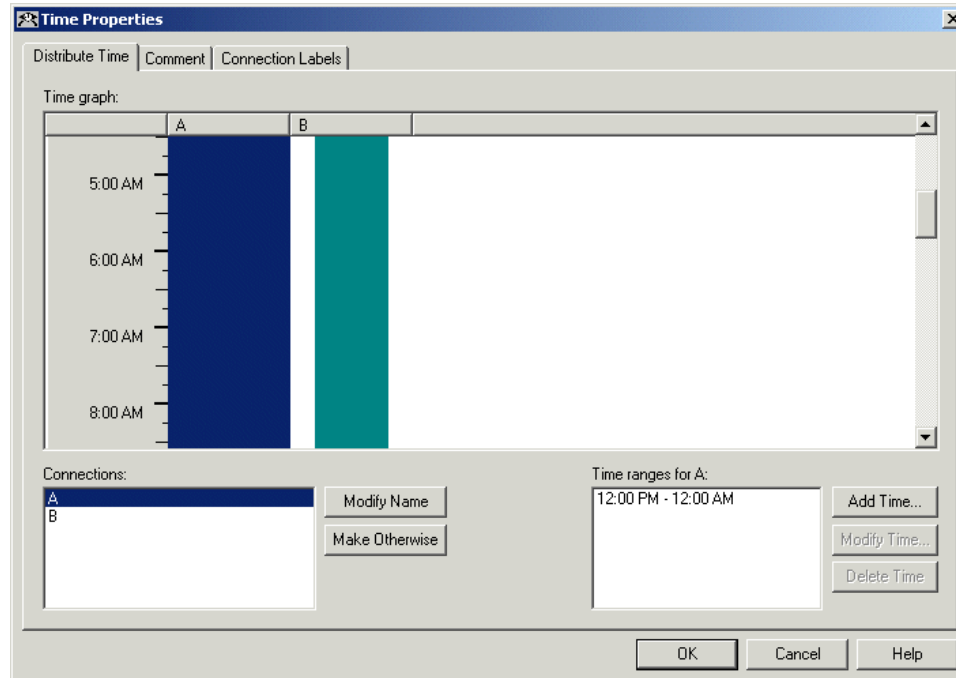
You must insert targets and connections from the Time node before you can define the node's properties.

Then define Time node properties as follows:

1. For each branch listed in the Connections list, define a Time Range. You can define multiple time ranges for a single branch. Click **Add Time** to add a new time range to the branch, or select a time range listed and click **Modify Time** to modify it. A dialog box opens in which you can define the time range (the Add Time dialog box is shown below; the Modify Time dialog box looks and functions similarly):

Figure 26: Add Time Dialog
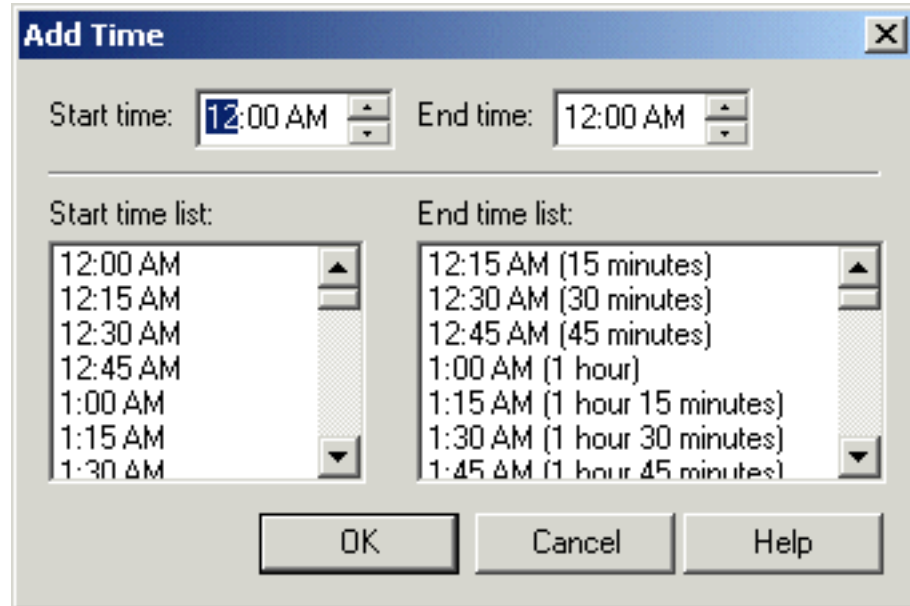
2. To delete a time associated with the branch, select the time and click **Delete Time**.

3. You can define a branch as Otherwise by selecting the branch and clicking **Make Otherwise**. Execution follows this branch if none of the specified time ranges apply. You can specify only one Otherwise branch for the node. To no longer define the branch as otherwise, select the branch and click **Delete Otherwise**.

4. Optionally, add comments and connection labels.

Note: If you delete a connection, the time range information you specified in the Properties dialog box is also deleted.

How to Categorize a Contact by the Day of Week

You use the Day of Week node (in the General tab of the Palette) to transfer control to one of several branches depending on the current day of week (Sunday, Monday, etc.).

Figure 27: Day of Week Icon

Following is the Properties dialog box of the Day of Week node:

Figure 28: Day of Week Properties
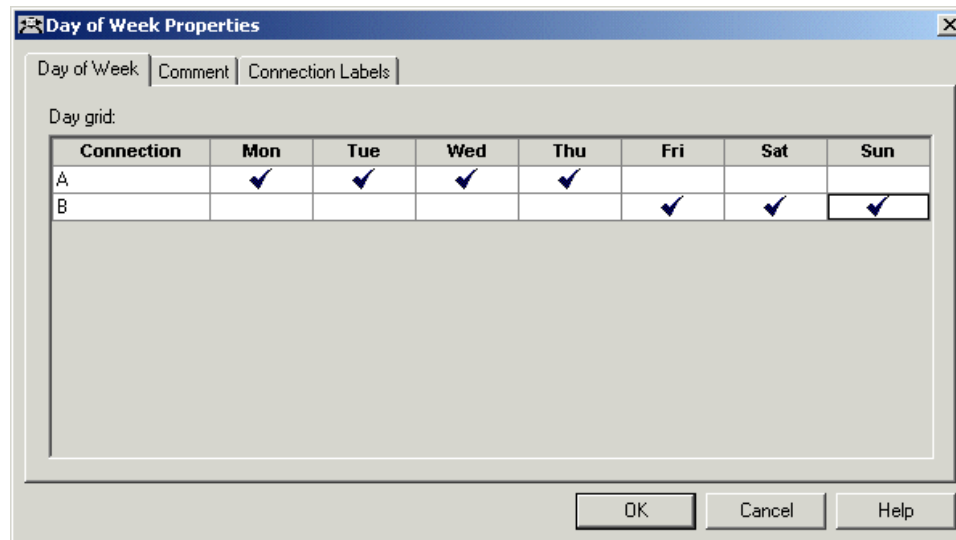
You can define multiple output connections from the Day of Week node and associate each with one or more days of the week.

You must insert targets and connections from the Day of Week node before you can define the node's properties.

Define Day of Week node properties as follows:

1. For each branch listed in the Connection list, check the days of the week in which processing should continue on that branch. To check the day for that connection, left-click in a spot in the grid corresponding to that connection and day. A check mark appears in the grid.

   You can associate each day of week with exactly one connection. However, you can associate each connection with one or more days of the week.

2. Optionally, add comments and connection labels.

Note: If you delete a connection, the day of the week information you specified in the Properties dialog box is also deleted.

Categorizing by Branching

Within a script, you can create multiple branches to direct script processing based on certain conditions. Branching allows you to use a single script that will process contacts differently depending on data associated with the contact, or on conditions at the contact center.

For example, you can design a script named "EMAIL_SCRIPT" that processes all requests to route e-mail messages. Within that script, you can detect if the e-mail message was marked as "Urgent" by Cisco E-Mail Manager. If the message was marked as "Urgent", the script can pass

the contact to a different script called "URGENT_EMAIL_SCRIPT that is designed and maintained specifically to process the most important e-mail messages.

How to Execute a Different Script

You use the Go To Script node (in the General tab of the Palette) to direct contact processing to another script without changing the call type. When ICM software encounters a Go To Script node, it stops executing the current script and starts the script indicated in the node.

Figure 29: Go To Script Icon

For example, you might have several scripts that check for exception conditions and, if none are found, execute a standard subroutine. Instead of including that subroutine as a branch from the failure output terminal of each of the exception conditions, you could use a Go To Script node pointing to a separate script containing the subroutine.

Following is the Properties dialog box of the Go To Script node:

Figure 30: Go To Script Properties
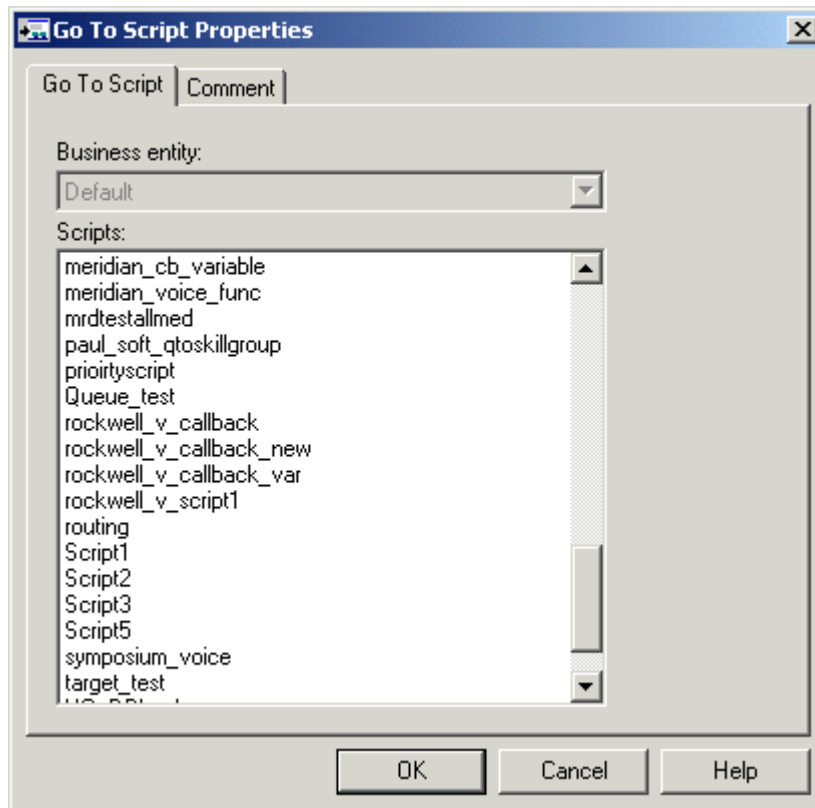
Define Go to Script node properties as follows:

1. Select the Business entity that owns the script that the node should execute.

By default, ICM software consists of one business entity. Multiple business entities are allowed only if you enable partitioning. For more information on partitioning, see the Cisco ICM Software Security Guide.

2. Select a script from the Scripts list.

   From within an administrative script, you can go to only another administrative script. Within a routing script, you can go to only another routing script.

3. Optionally, add comments and connection labels.

How to Direct Script Execution to a Specific Branch

You use the Switch node (in the General tab of the Palette) to direct script execution to its active output connection. You can define multiple output connections from the Switch node; one of them must be specified as the active connection.

Figure 31: Switch Icon

Following is the Properties dialog box of the Switch node:

Figure 32: Switch Properties
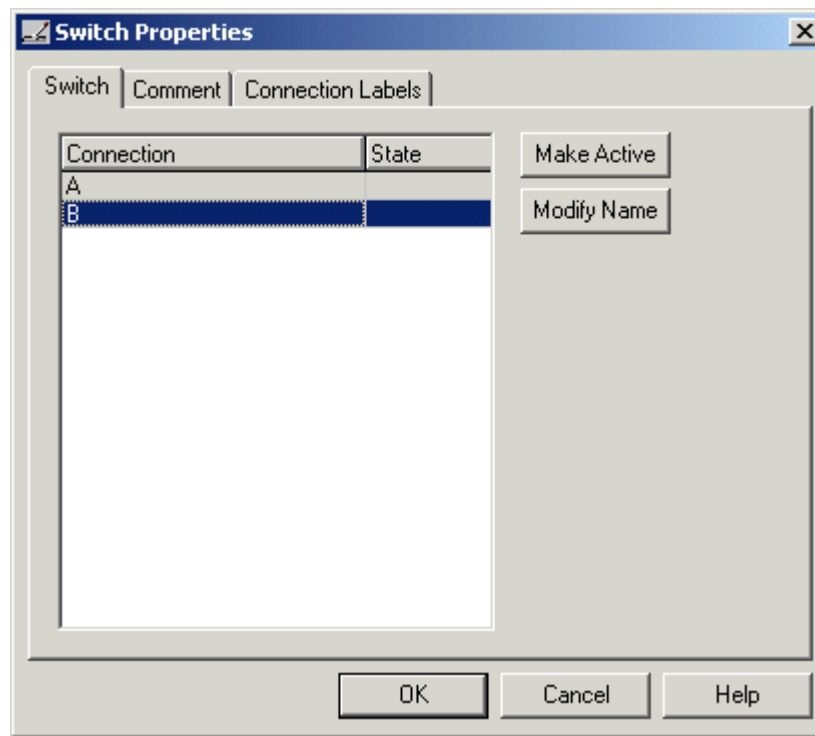
You must insert targets and connections from the Switch node before you can define the node's properties.

Then define Switch node properties as follows:

1. By default, Connections are labeled "A", "B", etc. To re-label a Connection, click **Modify Name** and make changes to the name.

2. To make a connection active, select a Connection and click **Make Active**.

   Only one connection can be active at any time. To change the active branch, you must re-edit the script and create a new script version.

3. Optionally, add comments and connection labels.

How to Direct Script Execution to Different Branches by Percentage

You can direct specific percentages of contacts to different branches in a script by using the Percent Allocation node (in the General tab of the Palette).

Figure 33: Percent Allocation Icon
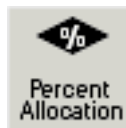
Each branch may lead directly to a target, or may include additional processing. Because contacts are distributed by percentage and without tests of the targets' data, distributing by percentage never fails.

For example, in a geographically diverse environment, you can create a script that sends 10% of contacts to Boston, 5% to Chicago, and distributes the remaining 85% to another set of targets.

Warning: Unlike selecting targets by rules or distributing contacts to targets, distributing contacts does not consider real-time contact center conditions and therefore may lead to load imbalances.

Following is the Properties dialog box of the Percent Allocation node:

Figure 34: Percent Allocation Properties
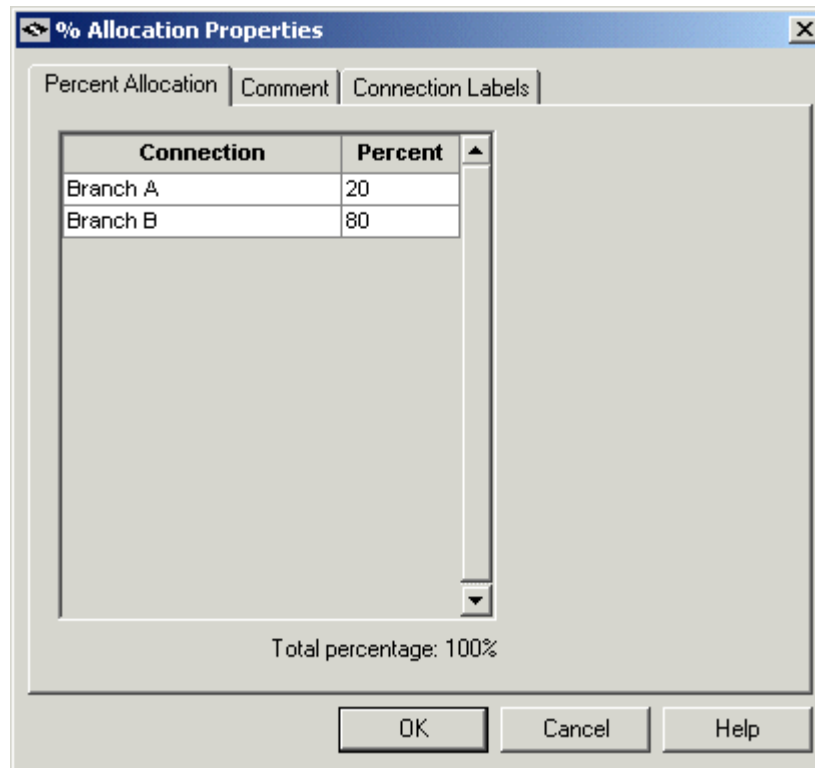
Define Percent Allocation node properties as follows:

1. In the Percent column for each connection, enter a percent number for the percentage of contacts to process on that branch. Note: The percent total for all rows must equal 100.

2. Optionally, modify the Connection name. Changes appear in the connector labels when you save the properties and view the script.

3. Optionally, add comments and connection labels.

How to Categorize a Contact based on a Condition

You use the If node (in the General tab in the Palette) to direct script execution to one of two branches based on the result of an evaluation. You can use formulas to define the If node.

Figure 35: If Icon

When ICM software executes an If node, it first evaluates the condition specified in the node Properties dialog box Define condition field. If ICM software determines that the condition is true, control flows through the success output terminal; if it determines the condition is false, control flows through the failure output terminal.

Following is the Properties dialog box of the If node:

Figure 36: If Properties
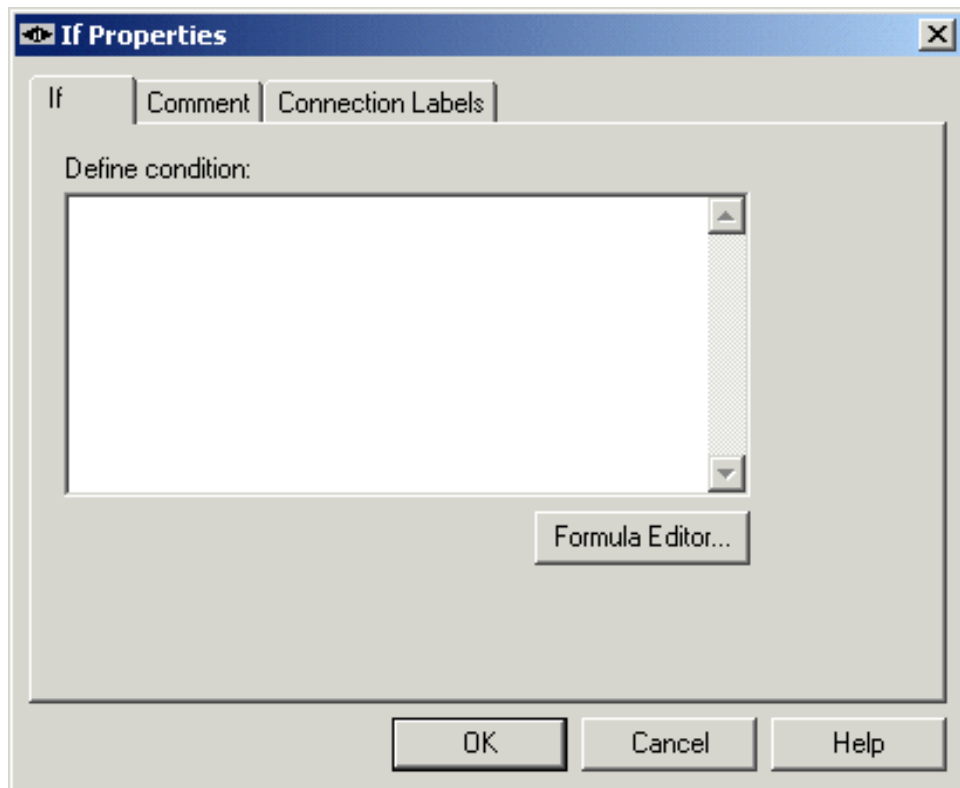
Define If node properties as follows:

1. In the Define condition field, enter a condition or use the Formula Editor to create a formula.

2. Optionally, add comments and connection labels.

How to Categorize a Contact Based on its Media Routing Domain

You use the Media Routing Domain node to categorize contacts based on their media routing domains. This node is described in the Universal Queue section of this document.

Categorizing By External Data

Scripts can categorize a contact based on data stored in a database that is not part of ICM software, such as a Customer Relationship Management (CRM) system.

For example, a script that processes incoming phone calls can query a CRM using the CLID to determine if the customer should receive Premium or Standard phone support. The records for the customer with the phone number matching the CLID are retrieved, and the value of the Support column is returned to the script. If the value indicates that the customer has paid for Premium support, script processing continues down one branch that assigns the phone call to a

skill group dedicated to responding to Premium customers; otherwise, script processing continues down another branch that assigns the phone call to a more general skill group, where the wait time is expected to be longer.

Note: You must use the ICM Configuration Manager to define the external database table and the columns that are to be referenced in scripts. For more information, see the Cisco ICM Software Configuration Guide.

You can categorize a contact by retrieving external data with the DB Lookup Node, then referencing the retrieved external data on the DB Lookup node success branch.

How to Modify the CallRouter Registry to Provide for DB Lookup Authentication on the Remote Database Target

By default, when attempting to access a remote database, the ICM CallRouter authenticates itself as:

- Username = SA

- Password = blank

If different account information is required to access the database, you must specify this by editing the Windows registry on the CallRouter to include a SQLLogin Registry Key at the following location:`HKEY_LOCAL_MACHINE\SOFTWARE\Cisco.System,INC.\ICM\<instancename>\RouterA\Router\CurrentVersion\Configuration\Database`.

Create the SQLLogin key as follows: `SQLLogin=\\<DBMachine>\<DBName>=(<DB_username>,<DB_password>)`. Use a comma as the delimiter to separate data for separate databases.

The following example shows login credentials for two external databases: `SQLLogin=\\MachineA\DatabaseA=(username,password),\\MachineB\DatabaseB=(username,password)`.

How to Categorize a Contact by External Data

You use the DB Lookup node (in the General tab of the Palette) to query a specific row of data from an external database. You can then reference columns from that row.

Figure 37: DB Lookup Icon

Following is the Properties dialog box of the DB Lookup node:

Figure 38: DB Lookup Properties
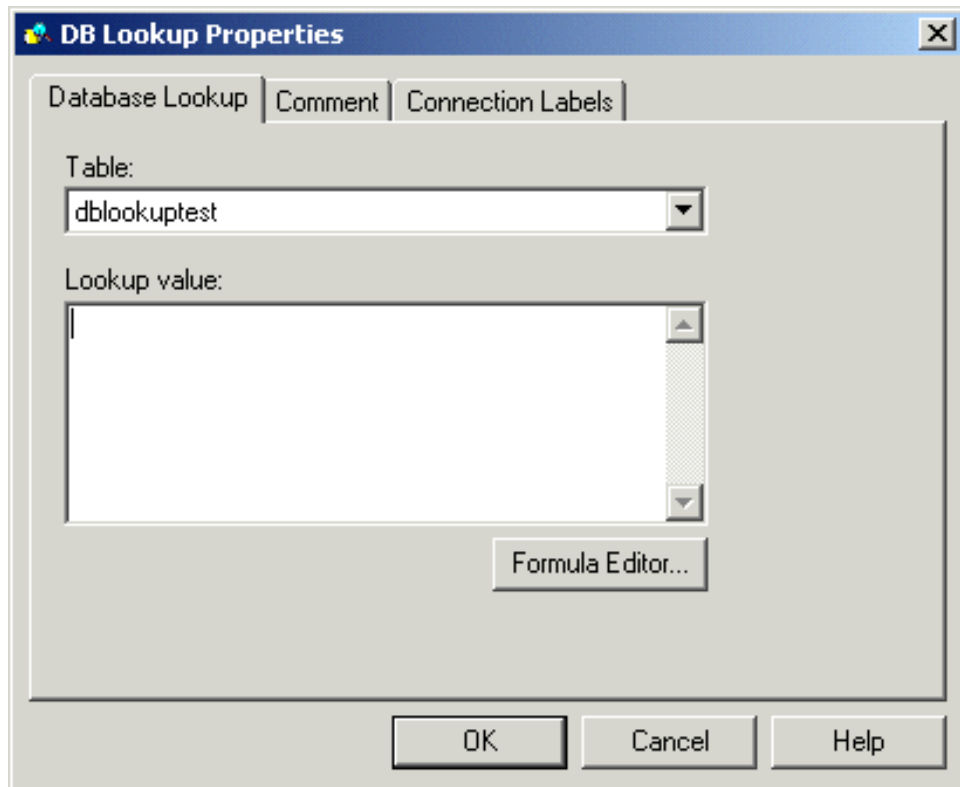
Define the DB Lookup node properties as follows:

1. Select the database Table you want to query. (The drop-down list contains the enterprise names of all lookup tables defined in ICM software.)

   Note: Define all integer fields in tables accessed by a DBLookup node as NOT NULL. Only the following data types are supported for SQL databases: SQLINT1 (tinyint), SQLINT2 (smallint), SQLINT4 (int), SQLCHAR (char), SQLVARCHAR (varchar), SQLFLT4DBFLT4 (real), SQLFLT8DBFLT8 (float), and SQLDATETIME (datetime). All fields except SQLDATETIME, SQLVARCHAR, and SQLCHAR must be defined as NOT NULL fields. These three fields can be defined as NULL.

2. In the Lookup value field, define constant or expression to match the key value in the row you want to retrieve. The value must be of the appropriate data type to match the key field in the table. You can use formulas to define the expression. For example, if phone_number is a key field in the database table, you might use Call.CallingLineID as the Lookup Value.

3. Optionally, add comments and connection labels.

How to Reference Retrieved External Data

To access retrieved data, you use the following syntax in nodes that reference a column:

```
Database.table-name.column-name
```

Where:

- **`table-name`** is the enterprise name of the table as defined through the Configuration Manager.

- **`column-name`** is the name of the column from the table, which is also defined through the Configuration Manager.

For example, if the table Customers contains a column named Priority, you would reference that column in an If expression as follows:

`Database.Customers.Priority = 1`

For more information, see .

Categorizing by External Applications

You can categorize a contact based on data returned from an application external to ICM software by using the Application Gateway node (in the General tab of the Palette).

Figure 39: Gateway Icon

For example, a script that processes incoming phone calls can send the caller's account number to an external application, which returns to the script the caller's account balance. The script can then branch on the value of the account balance, providing premium service to callers with higher account balances.

Note: You must use the ICM Configuration Manager to define the external application. For more information, see the Cisco ICM Software Configuration Guide.

Define the Application Gateway properties as follows:

1. In the Send tab:

a. *Figure 40: App Gateway Properties - Send*
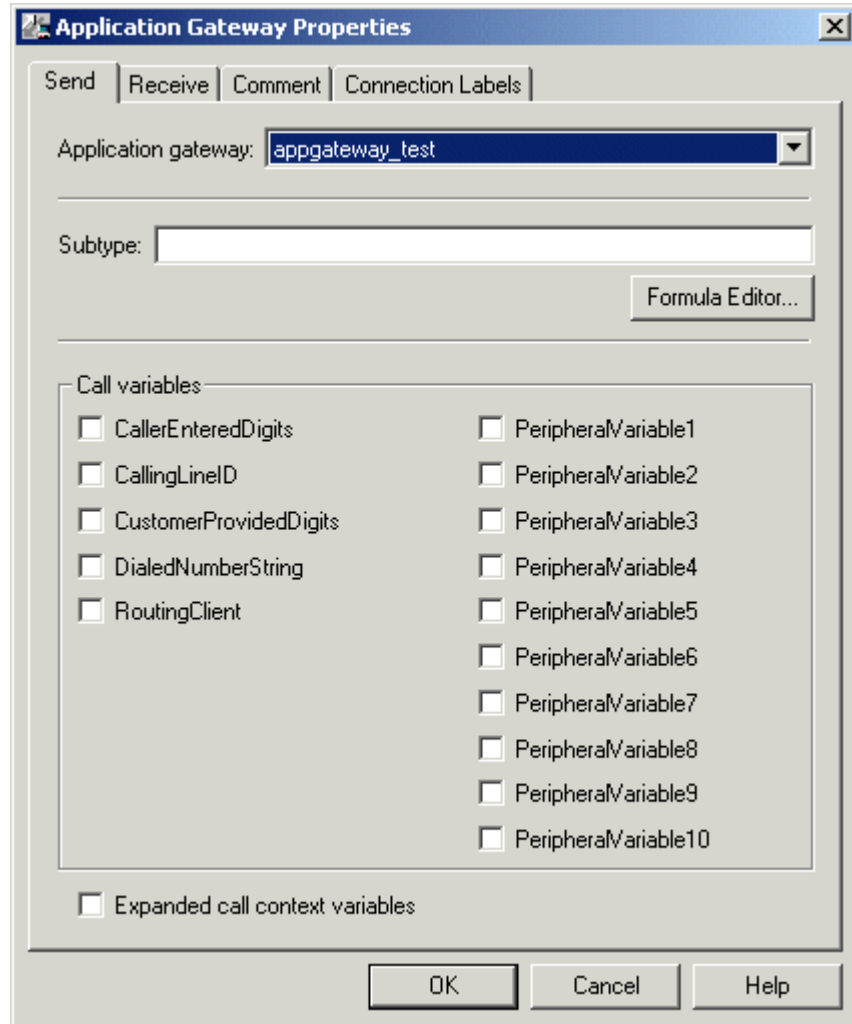
In the Application Gateway drop-down list, select the gateway.

b. In the Subtype filed, enter the string that is to be sent to the external application, or use the Formula Editor to write an expression that evaluates to a string.

c. In the Call variables list, check the call variables to send to the external application.

d. To send expanded call variables to the external application, check Expanded call context variables.

2. In the Receive tab:

— *Figure 41: App Gateway Properties - Receive*
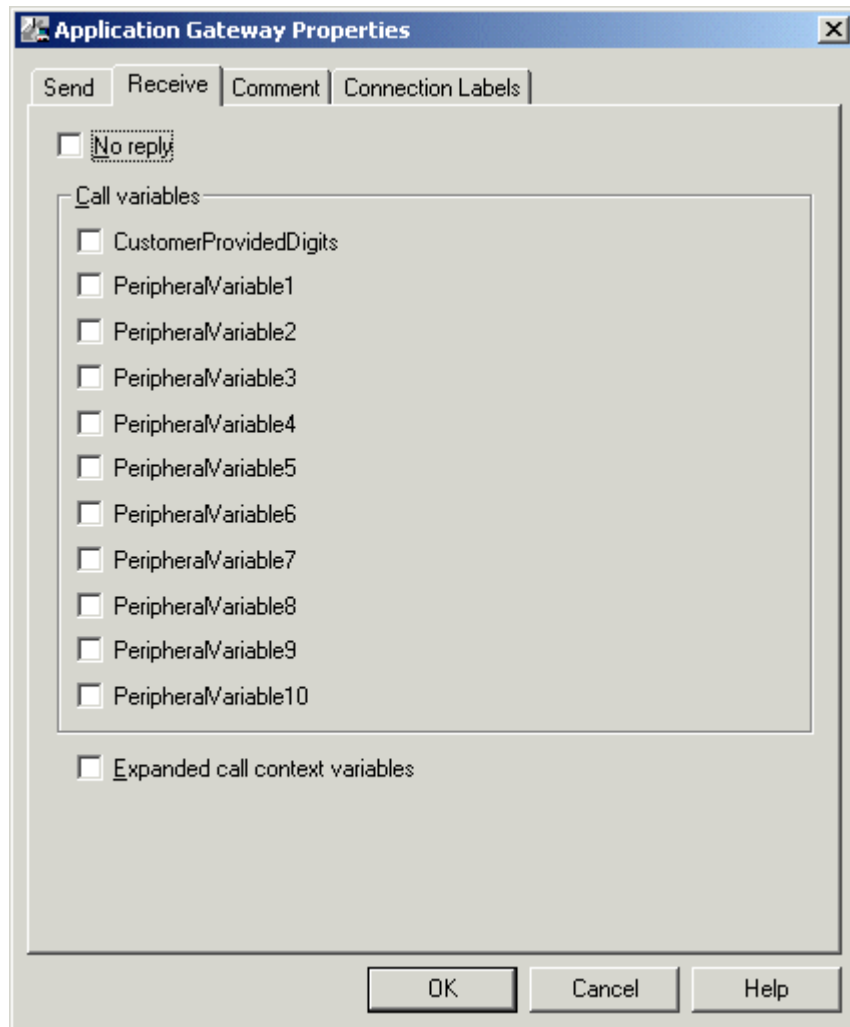
Check No Reply if the external application is not to return data to the script.

Note: If you select this option, ICM software is not able to retrieve any data from the external application.

— In the Call variables list, check variables that the external application may modify.

— Check Expanded call variables if the external application will modify and return values for the expanded call variables.

3. Optionally, add comments and connection labels.

Chapter 4

Selecting Routing Targets

After defining how a script is to categorize contacts, you typically use the nodes available in Script Editor to specify how the contact is to be routed to a target. By selecting routing targets, you determine the destination for contacts.

This chapter contains the following topics:

What is a Routing Target?

A routing target is an entity to which ICM software can route a contact. The routing target receives the contact and processes it accordingly.

There are two types of routing targets: Skill targets and Network targets.

What is a Route?

A value returned by a routing script that maps to a target at a peripheral, such as a service, skill group, agent, or translation route to a label.

Note: You create routes through Configuration Manager before writing routing scripts. For more information, see the *Cisco ICM Software Configuration Guide*

What is a Translation Route?

A translation route is a target at a peripheral that does not map to a specific service, skill group, or agent. When a contact arrives with the trunk group and DNIS that correspond to a translation route, the Peripheral Gateway (PG) is responsible for determining the ultimate target. When ICM software routes a call to a translation route, it sends a message to the PG. This message contains the ultimate target and further instructions for the PG. For example, the PG might be instructed to coordinate with a host computer so that the callers account number is displayed on the teleset of the agent who picks up the call.

Note: You create translation routes through Configuration Manager before writing routing scripts. For more information, see the *Cisco ICM Software Configuration Guide*

What is a Target Set?

A target set is a list of possible targets. During script processing, the actual target is chosen from the set by the preceding node on the script branch, a Select Node or Distribute Node.

Skill Target

A skill target is an entity at a peripheral or in the enterprise to which ICM software can route a contact. There are two types of skill targets: **Peripheral-level skill targets** and **Enterprise-level skill targets**.

Peripheral-level skill targets include:

- Agents

- Skill groups

- Services

Enterprise-level skill targets include:

- Enterprise skill groups

- Enterprise services

Agents: How to Define a Set of Agents Who Can Receive the Contact

There are three nodes available for agent routing:

1. Queue to Agent Node. See Queuing to Agents (page 128)

2. Agent to Agent Node. See Transferring a Call from Agents to Agents (page 82)

3. Agent Node

Agent Node

You define a set of agents who can receive the contact by using the Agent node in the Targets tab of the Palette.

Note: Do not use this node in IPCC environments. It is intended for use with ACDs.

Figure 42: The Agent Icon

The script can determine the target agent from the set by one of the following methods:

- Selecting the target by rules (Select node)

- Distributing contacts to targets in the set (Distribute node)

- A combination of selecting the target and distributing contacts (Route Select node)

Following is the Properties dialog box of the Agent node:

Figure 43: Agent Properties - Routing Target Tab
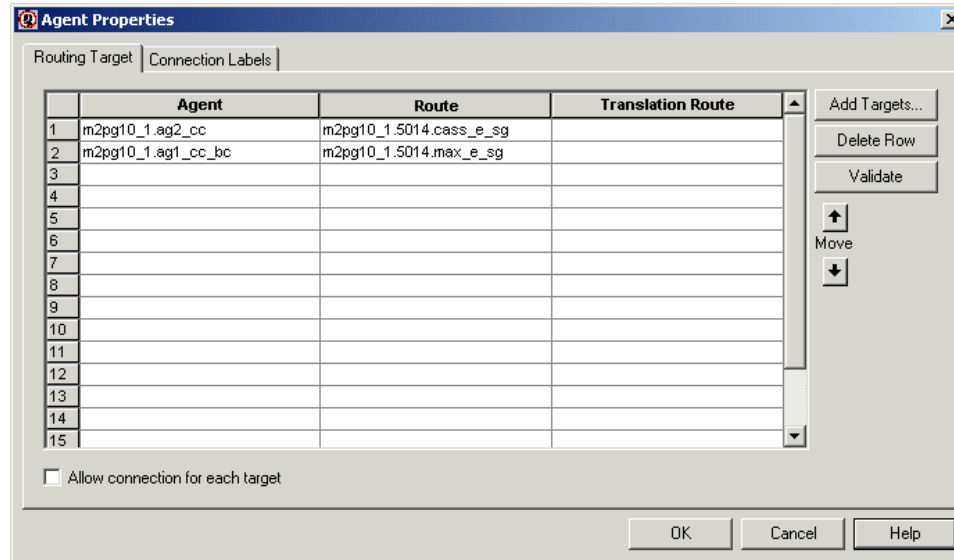
Define Agent node properties as follows:

1. For each agent in the target set:

   – In the Agent column, for each row used, select the agent to which the contact can be routed. You can use the drop-down list for each table cell, or select multiple agents by clicking Add Targets and using the dialog box that opens to select multiple agents.

   – In the Route column, select the route that maps to a specific target at the peripheral.

   – Optionally, in the Translation Route column, select a translation route.

2. Optionally, check Allow connection for each target to have an output terminal appear to the right of each individual target defined in the node. Control passes through this terminal when the associated target is chosen. When the script terminates, the route for the selected agent is still used.

3. Click Validate to check whether the targets you defined are valid. Correct any errors that are flagged.

4. Optionally, add connection labels.

Skill Groups: How to Define a Set of Skill Groups That Can Receive the Contact

You define a set of skill groups that can receive the contact by using the Skill Group node in the Targets tab of the Palette.

Figure 44: The Skill Group Icon

You use the Skill Group node to define the set of skill groups that can receive the contact. The script can determine the target skill group from the set by one of the following methods:

• Selecting the target by rules (Select node)

• Distributing contacts to targets in the set (Distribute node)

• A combination of selecting the target and distributing contacts (Route Select node)

Following is the Properties dialog box of the Skill Group node:

Figure 45: Skill Group Properties - Routing Target Tab
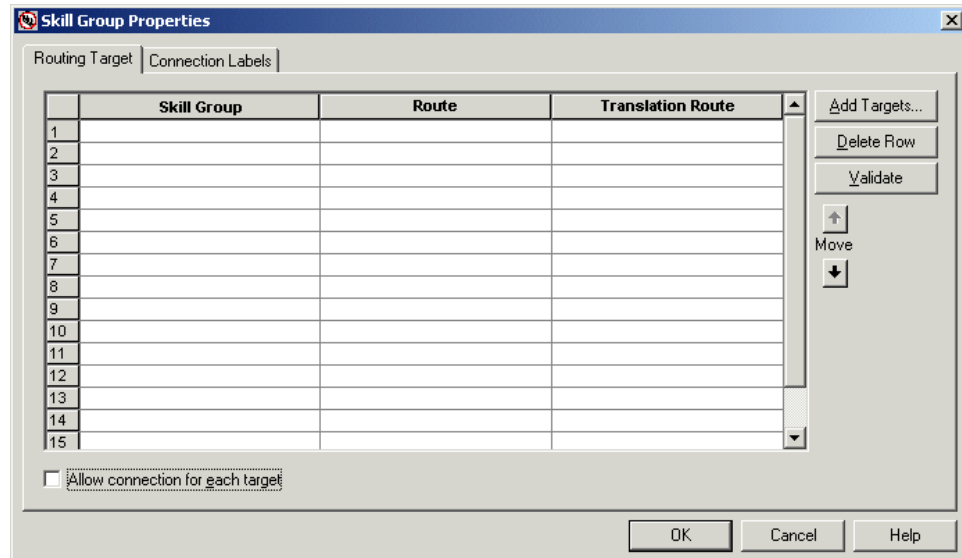
Define Skill Group node properties as follows:

1. For each skill group in the target set:

   – In the Skill Group column, for each row used, select the skill group to which the contact can be routed. You can use the drop-down list for each table cell, or select multiple skill groups by clicking Add Targets and using the dialog box that opens to select multiple skill groups.

   – In the Route column, select the route that maps to a specific target at the peripheral.

   – Optionally, in the Translation Route column, select a translation route.

2. Optionally, check Allow connection for each target to have an output terminal appear to the right of each individual target defined in the node. Control passes through this terminal

when the associated target is chosen. When the script terminates, the route for the selected skill group is still used.

3. Click Validate to check whether the targets you defined are valid. Correct any errors that are flagged.

4. Optionally, add connection labels.

Services: How to Define a Set of Services That Can Receive the Contact

You define a set of services that can receive the contact by using the Service node in the Targets tab of the Palette.

Figure 46: The Service Icon

The script can determine the target service from the set by one of the following methods:

- Selecting the target by rules (Select node)

- Distributing contacts to targets in the set (Distribute node)

- A combination of selecting the target and distributing contacts (Route Select node)

Following is the Properties dialog box of the Service node:

Figure 47: Service Properties - Routing Target Tab
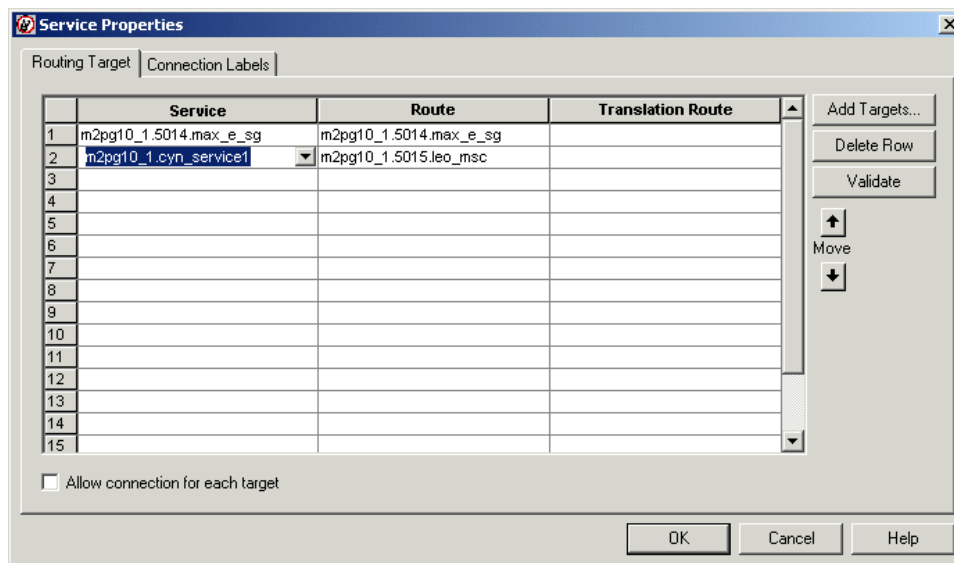
Define Service node properties as follows:

1. For each service in the target set:

   – In the Service column, for each row used, select the service to which the contact can be routed. You can use the drop-down list for each table cell, or select multiple services by clicking Add Targets and using the dialog box that opens to select multiple services.

   – In the Route column, select the route that maps to a specific target at the peripheral.

   – Optionally, in the Translation Route column, select a translation route.

2. Optionally, check Allow connection for each target to have an output terminal appear to the right of each individual target defined in the node. Control passes through this terminal when the associated target is chosen. When the script terminates, the route for the selected service is still used.

3. Click Validate to check whether the targets you defined are valid. Correct any errors that are flagged.

4. Optionally, add connection labels.

Enterprise Skill Groups: How to Define a Set of Enterprise Skill Groups That Can Receive the Contact

You define a set of enterprise skill groups that can receive the contact by using the Enterprise Skill Group node in the Targets tab of the Palette.

Figure 48: The Enterprise Skill Group Icon

The script can determine the target enterprise skill group from the set by one of the following methods:

- Selecting the target by rules (Select node)

- Distributing contacts to targets in the set (Distribute node)

- A combination of selecting the target and distributing contacts (Route Select node)

Following is the Properties dialog box of the Enterprise Skill Group node:

Figure 49: Enterprise Skill Group Properties - Routing Tab
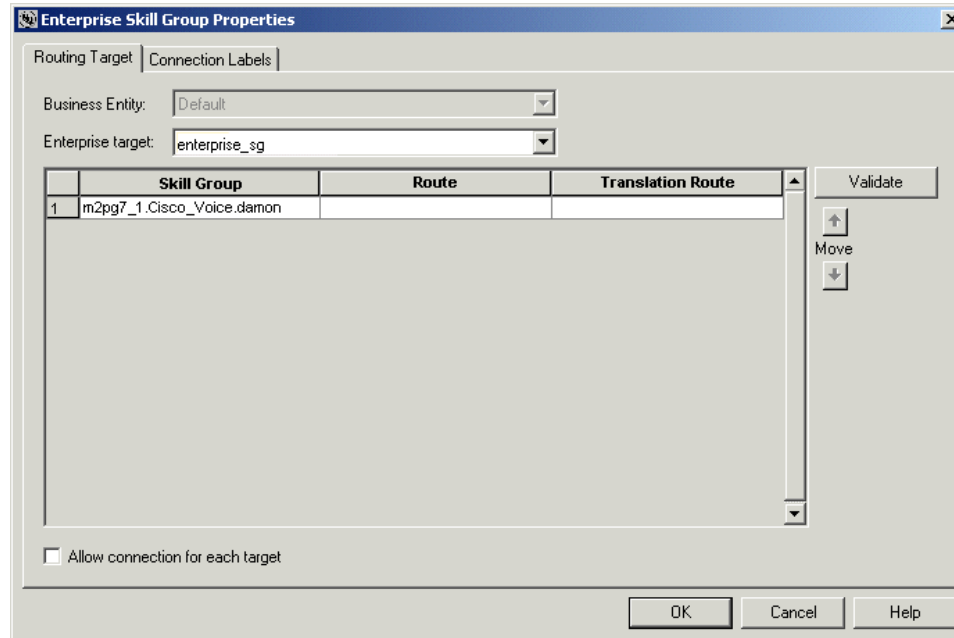
Define Enterprise Skill Group node properties as follows:

1. From the Business Entity drop-down list, select the business entity for the enterprise skill groups.

2. From the Enterprise target drop-down list, select the enterprise target for the enterprise skill groups.

3. For each enterprise skill group in the target set:

   – In the Skill Group column, for each row used, select the enterprise skill group to which the contact can be routed.

   – In the Route column, select the route that maps to a specific target at the peripheral.

   – Optionally, in the Translation Route column, select a translation route.

4. Optionally, check Allow connection for each target to have an output terminal appear to the right of each individual target defined in the node. Control passes through this terminal when the associated target is chosen. When the script terminates, the route for the selected enterprise skill group is still used.

5. Click Validate to check whether the targets you defined are valid. Correct any errors that are flagged.

6. Optionally, add connection labels.

Enterprise Services: How to Define a Set of Enterprise Services That Can Receive the Contact

You define a set of enterprise services that can receive the contact by using the Enterprise Service node in the Targets tab of the Palette.

Figure 50: The Enterprise Service Icon

The script can determine the target enterprise service from the set by one of the following methods:

- Selecting the target by rules (Select node)

- Distributing contacts to targets in the set (Distribute node)

- A combination of selecting the target and distributing contacts (Route Select node)

Following is the Properties dialog box of the Enterprise Service node:

Figure 51: Enterprise Service Properties - Routing Target Tab

Define Enterprise Service node properties as follows:

1. From the Business Entity drop-down list, select the business entity for the enterprise services.

2. From the Enterprise target drop-down list, select the enterprise target for the enterprise services.

3. For each enterprise service in the target set:

   – In the Service column, for each row used, select the enterprise service to which the contact can be routed.

   – In the Route column, select the route that maps to a specific target at the peripheral.

   – Optionally, in the Translation Route column, select a translation route.

4. Optionally, check Allow connection for each target to have an output terminal appear to the right of each individual target defined in the node. Control passes through this terminal when the associated target is chosen. When the script terminates, the route for the selected enterprise service is still used.

5. Click Validate to check whether the targets you defined are valid. Correct any errors that are flagged.

6. Optionally, add connection labels.

Network Targets

A network target is an end point on the network to which a script can send a contact. Following are the supported network targets:

- Announcement

- Scheduled Select

- Ring

- Busy

- Label

- Divert Label

How to Route a Call to an Announcement

You can route a call to an announcement on the network by using the Announcement node in the Targets tab of the Palette.

Figure 52: Announcement Icon

When executing the Announcement Node, ICM software returns the label associated with the announcement to the routing client. The Announcement Node terminates the script.

You might use the Announcement Node to:

- Explain why the call cannot currently be handled.

- Direct the caller to another phone number or to another way of contacting the company.

Note: You must configure Announcements and associated them with labels using the ICM Configuration Manager. For more information, see the *Cisco ICM Software Configuration Guide*.

Following is the Properties dialog box of the Announcement node:

Figure 53: Announcement Properties - Announcement Tab
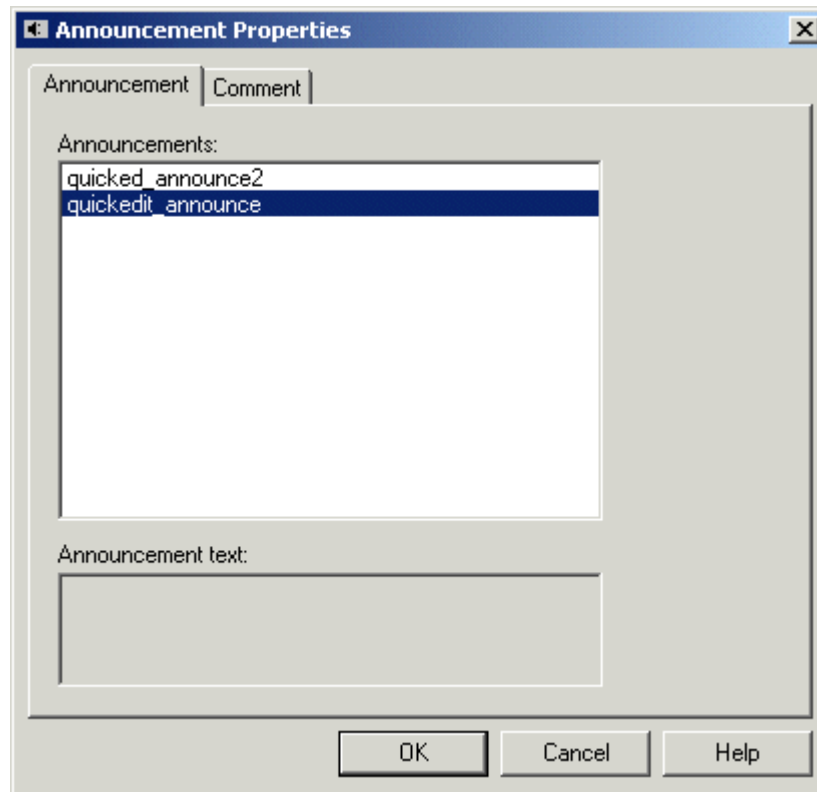
Define Announcement node properties as follows:

1. Select an announcement from the Announcements list.

2. Optionally, add comments.

How to Route a Call to a Scheduled Target

You can send a call to a target that is configured to handle a specific number of simultaneous contacts based on a periodic schedule by using the Scheduled Select node in the Targets tab of the Palette.

Figure 54: Schedule Select Icon

ICM software keeps track of the schedule and the number of calls sent to the target. The routing client informs ICM software when a call at the target ends, so ICM software always knows how many calls are currently in progress at the target and whether it can handle an additional call.

When ICM software executes the Scheduled Select Node, it searches the list of selected targets for one that is capable of handling the contact, based on its current schedule. If a target is found, the routing script ends and returns a label associated with the target to the routing client. The routing client then translates the label to a peripheral target.

Note: You must configure Scheduled targets and associated them with labels using the ICM Configuration Manager. For more information, see the *Cisco ICM Software Configuration Guide*.

Not all routing clients support the Scheduled Select node.

Following is the Properties dialog box of the Announcement node:

Figure 55: Schedule Select Properties
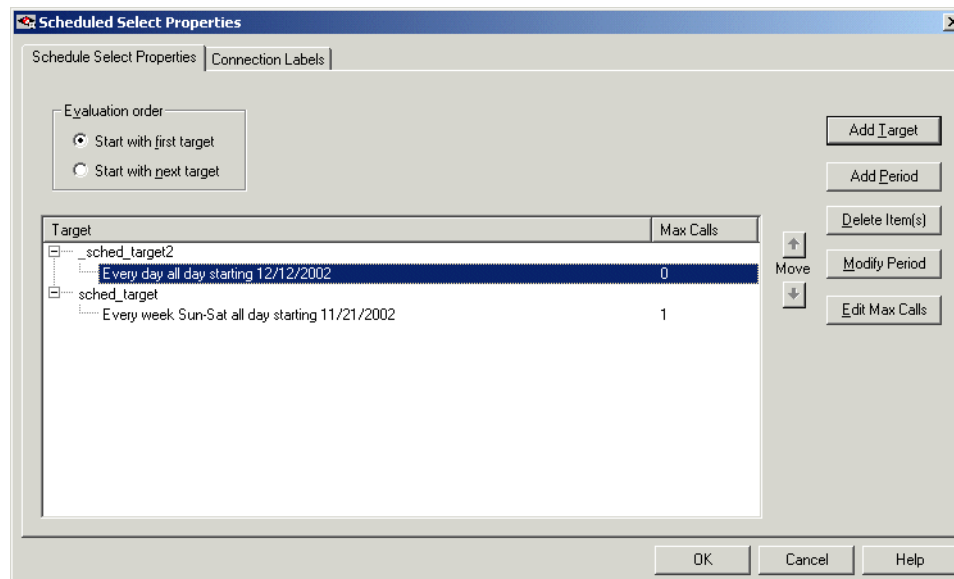
Define Scheduled Select node properties as follows:

1. In the Evaluation Order field select:

   – **Start with first target** to have ICM software always start the search from the first target in the list.

   – **Start with next target** to have ICM software start the search from the first target after the last chosen target.

2. Click Add Target to add a new scheduled target. In the Add Schedule Targets dialog box, select targets to add from the Available targets list, and click Add> to move them to the Add targets list. When finished, click OK. The targets are added to the list.

3. To add a time period for the scheduled target, select the target and click Add Period. The Add Periodic Schedule dialog box opens. Define the time period and click OK.

4. To modify a time period for the scheduled target, select the time period and click Modify Period. The Modify Periodic Schedule dialog box opens. Modify the time period and click OK.

5. To edit Max Calls for a time period, select the time period and click Edit Max Calls. The number in the Max Calls column is now editable. Modify the value as needed

6. Optionally, add connection labels.

How to Route a Call to an Unanswered Ring

You can route a call to an unanswered ring by using the Ring node in the Targets tab of the Palette.

Figure 56: Ring Icon

When ICM software executes a Ring node, it returns the first Ring label associated with the routing client; the routing client then plays an unanswered ring for the caller.

Note: You must define a Ring label using the ICM Configuration Manager. For more information, see the *Cisco ICM Software Configuration Guide*.

Note: Not all routing clients support the Ring Node.

To define the Ring node properties, you simply add comments to the node.

How to Route a Call to a Busy Signal

You can route a call to a busy signal buy using the Busy node in the Targets tab of the Palette.

Figure 57: Busy Icon

When ICM software executes a Busy node, it returns the first Busy label associated with the routing client; the routing client then plays a busy signal for the caller.

Note: You must define a Busy label using the ICM Configuration Manager. For more information, see the *Cisco ICM Software Configuration Guide*.

Note: Not all routing clients support the Busy Node.

To define the Busy node properties, you simply add comments to the node.

How to Return a Label to a Routing Client

You can return a label to a routing client by using the Label node in the Targets tab of the Palette.

Figure 58: Label Icon

When ICM software executes a Label node, it returns the first valid label for the routing client.

This differs from the Divert Label Node, which returns all the values in the Selected Labels list to the routing client.

If ICM software finds no valid label, it returns the default label for the dialed number. In either case, the Label node terminates execution of the script

When you define a Label Node, you can select Configured Labels and Dynamic Labels. Configured labels are static, defined through the Configuration Manager. Dynamic Labels are expressions the CallRouter processes in real time, converting an expression into a character string that is then returned to the routing client as a label. You use formulas to create a Dynamic Label.

Note: The Label node supports Target Requery.

Following is the Properties dialog box of the Label node:

Figure 59: Label Properties - Label Tab
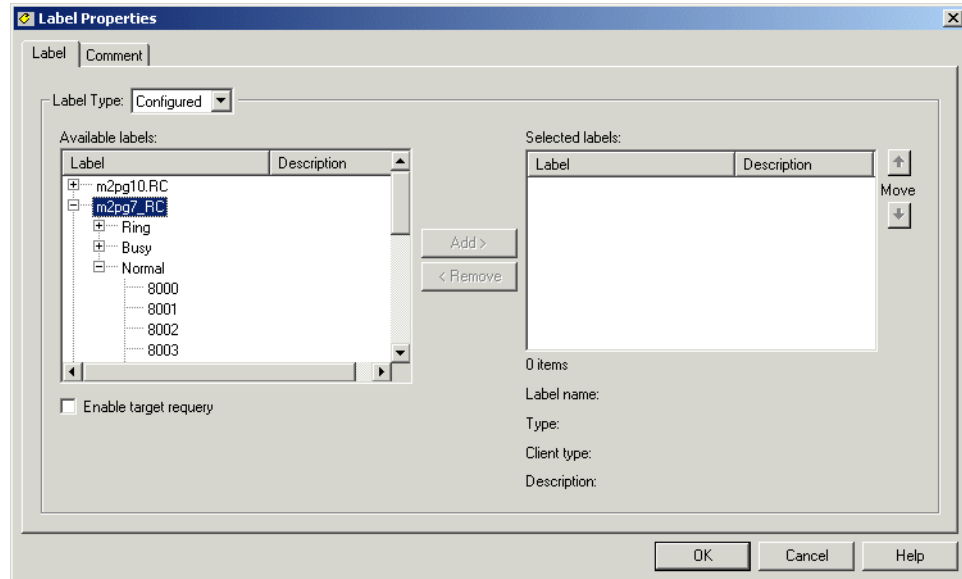
Define Label node properties as follows:

1. Select the Label Type:

   – Configured, to select from a list of configured labels.

   – Dynamic, to define an expression that is to be returned as a label.

2. If you select Configured, select labels from the Available labels list and click Add> to add them to the Selected labels list.

3. If you select Dynamic, enter a Label Expression, optionally using the Formula Editor.

4. Optionally, check Enable target requery.

5. Optionally, add comments.

How to Return Multiple Labels to a Routing Client

You can return multiple labels to a routing client by using the Divert Label node in the Targets tab of the Palette.

Figure 60: Divert Label Icon

When ICM software executes a Divert Label node, it returns all the values in the Selected Labels list to the routing client.

This differs from the Label node, which returns the first valid label for the current routing client.

ICM software then tries each label until it finds one that does not produce a Busy or Ring Tone No Response.

In most cases, you can specify up to ten labels in the Divert Label node. The exception is in NAM/CICM configurations, where you can specify up to nine labels; one label must be reserved for use by the NAM.

When you define a Divert Label node, you can select Configured Labels and Dynamic Labels. Configured labels are static, defined through the Configuration Manager. Dynamic Labels are expressions the CallRouter processes in real time, converting an expression into a character string that is then returned to the routing client as a label. You use formulas to create a Dynamic Label.

Notes:

- With the Divert Label Node, you can only select labels for routing clients with client types that support DivertOnBusy. The IPCC client type does not support DivertOnBusy. The following client types do support DivertOnBusy: BT INAP NIC, CRSP, DEUTSCHE TELEKOM, CWC NIC, ENERGIS INAP NIC, AUCS INAP NIC, CAIN, TELFORT INAP, BTV2 INAP NIC, GKTMP NIC, SS7IN NIC, and NTL NIC.

- When using a Divert Label node, arrange the labels so that a label defined with a type Busy or Ring the last label in the Divert Label Selected labels list. The reason for this is that a Busy or Ring label terminates the call; any label appearing after Ring or Busy will never be used. Also, never use a Divert Label node when only one label exists in the list; use the Label node, instead.

- Never use a Divert Label node when only one label exists in the list; use the Label node instead.

Define Divert Label node properties as follows:

1. Select the Label Type:

   – Configured, to select from a list of configured labels.

   – Dynamic, to define an expression that is to be returned as a label.

2. If you select Configured, select labels from the Available labels list and click Add> to add them to the Selected labels list.

3. If you select Dynamic, enter a Label Expression, optionally using the Formula Editor.

4. Optionally, add comments.

Selecting Targets by Rules

You can create a script to select a specific target from set of targets based on the rule you select or define.

For example, you can create a script that selects as a target the skill group with longest available agent (LAA) from a set of skill groups.

When defining scripts to select targets by rules, you must be aware of the following:

- Types of Target Searching

- Standard Selection Rules

- Custom Selection Rules

Types of Target Searching

When you use a Select node, you specify if the script is to search for the target that matches the criteria you define starting with the first target in the list, or the next target after the previously chosen target:

Start with first target - Also referred to as a homing search, this option has the script search for the target that meets the criteria you define starting with the first item in the list. For example, you can define a script to select the longest available agent from either the TechnicalSupport skill group (the first skill group in the list) or the BackUp skill group (the second, and last, item in the list). When you select this option, whenever the script can find an available agent in the TechnicalSupport skill group, the contact is routed to that agent. Agents from the BackUp skill group are only selected as targets when the script fails to find an available agent in the TechnicalSupport skill group. When you select this option, targets towards the top of the list will typically receive a higher percentage of the contacts.

Start with next target - Also referred to as a rotary search, this option has the script search for the target that meets the criteria you define starting with the item in the list after the target previously selected. For example, you can define a script to select the longest available agent from one of three separate technical support skill groups. When you select this option, after a script selects an agent from the first skill group, for the next contact, the script starts looking for an agent in the second skill group. When you select this option, contacts are distributed more evenly among the potential targets.

Standard Selection Rules

When you use the Select node, you can choose from one of the following standard selection rules:

Table 1: Select Node Standard Selection Rules

Selection Rule	Applicable Targets	Formula	Description
Always Select	Any target		Selects the first target that passes the specified acceptance rule.
Longest Available Agent (LAA)	Skill Groups and Enterprise Skill Groups	Consider if: *.AgentsAvail > 0 Evaluate: MAX (*.LongestAvailable)	Selects the target with the agent who has been available for the longest time. This selection rule helps to ensure that all agents in the skill group set are kept equally busy. It does not ensure that a particular agent is assigned the contact. If the target set includes a skill group that has subgroups (.pri, .sec, etc.), only agents logged in to the base group are considered. Since agents do not normally log in to the base group, specify the base groups you want to consider.
Next Available Agent (NAA)	Skill Groups and Enterprise Skill Groups	Consider if: *.AgentsAvail > 0 Evaluate: MAX (*.LongestAvailable/*.AgentsSignedOn)	Selects the target with the highest percentage of available agents.
Minimum Average Speed Answer (Min ASA)	Services and Enterprise Services	Evaluate: MIN (*.AvgSpeedAnswerTo5)	Selects the target in the set that is, on average, answering contacts most quickly. Because this selection rule evaluates the historical average, it does not select a target based on the current or expected future state of the contact center. Therefore, unexpected load imbalances may occur when you use this rule. To avoid this potential problem, you can use the Minimum Expected Delay selection rule instead.
Minimum Calls in Queue Per Position (Min C/Q)	Services and Enterprise Services	Evaluate: MIN (*.CallsQNow/*.AgentsReady)	Selects the target in the set with the lowest ratio of calls waiting and staffed stations. If agents are equally efficient at each target in the set, this rule tends to lead to the shortest average hold times. However, if agents are not equally efficient, some customers might wait longer than necessary at the less efficient target. To avoid this potential problem, you can use the Minimum Expected Delay selection rule instead.
Minimum Average Queue Delay (Min AvgQD)	Services and Enterprise Services	Evaluate: MIN (*.AvgDelayQTo5)	Selects the target in the set with shortest average hold times, assuming that agents at each target are equally efficient. Because this selection rule evaluates the historical average, it does not select a target based on the current or expected future state of the contact center. Therefore, unexpected load imbalances may occur when you use this rule. To avoid this potential problem, you can use the Minimum Expected Delay selection rule instead.

Selection Rule	Applicable Targets	Formula	Description
Minimum Longest Delayed Call (Min Delay)	Services and Enterprise Services	Evaluate: MIN (*.LongestCallQ)	Selects the target with the shortest longest delayed call. Note: This selection rule evaluates the historical average, not the current or expected future state of the contact center. Routing contacts to the target with the shortest longest delayed call does not immediately change the longest delay value. Therefore, this selection rule may route a disproportionately large number of calls to a single target. To avoid this potential problem, you can use the Minimum Expected Delay selection rule instead.
Minimum Expected Delay (MED)	Services and Enterprise Services	Evaluate: MIN (*.ExpectedDelay)	Selects the target with the shortest expected delay. In making this evaluation, this selection rule considers the average handle time, the number of contacts in queue, and the number of positions staffed. This rule is usually the most effective rule for keeping queue times to a minimum. The MED algorithm is not supported on IPCC.

Caution: Values used by the standard selection rules Minimum Average Speed Answer, Minimum Average Queue Delay, and Minimum Longest Delayed Call change slowly. Because the averages in these rules only consider what has happened in the past rather than what is currently happening, using these rules inappropriately can lead to load imbalances as newly routed contacts have little immediate effect on the values used to route later contacts. In contrast, the standard selection rule Minimum Expected Delay takes into account each contact as it is routed. Selecting the service using the Minimum Expected Delay rule usually provides the best balance among the services in the target set.

Custom Selection Rules

Instead of using one of the standard selection rules, you can create a custom selection rule. To write custom selection rules, you must be familiar with using formulas (page 133).

A custom selection rule has three parts:

- **Consider if statement** 0. This expression ensures that only skill groups with at least one available agents are considered as targets; if a skill group has no available agents, the expression returns False and the skill group is not considered.

- **Selection criteria**. A numeric expression plus an evaluation rule. The numeric expression determines a certain value of each target for which the Consider if statement returned True. The evaluation rule determines if the target with the minimum or maximum value returned is selected. If you do not define selection criteria, the first target to return True for the Consider if statement is selected. For example, the Selection criteria for the predefined Longest Available Agent selection rule is MAX (*.LongestAvailable). This expression determines

the highest LongestAvailable value of all the skill groups in the target set for which the Consider if statement returned True.

- **`Accept if` statement**. A Boolean expression that returns True or False for the target selected by the selection criteria. If the Accept if statement returns True, the target is selected; if it returns False, no target is selected by the Select node.

How to Select Targets by Rules

You can select targets by rules buy using the Select node in the Routing tab of the Palette.

Figure 61: Select Node Icon

The Select node sets up a rule by which the node chooses from a set of routing targets for the contact. You can select a standard rule or define your own custom rule. This node also has target requery capabilities.

Notes:

- You can follow the instructions in this section to select targets by rules using the Select node. You can also use the Route Select node to select targets as well as distribute contacts to targets.

- You must add a skill target and create a connection from the Select node's success terminal, and define at least one target in the target set, before defining the Select node.

- The Select Node supports Target Requery.

Following is the Properties dialog box of the Select node:

Figure 62: Select Properties - Select Tab
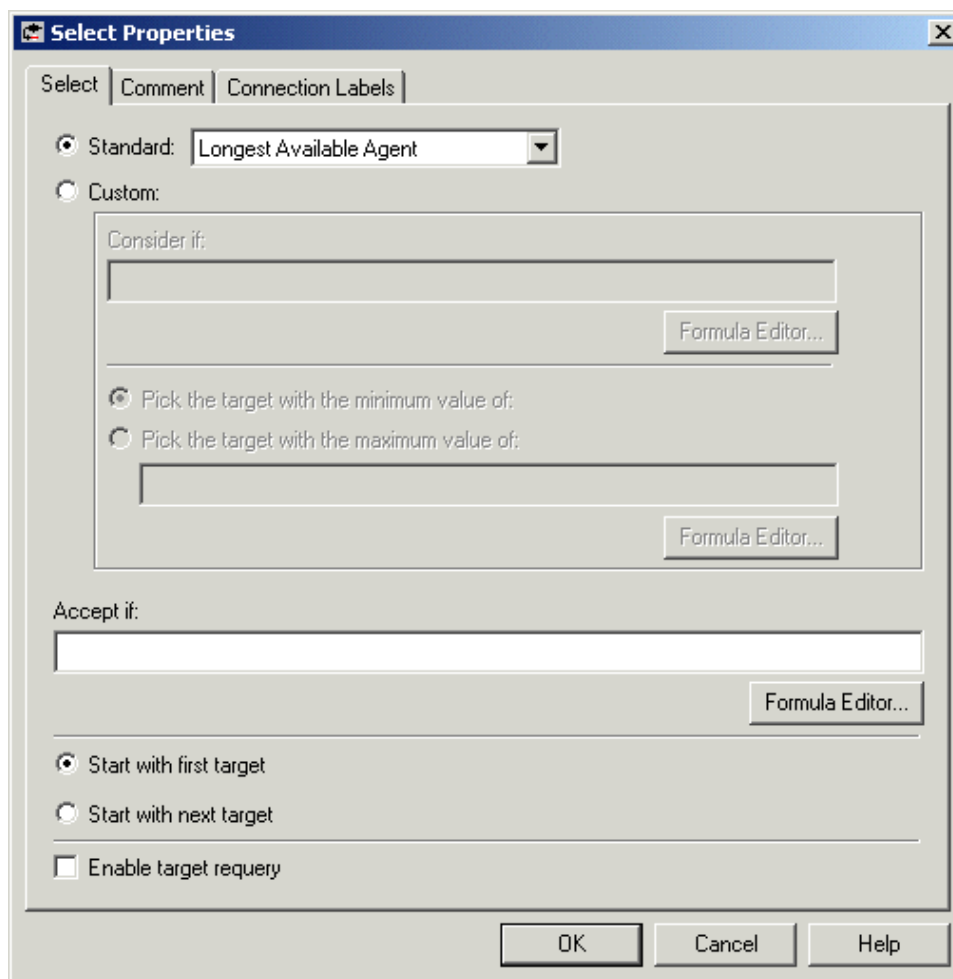
Define Select node properties as follows:

1. Select from the list of Standard rules, or select Custom and:

   – In the Consider if field, enter a Boolean expression. A target is considered for selection only if the Consider If expression is true for that target. Optionally, use the Formula Editor.

   – Select Pick the target with the minimum value of or Pick the target with the maximum value of and enter an expression to select the target.

2. Optionally, in the Accept if field, enter a Boolean expression that must evaluate to true for the target to be selected.

3. Select Start with first target or Start with next target to indicate how ICM software looks for targets. See Types of Target Searching.

4. Optionally, check Enable target requery.

5. Optionally, add comments and connection labels.

Distributing Contacts to Targets

You can create a script to allocate contacts among the targets in the set based on current information about each target by using the Distribute node in the Routing tab of the Palette.

Figure 63: Distribute Icon

For example, you can distribute contacts among a set of services based on the number of agents in the service in the Ready state. Services with more agents in the ready state will be routed more contacts than services with fewer agents in the Ready state, thus keeping the load balanced.

You can distribute contacts to any of the following types of target sets: Agent, Skill Group, Service, Enterprise Skill Group, or Enterprise Service.

When creating a script to distribute contacts to targets, you must define the following for the Distribute node using formulas (page 133).

- **`Consider if` statement**. A Boolean expression that returns True or False for each target in the target set. Only targets in the set for which the expression is True are eligible to be distributed contacts. If you do not define an expression for the Consider if statement, all targets in the target set are considered. For example,

- **`Distribute by` statement**. A formula used to distribute contacts to targets

Notes

- You can follow the instructions in this section to distribute contacts to targets using the Distribute node. You can also use the Route Select node to select targets as well as distribute contacts to targets.

- You must add a skill target and create a connection from the Distribute node's success terminal, and define at least one target in the target set, before defining the Distribute Enter node.

Following is the Properties dialog box of the Distribute node:

Figure 64: Distribute Properties - Distribute Tab
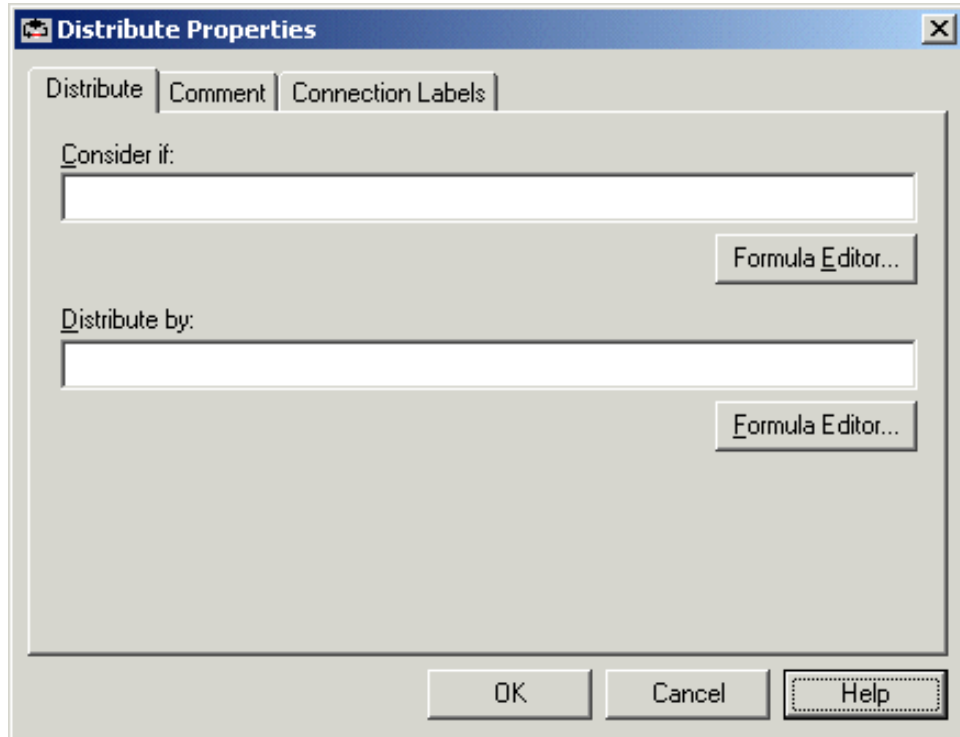
Define Distribute node properties as follows:

1. Enter a condition in the Consider if field to test potential targets against.

2. Enter a formula by which to distribute contacts in the Distribute by field.

3. Optionally, add comments and connection labels.

Selecting Targets and Distributing Contacts Using One Node

You can combine the functionality of selecting targets by rules with the Select node and distributing contacts to targets with the Distribute node by using the Route Select node in the Targets tab of the Palette.

Figure 65: Route Select Icon

Note: The Route Select Node supports Target Requery.

Following is the Properties dialog box for the Route Select node:

Figure 66: Route Select Properties - Route Tab
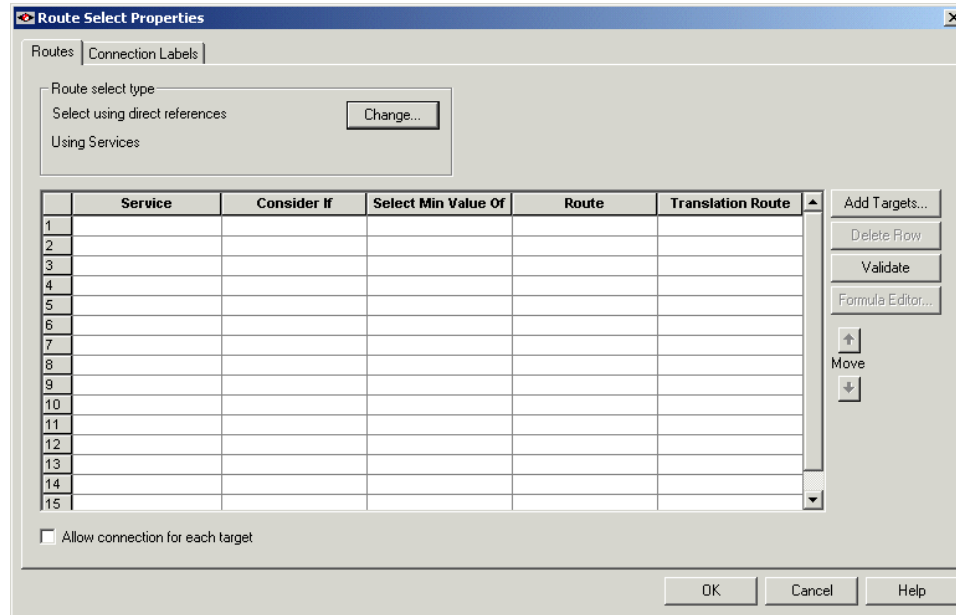
Define Distribute node properties as follows:

1. To select the Route select type, click Change. The Route Select Type dialog box opens:

— *Figure 67: Route Select Type*

For Target Type, select **Agent**, **Enterprise Service**, **Enterprise Skill Group**, **Service**, **Service Array**, or **Skill Group**.

— If you selected Enterprise Service or Enterprise Skill Group, select a Business Entity and Enterprise target.

— Select to **Distribute among targets** or **Select most eligible targets**.

— If you selected **Select most eligible targets**:

Select **Pick the target with the minimum value** or **Pick the target with the maximum value**.

In the Accept target if field, enter a condition that the target must meet to be selected.

Select **Start with first target** or **Start with next target**. See Types of Target Searching.

— In the Target references field, select **Explicit target references** to use direct references to targets, or **Lookup target references by expression** to use expressions that evaluate to names of targets.

> – Optionally, check **Enable target requery**.
>
> – Click **OK**.

2. The fields in the Route Select Properties dialog box change depending on your route select type selections. Enter and select data appropriate for the type you selected.

3. Click **Validate** to check whether the targets you defined are valid. Correct any errors that are flagged.

4. Optionally, add connection labels.

Transferring Calls from Agents to Agents

You can transfer a call from an agent to an agent by using the Agent to Agent node in the Targets tab of the Palette.

Figure 68: Agent to Agent Icon

The Agent to Agent node routes the call to the specified agent. You define the agent either by directly selecting the agent from the database or by providing an expression using a formula. The expression must translate to agent peripheral number or SkillTargetID. The router then finds a valid label for the agent. If there are no labels configured for the specified agent, the failure node of the Agent to Agent node is executed.

Following is the Properties dialog box for the Agent to Agent node:

Figure 69: Agent to Agent Properties
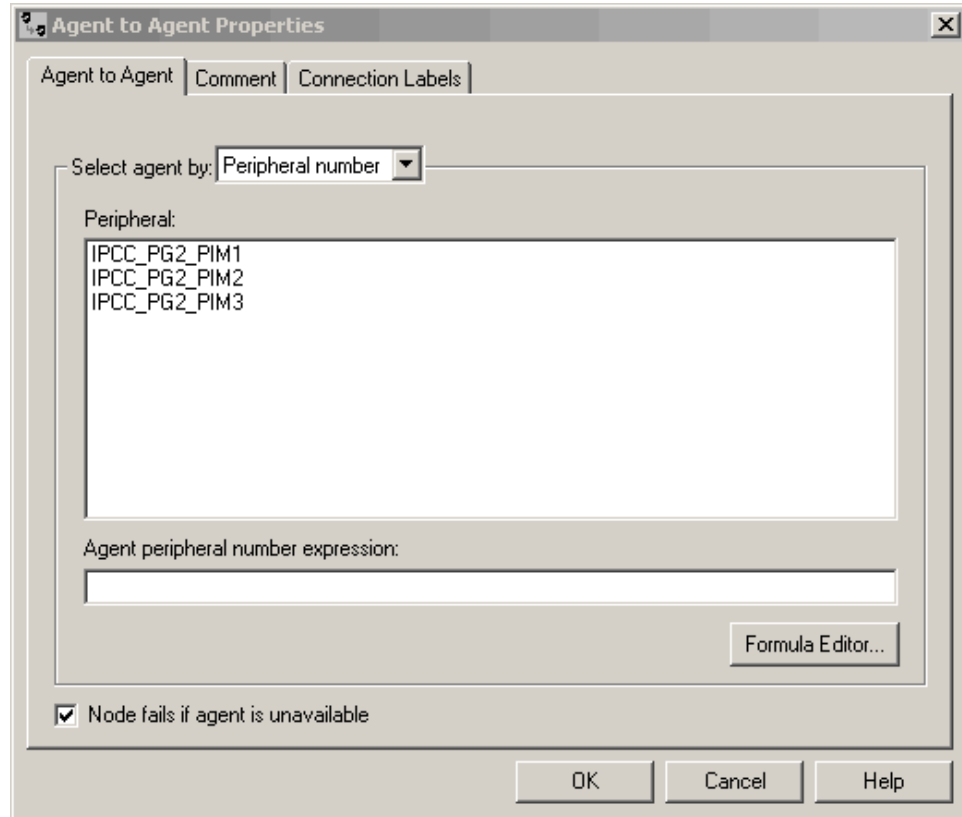
Define Agent to Agent node properties as follows:

1. Select an option from the Select agent by drop-down list:

    – Peripheral number - To select a peripheral and a provide formula that translates to the agent's peripheral number.

    – Enterprise Name - To select the agent from the list of configured agents.

    – Skill target ID To select the agent by providing an expression that translates into the agent's SkillTargetID. In the IPCC supervisory case, the expression should use the call's PreferredAgentID.

2. Based on your selection in Step 1, select the peripheral or agent, or enter an expression, as necessary.

3. Optionally, check or uncheck Fail node if agent is unavailable:

    – When checked, the success branch of the Agent to Agent node is executed and the router sends the call if:

    The router finds a valid label for the agent, and

    The agent is available, and

    The agent state is Ready.

– The failure branch of the Agent to Agent node is executed if:

The router does not find a valid label for the agent, or

The agent is not available or

The agent is in TempUnavailable mode (the router has just send a call to the agent).

– When not checked:

The success branch of the Agent to Agent node is executed and the router sends the call if the router finds a valid label for the agent.

The failure branch of the Agent to Agent node is executed if the Router does not find a valid label for the agent.

4. Optionally, add comments and connection labels.

Sending a Contact to a Different ICM System

A script may determine that the contact should be routed by a different ICM system.

You send a contact to a different system by using the ICM Gateway node in the Routing tab of the Palette.

Figure 70: ICM Gateway Icon

The ICM Gateway node passes a routing request to the selected ICM system.

Note: Before you can successfully use an ICM Gateway node in a script, you must use the Configuration Manager to configure a gateway to the ICM system to which you send the request. For more information, see the Cisco ICM Software Configuration Guide.

Define the ICM Gateway node properties as follows:

1. In the Send tab:

– Select the gateway to the ICM system (and hence the specific ICM instance) from the ICM Gateways list to which you want to send the request.

– Check Validate returned labels to have ICM software validate returned labels.

– Specify whether Calling Line ID masking instructions should be applied before the request is passed to the other ICM system. The Calling Line ID masking refers to when the caller's phone number is modified so that the ICM application does not display all of the digits; this is used in a NAM environment, where NAM sends the call to a customer ICM.

Select one of the following:

Do not apply masking rule - If selected, masking instructions are ignored.

Apply masking rule if call is presentation restricted - If selected, applies masking instructions if the call variable CLIDRestricted is set to 1.

Always apply masking rule - If selected, masking instructions are always applied.

Note: The Calling line ID masking rule is set through the Configuration Manager's System Information dialog box.

2. In the Default Label tab:

   – *Figure 71: ICM Gateway Properties*
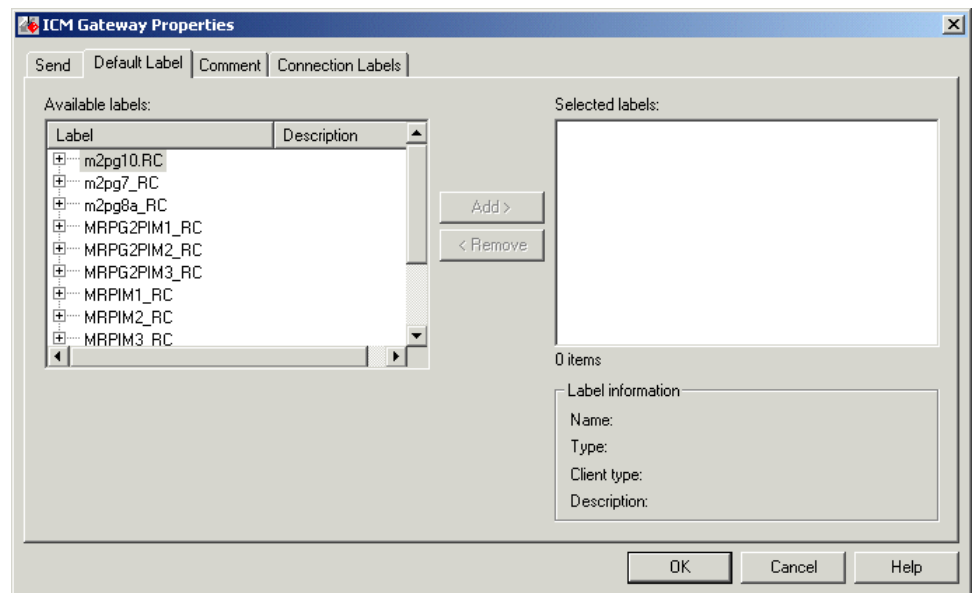
   In the Available Labels list, select one default label for each routing client to be used if the targeted ICM system:

   Returns a list of labels for divert-on-busy processing

   Fails to return valid labels

   – Click Add to move the selected label to the Selected labels list.

3. Optionally, add comments and connection labels.

Stopping Script Processing

You can use three nodes to stop script processing:

End Node

Termination Node

Release Call Node

End Node

You can terminate the script by using the End node in the General tab of the Palette.

Figure 72: End Icon

If the script reaches the End node, it has failed to find a target for the contact and ICM software then uses the default route for the dialed number.

Several End nodes can appear in the same script. The End node is never required; a script can terminate with any node.

You do not define any properties for the End node. You can optionally add comments.

Termination Node

You can terminate the script and specify how to handle the contact by using the Termination node in the Targets tab of the Palette.

Figure 73: Termination Icon

The Termination node includes the following options to invoke a default contact processing action or route for the dialed number:

• Default Label - ICM software uses the default label configured for the dialed number.

• Network Default - The routing client uses its own default processing.

• Ring - Makes the Termination Node equivalent to a Ring Node.

• Busy - Makes the Termination Node equivalent to a Busy Node.

Following is the Properties dialog box for the Termination node:

Figure 74: Termination Properties
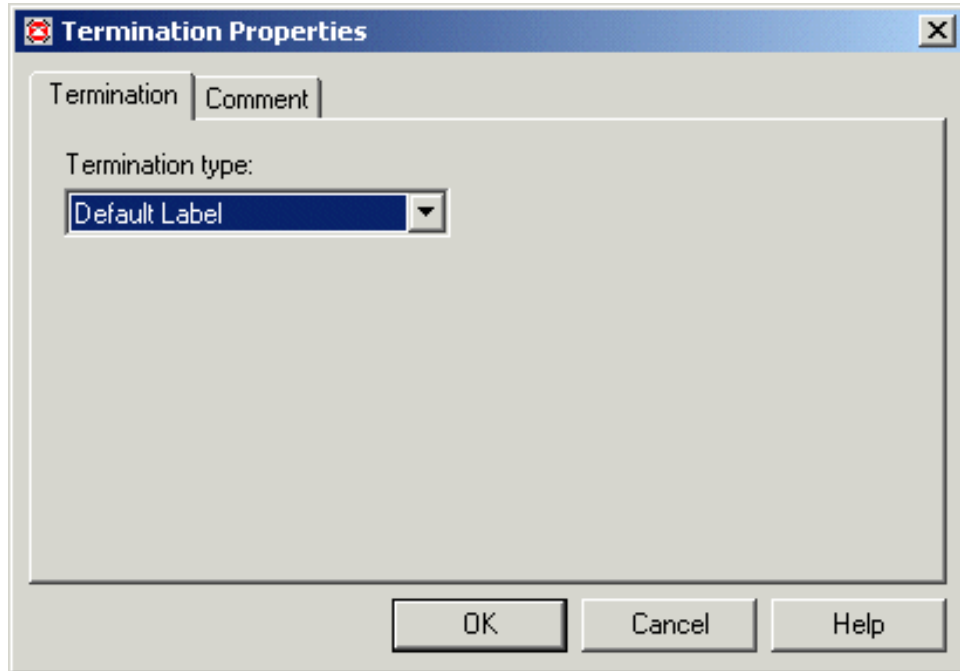
Define Termination node properties as follows:

1. Select the Termination type.

2. Optionally, add comments.

Release Call Node

You can terminate the script and disconnect the caller by using the Release Call node in the Targets tab of the Palette.

Figure 75: Release Call Icon

You can use a Release Call node in situations where the caller needs no further service after executing several IVR scripts.

You do not define any properties for the Release Call node. You can optionally add comments.

Using Target Requery

Target Requery is a script node feature that addresses routing failures due to transient failures in the network (such as network congestion). If the determined destination for a contact is available but not reachable, Target Requery attempts to find a different valid destination.

Target Requery is intended to address the following failures:

- Failure to deliver a call to an ACD agent.

- Failure to deliver a call to an individual Enterprise Agent (EA).

- Failure of the outbound leg of a blind-mode Network Transfer.

- Target Requery works on a per call basis; that is, the routing information for one call does not affect the state for other calls. If the first target selected for the contact was not reachable, the target is not eliminated from the potential routing destinations for other contacts.

Note: Target Requery does work in the NICM/CICM environment.

How Target Requery Works

In the ICM software system, when queried, the CallRouter returns a label to the routing client. The routing client then routes the call to the destination specified by the label. If the destination is not reachable (for example, because of a busy signal or no answer), the call is routed to the default destination.

With Target Requery, if the router fails to route to the target node, a second attempt is made. If the failure occurs a second time, then the router continues from the failure path in the node.

Note: Note that in a Queue Node, just one target is used and the failure path of the node is immediately taken. You can create a script that increases the priority and requeries the call from the failure path to the same queue.

In the event of a failure, you can handle requerying in the scripting environment, as you deem appropriate.

Target Requery:

- Is compatible with translation routes.

- Does not require different definitions for different failure cases. However, you can choose to handle different failures differently.

- Assigns the SERVICE_DIVERT_ON_BUSY service type for calls that use target requery. The Event Select in the connect message includes a REROUTE_REQUESTED_MASK. When a destination cannot be reached, the NIC queries the CallRouter for an alternative destination label.

How to Test the RequeryStatus Variable

You can test the error path of these script nodes using Target Requery to determine the specific network cause of failure and conditionally retry the attempt as necessary. This is done using an If node to check the value of the Call variable RequeryStatus. The decision path for the script is then determined by the value of the RequeryStatus variable.

The following are possible values for the RequeryStatus variable:

Table 2: RequeryStatus Variables

Requery Status Variable	Description
REQUERY_ANSWER (0)	CallRouter internal use. Script ends and the call was successfully sent to the chosen target. Not visible to users.
REQUERY_ROUTE_SELECT_FAILURE (1)	Routing client generated error code from ReRouteReq msg indicating a Route Select failure
REQUERY_CALLED_PARTY_BUSY (2)	Routing client generated error code from ReRouteReq msg indicating the called party is busy
REQUERY_NO_ANSWER (3)	Routing client generated error code from ReRouteReq msg indicating no answer
REQUERY_ERROR (4)	CallRouter generated error code. The attempt to send the call to target failed because the target was not reachable (i.e., busy, ring no answer).
REQUERY_TIMED_OUT (5)	CallRouter internal use. Script ends. The attempt to send the call to target failed because the Routing Client did not respond within the DivertOnBusyCallTimeout period. Not visible to users.
REQUERY_ABORTED (6)	CallRouter internal use. Script ends. The attempt to send the call to target failed because the caller hung up or the call was lost. Not visible to users. In the case of ABANDON and DISCONNECT, the CallRouter assumes the call has ended and ends the script. The RequeryStatus value is set to 6, indicating REQUERY_ABORTED. This is used for the internal workings of the CallRouter but is not visible to users. Which Nodes Support Target

Which Nodes Support Target Requery

The following nodes support Target Requery:

-
- Label

- Queue

- Queue to Agent

- Route Select

- Select

Note: Scheduled Targets do not support Target Requery

Target Requery with IPCC and IP IVR

Target Requery not supported when IP IVR is used as the queue point in an IPCC system.

IP-IVR cannot requeue the call. The call is requeued from the CallManager instead. Setting the Target Requery option on the script node has no effect on the requeuing of the call, and may lead to timeouts and other script errors for that particular call.

Do not enable Target Requery in script nodes when using IP IVR.

Using Target Requery

You define nodes to enable Target Requery.

For the Queue, Queue to Agent, and Route Select nodes:

1. Open the node properties.

2. Click **Change**. A dialog box opens.

3. Check **Enable target requery**.

4. Click **OK** to close the dialog box.

5. Click **OK** to close the properties dialog box.

For the Label and Select nodes:

1. Open the node properties.

2. Check **Enable target requery**.

3. Click **OK** to close the properties dialog box

Checking Targets

About Call Tracer

You can use the Call Tracer utility from within Script Editor to test and debug a routing script and to confirm that ICM software is selecting targets as you expect. Call Tracer simulates a contact to the CallRouter and generates a text-based description of how the contact was handled. The test contact is processed by the active CallRouter, using all the real-time data of the ICM software as it exists at that moment, but interactions with any peripheral is simulated.

How to Check Targets

To check targets:

1. Start the Call Tracer by double-clicking the Call Tracer icon in the ICM Admin Workstation group. Script Editor opens, bringing up Call Tracer with it. To open Call Tracer if Script

Editor is already open, double-click the Call Tracer icon in either the ICM Admin Workstation group or within the Script Editor toolbar.

Figure 76: Call Tracer Icon

2. In the Call Tracer window, choose a Media Domain, a Routing Client, and a Dialed Number.

3. Optionally, enter an ANI (callers telephone number).

4. If you want to test a response from a IVR/VRU routing script, enter values in CED (caller-entered digits) and VRU Responses for External Script.

5. To use Network Transfer Call, A feature that integrates the ICM Post-Routing function with a carrier network's call control ability so that a call can be transferred anywhere in the network without the use of transfer/connect services or inter-site tie lines:

   – Check Use Network Transfer

   – Select a routing client and a dialed number value.

6. Click Send Call to submit the request. The results appear in the Call Trace Results field.

7. To send additional calls, optionally change any of the call parameters and then click Send Call again. The Call Trace Results field is updated.

8. The ICM software executes only installed scripts in response to requests from the Call Tracer. Any uninstalled changes in scripts are not reflected in the Call Tracer results.

Example Results

In the following example results from Call Tracer, ICM software invokes a script called sales. It begins with the Start node and then executes a Percent Allocation node that has three branches allocated for 42%, 16%, and 42%, respectively. The asterisk indicates that for this call, the ICM software chooses the first branch. This branch leads to a target that maps to the service Scranton.Sales. Had this been an actual call, the call would have been delivered to that service.

As you continue to send calls to the ICM software, the numbers for each node change to indicate how calls have been distributed. Each time a call arrives at a Percent Allocation node, the ICM software calculates the percentage of calls previously sent to each branch

```
> DialedNumber (8005551212) using CallType (General_Sales - 6)

Start #3

__Sales\Sales__General (version 2)

Percent allocation #54
```

```
__*42% = 0

__  16% = 0

__  42% = 0

Target #64

__Service: Scranton.Sales R

Route: Scranton.Sales

Peripheral target: DNIS: 1111, Trunk group: Scranton.Incoming800

Label: 2010000000
```

In the following example, the new call is counted in the denominator for each branch, but not in the numerator. Therefore, the numbers add up to less than 1.0 (but the total gradually approaches 1.0 as you send more calls). The ICM software then picks the branch that is most below its allocation value. In this example, the ICM software picked the third branch (because 0.375 is well below the allocation of 42% or 0.42) and would send the call to the Gary.Sales service.

Traces are linked to live scripts. For example, clicking on the trace details of a Percent Allocation node immediately, if it is not already open, opens the script and locates that node in the Script Editor window. Previously, you needed to manually decode script, script version and node ID information, and then visually scan the script for the desired node. Node titles in the trace window support localization.

The Call Tracer results refer to script nodes by their internal integer identifiers. To find which node maps to each identifier, open the script and choose Display Node IDs from the Script menu.

```
> DialedNumber (8005551212) using CallType (General_Sales - 6)

Start #3

__Sales\Sales__General (version 2)

Percent allocation #54

__42% = 0.416667

__16% = 0.166667

__*42% = 0.375

Target #58

__Service: Gary.Sales

Route: Gary.Sales
```

```
Peripheral target: DNIS: 1111, Trunk group: Gary.Incoming800

Label: 3010000000
```

How to Check VRU Scripts

Call Tracer allows you to simulate responses from VRU routing scripts. For each Script node the ICM software encounters, you can specify one of the following responses:

- Fail. The ICM software treats the External Script nodes as though it had failed.

- No Response. The ICM software assumes it received no response from the Run External Script node.

- Response. You can specify a CED value and values for Variable1 through Variable10. The ICM software assumes these values were returned by the External Script node.

To check VRU scripts:

1. In the Call Tracer dialog box, click **Add**. The VRU Response dialog box opens.

2. Select the type of response (Fail, No Response, Response) the Call Tracer is to simulate.

3. If you chose Response in Step 2, optionally specify values to be returned for CED and Variable1 through Variable10.

4. Click **OK** to apply changes and close the dialog box.

Note: You can also set the initial condition of the 10 call variables for the first run of the External Script.

Checking Targets

Chapter 5

Network VRUs

Through routing scripts, you can divert a call to a Network VRU for additional call processing.

This chapter contains the following topics:

What is a VRU

A VRU, or Voice Response Unit, is a telecommunications device, also called an Interactive Voice Response Unit (IVR), that plays recorded announcements and responds to caller-entered touch-tone digits. A VRU can also be equipped with Automatic Speech Recognition (ASR) or Text-to-Speech (TTS) capabilities.

What is a Network VRU?

A Network VRU supports ICM software's service control interface. An ICM routing script can divert a call to a Network VRU and instruct the VRU to perform specific processing before ICM software determines the final destination for the call.

Configuring VRUs

You configure Network VRUs through the ICM software Configuration Manager. Following are the general steps you must take to configure VRUs before you can write routing scripts. There are different Network VRU types. Each type represents a different architecture and call flow. These are summarized below and described in detail in the Cisco ICM Software Configuration Guide.

Before you can write routing scripts, you must do the following through the Configuration Manager:

1. Configure the Network VRU.

2. Associate the Network VRU with a Network VRU Label.

   This is step is required for Type 3, Type 5, and Type 7 VRUs.

3. Depending on the type of Network VRU configured (see the Cisco ICM Software Configuration Guide) various additional items need to be configured.

4. Configure Network VRU scripts.

   Note: You create and maintain VRU scripts on the VRU; however, you must define these scripts through the Configuration Manager so that ICM software is aware of these scripts so they can be called from ICM software routing scripts.

Network VRU Types and Script Nodes

Before a routing script can run a Network VRU script or queue a call at a Network VRU, it might need an explicit node to force the routing client to connect the call to the Network VRU. The node used to send the call to the Network VRU depends on the Network VRU type, as shown in the following table:

Table 3: Network VRU Types

Type	Description	Routing Script Node to Use to Send Call to VRU
2	A Network VRU at the customer premises. In a NAM environment, a Type 2 VRU is connected at the ICM and not at the NAM. : IP IVR in an IPCC solution.	Translation Route to VRU
3	A Network VRU connected to a NAM where there is a separate routing client that controls the connection to the VRU. : Use this type rather than Type 7 when the routing client can automatically take back the call from the VRU when ICM software returns a destination. : A Network VRU with a SS7 IN routing client.	Send to VRU
5	A Network VRU that is controlled by the routing client. : Use this type (rather than a Type 3 or Type 6) when the routing client has access to multiple VRUs.	Send to VRU
6	A Network VRU that receives the call and then sends a route request to ICM software.	Send to VRU
7	A Network VRU controlled by the routing client. Note: Use this type (instead of Type 3) when the routing client cannot take back the call from the VRU. That is, ICM software automatically instructs the VRU to release when it sends a route response to the routing client. : Cisco Internet Service Node using a Cisco IOS-based Voice Browser.	Send to VRU
8	Similar to Type 2, but a Type 8 VRU is used when the NAM has a routing client that controls the call to the VRU.	Translation Route to VRU
9	Queuing for System IPCC.	Send To VRU node, or any queuing node

Accessing VRU Scripts in ICM Scripts

After you have configured VRUs through the Configuration Manager, you can write routing scripts that send calls to the VRU, queue the call, and invoke specific VRU scripts.

Sending a Call to a VRU with the Send to VRU Node

You can send a call to a VRU for further processing by using the Send to VRU node (in the Queue tab of the Palette).

Figure 77: The Send to VRU Icon

The Send to VRU node is also valid for use with Type 3, Type 5, Type 6 and Type 7 VRUs.

When ICM software executes a Send to VRU Node, it looks up the call's Dialed Number, the Dialed Number's Customer and the Customer's Network VRU. If that fails to retrieve a Network VRU, the router uses the system default Network VRU:

- If the Network VRU is of Type 3 or Type 7, the router sends an instruction to the routing client to connect the call to the Network VRU. The instruction contains the label configured for this Network VRU as well as a correlationID to allow the router to uniquely identify the dialog with the routing client and the dialogue with the VRU PG. The router waits until it receives a RequestInstruction message from one of the connected VRU PGs with the same correlationID as a sign that the call is now connected to the VRU.

- If the Network VRU is of Type 5 the router sends an instruction to the routing client to connect the call to a VRU resource. The instruction contains the label configured for this Network VRU. The router waits for a response back from the routing client that the call is properly connected to the VRU resource.

- If the Network VRU is of Type 6, the router simply continues without further action, because Type 6 VRUs do not need explicit instructions to connect the call to the VRU resource.

There are two failure cases:

- If the label does not exist, script execution continues with control flowing through the nodes failure output terminal.

- If ICM software does not receive confirmation, execution continues with control flowing through the nodes failure output terminal.

In all other cases script execution continues with control flowing through the nodes success output terminal.

Notes:

- If the Run External Script, Play, Menu, Collect Data, or Queue node is used in a script before a Send To VRU node, an implicit Send To VRU node is assumed. For Network VRUs of types 3, 5, 6 or 7 you do not have to use the Send To VRU node. However, it is recommended that you make a practice of including the node in routing scripts, as it can act as a visual aid if you ever need to troubleshoot the script.

- If the call is delivered to the VRU but then abandoned, script execution ends. In monitor mode, a special label on the Send To VRU node accounts for these cases.

You do not need to set properties for the Send to VRU node. However, you can optionally add comments or connection labels.

Sending a Call to a VRU with the Translation Route to VRU Node

For Type 2 and Type 8 VRUs, you can send a call to a VRU by using the Translation Route to VRU node (in the Queue tab of the Palette).

Figure 78: The Translation Route to VRU Icon

When ICM uses the translation route to VRU method to route a call to a Type 2 SCI VRU, the VRU becomes the new routing client (dynamic routing client) for the call. This means that the next Connect message is sent to the VRU from the CallRouter to connect the call (to an agent, for example).

Note: In order for the dynamic routing client feature to work, the routing target (network VRU, device targets, services, etc.) need to have labels configured for the type 2 VRU as a routing client.

This feature is applicable for following call scenarios:

- Receiving a call on a converged enterprise network after a carrier pre-route using ISN (release 2.1 and later) to control the call.

- Supporting Outbound Option for Hosted IPCC where a shared Network VRU is used for providing announcements to targeted phones/answer machines.

- Using pre-routing for load balancing purposes before the call is sent to an ISN controlled network, with call context and cradle to grave reporting capabilities.

In an environment where the routing client is not capable of Network Transfer (such as GKTMP NIC) all the pre-route calls must be transferred to ISN via Translation Route To VRU to take advantage of network transfer feature.

Typical Call Flow Example:

1. A call is pre-routed by ICM using a carrier NIC (for example, AT&T) to a Type 2 premise VRU using the TranslationRouteToVRU mechanism. The VRU becomes the new routing client.

2. When the VRU is done with the call, the script connects the call to an agent on a TDM ACD (using translation routing again) and the VRU connects the call to the agent.

3. The agent answers the call and talks to the caller.

4. The agent initiates a transfer and the script and configuration execute a network transfer. The Connect message is sent to the Type 2 VRU, and the VRU connects the call to the new destination.

Following is the Properties dialog box for the Translation Route to VRU node:

Figure 79: Translation Route to VRU Icon Properties
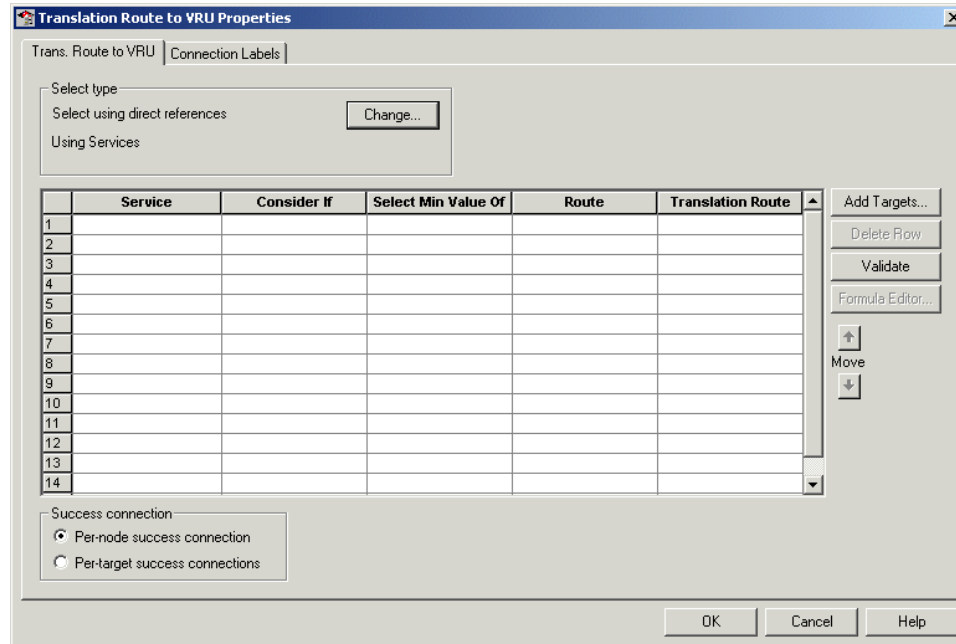
Define Translation Route to VRU node properties as follows:

1. To change the type of target:

   – Click Change. The Select Type dialog box opens.

   – Select the Target Type (Enterprise Service, Service, or Service Array).

   – If you selected Enterprise Service, select a **Business Entity** and **Enterprise target**.

   – Specify whether the Translation Route to VRU node is to act like a Select or Distribute node.

     Distribute Among Targets. The Translation Route to VRU node is to act like a Distribute node, distributing calls among the targets based on the relative values.

     Select Most Eligible Target. (Radio button.) The Translation Route to VRU node is to act like a Select node. If you select this option, you:

     Define whether to pick the target with the maximum value or the minimum value.

     Define a formula that determines which target is to be accepted.

     Define the type of target search.

2. To add targets, click Add Targets. The Add Targets dialog box opens. Use the Available Targets list and the Add button to select targets.

   Note: If you choose Enterprise Service as a target type, you can select just one item from the list. If you choose Service or Service Array, you can select one or more items from the list.

3. Click OK to close the Add Targets dialog box. The target members you selected appear in the Properties dialog box.

4. Continue defining Target information for each target:

   – **Consider If**. (Optional.) A formula that must evaluate to true for the target when ICM software executes the Translation Route to VRU node, or that target will not be considered.

   – **Select Max/Select Min Value of**. A formula that determines which of the targets is selected.

   – **Route**. (Drop-down list.) The route on which to send the call if this target is selected. (The list contains all routes associated with the target.)

   – **Translation Route**. (Drop-down list.) The route to send the call for initial VRU processing if this target is selected. (The list contains all translation routes associated with the same peripheral as the target.)

     Note: You must specify a value for this field. When a call is sent to a translation route, the PG retrieves the final route from ICM software and coordinates the other processing with the VRU.

   – **Per-node success connection**. (Radio button.) Select this option to attach one success output terminal to the node. This terminal is used regardless of which target is selected.

   – **Per-target success connection**. (Radio button.) Select this option to attach a success output terminal to each target in the node.

     Note: This option is useful in situations where you want to use different scripts depending on the selected target for a call.

5. Optionally, click Validate to validate the node properties.

6. Optionally, add connection labels.

Sending a Call to a VRU after a Translation Route to VRU

A SendToVRU node first checks for the Dialed Number. If the Dialed Number is not configured, the node immediately fails, interrupting the dialogue. If the Dialed Number is configured, the node checks for the Customer.

If the Customer is not defined, the node tries to send the call to Default Network VRU. If the Customer is defined, the node tries to find the Customer's Network VRU.

If the SendToVRU node finds the Customer's Network VRU , then it sends the call to Customer's Network VRU. If there is no Customer Network VRU, the node sends the call to Default Network VRU.

An explicit SendToVRU node only exits if the call is already at the correct VRU, as identified in the usual ways (for example, the dialed number's customer Network VRU), instead of exiting if the call is already at any VRU. If it's at a VRU, but it is the wrong VRU, the call is transferred to the correct VRU.

Example Call Flows:

1. A call is moved to a Type 2 VRU via a Translation Route to VRU. The call can then be moved to a Type 3 VRU via the SendToVRU node, if the Type 3 VRU is configured to be the network VRU for this call.

2. A call arrives from a Type 6 VRU (CallAtVRU = true). SendToVRU then attempts to send the call to the configured network VRU if the network VRU is different from the routing client's VRU.

3. SendToVRU fails the node if the transfer to the network VRU fails.

    – If the configuration is wrong or missing, (for example, if a network VRU is not define, or proper label is not defined) then the fail path is executed and the call remains at the VRU.

    – If the CallRouter succeeds in sending a Connect message, the CallRouter resets the previous VRU information and waits for RequestInstruction to set up the new VRU connection. If the operation times out, (RequestInstruction never arrives) then the fail path of the SendToVru node is taken and the call is assumed not to be at a VRU. The same behavior applies if CallRouter receives a ReRoute instead of a RequestInstruction command.

4. A registry flag enables/disables this feature. The registry key is called *NetworkVRUCheckEnabled* and is located at *...\Router\CurrentVersion\Configuration\Global*. The default value is 1 (enabled), setting it to 0 disables it.

Running External Scripts

You can instruct a Network VRU to run a specific script by using the Run External Script node (in the Queue tab of the Palette).

Figure 80: The Run External Script Icon

You can use multiple Run External Script nodes to execute a series of scripts on the VRU.

The Run External Script node is valid for use with all the VRU types.

Note: When Cisco Collaboration Server is integrated with ICM software, the Run External Script node can also be used to push a URL to the caller's Web browser. To do this, an entry in the Network VRU list must point to the URL map file on the Collaboration Server. For more information,, see the Cisco Collaboration Server Administration Guide and the Cisco ICM Software Configuration Guide.

The execution of ICM software routing script waits for the external script to finish:

- If the external script runs successfully, control then passes through the success branch of the Run External Script node.

- If the external script does not run successfully for any reason, then control passes through the failure branch of the Run External Script node.

Notes:

- If the current call is not at a VRU when the Run External Script node executes, ICM software sends the call to the associated Network VRU, as executing a Send to VRU node.

- Design scripts so that the Failure branch of a Run External Script Node contains a test for the Call.VRUStatus variable. If the value is 2, the VRU is likely to be not functioning properly. Therefore, the script avoids executing any subsequent Run External Script nodes on this Failure branch.

Note: When an uninterruptible script is used in a Run External Script node, the CallRouter waits for the script result from the VRU. It then executes the next node. Calls can only be routed when they reach an interruptible node. The Wait node and interruptible Run External Script node (micro apps) are interruptible. Every other node is uninterruptible.

Following is the Properties dialog box for the Run External Script node:
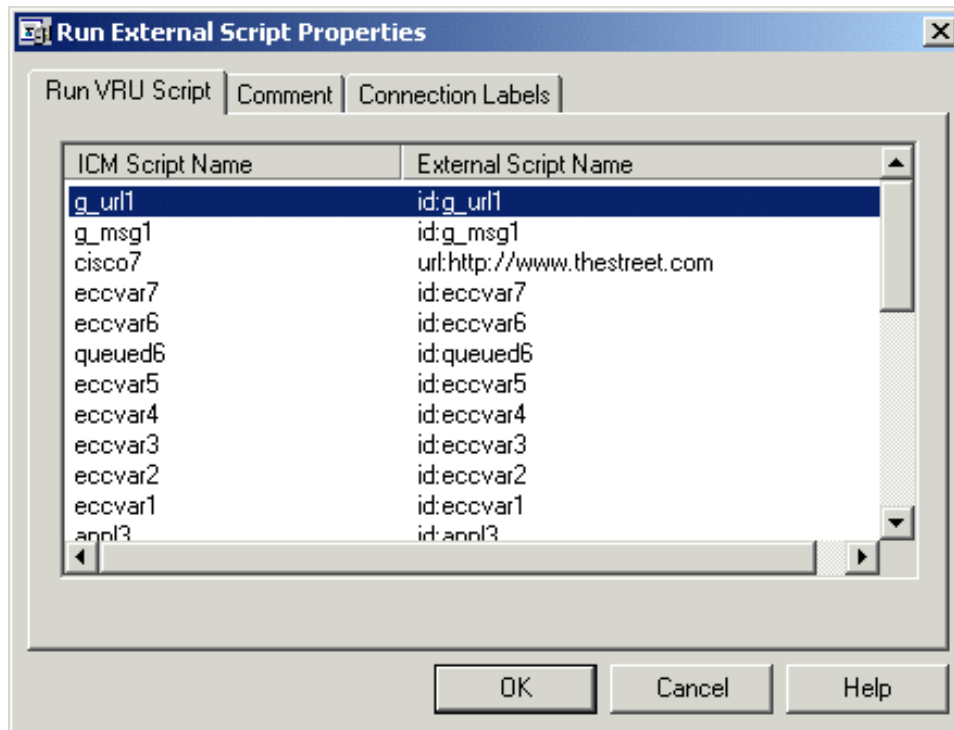
Define Run External Script node properties as follows:

1. Select the ICM Script/External Script Name you want to execute.

2. Optionally, add comments and connection labels.

Checking for VRU Errors

The following table lists the possible values for the VruStatus variable:

Table 4: VruStatus Variable Codes

Value	Meaning	Description
0	VRU_SUCCESS	The last VRU node was successful.
1	VRU_ERROR	The last VRU node failed because of a routing or configuration error
2	VRU_TIMEOUT	The last Send To VRU or Translation Route to VRU node failed because the routing client did not respond within 20 seconds or the last Run External Script node failed because the timeout limit defined for the script expired.
3	VRU_ABORTED	The last VRU node did not complete because the caller hung up or was otherwise lost. (Because this causes the routing script to terminate immediately, this value is never seen.)
4	VRU_DIALOG_ FAILED	The last VRU node failed because communication with the VRU ended unexpectedly

Value	Meaning	Description
5	VRU_SCRIPT_ NOT_FOUND	The VRU failed because the referenced VRU script was not found in ICM software configuration.

Queuing Calls at VRUs

You can queue a call at a Network VRU until a specific resource becomes available. A call can be queued for one or more skill groups, an enterprise skill group, or one or more scheduled targets. As soon as an agent becomes available at one of the specified targets, the call is removed from queue and sent to the target.

Specifically, you can:

- Place the call in a queue for one or more skill groups, an enterprise skill group, or one or more scheduled targets.

- Adjust the priority of call in a queue for one or more skill groups or scheduled targets.

- Remove the current call from any queues to which it has been assigned.

For example:

1. The call is first sent to the Network VRU. This step is required before you queue the call.

2. The call is queued for three skill groups.

3. If the call is successfully queued, the script cycles between a Wait node and a Run External Script node so that the caller hears an announcement every 30 seconds. If an agent in one of the skill groups becomes available, the call is removed from queue and taken back from the VRU. Routing script execution ends and the call is delivered to the target.

Note: In this scenario, you would likely make the VRU script interruptible so that the routing script can retrieve the call immediately when the resource becomes available.

Placing a Call in Queue

You can place a call in queue at a VRU for one or more skill groups, enterprise skill group, or one or more scheduled targets using the Queue node (in the Queue tab of the Palette).

Figure 82: The Queue Icon

If an agent becomes available in one of the skill groups or at one of the scheduled targets, the call is routed to that resource.

Note: If the current call is not at a VRU when the Queue node executes, ICM software sends the call to the associated Network VRU. (This does not apply to Type 2 or Type 8 VRUs, which are VRUs at customer premises.)

You cannot reference two types of targets (skill groups and scheduled targets, for example) within a single Queue node. However, you can execute multiple Queue nodes sequentially to queue a call to different target types.

The Queue node includes a Priority field, which sets the initial queuing priority for the calls processed through this node versus other calls queued for the same target. The Priority is expressed as an integer from 1 (top priority) to 10 (least priority). The default value is 5.

If more than one call is queued to a group when an agent becomes available, the queued call with the lowest priority number is routed to the target first. For example, assume an agent in a skill group becomes available and two calls are queued to that skill group. If one call has priority 3 and the other has priority 5, the call with priority 3, the lower value, is routed to the skill group while the other call continues to wait.

Note: The Queue node does not actually result in instructions being sent to the VRU. When queuing occurs the Queue node exits immediately through the success branch and the call is assumed to be at the VRU; the script should then continue with a Run External Script node to instruct the VRU what to do while holding the call until an agent to becomes available. Typically this would invoke a Network VRU Script that plays music-on-hold, possibly interrupted on a regular basis with an announcement.

Following is the Properties dialog box for the Queue node:

Figure 83: Queue to Skill Group Properties
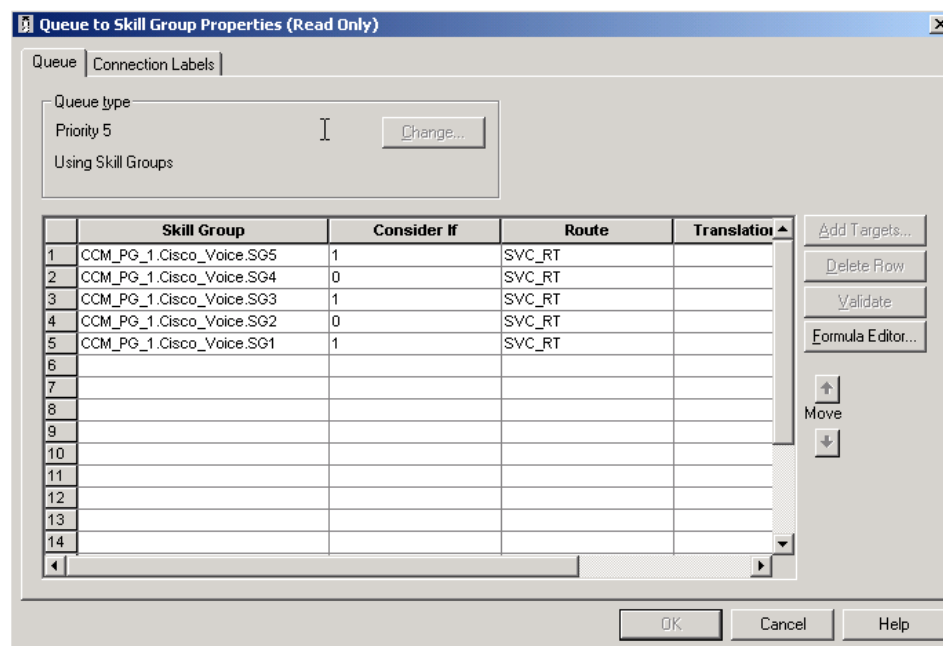
Define Queue node properties as follows:

1. To change the queue type:

- Click Change. The Queue Type dialog box opens.

- Select a Target Type (Enterprise Skill Group, Scheduled Target, or Skill Group). You cannot reference more than one type of target within a single Queue node. To queue a call to more than one target type, execute multiple Queue nodes sequentially.

- Optionally, select a Business Entity and Enterprise Target.

- Optionally, select a Priority to set the initial queuing priority for calls processed through this node versus other calls queued for the same target: 1 for top priority to 10 for least priority. (The default is 5.)

- Optionally, check Enable Target Requery. **Note**: When Target Requery is enabled in a Queue node and a Requery happens, for example because the call is presented to an available agent, but the agent does not answer, the script continues through the failure terminal. The script can then inspect the call variable RequeryStatus to determine what to do next. The typical action in case of a No Answer would be to Queue the call again, possibly to other skill groups, and possibly increase the priority so that it is taken out of the queue before regular queued calls.

- Click OK to close the Queue Type dialog box.

2. To add targets:

- click Add Targets. The Add Targets dialog box opens, listing available targets of the type you specified.

- Use the Available Targets list and the Add button to select targets.

- Click OK to close the Add Targets dialog box. The target members you selected appear in the Properties dialog box.

3. Optionally, continue defining Target Type information for each target member:

- Route. (Drop-down list.) The route to send the call to when an agent in the target type becomes available. (The drop-down list includes all routes associated with the target.)

- Translation Route. (Drop-down list.) The route to send the call for initial VRU processing if this target is selected. (The list contains all translation routes associated with the same peripheral as the target.)

- Scheduled Target. Individual targets to which the call is queued, if the Target Type is Scheduled Target.

4. Optionally, add connection labels.

Note: When processing a Queue node, the router first checks for an available target, if there is none available then the router attempts to queue the call. The call does not move to the VRU if there is an available agent.

Adjusting the Priority of a Call in a Queue

You can override the priority of a call in queue set by the Queue node by using the Queue Priority node (in the Queue tab of the Palette).

Figure 84: The Queue Priority Icon

For example:

1. The original priority of the call in queue is set by the Queue to Skill Group node.

2. The call waits in queue for 20 seconds while the caller listens to an announcement.

3. Call control passes to a second Wait node.

4. If 20 more seconds pass without an agent becoming available, the Queue Priority node executes and raises the call's priority in queue.

Notes:

- Only use the Queue Priority node after a Queue to Skill Group node. Any subsequent use of the Queue to Skill Group node results in setting the queue priority back to the original setting for that Queue to Skill Group node.

- The Queue Priority node sets the priority for a call within all queues that the call has been placed into. If a call only needs the priority raised in one queue only, a subsequent Queue to Skill Group node for that skill group/queue only (with the new priority) should be used.

- Queuing priorities should be handled very carefully. Just increasing Queue priority will not get a call handled sooner. The effect depends on the other call in the queue. For example, if all calls are treated using the example above, the priority increase will have no net effect. If the script above is only used for the Platinum customers while the Standard customers script leaves them at the default priority level, the effect is that all Platinum customer that have been in queue for more than 20 seconds will be handled first regardless of other customers in queue. This can be a dangerous practice, because while the delay for Platinum customers is greater than 20 seconds, no Standard customer will ever be handled. The solution is to increase the priority level for Standard customers as well, but only after they have been in queue for a longer period, for example 3 minutes.

Following is the Properties dialog box for the Queue Priority node:

Figure 85: Queue Priority Properties
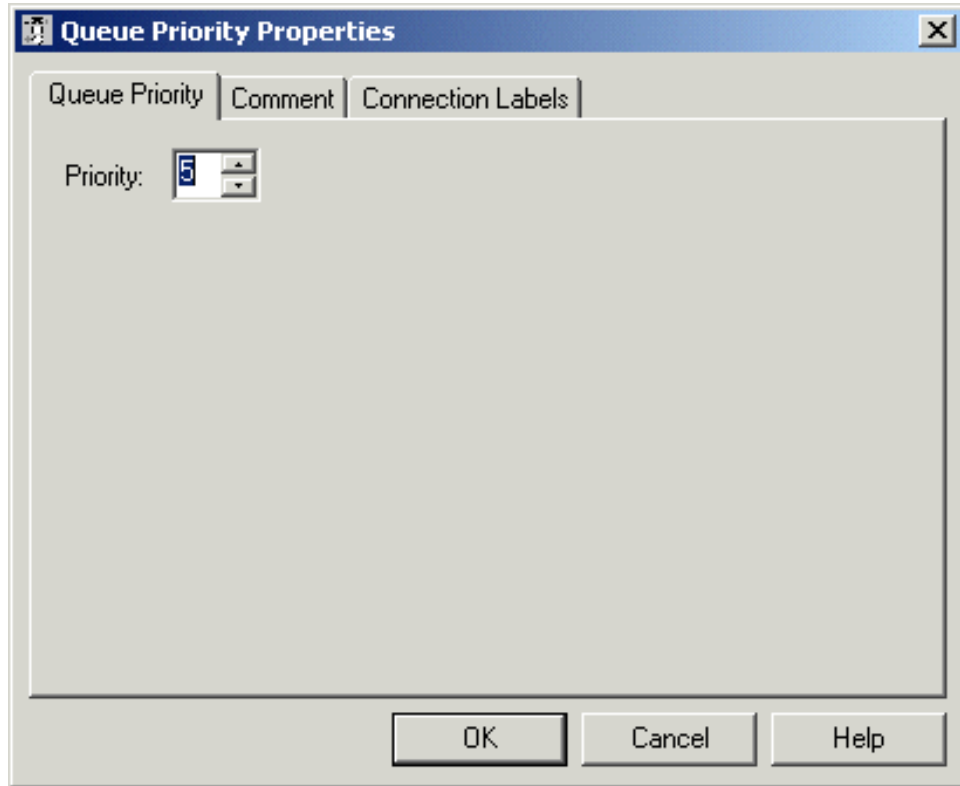
Define Queue Priority node properties as follows:

1. Specify the queuing priority for a call processed through this node: 1 for top priority to 10 for least priority. This determines the priority the call has versus other calls queued for the same target.

2. Optionally, add comments or connection labels.

Removing the Call from a Queue

You can remove a call from any queues by using the Cancel Queuing node (in the Queue tab of the Palette)

Figure 86: The Cancel Queuing Icon

. You do not have to define properties for the Cancel Queuing node. You can optionally add comments or connection labels.

VRU MicroApp Nodes

Note: These nodes can only be used if supported by your VRU.

The three VRU MicroApp nodes (Collect Data, Menu, Play) are essentially specialized Run External Script nodes. They allow all details of the interaction to be specified in the script node, rather then using Configuration Manager to specify the Network VRU Script.

The three MicroApp nodes each represent a specific sort of VRU interaction:

- Collect Data instructs the VRU to collect data from a caller after playing a prompt. The prompt can be played using a recorded announcement or using TTS. The data collected can be collected using touch-tone or using Automatic Speech Recognition (ASR).

- Menu is a simplified form of Collect Data that instructs the VRU to build a menu. It prompts the caller (either using a recorded announcement or TTS) and collects a single digit (either using touch tone or ASR). At the Menu node the script branches, depending on the input from the caller.

- Play instructs the VRU to speak a recorded announcement or a data element, such as a number or date. This can be done either by using recorded announcements or through Text-to-Speech (TTS) capabilities.

Collecting Data from the Caller

You can have a script play a prompt and instruct the caller to enter information by using the Collect Data node (in the Queue tab of the Palette). The caller-entered data can then be used to redirect the call to the appropriate destination.

Following is the Properties dialog box for the Collect Data node:

Figure 87: Collect Data Properties

Define Collect Data node properties as follows:

1. In the File Name field, enter the name of the media file to be played to the caller.

2. In the Library drop-down list, select the location of the file. You can select:

   – System

   – Application (default)

   – None

3. In the Protocols drop-down list, select the data transmission convention to use for the media file contents. You can select:

   – HTTP: (Hypertext Transfer Protocol, the default)

   – RSTP: (streaming)

   – file:

    –   other

4. Specify the number of digits a caller can enter:

   – In the Minimum number of digits field, select a number from 1 to 32 to indicate the minimum number of digits the caller must enter. The default is 1.

   – In the Maximum number of digits field, select a number from 1 to 32 to indicate the maximum number of digits the caller must enter. The default is 1.

   – In the Termination key field, enter the key that the caller presses to signify the end of digit entry. Value options are the digits 0-9, # (pound, the default) or * (asterisk). For variable-length data entry, only. A key which the user presses to signify the end of digit entry. Valid options: The digits 0-9, # (pound, the default) or * (asterisk).

   Note: If the minimum number value equals the maximum number value, this field is grayed out.

5. Optionally, check Use ASR to use automatic speech recognition. If you check this option:

   – Caller-entered information is obtained from spoken input as well at DTMF entry.

   – In the Automatic speech recognition grammar field, enter a grammar against which caller spoken-input is matched.

6. Optionally , check Allow barge-in to have any digit entry by the caller interrupt the media playback.

   The Cisco Internet Service Node (ISN) deals with barge-in as follows: If barge-in is not allowed (not checked), the Voice Browser continues to prompt play when a caller starts entering digits. If barge-in is allowed (checked), the Voice Browser discontinues prompt play when the caller starts entering digits.

7. Optionally, click Advanced Properties to specify how the micro-application should handle invalid or timed-out entries.

   The Advanced Properties override the VRU Default settings configured in the ICM software configuration database. The values you set apply only to the current node; other Collect Data nodes are not affected.

   Initial values for timeouts and number of tries on this dialog show the current values, obtained either from the database defaults or from a VRU Settings node.

   Invalid entry and No entry Media file names may be defined on the VRU device and invisible to the ICM environment. If this is the first micro-application operation in the script, the value for these fields is Default; it remains this value unless a VRU Settings micro-application changes it.

   – Optionally, clear ICM may interrupt. When checked (the default), the operation of the node can be interrupted by the ICM router.

- Optionally, clear Inter-digit timeout. When checked (the default), enter the number of seconds allowed between entering digits before the system assumes the caller is finished. Valid options: 1-99, default: 3.

- In both the Invalid Entry Media Properties and No Entry Media Properties sections, enter the file names of the files to play if the caller enters invalid data or if the caller enters no data.

   Note: To use the default settings on the VRU device, enter the word Default in these fields.

- Select the locations of the files in the Library drop-down lists. Valid options: System, Application (default), None.

   Note: If **None**, include the full path file name in the File name value; for example: http://www.xyzcorp.com/Media_Folder/File.wav.

- Select the data transmission convention to use in the Protocol drop-down lists. Valid options: HTTP: (Hypertext Transfer Protocol, the default), RSTP: (streaming), file:, other.

- Enter a number in the Number of tries field to indicate the number of times the Collect Data or Menu MicroApp will query the user for data when the user enters invalid data or does not enter data. Valid options: 1-9 (default: 3).

   Note: Optionally, you can clear the check box to have the default value defined in the database used.

- In the No Entry Media Properties section, enter a number in the Timeout field to indicate the number of seconds a caller is allowed to begin entering digits. If exceeded, the system times-out. Valid options: 1-99, default: 5.

   Note: If not checked, the default value defined in the database is used.

- When you have finished specifying advanced properties, click OK.

8. Optionally, add comments or connection labels.

Prompting a Caller to Select from a Set of Options

You can have a script play a prompt and instruct the caller to select from a list of options using the Menu node (in the Queue tab of the Palette).

Figure 88: The Menu Icon

The caller-entered data can then be used to redirect the call to the appropriate destination.

Following is the Properties dialog box for the Menu node:

Figure 89: Menu Properties

Note: You must define one or more success connections from the Menu node before you can define Menu node properties.

Define Menu node properties as follows:

1. In the File Name field, enter the name of the media file to be played to the caller.

2. In the Library drop-down list, select the location of the file. You can select:

    – System

    – Application (default)

    – None

3. In the Protocols drop-down list, select the data transmission convention to use for the media file contents. You can select:

- HTTP: (Hypertext Transfer Protocol, the default)

- RSTP: (streaming)

- file:

- other

4. Define menu choices. For each row beginning with the possible caller entered digit (which you cannot edit):

   - In the Case column, select a number from the drop-down list. The available numbers correspond to the success branches from the Menu node. The number indicates that if the caller enters the digit in that row, the call processing continues down that success branch.

   - Enter a description from that menu choice.

5. Optionally, check Use ASR to use automatic speech recognition. If you check this option:

   - Caller-entered information is obtained from spoken input as well at DTMF entry.

   - In the Automatic speech recognition grammar field, enter a grammar against which caller spoken-input is matched.

6. Optionally , check Allow barge-in to have any digit entry by the caller interrupt the media playback. Note: ISN deals with barge-in as follows: If barge-in is not allowed (not checked), the Voice Browser continues to prompt play when a caller starts entering digits. If barge-in is allowed (checked), the Voice Browser discontinues prompt play when the caller starts entering digits.

7. Optionally, click Advanced Properties to specify how the micro-application should handle invalid or timed-out entries.

   The Advanced Properties override the VRU Default settings configured in the ICM software configuration database. The values you set apply only to the current node; other Collect Data nodes are not affected.

   Initial values for timeouts and number of tries on this dialog show the current values, obtained either from the database defaults or from a VRU Settings node.

   Invalid entry and No entry Media file names may be defined on the VRU device and invisible to the ICM environment. If this is the first micro-application operation in the script, the value for these fields is Default; it remains this value unless a VRU Settings micro-application changes it.

   - Optionally, clear ICM may interrupt. When checked (the default), the operation of the node can be interrupted by the ICM router.

   - Optionally, clear Inter-digit timeout. When checked (the default), enter the number of seconds allowed between entering digits before the system assumes the caller is finished. Valid options: 1-99, default: 3.

- In both the Invalid Entry Media Properties and No Entry Media Properties sections, enter the file names of the files to play if the caller enters invalid data or if the caller enters no data.

  Note: To use the default settings on the VRU device, enter the word Default in these fields.

- Select the locations of the files in the Library drop-down lists. Valid options: System, Application (default), None.

  Note: If **None**, include the full path file name in the File name value; for example: http://www.xyzcorp.com/Media_Folder/File.wav.

- Select the data transmission convention to use in the Protocol drop-down lists. Valid options: HTTP: (Hypertext Transfer Protocol, the default), RSTP: (streaming), file:, other.

- Enter a number in the Number of tries field to indicate the number of times the Collect Data or Menu MicroApp will query the user for data when the user enters invalid data or does not enter data. Valid options: 1-9 (default: 3).

  Note: Optionally, you can clear the check box to have the default value defined in the database used.

- In the No Entry Media Properties section, enter a number in the Timeout field to indicate the number of seconds a caller is allowed to begin entering digits. If exceeded, the system times-out. Valid options: 1-99, default: 5.

  Note: If not checked, the default value defined in the database is used.

- When you have finished specifying advanced properties, click OK.

8. Optionally, add comments or connection labels.

Playing Specific Recordings to the Caller

You can instruct the VRU to play a series of media files and/or data to the caller by using the Play node (in the Queue tab of the Palette).

Figure 90: The Play Icon

Data can be a literal string or a formula that evaluates to a string. The data, its type, and the format it is to be played in is sent to VRU as part of the play request.

Note: The CallRouter will not verify the format setting to see if it is valid for the specific data type or micro-application. If the data format is invalid, the micro-application's result code will indicate such an error.

Following is the Properties dialog box for the Play node:

Figure 91: Play Properties
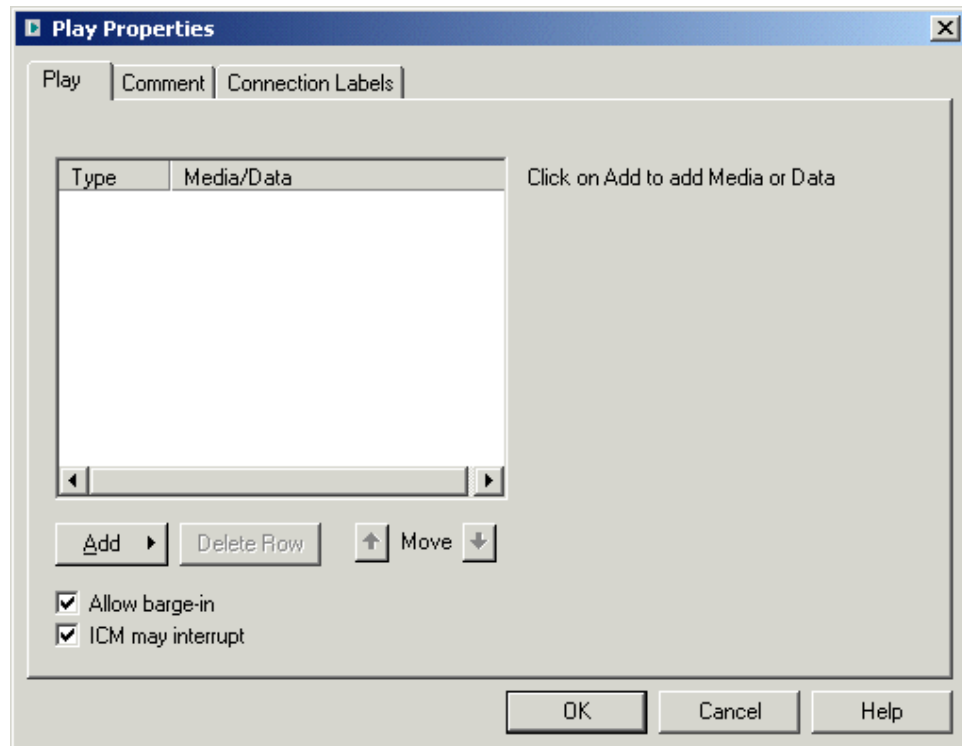
Define Play node properties as follows:

1. Click Add to add a new file or data element to play to the caller.

2. In the pop-up menu, select Media or Data.

3. If you selected Media in Step 3:

   – Enter the name of the file to play in the File name field.

   – In the Library drop-down list, select the location of the file. You can select:

   System

   Application (default)

   None

   – In the Protocols drop-down list, select the data transmission convention to use for the media file contents. You can select:

   HTTP: (Hypertext Transfer Protocol, the default)

RSTP: (streaming)

file:

other

4. If you selected Data in Step 3:

   – In the Data field, enter a string (or a formula that evaluates to a string) to be played by the VRU.

   – In the Data type drop-down list, select the type of data to be played: Type of data to be played by the VRU. Valid options are:

     Number - Numeric

     Char - Character

     Etime - Elapsed timeTOD - Time of day (12 hr)

     TOD - Time of day

     24TOD - Time of day (24 hr)

     DOW - Day of week

     Date - Entire date

     Currency - Money units

     Text - text

   – If in Step b above you selected Etime, TOD, or 24TOD, in the Time format drop-down list, select the time format. Valid options are:

     HHMM (default) - Hours and minutes

     HHMMSS - Hours, minutes, and seconds

     HHMMAP (TOD format, only) - Hours and minutes, A.M. or P.M.

     Note: If the minimum number value equals the maximum number value, this field is grayed out.

5. Optionally , clear Allow barge-in to not have any digit entry by the caller interrupt the media playback.

   ISN deals with barge-in as follows: If barge-in is not allowed (not checked), the Voice Browser continues to prompt play when a caller starts entering digits. If barge-in is allowed (checked), the Voice Browser discontinues prompt play when the caller starts entering digits.

6. Optionally, clear ICM may interrupt. When checked (the default), the operation of the node can be interrupted by the ICM router.

7. Optionally, add comments or connection labels.

Overriding VRU Settings

You can override a default VRU setting on a call-by-call basis by using the VRU Settings node (in the Queue tab of the Palette).

Figure 92: The VRU Settings Icon

You can only override one VRU setting with the VRU Settings node; you must use additional nodes to override additional settings.

Following is the Properties dialog box for the VRU Settings node:

Figure 93: VRU Settings Properties
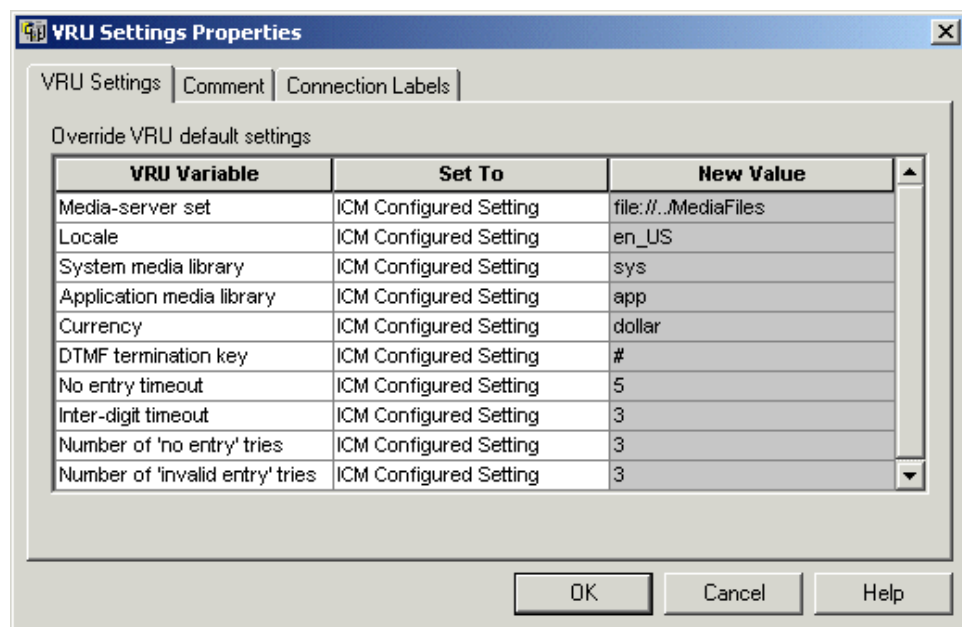
Define VRU Settings node properties as follows:

1. In the VRU Variable list, select the VRU variable to override.

2. Select an option in the Set To field:

   – Select ICM Configured Setting to have Script Editor, at runtime, set the variable to the value stored in the VRU_Defaults table.

– Select VRU Peripheral Setting to have Script Editor, at runtime, leave the variable's value as is.

Note: This value might be from the VRU_Defaults table, or consist of a value defined through a previous VRU Settings node.

– Select Customer Defined Value to have Script Editor, at runtime, set the variable to the value specified in the New Value field.

If you select this setting, the New Value field opens for editing. Specify a value in this field. (The variable type determines the type of data you can enter.)

3. Optionally, add comments or connection labels.

Temporarily Halting Script Execution

You can halt script execution for a specified number of seconds by using the Wait node (in the Queue tab of the Palette).

Figure 94: The Wait Icon

The Wait node simply stops script executing for the specified number of seconds. In the mean time, the Network VRU is waiting for instructions.

Warning: The protocol time-out variables in the VRU system need to be set to a value greater than the longest wait node used in the script.

Following is the Properties dialog box for the Wait node:

Figure 95: Wait Properties
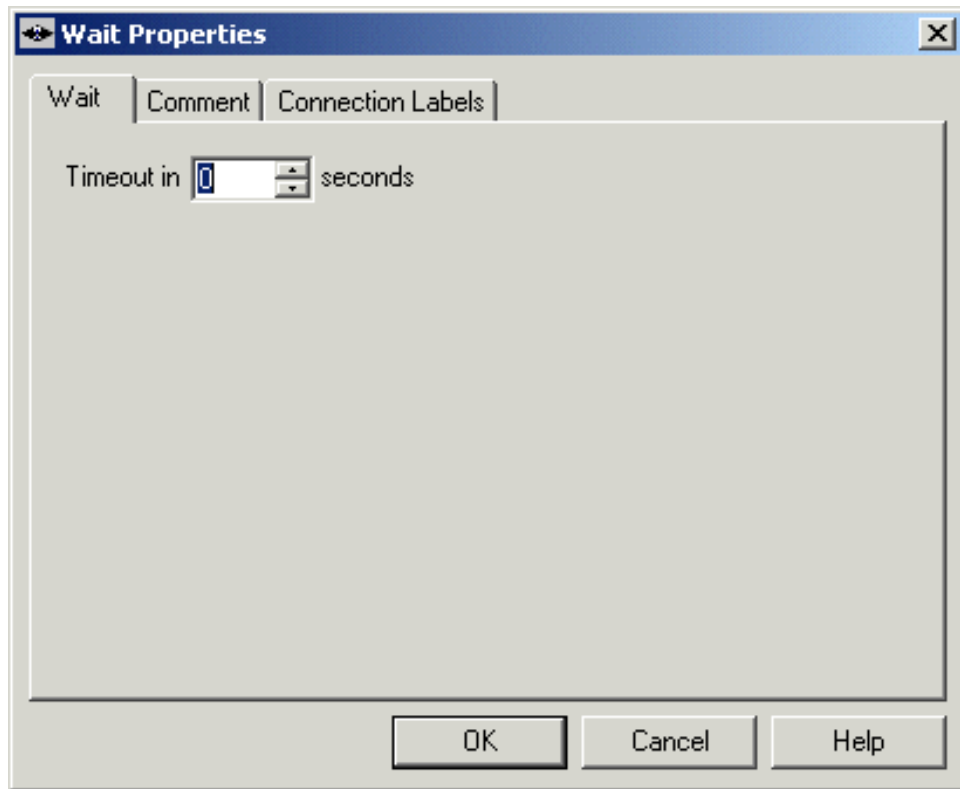
Define Wait node properties as follows:

1. In the Timeout field, specify an interval to wait, in seconds.

2. Optionally, add comments or connection labels.

Temporarily Halting Script Execution

Chapter 6

Multichannel Routing

Overview of Multichannel Services

When ICM software is integrated with Collaboration Server or Cisco E-Mail Manager, you write routing scripts to route contacts that are handled by these applications.

Supported Route Requests

ICM Software supports the following types of multichannel route requests:

- **Web callback** - A Web callback request is one that does not involve Web collaboration. A customer clicks a button on a Web site that says, "Call me back." Then the caller and agent simply talk on the phone.

- **Blended collaboration** With blended collaboration, the caller and agent talk on the phone and are linked in a collaborative Web session. They can share Web pages, forms, and applications, while at the same time conducting a voice conversation.

- **Text chat**The caller and agent can conduct a text chat session when a telephone call is not desired or not possible. They can both chat and collaborate on the Web.

- **E-mail message** The customer and agent communicate using electronic mail.

Routing Application Requests

The Collaboration Server and E-Mail Manager applications route requests to the ICM MR-PG. The Media Routing Peripheral Interface Manager (MR-PIM) on the MR-PG provides a generic interface to queue and route requests. The MR-PIM communicates with the ICM CallRouter, which runs a routing script to determine how best to handle the request.

ICM software uses a media class ID to identify the type of media or channel. A media class is a communication channel that is correlated to an application. There are five predefined media classes in ICM:

- **Cisco_Multi_Session_Chat** - Cisco_Single_Session_Chat

- **Cisco_Single_Session_Chat** - single-session chat requests

- **Cisco_Blended_Collaboration** - blended collaboration requests with IPCC

- **Cisco_Voice**- Web and delayed callbacks requests, blended collaboration requests with a legacy ACD, and basic ICM inbound and outbound voice calls

- **Cisco_Email**- e-mail requests

Each media class has at least one media routing domain (MRD), which is a collection of skill groups and services associated with a medium. ICM software uses the MRD to route a task to an agent who is associated with a skill group and a particular medium. Each MRD requires an ICM script, but it is possible to route requests from different MRDs using one script.

Synchronized Agents and Skill Groups

Agents are common across the multichannel software, but skill groups are application-specific. Agents can be created in the Collaboration Server or E-Mail Manager applications or in ICM software, and agents can be shared across applications. When agents or skill groups are created in either Collaboration Server or E-Mail Manager, they are simultaneously created in ICM software. If an agent is created in ICM software, the agent must be enabled in the Collaboration Server and E-Mail Manager applications if the agent wants to log into and work on those applications.

Skill groups are application-specific, and even though they are simultaneously created in ICM when they are created in Collaboration Server or E-Mail Manager, do not create, modify, or delete them in ICM software. You cannot enable skill groups in the applications.

Independent Media Queues

You can configure the multichannel software to route all media through independent queues defined by media class. Agents can be configured to log into only one media type to take either e-mail, text chat, blended collaboration, or voice. In this configuration, requests are queued only to agents who have logged into the corresponding media application.

You can use independent media queues in both IPCC and ACD environments.

Universal Queue

What is Universal Queue?

Universal Queue is the term used to describe ICM software's ability to route requests from different channels to agents who work with customer contacts in multiple media. With Universal Queue, ICM software treats requests from different channels as part of a single queue. Routing scripts can send requests to agents based on business rules regardless of the channel from which the request came. For example, ICM software can route phone, single session chat, and e-mail message requests to an agent who works with all these channels, based on the agent's skills and current tasks.

Universal Queue IPCC Requirement

Because Universal Queue must coordinate an agent's work on multiple tasks across different channels, Universal Queue requires that ICM software have complete control over how agents are assigned tasks Therefore, Universal Queue requires that ICM software be used in an IPCC environment.

Universal Queue is not supported in legacy ACD environments.

Universal Queue Configuration Overview

When you use IPCC, you can configure the multichannel software to manage a single universal work queue for all requests. You can configure agents to handle all media types, switching media on a task-by-task basis. For example, you would configure an agent as a member of three skill groups if the agent handles voice, e-mail, and chat, and the agent would log into the Softphone, E-Mail Manager, and Collaboration Server. The agent is assigned the longest waiting request from any of the three skill groups, or you can choose to prioritize the requests using the multichannel software scripting environment.

Multichannel Scripting and Media Routing Domains

What is a Media Routing Domain?

A Media Routing Domain (MRD) is a collection of skill groups and services associated with a specific communication channel. For example, E-Mail Manager uses the ICM software uses a MRD to route a task to an agent who is associated with a skill group and a particular channel. MRDs are defined in ICM configuration and have unique IDs across the enterprise.

Media Routing Domains and Interruptibility

When you configure MRDs, you indicate whether tasks for the MRD are interruptible. If the MRD is not interruptible, and agent working on tasks for that MRD is not assigned tasks from other MRDs. If the MRD is interruptible, the agent may be assigned tasks from another MRD.

Typically, voice calls and Web collaborations tasks are not interruptible, as agents are interacting with customers synchronously. E-mail messages, on the other hand, are typically interruptible because contact with the customer is asynchronous. Therefore, an agent responding to an e-mail message may be interrupted by a phone call or chat session.

How to Use Media Routing Domains to Categorize Contacts

You can categorize contacts based on the Media Routing Domain (MRD) based on the MRD making the route request by using the Media Routing Domain node (in the Routing tab of the Palette).

For example, you would have different MRDs for an E-Mail Manager instance and Web Collaboration single session chat. You could have a single script for both types of requests that branches so that it routes e-mail messages and single session chats to different targets.

You must insert targets and connections from the Media Routing Domain Node before you can define the node's properties.

Note: A branch can include multiple MRDs, but a single MRD can only be associated with one branch.

Following is the Properties dialog box for the Media Routing Domain node:

Figure 96: MRD Properties
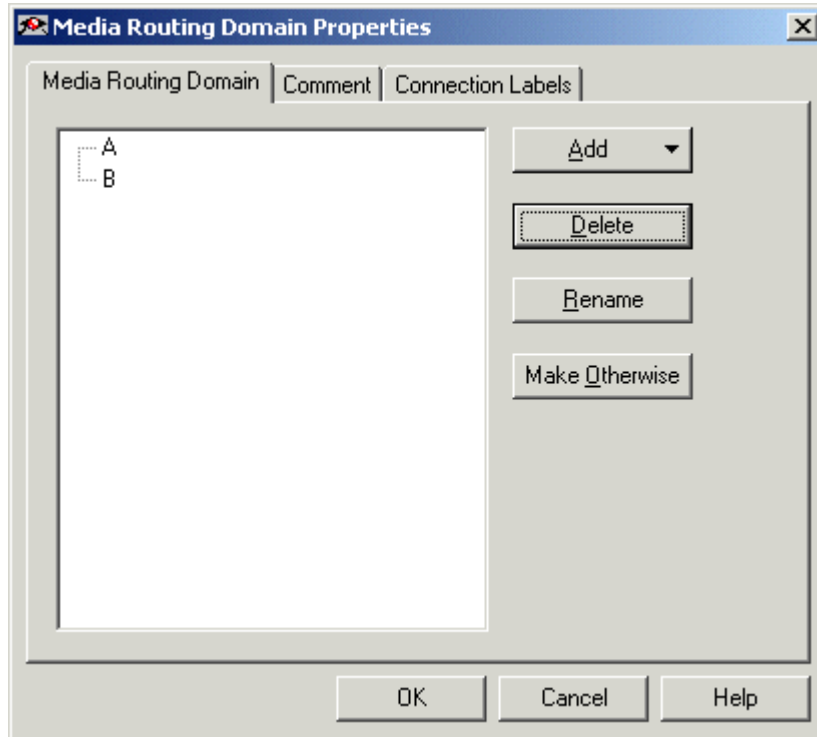
Define Media Routing Domain node properties as follows:

1. To associate a MRD with a branch:

   Select the branch.

   – Click **Add**.

   – Select a MRD from the drop-down list.

2. To delete a branch, select it and click **Delete**.

3. To rename a branch, select it, click **Rename**, and type the new name.

4. You can define a branch as Otherwise by selecting the branch and clicking **Make Otherwise**. Execution follows this branch if none of the specified time ranges apply. You can specify only one Otherwise branch for the node.

See Also

Cisco ICM Software Configuration Guide

Queuing to Agents

You can queue a contact directly to an agent by using the Queue to Agent node (in the Queue tab of the Palette).

You can change the Queue to Agent type to:

- Specify an agent directly

- Select an agent by expression

How to Change the Queue to Agent Type

Step 1   In the Queue to Agent properties dialog box, click Change. The Queue Agent Type dialog box opens:

Figure 97: Queue Agent Type
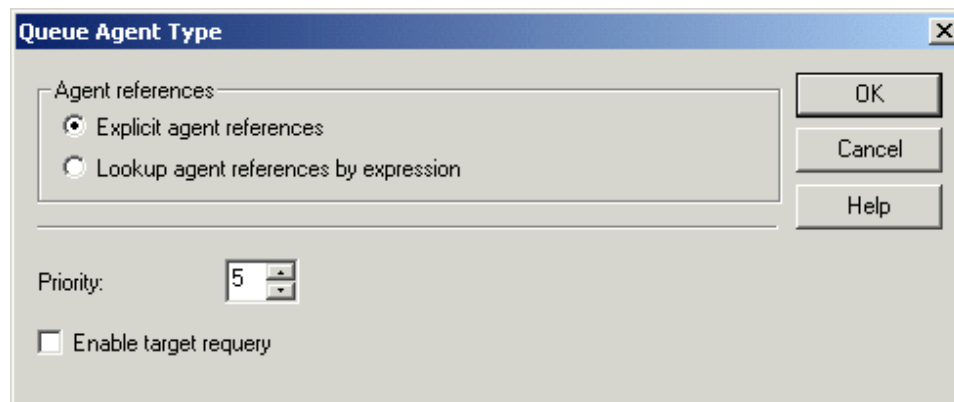
Step 2   To select a specific agent, check **Explicit agent references**.

Step 3   To select and agent by an expression, select **Lookup agent references by expression**.

Step 4   Select a **Priority** between 1 (the highest) and 10 (the lowest).

Step 5   Optionally, check **Enable target requery**.

How to Specify an Agent Directly

Following is the properties dialog box of the Queue to Agent node when you have selected to specify agents directly:

Figure 98: Agent Direct Properties
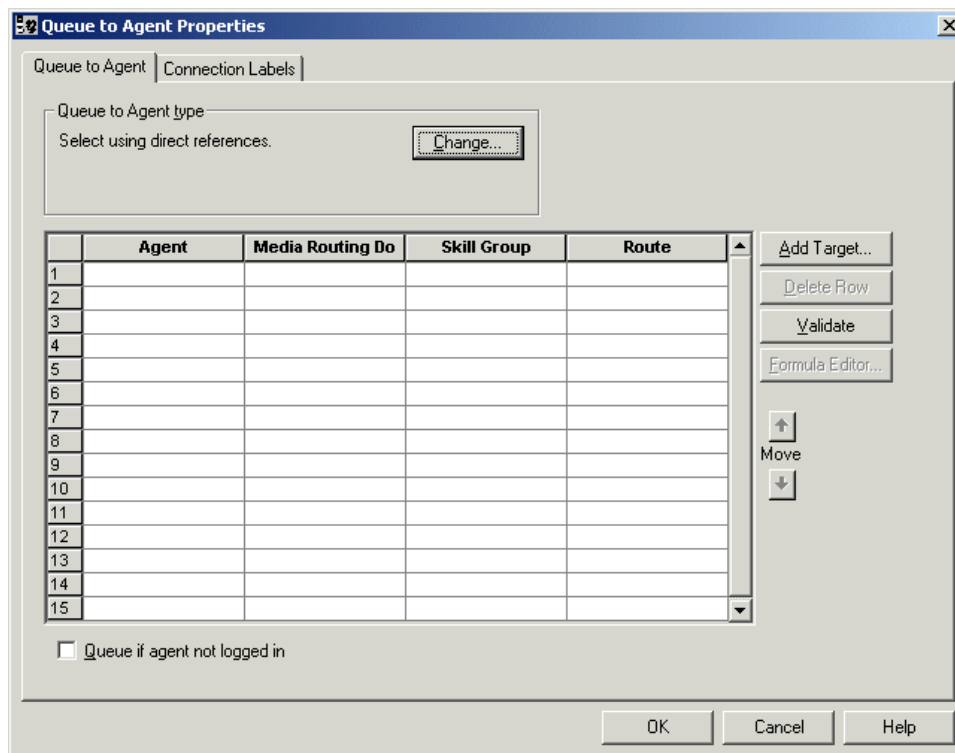
To specify agents directly:

Step 1   If necessary, change the Queue to Agent type to **Explicit agent references**.

Step 2   In the **Agent** column, select an agent.

Step 3   In the **Media Routing Domain** column, select the media routing domain for the selected agent.

Step 4   In the **Skill Group** column, select the skill group for the selected agent and media routing domain.

Step 5   In the **Route** column, select the route for the selected agent and media routing domain.

Step 6   Optionally, check **Queue if agent not logged in**, to have the contact queued to the agent even if the agent is not currently logged in.

Step 7   To test the data you entered, click **Validate**.

Step 8   Optionally, modify **connection labels**.

How to Select an Agent by an Expression

Following is the properties dialog box of the Queue to Agent node when you have selected to use an expression:

Figure 99: Agent Expression Properties
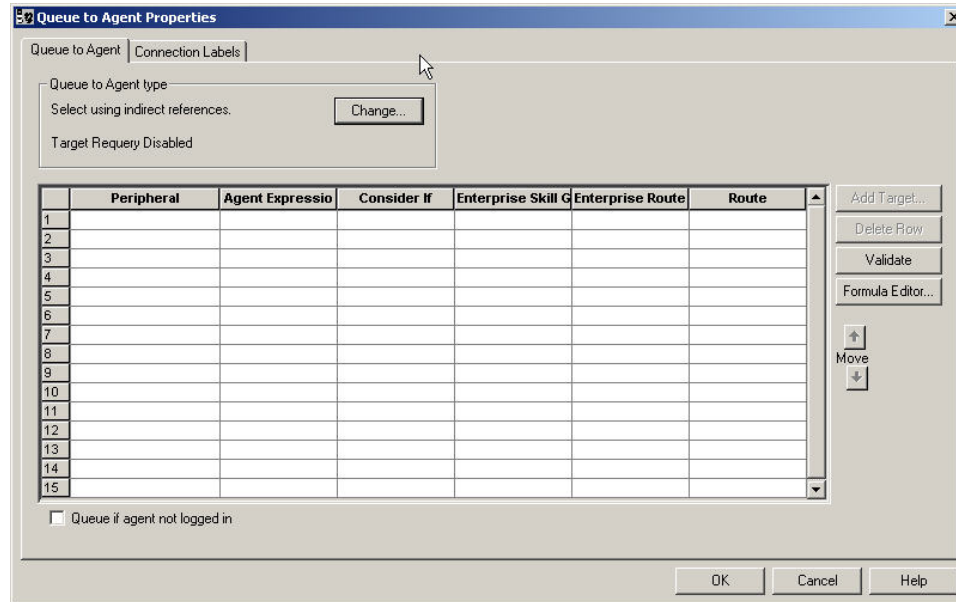
To specify agents by expression:

Step 1   If necessary, change the Queue to Agent type to **Lookup agent references by expression**.

Step 2   In the **Agent Expression** column, write an expression to select an agent. Click Formula Editor for help building a formula.

Step 3   Select the **Enterprise Skill Group** that includes the appropriate skill groups to cover all media routing domain cases for the selected Agent.

Step 4   Select the **Enterprise Route** that has an appropriate collection of routes, or the **Route**, matching the agent and media routing domain.

The **Enterprise Route** column lists collections of standard ICM and IPCC routes. The **Route** column lists single routes for System IPCC as there is only one PG.

If you do not select either an **Enterprise Route** or a **Route**, the following error message appears:

Figure 100: No Enterprise Route or Route Chosen

If you select both an **Enterprise Route** and a **Route**, the following error message appears:

Figure 101: Both an Enterprise Route and a Route Chosen

The specified **Enterprise Route** or **Route** is used to send the call to an agent.

Step 5   Optionally, check **Queue if agent not logged in**, to have the contact queued to the agent even if the agent is not currently logged in.

Step 6   To test the data you entered, click **Validate**.

Step 7   Optionally, modify connection labels.

Queuing to Agents

Chapter 7

Using Formulas

You can use formulas in many routing nodes to both categorize contacts (page 35) and select routing targets (page 57).

This chapter contains the following topics:

What is a Formula?

A formula consists of one or more expressions that ICM software evaluates to produce a value that it can use for subsequent script processing. You define expressions - made up of variables (page 134), constants, operators, and functions - as part of custom selection rules (page 73) or distribution criteria (page 78) in scripts.

Formula Example

Following is a simple example of a formula:

```
CallerEnteredDigits == 1
```

In this example:

- The Left-value, CallerEnteredDigits, is a variable . More specifically, it is a call control variable.

- The operator is the "Equal To" equality operator.

- The Right-value is the number 1.

If the value of CallerEnteredDigits is 1, the formula returns true; otherwise, the formula returns false.

Variables

What is a Variable?

A variable is a named object that holds a value. You use variables in formulas to select targets and help in call tracking.

Variable Syntax

Following is the syntax for using a variable in a formula:

Following is the syntax for using a variable in a formula:

object-type.object-name.variable-name

Where:

- The object-type is an object category, such as Service.

- The object-name is the name of an object contained in the ICM database, such as the name of a Service (for example, BosSales).

- The variable-name is the name of an object that can hold a value, such as a call control variable (for example, (CallerEnteredDigits).

- Each component in the variable is separated by a period (.).

Single-Target Variables

A single-target variable examines data for one specified routing target. For example, the variable:

Service.BosSales.ExpectedDelay

examines the expected delay for the BosSales service.

Multiple-Target Variables

A multiple-target variable examines data across multiple routing targets. For example, the function:

Max(SkillGroup.*.LongestAvailable)

finds the skill group, from all skill groups defined in the target set for the script node that calls the function, with the longest available agent.

You use an asterisk (*) as the object-name value to indicate that the variable is to examine data across multiple targets.

What are Call Control Variables?

Call control variables provide information about the current contact that is being routed by the script. Call control variables include information about where the route request came from, contact classification data, and data to be passed to the peripheral that receives the contact.

Variable	Data Type	Description	User Setable
CallerEnteredDigits	String	Digits caller entered in response to prompts.	Yes
CallingLineID	String	Billing telephone number of the caller.	No
CLIDRestricted	Integer	If 1, CLID presentation should be restricted. If 0, CLID presentation should not be restricted.	Set in Configuration Manager. Open Tools > Miscellaneous Tools > System Information. Check Enabled in the CLID Masking section of the screen to turn on.
CustomerProvidedDigits	String	Digits to be passed to the routing client for forwarding to the call recipient.	Yes
DialedNumberString	String	Telephone number dialed by the caller.	No
ExpCallVarName	String	Expanded Call Context (ECC) variable value assigned in scripts and passed with contact.	Yes
NetworkTransferEnabled	Integer	If 1, network transfer is enabled. If 0, network transfer is not enabled.	Yes
PeripheralVariable1-PeripheralVariable10	String	Values passed to and from the peripheral.	Yes
RequeryStatus	Integer	Provides the ability to test the error path of the Label, Queue, RouteSelect, and	No

Variable	Data Type	Description	User Setable
		Select nodes to determine the specific network cause of failure and conditionally retry the attempt as necessary.	
RouterCallDay	Integer	An encoded value that indicates the date on which ICM software processes the call.	No
RouterCallKey	Integer	A value that is unique among all calls ICM software has processed since midnight. RouterCallDay and RouterCallKey combine to form a unique call identifier.	No
RoutingClient	String	The name of the routing client that made the route request.	No
TimeInQueue	Integer	Number of seconds a call has been queued.	No
UserToUserInfo	String	ISDN private network User to User information	Yes
VruStatus	Integer	Indicates the result of a previous VRU node.	No

Note: For a Post-Routing® request from an Aspect ACD, PeripheralVariable1 through PeripheralVariable5 map to the Aspect variables A through E. The Aspect routing client passes these variables to ICM software as part of the request and ICM software returns them with the response. Other routing clients might use some of these variables for other purposes. The values of these variables are also stored in the Route_Call_Detail table of the ICM database.

What are Expanded Call Context (ECC) Variables?

Expanded Call Context (ECC) variables store values associated with the contact.

ECC values are written to Termination Call Detail records only if, and when, an ECC value is explicitly set (which can be done any numbers of ways, such as using a script, an IVR, a NIC, CTI, and so on). This applies to null values as well as non-null values.

If an ECC variable has been defined, but never assigned a value, it will not have a row in the Termination Call Variable table when a Termination Call Detail record is written.

Generally speaking, ECC values are passed from leg to leg on the call. Once a value has been assigned, the value is recorded in the Termination Call Variable for every Termination Call Detail Segment. Note, however, that this depends on how each new call segment is created. If it does not involve translation routes or IPCC, and is outside the original peripheral, ECC variables, like all call variables, cannot be propagated.

ICM software is delivered with some ECC variables, and you can create others through Configuration Manager. For more information, see the **ICM Configuration Guide for Cisco ICM Enterprise Edition, Release 7.0(0)** (http://www.cisco.com/en/US/products/sw/custcosw/ps1001/tsd_products_support_series_home.html).

Persistent vs Non-Persistent Call Variables

When IPCC/ICM writes call data records to its historical database, it can store the values of all call variables. Storing excessive call variable data can degrade historical database performance. When defining a call variable (in ICM Config Manager or System IPCC Web Admin), you have the option of tagging it as either *persistent* or *non-persistent*. Only persistent call variables are written to the historical database. Non-persistent variables can be used in routing scripts, but will not be written to the database.

Expanded Call Context Variables for E-Mail Manager Messages

Each E-Mail Manager message routed through ICM software has three associated ECC variables:

- cisco.cem.Priority - The priority of the message. The value of the Priority variable, which is "0" through "3"

    - "0" for Normal

    - "1" for High

    - "2" for Very High

    - "3" for Urgent

- **Note:** The priority if set through the E-Mail Manager rules. The value of the variable can be used to categorize the contact in a script.

- cisco.cem.Category - The categories of the message. The categories variable is an array containing up to 10 category values. Category values are configured by the E-Mail Manager administrator. The value of the variable can be used to categorize the contact in a script.

- cisco.cem.MessageKey - The unique identifier of the message that E-Mail Manager is requesting ICM software to route. While typically the Message Key would not be used to categorize a contact in a script, it may be useful when ICM software is integrated with a CRM application; the Message Key could be recorded in the CRM database for future reference to e-mail correspondence with a customer.

The ECC variables for E-Mail Manager messages are created by default during ICM software installation.

For more information on routing e-mail messages, see the *Cisco E-Mail Manager Administration Guide*.

Expanded Call Context Variables for Web Callback

You must create two ECC variables if you intend to use Blended Collaboration and/or Voice Media Routing Domains to route Delayed Callback requests. The ECC variables are:

- user.cisco.cmb - Specified by the peripheral.namedvars property in Media Blender, enabling Web callback requests to be routed through ICM software.

- user.cisco.cmb.callclass - Specified by the peripheral.namedvars property in Media Blender, used to send the call class from Collaboration Server to Media Blender by way of the pre-call message.

For more information on Web Callback, see the *Cisco Media Blender Administration Guide*.

For more information on configuring ECC variables, see the *Cisco ICM Software Configuration Guide*.

What are User Variables?

User variables are variables you create to serve as temporary storage for values you can test with an **If Node**. For example, you could create a user variable called usertemp to serve as a temporary storage area for a string value used by an If node.

You create user variables through the ICM Configuration Manager. For more information, see the *Cisco ICM Software Configuration Guide*.

Each user variable must:

- Have a name that begins with user.

- Be associated with an object type, for example, Service. (This enables ICM software to maintain an instance of that variable for each object of that type in the system.)

- Be identified as a persistent (retains value across CallRouter restarts) or non-persistent (does not retain value across CallRouter restarts) variable.

- A user variable can store a value up to 40 characters long.

After you have defined a variable, you can then use the Formula Editor to access the variable and reference it in expressions, just as you would with a built-in variable.

How to Set Variable Values with the Set Variable Node

Figure 102: Set Properties window
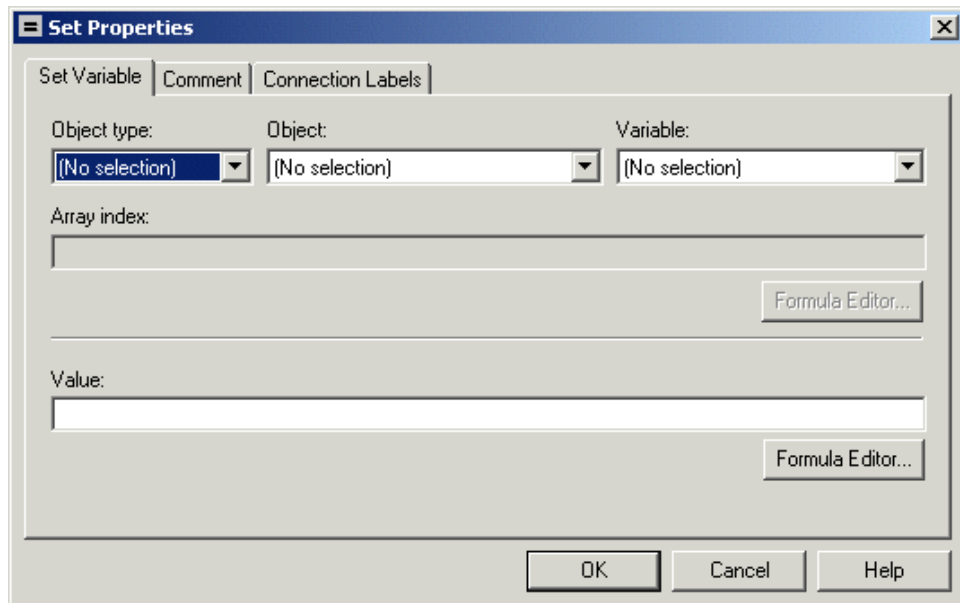
You can set the value of a variable with the Set Variable node:

- Object type - Select the type of object the variable is associated with.

- Object - Select the specific object the variable is associated with.

- **Note:** If you choose Call as the Object Type, this field does not apply.

- Variable - The specific variable you want to set.

- **Note:**

  - The variables that are available are determined by the value you choose in the Object Type field.

  - Define all integer fields in tables accessed by a Set Variables node as NOT NULL.

- Array index - Enter an integer or an expression that evaluates to an integer. For example, if the Array Index expression evaluates to 2, then the Set Variable node sets the second element of the variable array.

- **Note:** This field is only available if you select an array variable in the Variable field.

- Value - Enter the value to assign to the variable. The value can be:

  – A constant

  – A reference to another variable

– An expression

SkillGroup.Avail And SkillGroup.ICMAvailable Variables

When the ICM system includes only the voice channel, the value of the SkillGroup.Avail variable is the number of agents in the available state, meaning that the agents are able to accept new calls.

However, when the Web or E-mail channel is used with non-voice Media Routing Domains and agents log in to multiple domains, the value of the SkillGroup.Avail variable is calculated differently. There is also a SkillGroup.ICMAvail variable.

The following table describes the difference between the SkillGroup.Avail and the SkillGroup.ICMAvail variables:

Case	SkillGroup.Avail	SkillGroup.ICMAvailable
Only voice domain is used	Number of agents in the Available state.	Same
Multiple Domains are used	Number of agents in the Available state, regardless of what they may be doing in this or other domains.	Number of agents who can actually handle an additional task or call in the domain.

SkillGroup.ICMAvailable Variable

The value of the SkillGroup.ICMAvailable variable is the actual number of agents logged into the skill group who who can take new calls or tasks. Such agents must meet all the following criteria:

1. They are routable in the domain.

2. The agent's state in the domain is something other than "Not-Ready".

3. The agent is below the maximum task limit.

4. **Note:** For most domains (that is, if the agent is not a Collaboration Server Multi-session agent), the maximum task limit is 1, and an agent is below the maximum only when he is not working on any call or task.

5. The agent is not working on another task in a non-interruptible domain.

SkillGroup.Avail Variable

SkillGroup.Avail is the number of agents in the skill group who are not doing anything in the domain. An agent who is logged into two domains can be counted as Avail in one domain even though that agent is handling a task in another non-interruptible domain. An agent in a domain that handles multiple tasks (such as multi-session chat) is not counted as Avail if that agent is handling a task, even though the agent has additional capacity for more tasks.

The following table shows some possible values for these variables. Assume three agents are logged into a voice skill group, and the same three agents are also logged into another non-interruptible domain, such as Collaboration Server Single-session Chat. This table shows the voice skill group states.

Case	SkillGroup.Avail	SkillGroup.ICMAvailable
Initial state	3	3
First agent handles a call	2	2
Second agent handles a Collaboration chat session	2 (because there are two agents doing nothing in the domain)	1 (because there is really only one agent left to handle voice calls)
Voice call ends	3	2
Collaboration chat ends	3	3

If a routing script needs to check the number of available agents, the correct variable to use is SkillGroup.ICMAvailable.

SkillGroup.Avail>0 does not guarantee that there are really available agents, as those apparently available agents may be handling tasks in another domain.

Following is another example showing agents handling multi-session chat. Assume three agents logged into a multi-session chat skill group, each capable of handling two chats. This table shows states for the multi-session chat group.

Case	SkillGroup.Avail	SkillGroup.TalkingIn	SkillGroup.ICMAvailable
Initial state	3	0	3
First agent handles a Collaboration chat session	2 (because the agent is now in the talking state)	1	3 (because all three agents can still handle additional chats)
Second agent handles a Collaboration chat session	1	2	3
Third agent handles a Collaboration chat session	0	3	3
First agent handles second Collaboration chat session	0	3 (even though a total of 4 chats are in progress, only 3 agents are doing the work)	2 (because only the second and third agents can handle an additional chat)

By default, Script Editor shows the ICMAvailable value instead of Avail value when displaying skill group real-time data.

Operators

Prefix Operators

The Prefix Operators in the following table take a single operand:

Operator	Meaning	Comments/Examples
+	Positive	Numeric values are positive by default, so the positive operator (+) is optional. Example: 2 and +2 represent the same value.
-	Negative	The negative operator (-) changes the sign of a value. Example: 2 represents a positive value; -2 represents a negative value.
!	Logical negation	A logical expression is any expression that evaluates to true or false. The logical negation operator (!) changes the value of a logical expression. Note: Numerically, a false value equates to 0 and a true value equates to a non-zero value. Example: If the current value of SkillGroup.Sales.Avail is 3, then SkillGroup.Sales.Avail > 0 is true and (SkillGroup.Sales.Avail > 0) is false.
~	One's complement	Operates on a bit value, changing each 1 bit to 0 and each 0 bit to 1. Note: This operator is rarely used.

Arithmetic Operators

The Arithmetic Operators in the following table take two operands:

Operator	Meaning	Comments/Examples
*	Multiplication	Arithmetic operators perform the basic operations of addition, subtraction, multiplication and division. You can use them in making calculations for a skill group, service, or route. Note: Multiplication (*) and division (/) operators are evaluated before addition (+) and subtraction (-) operators. Examples: returns the number of agents who are logged on to the service but not currently available. evaluates to because the multiplication is performed first.
/	Division	
+	Addition	
-	Subtraction	

Equality Operators

The Equality Operators in the following table take two operands:

Operator	Meaning	Comments/Examples
==	Equal to	Equality operators allow you to determine whether two values are equivalent or not. Examples: is true if any calls are currently queued for the service. is true if all agents logged on to the service are currently available.
!=	Not Equal To	

Relational Operators

The Relational Operators in the following table take two operands:

Operator	Meaning	Comments/Examples
>	Greater than	Relational operators allow you to perform a more sophisticated comparison than the equality operators. : is true if more members of the skill group are in the Not Ready state. is true if at least as many agents are Ready as Not Ready.
<	Less than	
>=	Greater Than or Equal To	
<=	Less Than or Equal To	

Logical Operators

The Logical Operators in the following table take two operands. Logical operators examine the values of different logical expressions:

Operator	Meaning	Comments/Examples
&&	And	The expression is true if both of the operands are true. If either is false, the overall expression is false. :The following is true if the skill group has at least one agent logged on and no agents are currently available:
||	Or	The expression is true if either or both of the operands is true. If both are false, the overall expression is false. :The following is true if the skill group has at least 5 agents logged on or no agents are currently available: However, the expression is false if less than five agents are logged on and there is an agent available.

Note: The equality (==) and relational (>) operators are evaluated before the logical operators (&& and ||).

Bitwise Operators

The Bitwise Operators in the following table take two operands.

Operator	Meaning	Comments/Examples
&	And	The & Bitwise Operator turns specific bits in a value on or off. :The The following expression turns off the six low-order bits of AvgTalkTimeTo5: Note the use of the complement operator (~) with the constant. This is equivalent to rounding the value down to the next multiple of 64.
|	Inclusive Or	Inclusive Or and Exclusive Or differ in the way they handle the case where bits in both values are 1: Inclusive Or evaluates the result as true and sets a 1 bit in the result. Exclusive Or evaluates the result as false and sets a 0 bit in the result. (An Exclusive Or applies the rule "one or the other, but not both.").
^	Exclusive Or	

Miscellaneous Operators

The following table lists miscellaneous operators:

Operator	Meaning	Comments/Examples
?	Conditional	The conditional operator (?) takes three operands. Its syntax is as follows: ICM software evaluates the expression by first examining the logical expression condition and then doing the following: If the result is true, then the overall expression evaluates to the value of the expression true-result. If the result is false, then the overall expression evaluates to the expression false-result. The following expression determines whether the number of agents available for skill group S1 is even or odd:
&	Concatenation	The concatenation operator (&) joins two strings end-to-end. returns the value
,	Sequential	The sequential or comma operator (,) takes two operands, each of which is an expression. ICM software evaluates the left expression first and then the right expression. The value of the overall expression is the value of the right expression. The first expression typically affects the valuation of the second.
<<	Shift left	The shift left (<<) and shift right (>>) operators shift the bits within a value. The following example shifts the bits in Avail two places to the left. The two right-most positions are filled with zeroes: The following example shifts the bits in Avail two places to the right. In this case, the positions are filled with sign bits (0 if the original value is positive or zero; 1 if the original value is negative).

Operator Precedence

The following table shows the order in which operators are evaluated.

Note: The operators with priority 1 are evaluated first, then those with priority 2, and so on. The order of evaluation within each priority level can also be important. Prefix operators are evaluated from right-to-left in an expression. Assignment operators are also evaluated from right-to-left. In all other cases where operators have equal priority, they are evaluated left-to-right.

Priority	Operator type	Operators
1	Prefix (unary)	+ - ! ~
2	Multiplication and division	* /
3	Addition and subtraction	+ -
4	Shift right and shift left	>> <<
5	Relational	< > <= >=
6	Equality	== !=
7	Bitwise And	&
8	Bitwise exclusive Or	^
9	Bitwise inclusive Or		
10	And	&&
11	Or	||
12	Conditional	?
13	Sequential	,

Built-in Functions

Date and Time Functions

The following table lists the built-in date and time functions:

Function	Data Type	Return Value/Example
date [(date)]	Integer	Returns the current system date or the date portion of a given date-time value. The given date can be a floating point value (as returned by the now function), a string of the form mm/dd/yy, or three integers: yyyy, mm, dd. date (with no arguments) returns the current date. = date(2001, 7, 15) tests whether the current date is July 15, 2001. **Note:** Do not use the slash (/) character in defining a date function. Because it is the division operator, the function would not return the results you are looking for. You could enclose the argument within a string----but the preferred format is: .
day [(date)]	Integer	Returns the day of month (1-31) for the current date or a given date. The given date must be an integer or a floating-point value, as returned by the date or now function. tests whether tomorrow is the first of the month.
hour [(time)]	Integer	Returns the hour (0-23) of the current time or a given time. The given time must be a floating-point value, as returned by the now function. tests whether the current time is before noon.
minute [(time)]	Integer	Returns the minutes (0-59) of the current time or a given time. The given time must be a floating-point value as returned by the time function. tests whether the current time is in the second fifteen-minute interval after an hour.
month [(date)]	Integer	Returns the month (1-12) of the current month or a given date. The given date must be a floating-point value, as returned by the date or now function. tests whether the current month is June.
now	Float	Returns the current date and time, with the date represented as an integer and the time represented as a fraction. Note: You can use the date or time functions without any arguments to return just the current date or time. This function is useful for comparing the current date and time to a specific point in time. to test whether the current date and time is later than 10 P.M., December 24, 2001, use the expression .
second [(time)]	Integer	Returns the seconds (0-59) of the current time or a given time. The given time must be a floating-point value, as returned by the time function. tests whether the current time is within the last ten seconds of a minute.
time [(time)]	Float	Returns the current system time or the time portion of a date-time value. The given time can be a floating point value, a string of the form hh:mm:ss, or two or three numeric values: hh, mm [, ss]. (with no arguments) returns the current time. tests whether the current time is after 2:00 PM.
weekday [(date)]	Integer	Returns the current day of week (Sunday=1, Monday=2, etc.) of the current date or given date. The given date must be an integer or floating-point value, as returned by the date or now function. tests whether today is Tuesday.

Function	Data Type	Return Value/Example
year [(date)]	And	Returns the year of the current year or given date. The given date must be a floating-point value, as returned by the date or now function. tests whether the millennium has passed.

Mathematical Functions

The following table lists the built-in mathematical functions:

Function	Data Type	Return Value/Example
abs(n)	Floating Point or Integer	Returns the absolute value of (the number with no sign). returns the value 15.
max(n1, n2 [,n3] . . .)	Floating Point or Integer	Returns the largest of the operands. Each operand must be numeric. returns the value 3.
min(n1, n2 [,n3] . . .)	Integer	Returns the smallest of the operands. Each operand must be numeric. returns the value -2.
mod(n1,n2)	Floating Point or Integer	Returns the integer remainder of n1 divided by n2. returns the value 99.
random()	Floating Point or Integer	Returns a random value between 0 and 1.
sqrt(n)	Floating Point or Integer	Returns the square root of . (The operand n must be numeric and non-negative.) returns the value 7.
trunc(n)	Floating Point or Integer	Returns the value of truncated to an integer. returns the value 28.

Miscellaneous Functions

The following table lists the built-in miscellaneous functions:

Function	Data Type	Return Value/Example
after(string1,string2)	String	That portion of string2 following the first occurrence of string1. If string1 does not occur in string2, the null string is returned. If string1 is the null string, string2 is returned. returns the value defg.
before(string1,string2)	String	That portion of string2 that precedes the first occurrence of string1. If string1 does not occur in string2, string2 is returned. If string1 is the null string, the null string is returned. returns the value abc.
ClidInRegion	Logical	Indicates whether the CLID for the current contact is in the geographical region specified by string. The value string must be the name of a defined region. You can use the Name variable of a region to avoid entering a literal value. tests whether the CLID is from the Maryland region.
concatenate(string1,string2, . . .)	String	Returns the concatenation of the arguments. The function takes up to eight arguments. returns the value abcde.

Function	Data Type	Return Value/Example
find(string1, string2 [,index])	Integer	Returns the starting location of string1 within string2. If you specify an index value, searching starts with the specified character of string2. returns the value 6.
if(condition,true-value,false-value)	Logical	Returns a value of true-value if the condition is true; false-value if the condition is false. returns the current hour in 12-hour format rather than 24-hour format.
left(string,n)	String	Returns left-most n characters of string. returns the value abc.
len(string)	Integer	Returns the number of characters in string. returns the value 3.
mid(string,start,length)	String	Returns a substring of string beginning with the start character and continuing for length characters. returns the value bcd.
result	Floating Point or Integer	Returns the result of the current Select node. (This function is valid only in a Select node.) If you are using the LAA rule in the Select node, result returns the number of seconds the selected agent has been available.
right(string,n)	String	Returns right-most n characters of string. returns the value cde.
substr(string,start [, length])	String	Returns a substring of string beginning with start character and continuing for length characters. returns the value 01851.
text(n)	String	Converts a numeric value to a string. returns the value "5".
valid(variable)	Logical	Returns whether variable has a valid value. tests whether the database C name is a valid value.
ValidValue(variable,value)	String	If variable has a valid value, returns that value; otherwise, returns "value". returns either a name from the database or the string value None.
value(string)	Floating Point or Integer	Converts a string to a numeric value. returns the value 5.

Custom Functions

This section contains the following information:

Note: <Skipped table here>

What are Custom Functions?

Custom functions are those functions you create for use within scripts, as opposed to built-in functions.

How to Add a Custom Function

Adding Custom Functions

Step 1 In Script Editor, from the **Script** menu, select **Custom Functions**. The Custom Functions dialog box opens, listing all the custom functions currently defined.

Step 2 Click **Add** to open the Add Custom Function dialog box.

Step 3 Specify the following:

    a.   Function name. All custom function names must begin with user.

    b.   Number of Parameters. The number of parameters to be passed to the function. A function may take 0, 1, or more parameters.

    c.   Function definition. The expression to be evaluated when the function is called. When entering the function definition, keep the following in mind:

The parameters to a function are numbered beginning with 1. To reference a parameter within the expression, surround it with percent signs (%). For example, %3% is a reference to the third parameter.

The lower portion of the dialog box is just like the Formula Editor. You can use it to help build the expression.

Step 4 When finished, click **Test**. The Test Function dialog box opens.

Step 5 Test the function by entering an example of how you might reference the function. Include a specific value for each parameter.

Step 6 Click **Evaluate** to see how the Script Editor interprets the function call and click **Close** to return to the Add Custom Function dialog box.

Step 7 Use one of the Validate buttons to validate the scripts that reference a selection function. (The Validate All button lets you validate all the scripts that reference any custom function.)

Step 8 When finished, click **OK** to apply changes and to close the dialog box.

How to Import a Custom Function

Importing Custom Functions

Step 1   In Script Editor, from the **Script** menu, select **Custom Functions**. The Custom Functions dialog box opens, listing all the custom functions currently defined.

Step 2   Click **Import**. The Import Custom Function dialog box opens

Step 3   Choose a file name with an ICMF extension (.ICMF) and click **Open**. The Script Editor examines the file for naming conflicts. If a conflict is found, a dialog box appears listing options for resolving the conflict.

Step 4   Choose one of the options and click **OK**.

Note: If you choose to rename the function, the new name must begin with user.

The Script Editor performs automapping and the following happens:

- If all imported objects were successfully auto-mapped, a message window appears prompting you to review the mappings. Click **OK** to access the Object Mapping dialog box.

- If some imported objects were not successfully auto-mapped, the Object Mapping dialog box appears, with all unmapped objects labeled Unmapped.

The Object Mapping dialog box contains three columns:

- Object Types. The type of imported objects.

  - Imported Object. Name of imported object.

    Mapped To. What this imported object will be mapped to.

- (Optional.) Click an Imported Object value. The Mapped To column displays all the valid objects on the target system.

- (Optional.) Select an object from the Mapped To columns drop-down list on the target system that you want to map the imported object to.

Note: Multiple objects may be mapped to the same target. Objects may be left unmapped; however, the resulting custom function will not be valid until all objects are mapped.

When the mapping is complete, click **Apply** and **Finish**.

How to Export a Custom Function

Exporting Custom Functions

Step 1 In Script Editor, from the **Script** menu, select **Custom Functions**. The Custom Functions dialog box opens, listing all the custom functions currently defined.

Step 2 Select the custom function(s) from the list and click **Export**. The Export Custom Function dialog box opens.

Note: If you selected a single function, that functions name appears in the File Name field. If you selected more than one function, the File Name field is blank.

Step 3 (Optional.) Change the File Name.

Note: You cannot change the file type; the script can only be saved in .ICMF format.

Step 4 Click **Save**. If the file name already exists, the system prompts you to confirm the save.

Step 5 If prompted, click **OK**. The custom function(s) are saved to the specified file in text format.

Chapter 8

Script Administration

This section describes Script Administration.

This chapter contains the following topics:

How to Check Script Routes

After you save a script, you should check that all routes referenced have valid labels for the routing clients and dialed numbers for which you have scheduled the script.

Note: You must have saved the script to the ICM database before you can check the routes.

For ICM software to route calls through a script, you must have defined a routing label for each route referenced in the script. Each label is valid only for specific routing clients and, optionally, for specific dialed numbers.

To check script routes:

Step 1   Start Check Routes from the ICM Admin Workstation group. The Check Routes window opens.

Step 2   Using the drop-down lists, select the following:

- Name of the routing client to send calls through the script.

- Dialed number for the call to be sent through the script.

- The routing script name.

- The version number of the routing script. (The default is the active version.)

Step 3 To validate the route of a network transfer call target, check Use Network Transfer specify a routing client and a dialed number.

Step 4 Use the drop-down list to select Routes Used Directly by Script (the default) or Translation Route Used in Script. The routes referenced in the script appear in the left column. (If any of these routes do not have an associated label that is valid for the routing client and dialed number you have chosen, an error message appears in the Errors field.)

Note: To see the configuration details for a translation route, select the route name and click View Translation Route. (This button becomes enabled when you choose a specific translation route.) The Translation Route dialog box opens.

Step 5 To see the specific peripheral targets associated with a route, select the route name in the left column. The associated peripheral targets appear in the center column.

The routes referenced in the script appear in the left column. If any of these routes does not have an associated label that is valid for the routing client and dialed number you have chosen, an error message appears in the Errors field at the bottom of the window.

Step 6 To see the specific labels associated with any of these peripheral targets, select the peripheral target. The associated peripheral targets appear in the right column.

The following symbols might appear next to a label:

- The label is not valid for the specified routing client.

- The label is not valid for the specified dialed number.

Step 7 To check configuration information for a route or peripheral target, double-click a route or peripheral target name. Check Routes to display the configuration information for that route or peripheral target.

Note: From the Peripheral Target dialog box, you can access information about the route by clicking the Route button.

Step 8 To correct any problems you find through Check Routes, make and save changes within the Script Editor or the Configuration Manager.

Step 9 To see the effect of changes, click Reload in the Check Routes window. (If you have created a new version of the script, be sure to update the Version field.) Check Routes reads the latest version of scripts and configuration data from the local database.

How to Set Active Scripts

Although the Script Editor may contain several versions of a script, only one version of a particular script can be active at one time: this is the version that ICM software runs if the script is currently scheduled.

You can use one of three methods to set the active version of a script:

- Preferences

- Make Active Version command

- Script Explorer

How to Use Preferences to Set an Active Script

Step 1   Within Script Editor, select Options > Preferences. The Script Editor Preferences dialog box opens.

Step 2   Optionally, check the Automatically Make a Script Active When Saved checkbox.

- When you select this checkbox, whenever you save a valid script, ICM software makes that new version the active version.

- When you clear this checkbox, you must manually activate the script after saving it.

How to use the Script > Make Active Version Command

Step 1   In an open script in edit mode, select Script > Make Active Version or click Make Active Version. ICM software makes the open script version the current active version.

How to View Enabled Scripts

Once a script has been saved and scheduled, it is considered enabled. Use the Enabled Scripts dialog box to examine all scripts currently enabled in the system.

To view all enabled routing and administrative scripts:

Step 1   Within Script Editor, select Scripts > Enabled Scripts. The Enabled Scripts dialog box opens listing all routing scripts that are scheduled for the current date and time.

The dialog box lists all call types and the script that is currently scheduled for each. The Call Count column lists the number of calls the script has processed.

Step 2   Optionally, use the Display Count From section of the dialog box to choose how to display the number of calls processed. Whichever option you choose, the counts are updated every 15 seconds. This allows you to see which scripts are currently handling calls.

Note: The Call Count values are associated with scripts, not necessarily with call types. If the script has been scheduled for more than one call type, the Call Count value includes all calls processed by the script regardless of call type.

Step 3   To see all administrative scripts scheduled for the current date and time, click the Administrative Scripts tab. This lists all administrative scripts.

How to Schedule an Administrative Script

To schedule an administrative script:

Step 1   Within Script Editor, select Script > Administrative Manager. The Administrative Manager dialog box opens, listing any administrative scripts that are currently scheduled.

Step 2   Click Add. The Add Administrative Schedule dialog box appears, opening at the Script tab.

Step 3   Select the script you want to schedule and click the Period tab.

Step 4   Specify when you want the script to be active:

a.   In the Date Range, Recurrence Pattern, and Duration sections, specify the range of times when this script may run.

b.   In the Frequency section, specify how often the script should run during the specified time range.

Step 5   Optionally, click the Description tab and add some descriptive text about this schedule.

Step 6   When you have finished, click OK to submit the schedule and return to the Administration Manager.

Step 7   Click OK to save your changes to the ICM database and close the Administrative Manager.

Notes:

• The first execution occurs at the start time of the schedule.

• The last execution occurs before or at the end time of the schedule.

• The timing of script execution might not be exact. Typically, scripts execute within a few seconds after the scheduled time. The last script execution might occur slightly after the scheduled end time.

Monitoring Scripts

After you save a routing or administrative script, you can observe how it runs. Watching routing requests moving through a script in real-time helps ensure that the routing script is operating as expected.

When you monitor a script, that is, view the script in Monitor mode, labels appear on each connection in the script.

Monitor Labels

Most monitor labels display the raw number and percentage of route requests that have passed through the connection since the start of the monitoring period.

Each target set also lists the number and percentage of calls routed to each of the targets in that set. Each statistic is updated automatically as new real-time data become available about every 15 seconds.

When you edit a script, position nodes so that there is enough space for the monitor labels to display. (Because you cannot make any changes to a script while in Monitor mode, you cannot rearrange the nodes at that time, unless you have enabled Quick Edit from Monitor Mode as described below.) Use the **Script > Display Monitor Labels** command while in edit mode to display blank monitor labels on each connection of the script.

Enabling Quick Edit from Monitor Mode

To enable Quick Edit from Monitor Mode:

- From the Options menu, select **Preferences**.

- Check **Allow quick edit from monitor mode**.

- Click **OK**.

You can now perform quick edits when monitoring a script.

Modified Monitor Labels

The monitor labels have been modified for the following seven nodes.

- Wait

- Queue

- Translation Route to VRU

- DB Lookup

- Gateway

- ICM Gateway

- Queue to Agent

- Collect Data (displays only the Abort meter)

- Menu (displays only the Abort meter)

- Play (displays only the Abort meter)

- Run External Script (Displays In Progress and Abort meter)

The modified monitor labels can display two of the following three meters:

- InQueue meter - Displays the number of calls in queue.

- InProgress meter - Displays the number of calls in progress in the node.

- Abort meter - Displays the number of calls aborted.

The InProgress meter is displayed in the top row of all the monitor labels for all the listed nodes except the Queue node. The top row of the Queue node monitor label displays the InQueue meter.

The second row of the monitor label for all of the listed nodes displays the Abort meter.

How to Adjust Monitor Label Location

To adjust the Monitor Label Location

1. In the Properties dialog box of a node within Script Editor, click the Labels tab.

2. Specify the following:

   – Label position. (Slider.) Choose whether to display connection labels close to the node (Origin), close to the targeted node (Destination), or half way in-between (Center).

   – Display monitor labels. (Checkbox.) If this box is checked, then when the Script Editor is in Monitor mode, labels display for each connection from the node. If the box is not checked, no labels display for connections from the node.

3. Click OK to apply changes and to close the Select Properties dialog box.

How to Access Monitor Mode

Within Script Editor, do one of the following:

1. Select File > Open to access the Open dialog box and choose the Monitor option from the Open Mode drop-down list.

2. From within an open script in the Script Editor workspace, select Script > Monitor Script or click Monitor Script in the toolbar.

3. From within the Enabled Scripts dialog box, select the Monitor radio button in the Open Script in Mode section and click Open.

The script displays in Monitor mode in the Script Editor workspace.

How to Set Monitor Mode Options

To set Monitor Mode Options

1. In Script Editor, do one of the following:

   – Select Script > Monitor Options.

   – Click Monitor on the status bar.

     A drop-down menu appears.

2. Choose among the following options:

   – Start of Day. Initial monitor values are totals since midnight. New values are added to these totals every 15 seconds.

   – Starting Now. Initial monitor values are all zeros. New values are added to these totals every 15 seconds.

   – For Each Interval. Initial monitor values are the values reported for the most recent 15-second interval. New values are added to these numbers every 15 seconds.

   – Percentages for Each Node. The percentages for each connection from a node are calculated by dividing the number of executions that passed though the connection by the number of executions that entered the node. All connections coming from each node add up to 100%.

   – Percentages for Entire Script. The percentages for each connection are calculated by dividing the number of executions that passed through that connection by the total number of executions handled by the script.

How to View Real-time Data

In viewing the number of contacts that pass through each branch of the script, you can also view real-time data about activity at the contact centers, including statistics for each skill group, service, call type, or scheduled target referenced in the script.

For specific information about the real-time data, see the descriptions of the Service_Real_Time and Skill_Group_Real_Time tables in the Cisco ICM Software Database Schema Handbook.

To view real-time service or skill group data:

1. While in Monitor mode in Script Editor, select **Script > Display Real-Time Data**. The Real-Time Data window opens.

   By default, the Real-Time Data window displays data about the Services referenced in the current script. (Scroll to the right to see additional columns.) The values in the screen are updated continuously as new real-time data arrives at the Admin Workstation.

2. To display information about skill groups or scheduled targets, rather than services, change the value in the field in the upper-left corner of the Real-Time Data window. The window contents automatically change to show skill group data.

3. When finished, click Close.

How to Change the Real-time Data Configuration

To change the Real-time Data Configuration:

1. Click **Configure** in the Real-Time Data window or select **Script > Configure Real-Time Data**. The Configure Real-Time Data dialog box opens.

   Use this dialog box to add new columns, remove existing columns, reorder the columns, or change the column labels.

2. Select a Routing target type (Service, Scheduled Target, or Skill groups). This updates the other fields:

   – The Real-Time Columns list contains all the real-time columns available in the database for the target type.

   – The Routing Target Columns list contains the columns to be displayed in the Real-Time Data window.

   – If you select a column in the Routing Target Columns list, the heading for that column appears in the Column Header field.

3. Use the Add and Remove buttons to move columns between the lists. Use the Move buttons to change the order of the selected columns. (To change back to the default column order, click Default Columns.)

4. Optionally, to edit a Routing Target Columns heading, make changes within the Column Header field. (To change back to the default header, highlight the Routing Target Column name and click the Default Header button.)

5. Optionally, click the Short Headers / Long Headers radio buttons to change between the full and abbreviated forms of the default column headers. The abbreviated forms are typically three to four letters. (For example, AHT is used for Average Handle Time.)

6. To save the changes, click OK. The settings apply to the current and future Script Editor sessions.

How to View Router Logs

You can view CallRouter log messages to determine how contacts have been routed and to see any errors ICM software has encountered in processing routing requests.

You Start Router Log Viewer from the ICM Admin Workstation group. The Router Log Viewer window opens.

The top field of the Router Log Viewer window displays information about each call ICM software has routed, including:

- The time that routing request was received

- The dialed number (DN), or script selector, and the caller's billing telephone number (ANI)

- Any caller-entered digits (CED)

- The label that ICM software returned to the routing client

The bottom field of the window displays any errors that ICM software has encountered in routing calls, including:

- The time the error occurred

- Text describing the error

How to Export a Script

To export a script:

1. Select File > Export Script. The Export Script dialog box opens with the name of the current script and version number in the File name field.

2. Optionally, change the File name.

You cannot change the file type; the script can only be saved in .ICMS format.

3. Click Save. If the file name already exists, the system prompts you to confirm the save.

4. If prompted, click OK.

How to Import a Script

To import a script:

1. Select File > Import Script. The Select Script to Import dialog box opens.

2. Choose a file name with an ICM Script extension (.ICMS) and click Open. The Script Editor performs automapping and the following happens:

   – If all imported objects were successfully auto-mapped, a message window appears prompting you to review the mappings. Click OK to access the Object Mapping dialog box.

   – If some imported objects were not successfully auto-mapped, the partially mapped script is opened as a new script in Import mode and the Object Mapping dialog box appears, with all unmapped objects labeled ?Unmapped.?

     The Object Mapping dialog box contains three columns:

   – Object Types. The type of the imported object.

   – Imported Object. The name of the imported object.

   – Mapped To. What this imported object will be mapped to.

3. Optionally, click an Imported Object value. The Mapped To column's drop-down list shows all the valid objects on the target system. (The Script Editor window also highlights the script nodes that refer to this object.)

4. Optionally, select an object from the Mapped To drop-down list on the target system that you want to map the imported object to.

   Note: Optionally, select an object from the Mapped To drop-down list on the target system that you want to map the imported object to.

5. When the mapping is complete, click Apply and Finish.

How to Modify Script Version and Schedule System Information

To modify script version and schedule system information:

1. Within the Configuration Manager, select Tools > Miscellaneous Tools > System Information. The System Information window opens.

2. Enter the Minimum Script Schedule Time.

3. Set the number of script versions to be retained in the Retain Script Versions field.

4. Click Save to apply your changes.

How to Modify Script Version and Schedule System Information

Chapter 9

Script Editor Feature Control

Script Editor Feature Control addresses the need of restricting users, or classes of users, from some or all of the functionality of the ICM Script Editor software. In a possible deployment scenario, an ICM software administrator can restrict certain people from doing specific types of script editing.

An administrator has two means to restrict access to the editing features of Script Editor and Internet Script Editor:

- Edit Options

- Script Node Control

Note: When ICM software is running on a partitioned system, users need at least reference access to objects to edit scripts that contain references to those objects. For example, a user needs at least reference access to skill groups to edit a script in which those skill groups are included in the Skill Group node.

It is also possible for an administrator to use a combination of both feature control options.

For more information about Feature Control, see the Cisco ICM Software Configuration Guide.

This chapter contains the following topics:

Edit Options

The administrator can assign one of two editing options:

- Full Edit (Includes Quick Edit)

- Quick Edit Only

You can only access the Full Edit mode or the Quick Edit mode from the Monitor or the Browse modes. The Full Edit and the Quick Edit modes cannot be accessed from each other.

Full Edit (includes Quick Edit)

Full Edit mode allows you to use Full Edit mode or Quick Edit mode when working with scripts. Both Full Edit and Quick Edit are enabled on the Script Editor toolbar. When changing from Monitor or Browse mode to Full Edit mode, the Script Editor workspace background goes from gray to white.

Your Full Edit mode editing capabilities are determined by whether or not you are assigned to a feature control set:

- If you are a full-edit user and are not assigned to a feature control set, you can add, edit, or delete any script or node.

- If you are a full-edit user and are assigned to a feature control set, you can add, edit, or delete any script. You only have Full Edit permissions for those scripts that do not contain nodes that are marked as unavailable to you in a feature set.

Full Edit gives you restricted editing capabilities if you are assigned a feature control set. You can still edit any script, however, your ability to edit specific script nodes is determined by which nodes are selected in the Node Control table (located on the Script Editor Feature Control dialog) of the assigned feature control set. This allows you to edit the structure of a script or to create, edit, and delete scripts containing the selected nodes.

If a script is opened that contains a disabled node, you can browse or monitor the script but you cannot put the script into edit mode. If you attempt to put this script into edit mode a message indicating you are not authorized to enter edit mode is displayed. However, you can still Quick Edit the script, just not the node.

You might want to use Quick Edit mode so as not to accidentally change the structure of a script.

Quick Edit Only

Quick Edit mode allows you (the feature-control-set member) to work only with scripts in Quick Edit mode. Edit is disabled and Quick Edit is enabled on the Script Editor toolbar. When changing

from Monitor or Browse mode to Quick Edit mode, the Script Editor workspace background goes from gray to yellow.

In Quick Edit mode:

- You cannot add or delete a node

- You can adjust most of the properties of the script nodes selected in the Node Control table of your assigned feature control set. However, any properties of the selected nodes that change the structure of a script or that reset previous reporting data cannot be edited in Quick Edit mode.

As a Quick Edit Only User:

- You can only edit scripts through Quick Edit mode.

- You cannot create or delete a script

- You can access the Properties of any script node in any mode by either right clicking on the node and selecting Properties, or by double-clicking on the node.

- You cannot edit the Call Type Manager dialog (Script > Call Type Manager).

- You cannot edit the Administrative Manager dialog (Script > Administrative Manager).

- You cannot edit the Custom Functions dialog (Script > Custom Functions).

- Script > Make Active Version is disabled.

- Script > Make Active Version is disabled.

- You cannot import scripts.

- You cannot use the File > Script Locks tool.

How to access Quick Edit Mode

Quick Edit can be accessed by performing any of the following steps:

- Select Script > Quick Edit.

- Click Quick Edit on the Script Editor toolbar.

- Right-click in the Script Editor workspace and select Quick Edit.

- From within the script in Monitor mode, select, then double-click the node you want to edit. (This is only available if Options > Preferences > Allow for Quick Edit from Monitor Mode is checked.)

When in Quick Edit mode, disabled nodes are removed from the object palette.

Script Editor Feature Control

Script Editor Feature Control allows an administrator to create feature sets that can be assigned to users. The feature set controls which script nodes are accessible to the user and which edit modes are available to the user.

Node Control Table

The Node Control table (on the Configuration Manager Script Editor Feature Control dialog) has two columns, the Node column and the Available column. This table allows an administrator to create feature control sets that can be assigned to users. The feature control set controls which script nodes are accessible to the user.

If a script is opened that contains a disabled node, you can browse or monitor the script but you cannot put the script into edit mode. If you attempt to put this script into edit mode a message indicating you are not authorized to enter edit mode is displayed. However, you can still Quick Edit the script, just not the node.

Node Column

A node is an executable element within a script. A script consists of nodes, connections, routing targets, and comments. Every script begins with a Start node. This column lists of all the nodes that can be used in a script.

Note: The Line Connector node is always available. Available Column

Available Column

Each checked node in this column appears on the editing palette of the feature-control-set user, regardless of the edit mode (Full Edit or Quick Edit Only).

Disabled nodes are removed from the object palette

How to Create a Feature Control Set

The system administrator can create a feature control set using the ICM Configuration Manager on the ICM distributor:

Step 1   Ensure the users the feature set is to be assigned to are configured.

Step 2   Start the Configuration Manager.

Step 3   Select Tools > List Tools > Feature Control Set List.

Step 4   In the Feature Control Set section (on the left), click Add.

Step 5   Enter the name of the feature control set. The name appears in the left section when Enter or Tab is pressed.

Step 6   Optionally, enter a description.

How to Assign Users to a Feature Control Set

Step 1   In Configuration Manager, use the User List tool to select a feature control set to associate with the user.

How to Select Script Nodes for a Feature Control Set

Step 1   In the Feature Control Set List dialog box, select the name of the feature control set to be assigned.

Step 2   Select Advanced (under Script Editor).

Step 3   In the Script Editor Feature Control dialog box, select the nodes for this feature control set and an edit option (Full Edit or Quick Edit).

Step 4   Click OK.

Step 5   Click Save.

Script Editor Feature Control

Chapter 10

Internet Script Editor (ISE)

Instead of or in addition to Script Editor, you can use Internet Script Editor to work with routing and administration scripts.

This topic contains information about Internet Script Editor in the following sections:

This chapter contains the following topics:

What Is Internet Script Editor?

This section describes the Internet Script Editor application.

Internet Script Editor is an application you can use to work with routing and administration scripts. Internet Script Editor provides the same functionality as the ICM Script Editor software, without the need for a full Administrative Workstation (AW).

Note: When ICM software is running on a partitioned system, you cannot edit security information for a script with Internet Script Editor; you must use Script Editor.

How Internet Script Editor Works

This section describes how Internet Script Editor works on and communicates with the ICM software distributor.

Internet Script Editor works through the IIS Web server on ICM software distributor. It uses HTTP to communicate with the ICM software distributor.

The Internet Script Editor and the ICM Script Editor GUIs are essentially the same. The menus, toolbars, palette, and work space are utilized in the same manner in both applications. The differences between the two occur primarily in the method by which each application communicates with the ICM software.

Internet Script Editor Requirements

This section describes Internet Script Editor requirements.

Internet Script Editor is supported on following operating systems:

- Windows 2000

- Windows 2003

- Windows XP

Secure Socket Layer (SSL) Requirements for ISE

Secure Socket Layer (SSL) is a protocol developed by Netscape for transmitting private documents via the internet. SSL works by using a private key to encrypt data that is transferred over the SSL connection. By convention, URLs requiring an SSL connection start with https: rather than http:.

Secure HTTP communications are enabled by the use of SSL 3.0 (in IIS 6.0) which implements encryption/decryption of client requests and server responses. An IIS Web Server is used in the ICM environment for a number of applications, ISE being one of them. This addresses the need to encrypt sensitive data (which includes user names and passwords, configuration data, call control data in scripts, etc.) being exchanged from a user running a web browser and the server hosting the application services.

The SSL Encryption Utility (SSLUtil.exe) provides the ability for both ICM Setup and System IPCC Setup to create and install a self-signed server certificate. When the certificate has been generated, imported to Local Machine Store, and installed on the Web Server; the virtual directories and/or web pages are enabled for SSL and get configured with 128-bit security for the entire session.

A digital certificate is an attachment to an electronic message used for security purposes. The most common use of a digital certificate is to verify that a user sending a message is who he or she claims to be, and to provide the receiver with the means to encode a reply. A self-signed certificate is not automatically recognized by a user's browser(s) and, a self-signed certificate does not provide any guarantee concerning the identity of the organization that is providing the website. Most Web browsers that support SSL have a list of trusted CAs (Certification Authoritys) whose certificates they automatically accept. If a browser encounters a certificate who's

authorizing CA is not in the list, the browser asks the user to choose whether to accept or decline the connection.

Install the standalone SSL Encryption Utility on the AW Real-time Distributor (in the AW Program Group) and the WebView Server (in the /icm/bin directory). This enables you to change the default SSL settings (implemented by Setup). This utility contains the functionality to regenerate the self-signed certificate and replace the IIS installed certificate as needed.

Internet Script Editor Client

While Cisco supported SSL on Windows 2000 for previous versions of ICM, HTTP was setup as the default connection using port 80. If the client required an SSL connection (using port 443), the server had to be manually configured, the certificate setup on the server, and the encrypted flag had to be set on the ISE client. In addition, the ISE client did not check to see if the certificate was valid or not.

With ICM/IPCC release 7.0(0) on Windows 2003, ICM setup now configures the SSL connection (using port 443) by default and sets up the certificate on the ISE server (the Distributor AW).

For an ICM/IPCC release 7.0(0) system on Windows 2000, ICM setup does not configure an SSL connection or setup the certificate. If you areinstalling a new ISE client, an HTTPS connection is configured by default.

An upgraded ISE client uses the connection that was previously configured. During upgrades, if you decide to keep the client at Windows 2000, the client always uses the default port setting (80) unless communications fail then both the server and the client communicate using HTTPS (still using port 80 - no change).

When the ISE client connects to the server, one of the following occurs.

1. If the encrypted flag is not set:

    – The client starts with setting up an HTTP connection.

      If the client successfully connects to the server, it sents the user account and password in plain text. It then establishes the session with the server via HTTP.

      If the client cannot connect to the server, it fails over and tries to connect to the server via HTTPS. If the HTTPS connection is setup successfully, the client sends the encrypted user account and password. It also sets the encrypted flag in the registry so that the next time it will use HTTPS.

    – After the HTTPS connection is established, the server sends the certificate to the client. The client presents the certificate prompt, unless it has been previously saved locally.

Figure 103: Security Alert Dialog Box
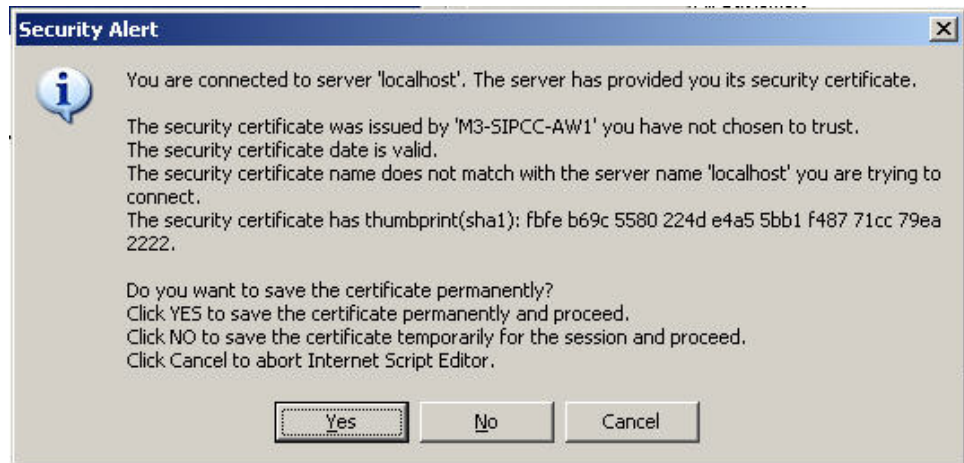
2. If the encrypted flag is set:

   – The client starts with setting up an HTTPS connection.

   – If the client successfully connects to the server, the client sends the encrypted user account and password.

   After the HTTPS connection is established, the server sends the certificate to the client.

   After the HTTPS connection is established, the server sends the certificate to the client. The client presents the certificate prompt (see the Security Alert Dialog Box (page 172)), unless it has been previously saved locally.

   If the client cannot connect to the server, it prompts you to determine if it should failover to try to connect to the server via HTTP, or not. If you select **Yes**, the client sends in the user account and password in plain text once the HTTP connection is established. However, it does not set the encrypted flag in the registry so that the HTTPS connection will still be initiated the next time.

   If you want to use the HTTP connection for all future connections, you must manully unset the flag in the login screen.

During runtime, the initialization is the same for every HTTP request.

During upgrades on clients with Windows 2003, the secure connection is automatically configured to the default setting of 443 for SSL.

During new installations on MS Windows 2000 and Windows 2003, the secure connection is automatically configured to the default setting of 443 for SSL.

Note: ISE client is able to revert back to unencrypted communication over port 80 if it fails to establish an HTTPS session.

How to Install and Upgrade Internet Script Editor

To install Internet Script Editor:

Step 1   Point your browser to **server-name/install/iscripteditor.htm**, where `server-name` is the name of the computer on which you installed the distributor with the Internet Script Editor client package.

Step 2   Enter your **Domain** and **User Name**, and **Password**, then click **OK**.

Step 3   Click **Download Internet Script Editor**. .

You can also open the **iscripteditor.exe** file directly from the Web page

Step 4   Navigate to the directory where you want to save **iscripteditor.exe**.

Step 5   Click **Save** to begin the download.

Step 6   When the download is complete, close the browser.

Step 7   On your desktop, navigate to **iscripteditor.exe** and execute the file.

Step 8   When the InstallShield Wizard for Internet Script Editor starts, click **Next** to continue.

Step 9   Select the default Destination Folder by clicking **Next**; or click **Browse** to navigate to the desired Destination Folder, and then click **Next**.

Step 10   When the InstallShield Wizard indicates the installation is complete, click **Finish**.

A shortcut for Internet Script Editor (IScriptEditor) appears on the desktop, as well as in the Start menu in the Programs/Cisco Systems Inc. program group.

To start Internet Script Editor

Step 1   Double-click the desktop shortcut for Internet Script Editor (IScriptEditor).

Step 2   Click **Connection**.

Step 3   Enter the correct **Address**, **Port**, and **ICM Instance** information.

Step 4   Click **OK**.

Step 5   Enter your **User Name** and **Password**.

Step 6   Enter the **Domain** of the ICM system.

Step 7   Click **OK**.

Step 8   Upgrade Internet Script Editor as necessary.

To upgrade Internet Script Editor

After you start Internet Script Editor, if there is a newer version, you receive a message informing you that you can upgrade Internet Script Editor.

Note: Some upgrades are optional; these upgrades typically contain GUI enhancements. Other upgrades, typically involving protocol or database changes, are mandatory. You cannot use Internet Script Editor until you accept mandatory upgrades

Step 1   Accept a software upgrade.

A Web page opens from which you can download the new Internet Script Editor.

Step 2   From this point, follow the directions on installing Internet Script Editor, above.

You are not able to use Internet Script Editor during the upgrade.

Troubleshooting Tools for Internet Script Editor

This section describes the tools that you can use to troubleshoot the Internet Script Editor

Client-Side Internet Script Editor Troubleshooting Tools

The following table describes the client-side Internet Script Editor troubleshooting tools:

Troubleshooting Method	Description
EMS trace files	Internet Script Editor writes to EMS logs and purges old logs on startup just as ICM Script Editor does. These logs are located in:
Dr. Watson	Internet Script Editor is built without symbol tables to keep it small. This makes Dr. Watson output more difficult to debug.
PCAnywhere	Because the client runs on hardware completely outside of Cisco control, no support for remote control debugging is provided.

Server-Side Internet Script Editor Troubleshooting Tools

The following table describes the server-side Internet Script Editor troubleshooting tools:

Troubleshooting Method	Description
IIS Logs	IIS logs its activity to system event log or to an ODBC data source.

Troubleshooting Method	Description
EMS trace files	ISAPI DLL generates trace output on the distributor. System administrators can use the Dumplog utility to display the contents of the logs.
PCAnywhere	Because the client runs on hardware completely outside of Cisco control, no support for remote control debugging is provided.

Troubleshooting Tools for Internet Script Editor

Chapter 11

Scripting in an IPCC Environment

IPCC Gateway

The IPCC Gateway PG allows either a site IPCC Enterprise system, or an IPCC Express system, to connect to an Enterprise ICM. The Enterprise ICM views either as an ACD.

Figure 104: IPCC Gateway PG
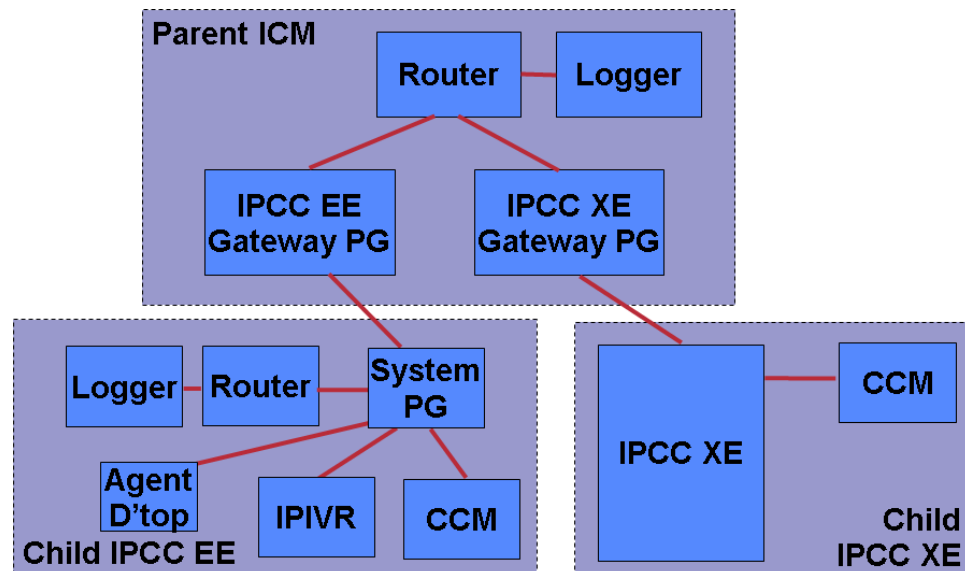
For IPCC Enterprise, this is important since it allows a "parent" ICM to monitor and send calls to a "child" IPCC Enterprise system. Previous to the IPCC Gateway feature, doing this using intelligent (PG based) routing was not possible. This allows better scaling, resiliency, and simpler, more standardized scripting across sites.

The Gateway PG looks to the ICM like any other PG, it does not look like IPCC. It has two different Peripheral types to allow connection to either a "child" IPCC Enterprise system, or an

IPCC Express system. These are listed as "IPCC Enterprise Gateway" and "IPCC Express Gateway" respectively in the setup screen for the PG. The Gateway PG does not support third party call control (CTI) so that all agent desktops, etc. are connected to the child IPCC Enterprise system, or the IPCC Express system. The link supports voice only. The child system may support multi-media but the ICM only routes voice calls to the child.

Object tracking is accomplished the same as on all legacy PIMs, through matching Peripheral Numbers on the Agent, Skill Group, and Service Table. Routing to the child is to peripheral targets (Skill Groups/Services) as with all legacy TDMs, in contrast to IPCC. All normal TDM functionality is supported: Pre-Routing, Pre-Routing with Translation routing, post routing, etc. Third party call control is the only exception. Full variable passing is done between the parent and child (you can send/receive call variables 1-10 and ECC variables).

Scripting is consistent with traditional PGs, not IPCC. Scripts use LAA, MED, target services, and skill groups; but not agents.

Note: NCT is not supported with either the Express or Enterprise Gateway. This is due to the CTI control being on the ACD (Child) rather than on the enterprise ICM.

Deploying IPCC Gateway for IPCC Enterprise

Where to deploy IPCC Gateway for IPCC Enterprise:

- The gateway PG is ideal for deployments that have several call centers spread geographically.

- The call centers are independent of the ICM Central Controller so in a WAN outage, the call centers (System PG systems) can operate totally independently of the ICM.

- The gateway PG is also an ideal way to integrate new IP call centers into an existing ICM environment with many TDM ACD sites.

  Note: In a single script node (MED) peripherals that are legacy and gateway type can be used to select the best "site" to send the call to.

How does it work:

- The gateway PG works by monitoring all the events that happen on the child IPCC system. This is identical in function to a PG/PIM connected to any other ACD. By knowing the active call and agent states the ICM router can have information available to decide if that "site" is the best to send a call to.

- The child system can also send up route requests to the parent ICM to decide where to send a call (Post/Translation routing).

- The event monitoring is accomplished by the PIM connecting to the child systems CTI Server modified version of the CTI Protocol.

Object Mapping Overview (Enterprise)

- Agents on the child map to Agents on the parent

- Skill Groups on the child map to Skill Groups on the parent

- Call Types on the child map to Services on the parent.

  Note: This is due to the child's reporting being based on CallTypes and the parents reporting being based on services.

IPCC Gateway for IPCC Express Features

- The gateway PG for IPCC Express allows an IPCC Express IP-ACD to be integrated into an ICM-Enterprise solution.

- Post and translation routing are supported.

- Call variable transfer is supported.

Object Mapping Overview (Express)

- Agents on Express child map to Agents on the parent by Agent ID.

- CSQs on the child map to Skill Groups on the parent by CSQ ID.

- Applications on the child map to Services on the parent via by Application ID.

- There is no direct mapping of Service Member. This needs to be done manually on ICM based upon the Select Resource steps in scripts on Express.

IPCC Enterprise

This topic contains the following information and recommendations for writing routing scripts when ICM software is part of an IPCC Enterprise environment:

Prioritizing Agents

You can prioritize agents within a skill group by:

1. Using a Select node and grouping the agents that have a higher priority in a skill group in the first Select Longest Available Agent (LAA) node,

2. then looking for available agents in a subsequent Select node that has another lower priority grouping of agents.

This is done in the case where there are idle agents when a call comes in. The order of the Select nodes within the script determines the agent prioritization.

Prioritizing Calls

When a call is queued to a skill group because there are no agents available, the Queue to Skill Group node sets the call's priority. The Queue Priority node can then promote the call's priority based on time the caller has waited. The call can be queued to multiple skill groups with the same or different priorities.

If there are calls in the agent's skill group queues when an agent becomes available, the agent will be presented with the highest priority (1-10 with 1 being the highest priority) call that has waited the longest within the skill group(s) that the agent is assigned to.

Checking for Available Agents

A script that routes to IPCC agents needs to check for an available agent within a skill group. If an agent is not available, then the script should use a Queue to Skill Group node. The script execution will end when an agent becomes available or when the caller disconnects.

Using the Select Node

You can use the Select node to check for the Longest Available Agent (LAA). However, the Select node cannot check for Minimum Expected Delay (MED) of a service, because IPCC skill groups do not have a valid expected delay because the calls in the queue are on the IVR PG service.

Do not put LAA Select nodes after a Queue to Skill Group node for the same skill group. This node can only be used if the call is not already queued to a skill group.

There is, however, a lengthy workaround to determine the MED for a call using an IF node. See Estimated Wait Time Queing (page 232)

Queuing to a Skill Group

If you use the Queue to Skill Group node instead of Queue to Enterprise Skill Group, then do not use the base priority within the Queue node unless the option is enabled in the router through the registry. If agents are assigned to base skill groups, use base skill groups in this node; if they are assigned to sub-skill groups, use sub-skill groups.

If the script has both Queue to Skill Group and Queue to Enterprise Skill Group nodes, then do not include skill groups defined within the Queue to Skill Group nodes that belong to the enterprise skill groups that are defined within the Queue to Enterprise Skill Group node.

Canceling Queuing

If the call needs to be taken out of a skill group, then use the Cancel Queuing node. The Cancel Queuing node takes the call out of all the skill groups it is queued to.

Using the Busy Node

Use the Busy node for initial overflow conditions. Do not use this treatment if the call has already received ringback tone or given announcements or music. If a Busy CTI Route point and CTI port are defined on IP IVR and CallManager, then IP IVR will tell CallManager to return a busy signal. If a Busy CTI Route point and CTI port are not defined on CallManager and IP IVR and the call is already connected to an IP IVR port, then the IP IVR port will play a busy tone from a .wav file for 30 seconds and then disconnect the call. If the IVR port plays a busy tone from a .wav file then, answer supervision will be returned to the far end. This would cause charges to occur for the call. If the call is a toll free call (for example, 800#), then the Contact Center will accrue the charges; otherwise the caller has to pay.

A Busy CTI route point and one CTI port need to be defined on CallManager and associated with the IP IVR user. The one CTI port can be used for multiple calls. In IP IVR Administration, a new CTI Route point of Type Busy and a CTI port need to be defined whose dialed numbers match what was previously defined on CallManager.

Define a label of type busy with a Label field that corresponds to the aforementioned CTI Route point dialed number in CallManager and IP IVR.

If the call is not at an IVR port and the Busy CTI Route point is not defined on CallManager and IP IVR and ICM returns a busy label to CallManager, the CallManager returns a fast busy to the caller for 30 seconds and then the call is disconnected.

Using the Ring Node

You can use the Ring node for diverting blacklisted callers. It is supported as a post route from CallManager. The call will be given ringback tone until the caller hangs up. No answer supervision will be returned for this call since it is not connected to a IVR port. This means that if this is the initial treatment given the caller, then no charges will be accrued for this call.

You need to define a Ring No Answer CTI port group in CallManager, but you do not have to define CTI ports. In IP IVR Administration, you need to define a new CTI Route Point of Type Ring No Answer whose dialed number matches what was previously defined on CallManager.

Define a label of type Ring with a Label field corresponds to the CTI Route Point dialed number in CallManager and IP IVR.

Using the Release Call Node

Use the Release Call node for initial overflow conditions. It is supported as a post route from CallManager. Do not use this node if the call has already received ringback tone or announcements or music. The call will be given busy tone for 30 seconds before it is disconnected on CallManager. No answer supervision will be returned for this call since it is not connected to a CTI port. This means that no charges will be accrued for this call.

No configuration needs to be done within CallManager or IP-IVR. No label needs to be defined within ICM.

Using the End Node

The End node will either try default routing, or if there is no default label, it sends an error (dialog fail) to the routing client.

Call Treatment Comparison

Attributes/Treatment	Busy Node	Ring Node	Release Node	End Node
Answer Supervision	# No if Busy CTI Route Point defined # Yes if Busy CTI Route Point not defined	No	No	No
Voice port required	Yes (only 1 CTI port needed)	No	No	No
Timing	Initial treatment only	Anytime	Initial treatment only	Initial treatment only
Applications	Overflow Conditions	Blacklisted Callers	Overflow Conditions	Overflow Conditions
Treatment provided by IP IVR	IP IVR		CallManager	CallManager

Using the Agent to Agent Node

You can use the Agent to Agent node for agent to agent transfers; the router checks agent availability before sending the call to the agent. If the agent is not available, the script queues the call to a skill group. The Agent to Agent node can also be used to send a call to the agent, the "caller" is not required to be an agent.

Using the Service and Enterprise Service Nodes

You do not need the Service node or Enterprise Service node to route calls to IPCC agents because services are not required for IPCC agents.

Using the Scheduled Select and Divert Label Nodes

CallManager does not support the Scheduled Select node and Divert Label node. You cannot use these nodes to route calls to IPCC agents.

Using IVR as a Queue Point

IPCC relies on the IVR to queue the call while it is waiting for an available agent.

ICM sends the call to the IVR port for queuing:

1. To provide the call with a termination point that will allow the VoIP Gateway to return the correct signals or messages back to the PSTN.

2. Provide announcements or music or expected wait time or initial position in queue to the caller while they are waiting for an agent. Allow the caller an option to leave a message if the caller does not want to wait for an agent.

3. To obtain further information from the caller that is not sent from the network

The IVR will let ICM know when the caller disconnects via the Event Report Message with Event Type of either DISCONNECT or ABANDON. When an agent becomes available, ICM will automatically instruct the IVR to route the call to the agent through the Connect message.

Interruptible vs. Non-Interruptible

If the VRU script is collecting digits from the customer in order to ascertain information regarding the caller that is crucial for a screen pop or call routing, put the VRU script in a non-interruptible mode.

If a call was queued to a skill group through a Queue to Skill Group or Queue to Enterprise Skill Group node and then sent to VRU in order to hear a non-interruptible VRU script, if during the time that the caller is interacting and listening to the non-interruptible VRU script, an agent becomes available, the call will not be connected to the agent. ICM only looks for available agents for that call when the VRU script is finished and the call executes an ?interruptible node? such as a Wait node or a Run External Script node for a VRU script that is interruptible. The call will, however, maintain its place in the queue so when the call does become available for an agent, it will be answered before calls that came in afterward it (assuming the same priority).

For announcement and music type of treatments, put the VRU Scripts in interruptible mode. This allows the call to be connected to the first available agent even while the caller is listening to a VRU script.

You set the interruptibility of a VRU script through the Configuration Manager, under the Add Network VRU script configuration. Neither the VRU or ICM script can overwrite this setting.

IVR (VRU) Types

Listed in the table below are the IVR types that are supported for IPCC These types are defined in ICM in the Network VRU Configuration. These IVR types are used by ICM to determine the routing client, routing type, script node used and IVR port status. Type 3, 7 can only be Network IVRs. However, Types 2 and 6 can be Network or On-premise IVRs.

IVR Type	Routing Client	Routing Type	Scripting Node Used to Send to IVR	ICM Knows IVR Port Status
2	Yes	Post-route from CallManager	Translation Route to VRU	Yes
3, 7	No	Pre-route	Send To VRU	No
6	Yes	Post-route from IVR	N/A	Yes
9	Yes	Queueing for System IPCC	Send To VRU, or any Queueing node	Yes

Usually the IVR has been configured to support only one IVR type, although IP IVR can support both Type 2 and Type 6 based on whether the call is routed directly to the IVR or the call comes into a Translation Route point. In this case, the IVR type must be defined as Type 2.

Using the Translation Route to VRU Node

The Translation Route to VRU node is used by Type 2 IVRs that are post routed from CallManager, when the call is not at the IVR, but at another routing client. The call can be a pre-route from the IXC or it can be a post route from a CallManager CTI Route point.

The Translation Route to VRU node is used to send the call to the IVR port. Translation routing is needed for:

- Cradle to grave reporting (Termination Call Detail Reporting)

- Ability to send call context data like the calling line ID or dialed number to the IVR

- Ability to check if the IVR peripheral is online or if all IVR ports are busy before sending the call.

The Translation Route to VRU node is typically followed by a Run External Script node. If some checking is done before the Run External Script node is executed, the time it takes ICM to execute the nodes in the script must not exceed the IVR Request Time-out timer. Information obtained from the caller during the IVR session can be passed to ICM for further processing.

No Default Skill Groups in Routing Scripts

Ensure your ICM routing scripts do not reference the default skill group. This ensures that the default skill group does not capture statistics for any ICM-routed calls.

See Also

Cisco ICM Software: IP Contact Center Installation and Configuration Guide
Cisco ICM Software: IP Contact Center Administration Guide
Cisco ICM Software: IP Contact Center Laboratory Guide
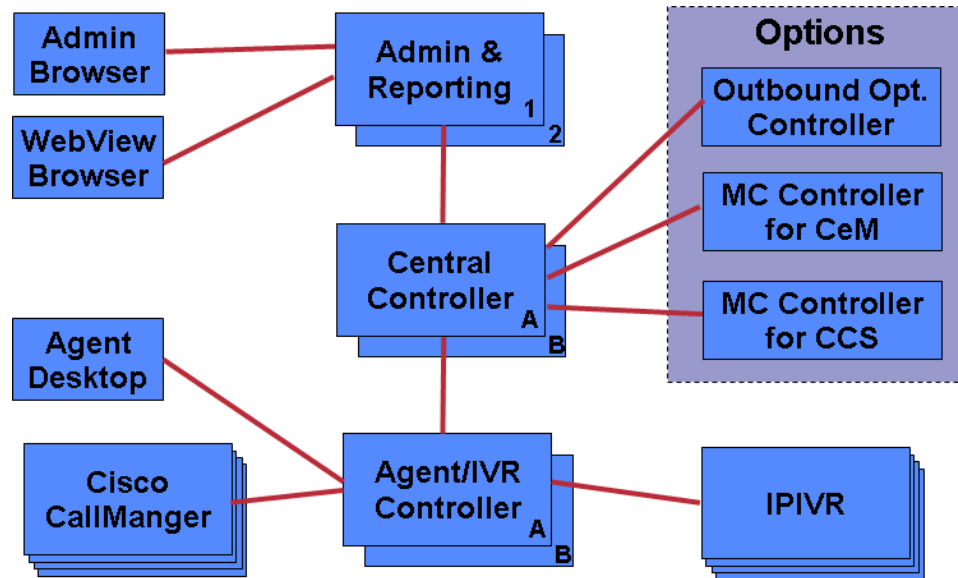
System IPCC

Deployments

- 3 Main IPCC deployments supported

    - Demo/Pilot System – not for production use

    - Small to Medium System – up to 300 agents

    - Large System – up to 1000 agents

- Each deployment can be duplexed

- Each deployment requires specific MCS hardware (or the exact equivalent)

    - Each deployment has a different box count

- Each deployment can optionally support

    - 1 Outbound option and/or

    - 1 E-Mail Manager option (version 7.0) and/or

    - 1 Web Collaboration option (version 7.0)

- Each deployment can talk to one CallManager cluster and up to 5 IP-IVRs

The following diagram shows the most distributed deployment of System IPCC. Not all of these components must be on separate boxes, but this is an overall diagram of how the components go together.

Figure 105: System IPCC Deployment

IPCC System PG

IPCC System PG can be used as:

- An integration point for the IPCC Gateway PG.

- A standalone IPCC PG providing seamless configuration and intelligent call reporting features.

Overview

- System IPCC is similar to Generic PG with co-resident CCM and IVR peripherals (PIMs).

- A mixture of CCM and IVR peripherals can operate within System IPCC, however, IVR peripherals are hidden behind CCM (single routing-client interface like the ACD model).

- CCM peripherals within System IPCC can share the common IVR resources if JTAPI triggers are partitioned properly.

- Seamless configuration is achieved through IPCC Web Administration Tool (one CCM peripheral and five IVR peripherals are pre-configured).

- Each IVR is configured in the Network VRU Bank table as Type-9 with a single network label to allow load-balancing and correlation-id based routing (no translation-routes).

- Same caller redirected between IVR and CCM is captured in single termination call detail record (TCD).

- Only IP IVR is supported in this release. CVP is not supported in the Network VRU Bank table.

- System IPCC presents a single peripheral interface to IPCC Gateway PG through consolidated reporting of agent and call states.

- Scripting and reporting are more intuitive, e.g. direct use of Queue to Skill Group node and integrated IVR and CCM reporting.

- Device Targets are no longer required.

Interworking System IPCC and IPCC Gateway

- IPCC Gateway PG is an all-events CTI client receiving consolidated state information through CTI Server attached to IPCC System PG.

- IPCC System PG diverts routing of all calls arriving on controlled DNs to IPCC Gateway PG.

- IPCC System PG feeds most configuration elements to CTI Server for retrieval by IPCC Gateway PG (e.g. auto-configuration and device monitoring).

- IPCC System PG ensures consistent reporting by sending events to CTI Server when certain call attributes change such as call type, queue point.

Summary

- System IPCC can be deployed standalone with a mixture of CCM and IVR peripherals.

- System IPCC presents a single peripheral model to IPCC Gateway PG.

- System IPCC presents a single routing client interface to Central Controller.

- System IPCC is easier to configure and script than IPCC Enterprise PG.

- Combined reporting of agent and IVR activity.

- CVP is not supported, IP IVR must use SCI.

System IPCC to ICM Object Mapping

Following is a list of ICM objects either not used or renamed in System IPCC:

- System IPCC does not use or require these ICM objects: services, labels, persons, device targets, translation routes, dialed number plan, and trunk groups.

- System IPCC does not support post routing.

- System IPCC does not support sub skill groups.

- System IPCC's call variables are equivalent to ICM's expanded call context variables.

- System IPCC's network IVR is equivalent to ICM's network VRU.

IPCC Express

In terms of scripting and routing, the IPCC Express environment is treated no differently than any other legacy ACD.

You can design ICM scripts to interact with IPCC Express scripts in an IPCC Express system integrated with an ICM system through the IPCC Gateway. In such an integrated system, the IPCC Express software is linked as an ACD to the ICM software.

Scripting in an IPCC Express Environment

The scripts on Parent ICM must match the scripts on the Child IPCC Express system. This means that, if the parent ICM script returns a route-point as label, the script on child IPCC Express system must be designed to handle that route-point label.

In an IPCC Express environment , the Parent can return the following labels to a Child IPCC express:

- Route-points or DN (local or remote)

  – The Child IPCC Express script must be designed to redirect the call to that route-point.

- CSQ ID

  – The Child IPCC Express script must be designed to use skill based routing and queue the call to that CSQ.

- Agent-ID

  – The Child IPCC Express script must be designed to use Agent based routing and send the call to that agent.

Using Variables in an IPCC Express Environment

In IPCC Express scripts that interact with ICM scripts through the IPCC Gateway, you can use three types of variables:

- Local variables

- Cisco predefined enterprise call variables

- Enterprise Expanded Call Context (ECC) variables

IPCC Express uses enterprise Call Variables and enterprise Expanded Call Context Variables when passing data between the integrated systems that make up an IPCC Gateway system:

- the Cisco ICM Enterprise Server

- the Cisco IPCC Express system

- the Cisco Agent Desktop

The Set Enterprise Call Info and Get Enterprise Call Info steps are used to take data stored in a local IPCC Express script variable and to make it available for display in the Cisco Agent Desktop or for use in an ICM script.

Cisco Predefined Enterprise Call Variables

You can use the following Cisco predefined strings as enterprise call variables in IPCC Express in the Set Enterprise Call Info and Get Enterprise Call Info steps to pass information between the ICM system and the IPCC Express system:

- VRU Script Name

- ConfigParam

- Call.CallingLineID

- Call.CallerEnteredDigits

- Call.PeripheralVariable1 to Call.PeripheralVariable10

- Call.AccountNumber

The Cisco ICM Enterprise Server, the Cisco IPCC Express system, and the Cisco Agent Desktop support these call variables for passing data among themselves.

If you need more call variables than those predefined in the General tab, use Expanded Call Context (ECC) variables.

Enterprise Expanded Call Context (ECC) Variables

The Cisco ICM Enterprise Server, the Cisco IPCC Express system, and the Cisco Agent Desktop can also pass ECC variables to each other.

Enterprise Expanded Call Context (ECC) data fields are used by all applications in the IPCC Express Cluster. There can be as many as 200 user-defined fields defined in the Field List (index numbers 0-199) of expanded call variables. These field values do not appear in the ContactCallDetail records as there are no fields reserved for them. IPCC Express has pre-defined ECC variables.

> **Note:** Every enterprise ECC variable must be separately defined on all parts of the system that sends and receives the variable data: the CRS Editor in IPCC Express, the Cisco Agent Desktop, and the ICM software.

> **See Also**

> *Cisco CRS Scripting and Development Series: Volume 2, Editor Step Reference Guide*

Example IPCC Gateway Post-routing Scripts

In IPCC Gateway deployments where the IPCC Express system is the child system, that is, connected to an ICM; the IPCC Express system can use the ICM system for post-routing. Post-routing is typically used to enable the ICM to determine the best routing solution based on the current situation of the call center.

The following are three sample post-routing scripts that illustrate three different ways of post-routing through the Cisco IPCC Gateway. In each of these examples, the script accepts the call from the Cisco CallManager in the IPCC Express system and then queries the ICM through the Request Route step. IPCC Express then routes the call based upon the return value of the Request Route step supplied by the ICM.

In the first sample script, the call is routed to a CSQ. In the second sample script, the call is routed to an agent. And in the third sample script, the call is routed to a route point.

These are the sample scripts:

- PostRouteSelectCSQ.aef

  - The RouteRequest step returns a label corresponding to a CSQ which is used in the select resource step.

- PostRouteSelectAgent.aef

  - The RouteRequest step returns a label corresponding to an agent extension which is used in the select resource step.

- PostRouteSelectSimple.aef

  - The RouteRequest step returns a label containing an IPCC Express route point which is used in the call redirect step.

Each of three sample IPCC Express scripts presume an ICM script designed to interact with the IPCC Express script, depending on what IPCC Express resource is wanted: a CSQ, an agent, or a route point.

> **Note:** The ICM script developer must work with the IPCC Express script developer so that the correct ICM script calls the correct IPCC Express script and vice versa.

Each of the following ICM scripts are extremely basic (simply returning a label to either a CSQ, an agent or a route point) and are used for example purposes only. The most important point

from these examples is that the ICM and IPCC Express script writers need to work together during both design and implementation to ensure that the correct type of information is returned by the ICM script via the IPCC Express Route Request step and is used properly by the IPCC Express script to route the call appropriately. The following figure displays labels used in these examples.

Figure 106: Label Node Properties
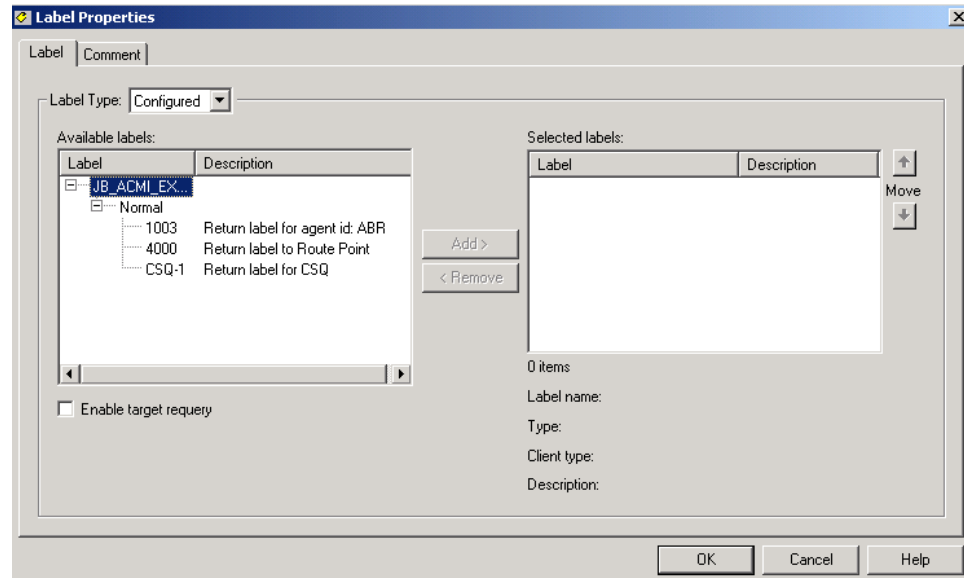
Example IPCC Express and ICM Scripts that Select a CSQ

The following figure displays an example script using the Select Resource step to route the call to a member of a CSQ, as it appears in the CRS Editor window. For additional information on this IPCC Express script, refer to the "Designing IPCC Gateway Scripts" chapter of: *Cisco CSR Scripting and Development Series: Volume 1, Getting Started with Scripts, Release 4.0(1)*.
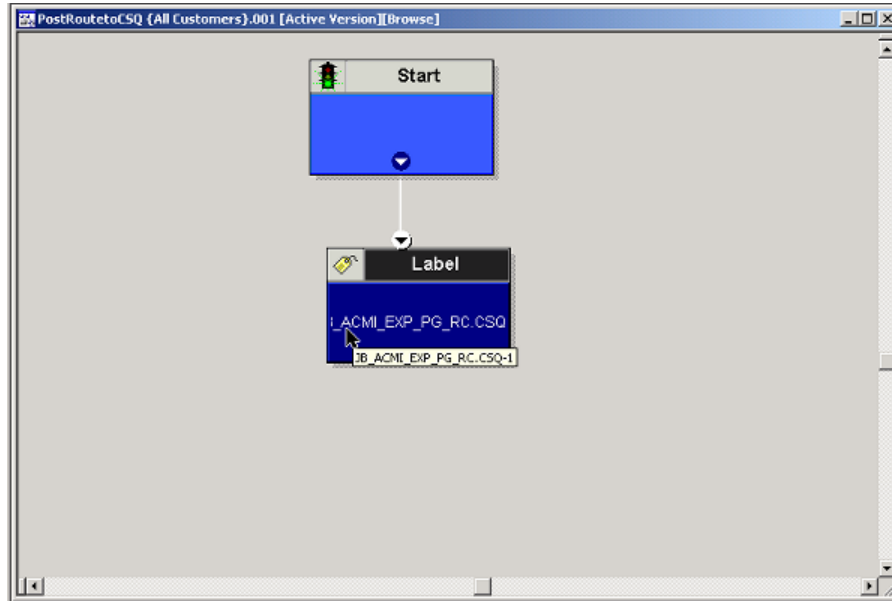
Example IPCC Gateway Post-routing Scripts

Figure 107: PostRouteSelect CSQ.aef

The following is an example of an ICM script that works with the example IPCC Express script above to route the call to a member of a CSQ.
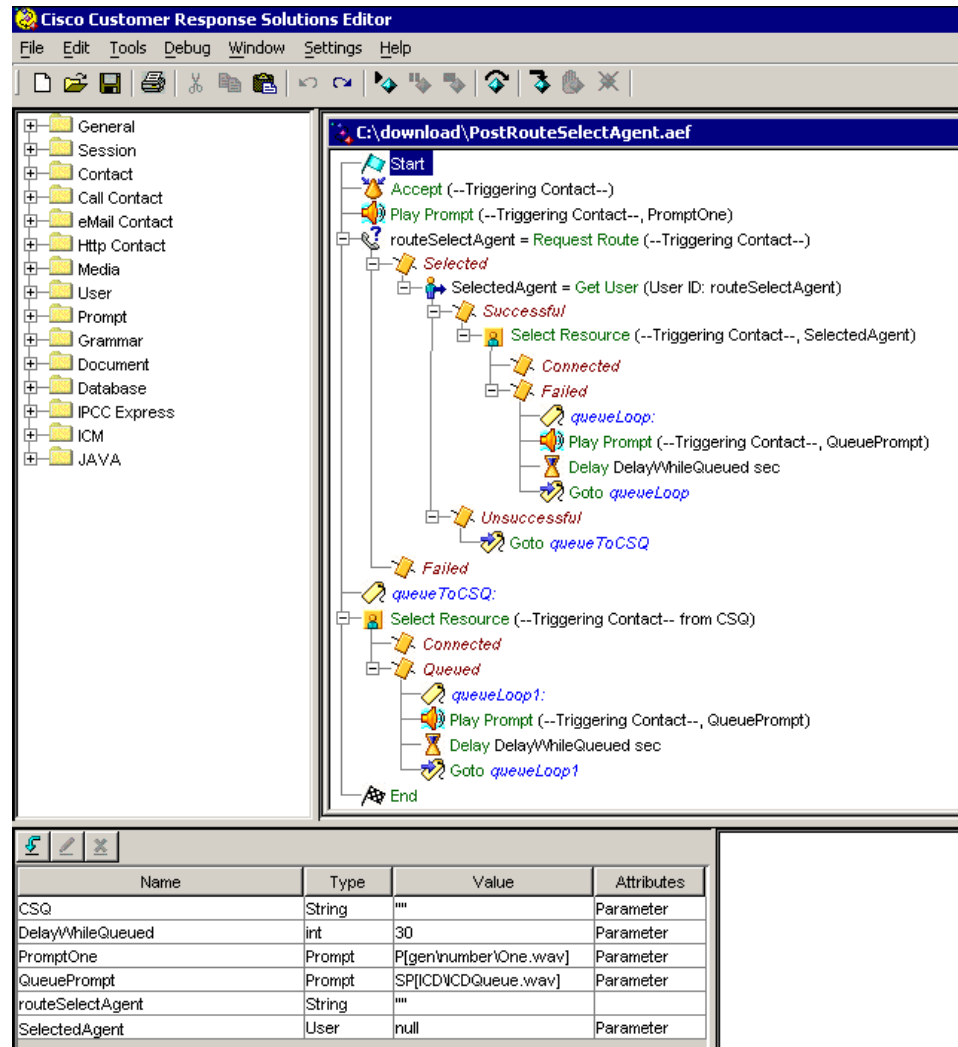
Figure 108: Post Route to CSQ

Example IPCC Express and ICM Scripts that Select an Agent

The following figure displays an example script using the Select Resource step to route the call to a specific agent (agent-based routing), as it appears in the CRS Editor window. For additional information on this IPCC Express script, refer to the "Designing IPCC Gateway Scripts" chapter of: *Cisco CSR Scripting and Development Series: Volume 1, Getting Started with Scripts, Release 4.0(1)*.

Note: For agent based routing to succeed, the state of the selected agent must be ready. The Select Resource step fails if the selected agent is in any state other than ready. The ICM can determine the current state of an agent although there is no guarantee that the state of the agent state will not change between when the ICM returns information and the IPCC Express script routes the call based on that information although, depending on the design of the script, the time between the two should be extremely small making it unlikely that this would occur.
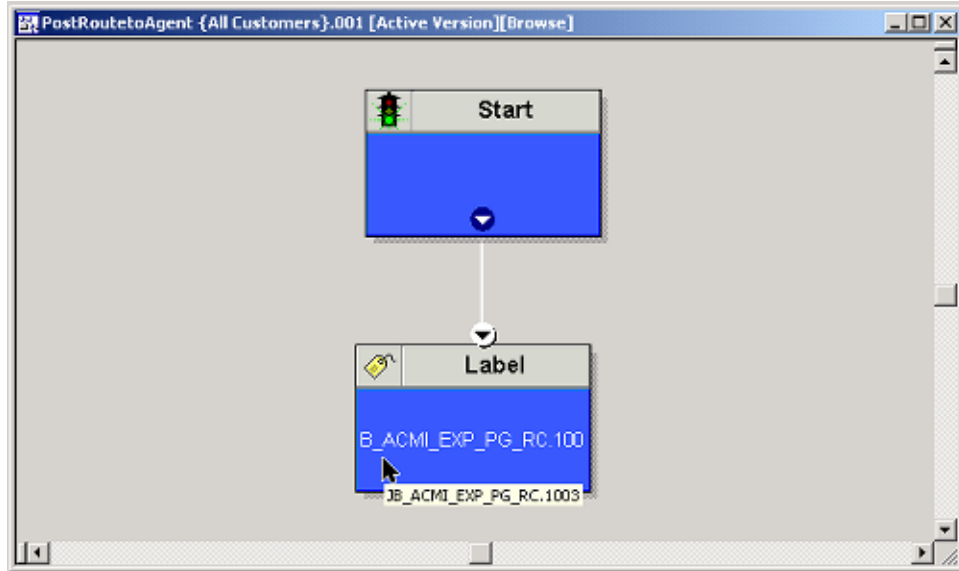
Example IPCC Gateway Post-routing Scripts

Figure 109: PostRouteSelectAgent.aef

The following is an example of an ICM script that works with the example IPCC Express script above to route the call to an agent.

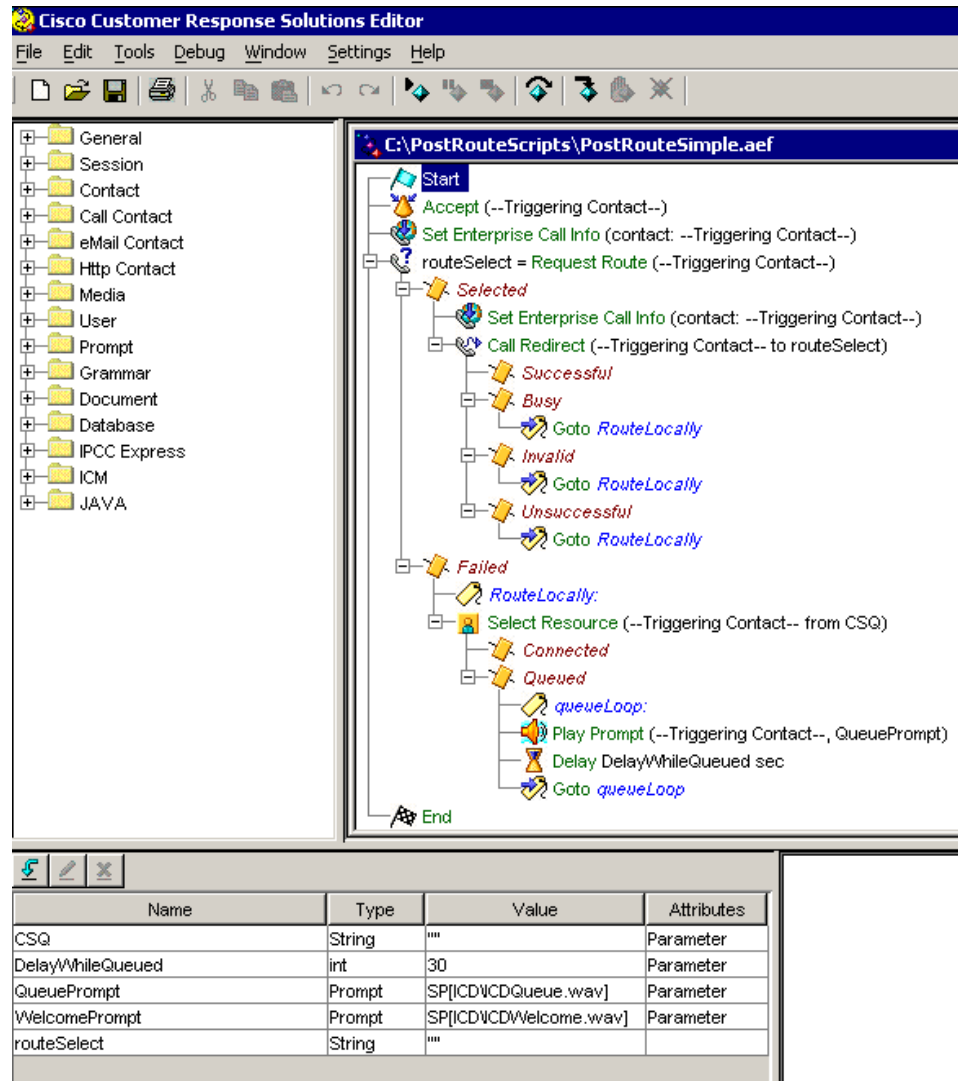*Figure 110: Post Route to Agent*

Example IPCC Express and ICM Scripts that Select a Route Point (Redirect)

The following figure displays an example script using the select from the Call Redirect step to place a new call to a route point, as it appears in the CRS Editor window. For additional information on this IPCC Express script, refer to the "Designing IPCC Gateway Scripts" chapter of: *Cisco CSR Scripting and Development Series: Volume 1, Getting Started with Scripts, Release 4.0(1)*.
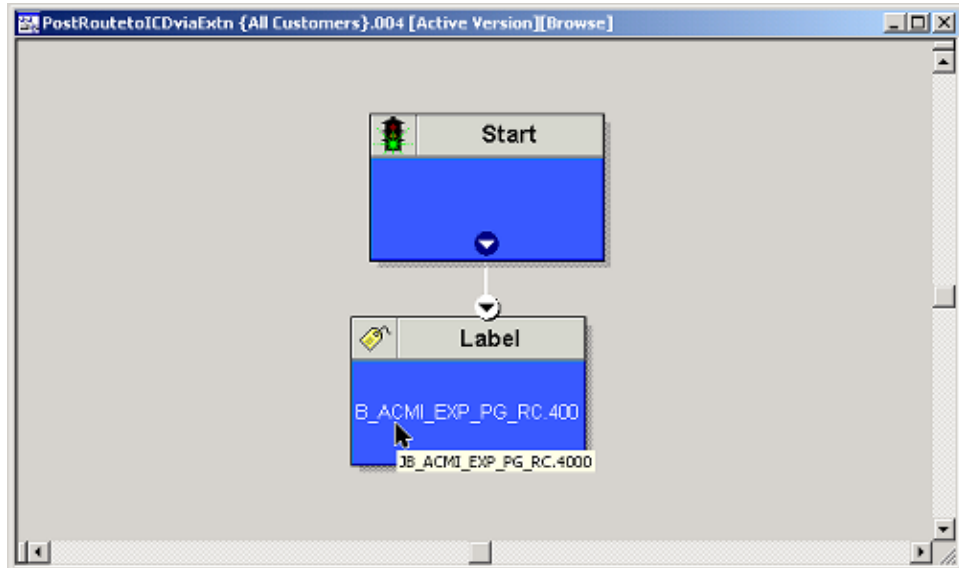
Example IPCC Gateway Post-routing Scripts

Figure 111: PostRouteSimple.aef

The following is an example of an ICM script that works with the example PostRouteSimple.aef IPCC Express script above to select a route point for the call.

Figure 112: Post Route to a Route Point

Scripting in an Outbound Environment

Outbound Option only supports both Type 2 (Generic PG environment) and Type 9 (IPCC System PG environment). For CVP, Outbound Option supports both Type 7 and Type 5 (CVP/ISN Comprehensive Model - Cisco CVP was formerly called Cisco ISN).

If using the transfer to IVR feature, create a transfer to IVR campaign. Transfer to IVR campaigns might require translation routes if using a Type 2 IVR, so refer to the IPCC Installation and Configuration Guide for Cisco IPCC Enterprise Edition for details about setting up translation routes. (Note that translation routes are not required when using the IPCC System PG.)

Note: The transfer to IVR feature is only supported for Outbound Option on Cisco IP Contact Center (IPCC Enterprise). You can not use this feature in the Direct Preview or the regular Preview modes.

The dynamic routing client feature is used when an outbound agent transfers a call using CVP/ISN to a Type 2 VRU. Make sure the routing client for the translation route labels is CallManager, which makes the outgoing call.

Controlling the OutboundControl Variable and Skill Group Reservation Percentage Using an Administrative Script

Use the ICM Script Editor application to create an administrative script for each skill group to control the OutboundControl variable and the skill group reservation percentage (one script can be used to control all Outbound Option skill groups or multiple scripts can control multiple Outbound Option skill groups). The Outbound Option Dialer looks at the value of the OutboundControl variable to determine which mode each skill group uses. If the OutboundControl variable is not set, the skill group defaults to inbound. If using the transfer to IVR feature, create a transfer to IVR campaign.

Note: The transfer to IVR feature is only supported for Outbound Option on Cisco IP Contact Center (IPCC Enterprise). You can not use this feature in the Direct Preview or the regular Preview modes.

How to create an administrative script to control the OutboundControl variable and skill group percentage:

1. Open the ICM Script Editor application.

2. Create an administrative script using a Start node, a Set node, an End node, and an If node (all required).

3. Set the OutboundControl variable.

   Note: Use the Set node to set skill group variables (OutboundControl and OutboundPercent).

   OutboundControl Variable Settings

   – INBOUND: Indicates that this skill group is disabled for outbound use and only takes inbound calls.

   – PREDICTIVE_ONLY: Dials several customers per agent. After reaching a live contact, the Predictive Dialer transfers the customer to a live agent along with a screen pop to the agent's desk. The predictive algorithm is designed to calculate the number of lines to dial per available agent to keep agent wait time to a minimum.

   – PREDICTIVE_BLENDED: Agents receive inbound calls, but could be used for an outbound call when available.

   – PREDICTIVE_BLENDED: Agents receive inbound calls, but could be used for an outbound call when available.

   – PREVIEW_ONLY: Reserves an agent prior to initiating an outbound call and presents the agent with a screen pop. The agent might then

     Accept the call: Dials the customer and transfers the call to the agent.

     Skip the call: Agent receives another customer call.

     Skip-Close the call: Skips the current preview call and closes the record so it will not be called again.

     Reject the call: Releases the agent. At this point, the system might deliver the agent another preview outbound call or a new inbound call.

     Reject-Close the call: Rejects the current preview call and closes the record so it will not be called again.

   – PREVIEW_BLENDED: Agents receive inbound calls, but could be used for an outbound preview call when available.

- PREVIEW_DIRECT_ONLY: Agents can only place outbound calls and hear ring tones, such as phone ringing or busy signal.

- PREVIEW_DIRECT_BLENDED: Agents can receive inbound calls, place outbound calls, and hear ring tones, such as phone ringing or busy signal.

- PROGRESSIVE_ONLY: Similar to PREDICTIVE_ONLY; however, lines to dial per agent is not calculated—users configure a fixed number of lines that will always be dialed per available agent.

- PROGRESSIVE_BLENDED: Similar to PREDICTIVE_BLENDED, but a fixed number of lines will always be dialed per available agent.

  Note: If the administrative script (where the OutboundControl variable or reservation percentage is set) is running, but the modes/percentages are not being updated at the Dialer, do the following: 1. Make sure that the skill group being controlled is the base skill group, and not the primary or secondary skill groups. Although agents may be logged into just the primary or secondary skill group, the outbound control variable must always be set on the base skill group. 2. Verify that the outbound control variable mode is spelled correctly.

4. Set the OutboundPercent variable by entering the agent percentage in the Value field of the Set Properties window.

   Note: Use the Set node to set skill group variables (OutboundControl and OutboundPercent).

   This variable controls the percentage of agents, logged into a particular skill group, to be used for outbound dialing. For example, if there are 100 agents logged into skill group N, and the OutboundPercent variable is set to 50%, 50 agents would be allocated for outbound dialing.

   Note: This variable does not allocate specific agents for outbound dialing, just a total percentage.

The following illustration displays a very simple administrative script where both the OutboundControl variable and the outbound percentage are set for a skill group. A script in a production call center would typically be more complex, perhaps changing these variables due to time of day or service level.

Note:

- Refer to the ICM Script Editor online help for detailed information about using ICM Script Editor.

- See the "IPCC System PG (page 197)" section for sample administrative and routing scripts for the IPCC System PG.

Figure 113: Setting Skill Group Variables (OutboundControl and OutboundPercent)
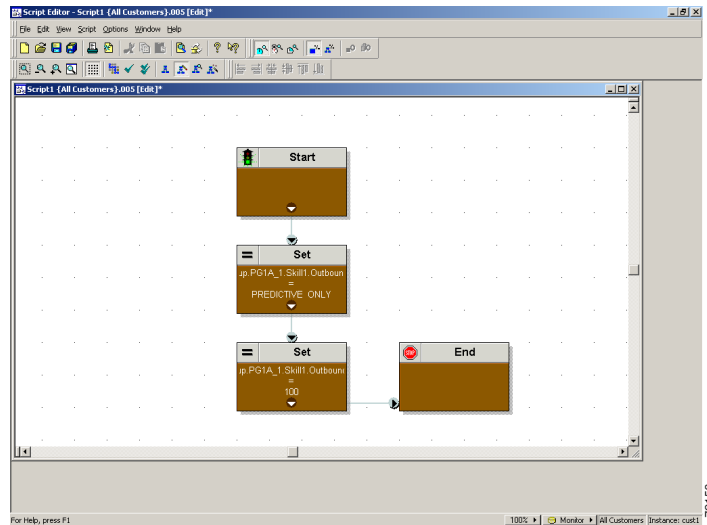
Transfer to IVR Using Outbound Option with IP IVR

The following illustration displays a routing script for a transfer to IVR campaign using Outbound Option with IP IVR. (Refer to the *Outbound Option User Guide for Cisco ICM/IPCC Enterprise & IPCC Hosted Editions* for detailed information about configuring a Outbound Option transfer to IVR campaign.)

Figure 114: Example Routing Script for a Transfer to IVR Campaign Using Outbound Option with IP IVR
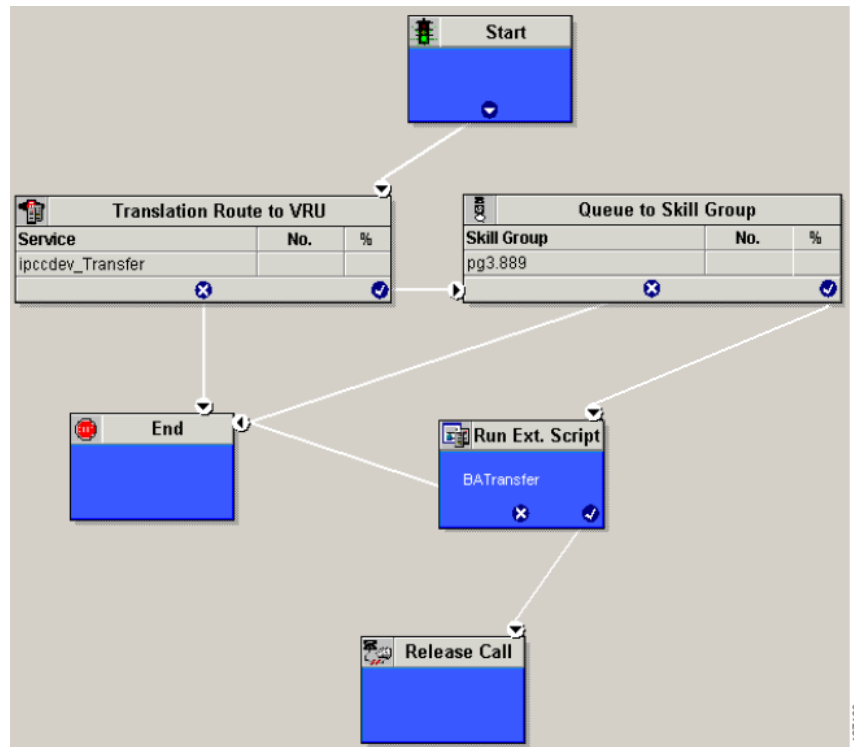
Transfer to IVR Campaign Using Outbound Option with CVP/ISN

The following illustration displays a routing script for a transfer to IVR campaign using Outbound Option with CVP/ISN. (Refer to the *Outbound Option User Guide for Cisco ICM/IPCC Enterprise & IPCC Hosted Editions* for detailed information about configuring a Outbound Option transfer to IVR campaign.)

Figure 115: Example Routing Script for a Transfer to IVR Campaign Using Outbound Option with CVP/ISN
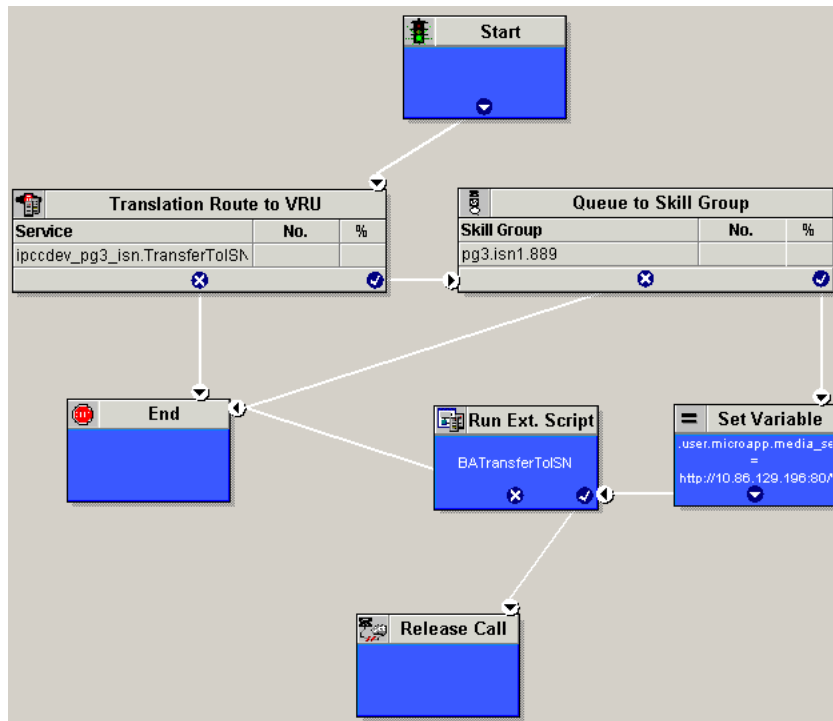
Use the ICM Script Editor application to create a routing script that uses the dialed number for the MR routing client, and routes through a Select node to the previously configured skill group.

Note: Refer to the ICM Script Editor online help for detailed information about using ICM Script Editor.

Use the ICM Script Editor Call Type Manager to associate the MR (and Personal Callback, if used) dialed number(s) with the configured call type and newly created routing script.

Note: Refer to the ICM Script Editor online help for information about using this application.

The following diagram displays an example routing script that uses the objects mentioned above.

*Figure 116: Example Routing Script Using the Dialed Number for the MR Routing Client*
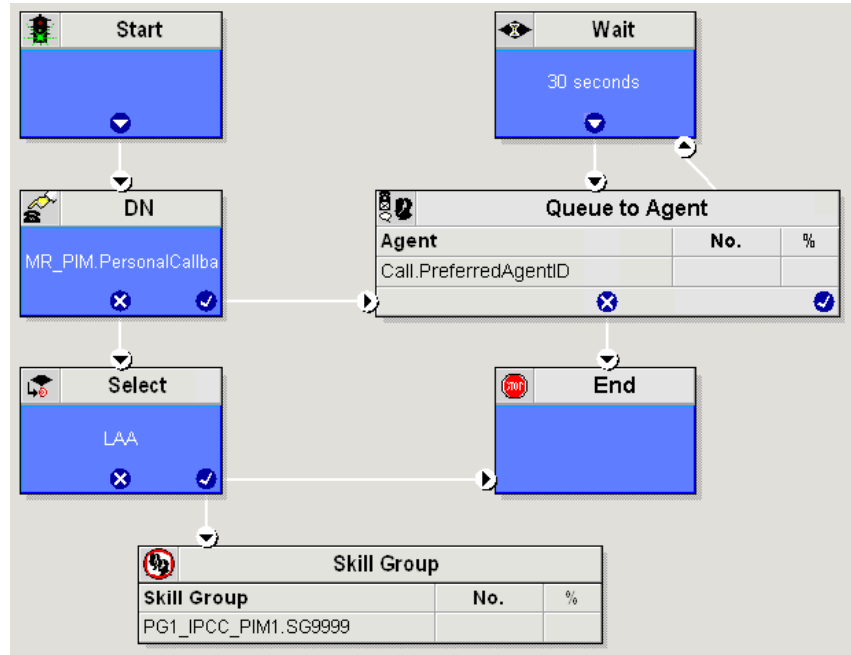
Note:

- Lines connecting objects cannot appear on top of objects and therefore, partially display under the objects; for example, the line connecting the "X" (output terminal failure) on the Select object to the End object runs partially under the Select object.

- Translation routes are not used in the IPCC System PG, so routing scripts using this PG do not need to use this object.

Configuring a Queue to Agent Node

The following illustration displays the Queue to Agent tab of the Queue to Agent Properties dialog box.

Figure 117: Queue to Agent Properties
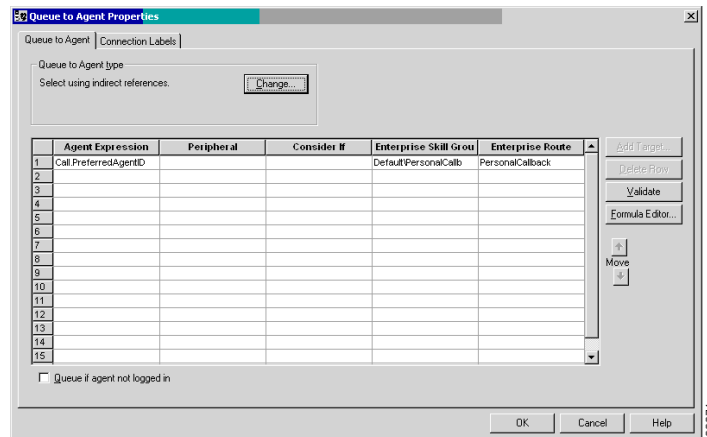
How to configure a Queue to Agent node:

1. Right-click the Queue to Agent node and select **Properties**.

2. Click **Change** in the "Queue to agent type" section.

3. Click **Lookup agent reference by expression**, then click **OK**.

4. Enter the agent expression **Call.PreferredAgentID**.

5. Make sure the Peripheral column is left blank.

6. Select the appropriate enterprise skill group.

7. Select the appropriate enterprise route.

8. Click **OK** to save the Queue to Agent node.

IPCC System PG for Outbound Option

The IPCC System PG combines two PGs (CallManager and IVR PGs) into one PG and is only supported with IP IVR.

Setting up an IPCC System PG for Outbound Option campagns consists of two tasks:

• Configuring the IPCC System PG.

• Adding the PG in ICM Setup.

Use the ICM Script Editor and create an administrative script. The following illustration is an example of an administrative script.

Figure 118: Example IPCC System PG for Outbound Option Campaign Administrative Script
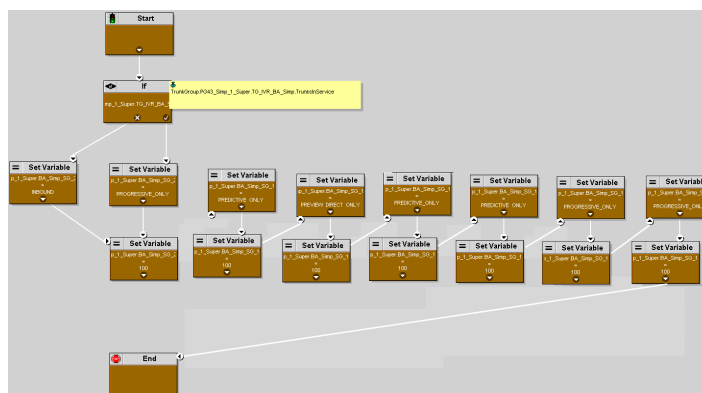
The following diagram displays an example routing script using the Queue to Skill Group node.

Scripting in an Outbound Environment

Figure 119: Example Routing Script Using the Queue to Skill Group Node
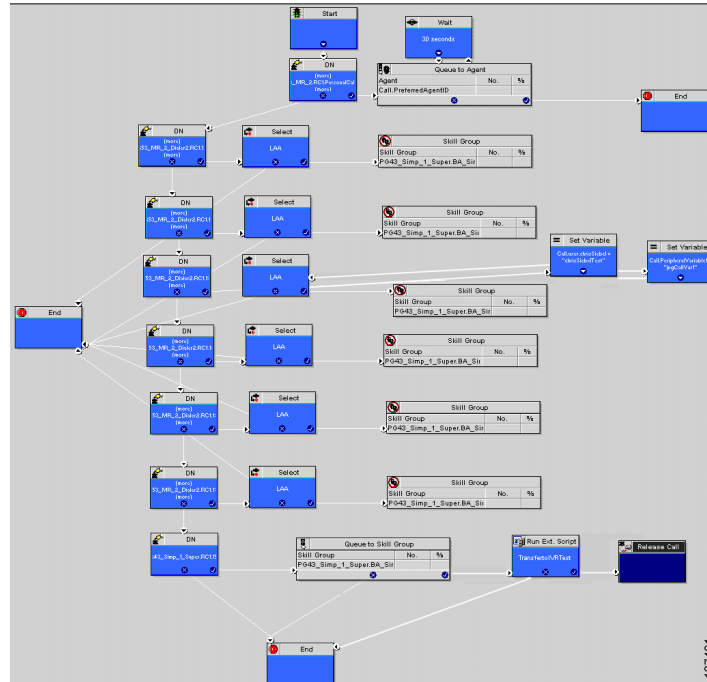
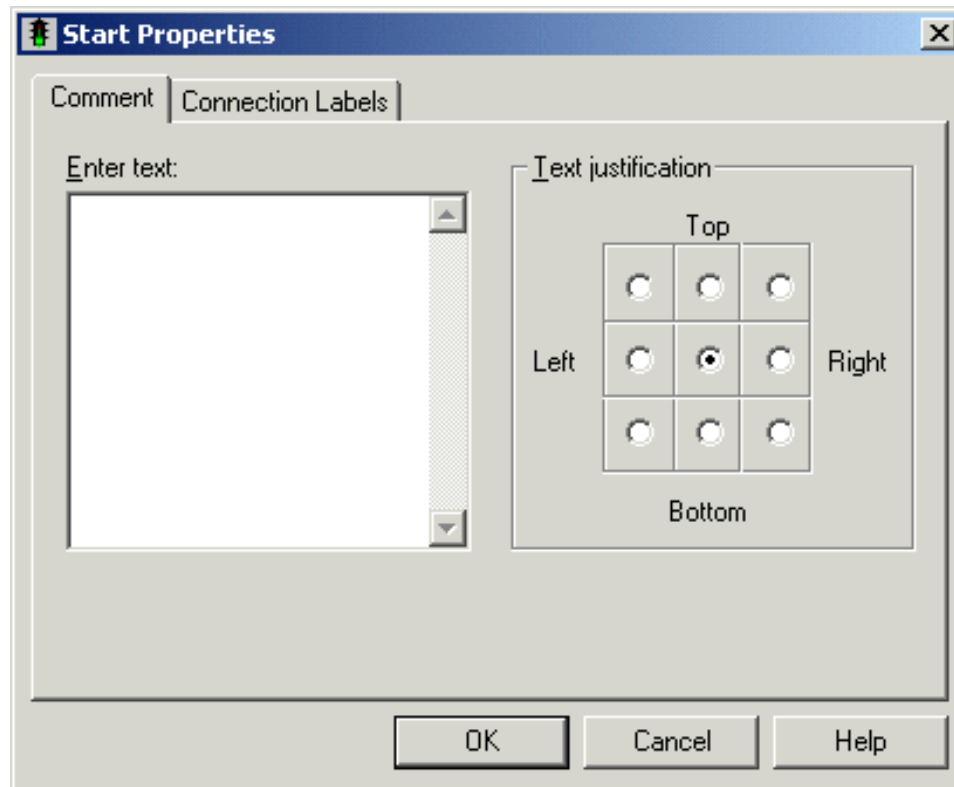# Chapter 12

Utility Nodes

Start Node

The Start node marks the beginning of a script. Script Editor automatically inserts the Start node when you create a new script; a script must have one and only one Start node.

You do not define any properties for the Start node. However, you can add comments and connection labels:
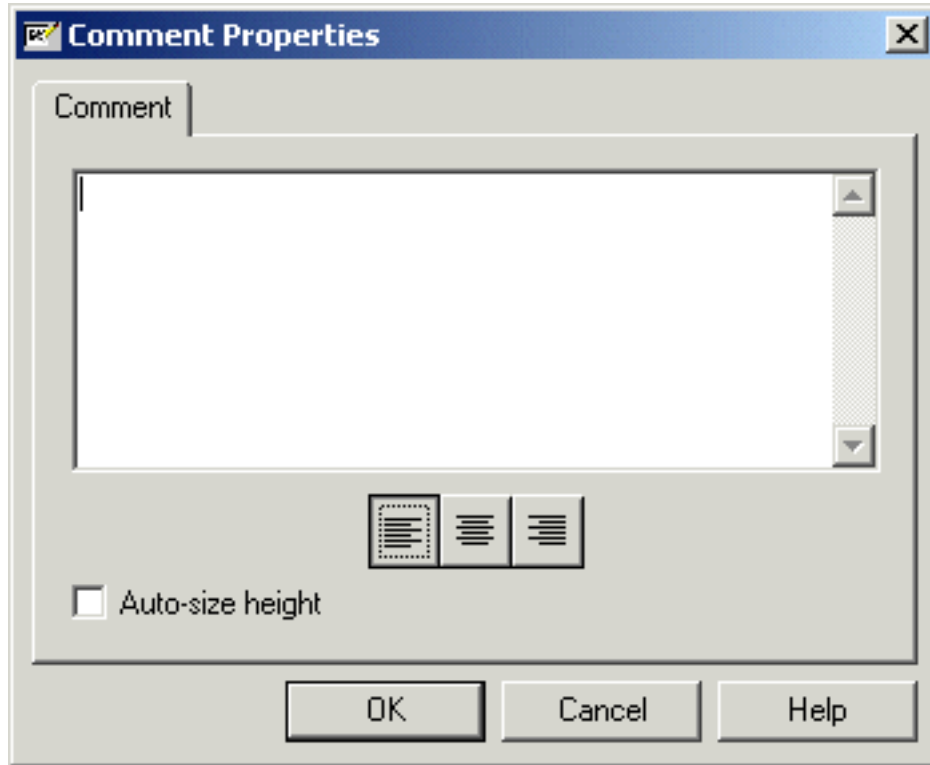
Figure 120: Start Properties

Comment Node

Use the Comment node (in the General tab of the Palette) to include a block comment in a script. A block comment provides general documentation for a script or section of a script:

Figure 121: The Comment Icon

Figure 122: Comment Properties

For example, you might add a comment describing the purpose of the script.

You can move and resize the comment box within the script.

Note: If you choose the Auto-Size Height option, you cannot adjust the height of the comment.

Line Connector Node

Use the Line Connector node (in the General tab of the Palette) to make routing and administrative scripts clear and understandable.

Figure 123: The Line Connector Icon

A script can be difficult to understand and the call flow hard to follow if:

• The connecting lines between nodes are too long.

• The connecting lines go in different directions.

• The connecting lines run over other nodes and other connection lines.

The Line Connector node allows you to break and reconnect lines using one or more of its multiple input connections and single output connection. Any request coming into this node (on any one of the multiple inputs) goes to the single output connection of the line connector node.

For the Line Connector node, you define the connection labels:

Figure 124: Line Connector Properties
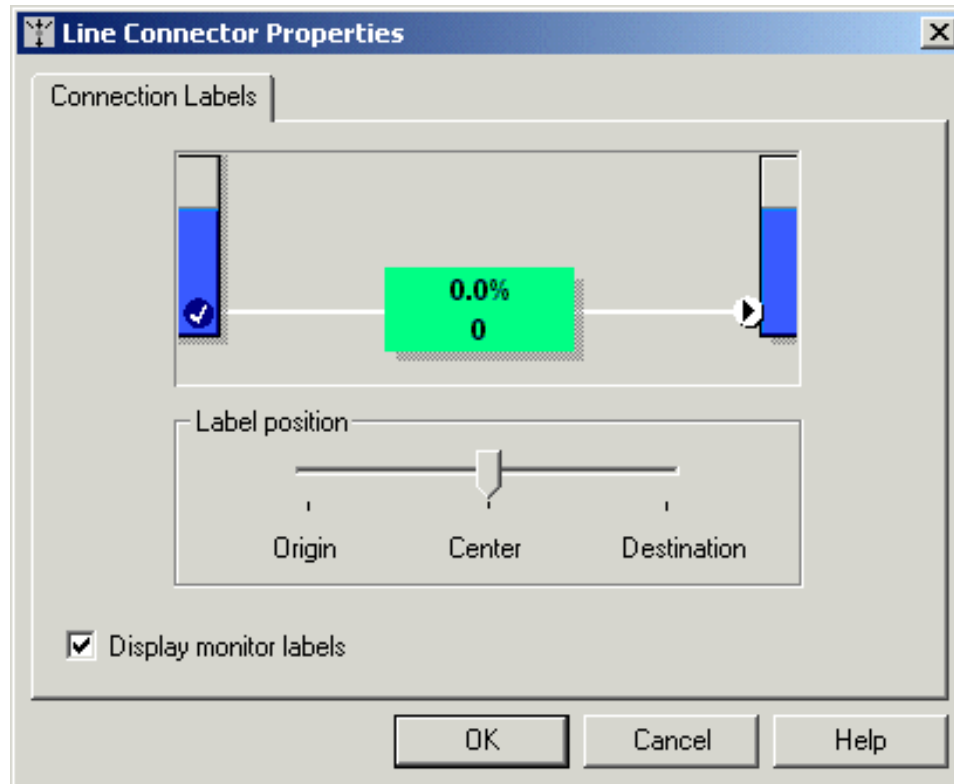
Chapter 13

Example Scripts

Example Web Collaboration Scripts

You can configure Cisco ICM software and Cisco Collaboration Server so that ICM software routes Web Collaboration requests that are processed by Cisco Collaboration Server.

Overview of Web Request Routing through ICM Software

With the Collaboration Server integrated with ICM software, ICM software can route single-session chat, multi-session chat, and blended collaboration Web requests.

The process for routing Web requests through the ICM can be divided into 5 parts, as shown in the following image:

Figure 125: Web Request Routing Process
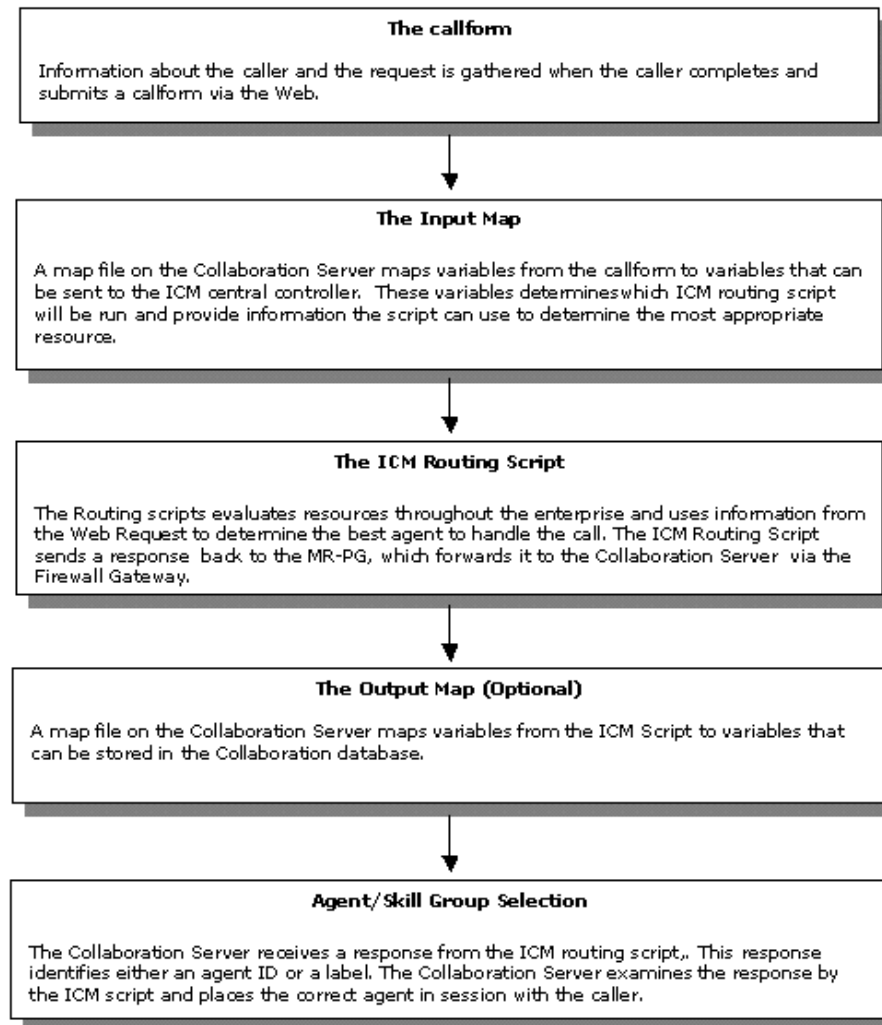
Web Requests and Media Routing Domains

Collaboration Server can take advantage of the following types of Media Routing Domains (MRDs):

- Non-voice MRDs

- The Voice MRD

For more information about MRDs, see the *Cisco ICM Software Configuration Guide*.

Non-voice MRDs

Collaboration server uses Non-voice MRDs for the following types of requests:

- Single-session chat

- Multi-session chat

- Blended Collaboration with IPCC

The Voice MRD

The Voice MRD handles these types of requests:

- Blended Collaboration with legacy ACDs (rather than IPCC)

- Blended chat with legacy ACDs (these are requests for chat, but with agent reservation on the ACD)

- Web callback and delayed callback with both ACDs and IPCC

Example - Queuing a Web Request to a Skill Group

The following script example shows how a Web request can be queued to a skill group:

Figure 126: Example - Queuing a Web Request to a Skill Group
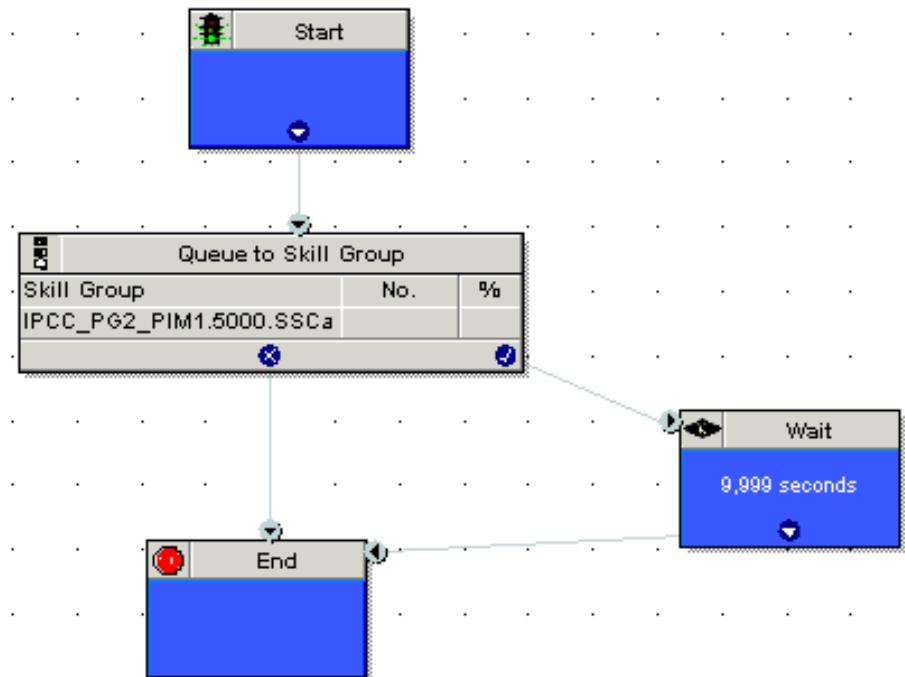
In this example:

1. The script is executed Cisco Collaboration Server processes a new single-session chat request. This script is scheduled to run for single-session chat requests. For more information on call types and scheduling Web requests routing scripts, see Example - How ICM Software Determines the Call Type for Web Request.

2. The script queues the request to a single-session chat skill group. At this time, ICM software attempts to find an available agent who is a member of that skill group. When the agent is found, ICM software returns the agent ID to the Collaboration Server.

Example - Pushing a URL to a Waiting Caller

The following script example shows how a script can push a URL to a waiting caller before the Web request is queued to a skill group:

Figure 127: pushiing URL to waiting caller
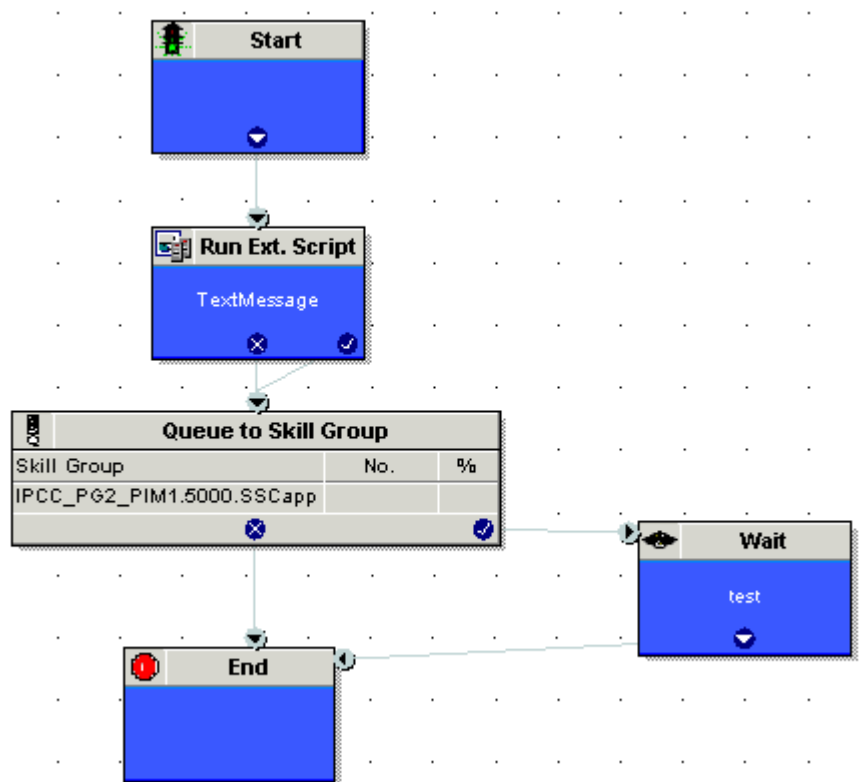
In this example, the Run External Script node pushes the selected URL to the caller's browser.

Note: For the Run External Script node to work, here must be an entry in the Network VRU list pointing to the URL map file on the Collaboration Server. For more information, see the *Cisco Collaboration Server Administration Guide*.

After the Run External Script node, the script functions just as the preceding example.

Example - Queuing Directly to an Agent

The following script example shows how a Web requests can be queued directly to an agent:

Figure 128: Queuing directly to an agent
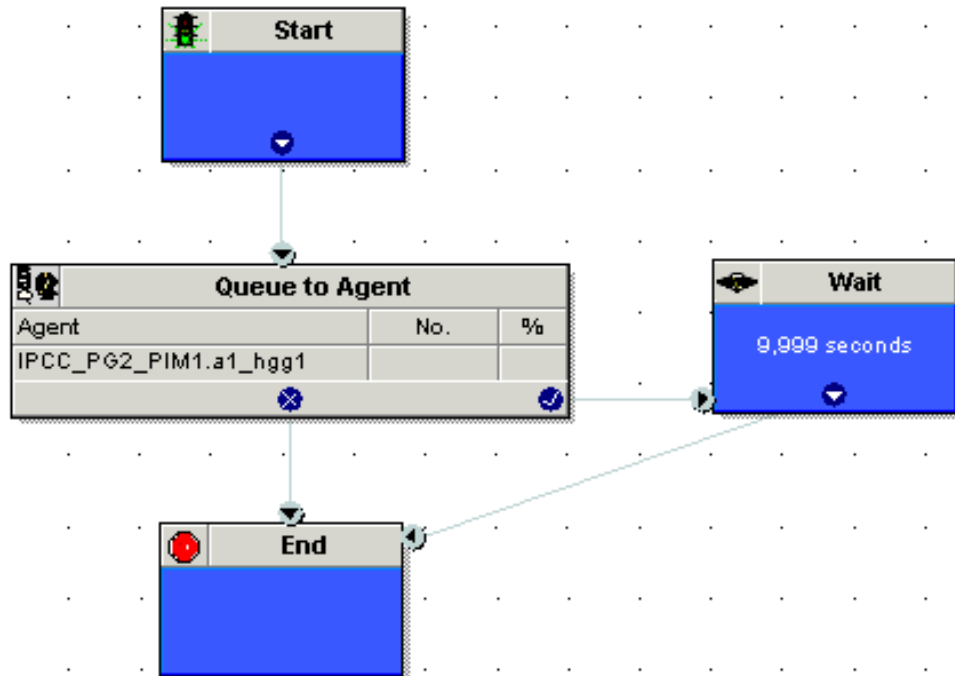
In this example:

1. This script can be scheduled to run for a particular type of Collaboration Server request.

2. The script attempts to queue the request directly to an agent using the Queue to Agent node. The Queue to Agent node uses a direct reference to the agent, as shown in its Properties dialog box:
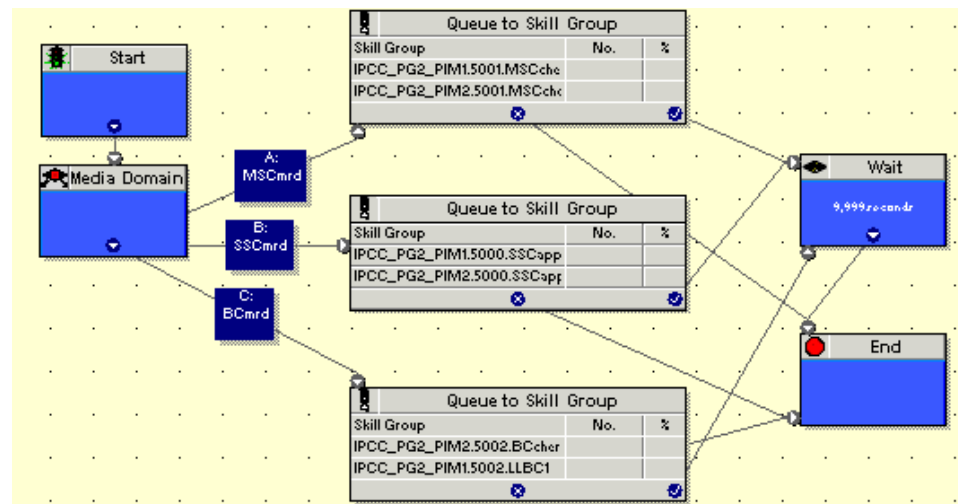
Figure 129: Direct Reference to Agent
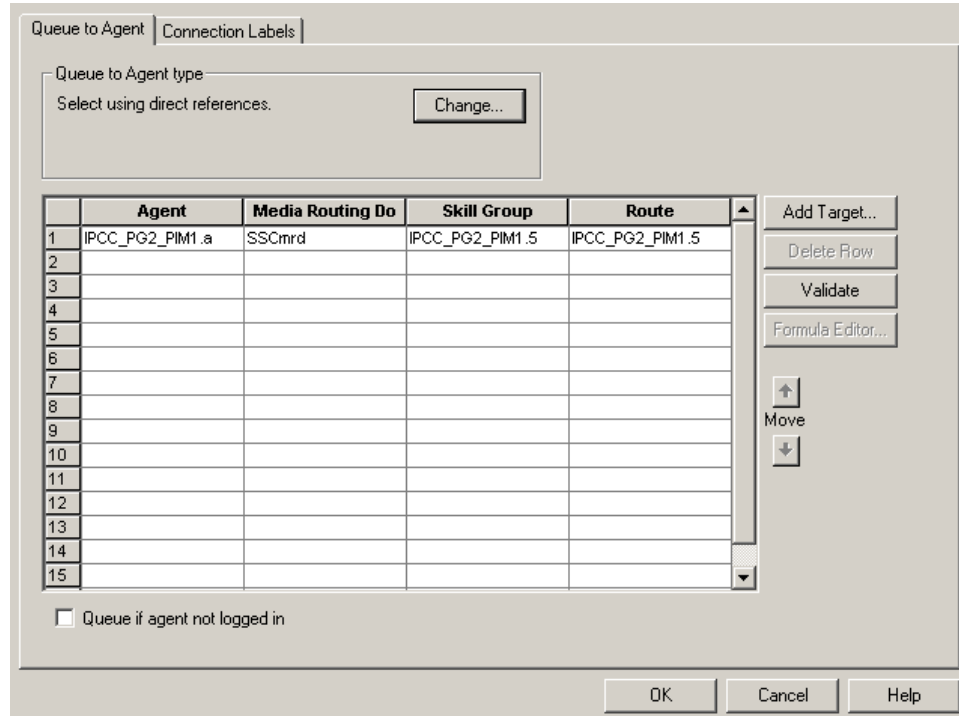
3. The script tries to do this for 9,999 seconds before ending, as defined in the Wait node.

Example - Routing Based on the Media Routing Domain

The following script example shows how a Web requests from different MRDs can be queued to different skill groups:

Figure 130: Routing based on MRD

In this example:

1. The script is executed Cisco Collaboration Server processes a new single-session chat, multi-session chat, or blended collaboration request. This script is scheduled to run for all these types of requests.

2. The script first detects the MRD of the request using the Media Routing Domain node. This node has three branches for the three possible MRDs of the request: the multi-session chat MRD (branch A), the single-session chat MRD (branch B), or the blended collaboration MRD (branch C).

3. The script queues the request to the appropriate skill group for that type of request. At this time, ICM software attempts to find an available agent who is a member of that skill group. When the agent is found, ICM software returns the agent ID to the Collaboration Server.

Example E-Mail Scripts

You can configure Cisco ICM software and Cisco E-Mail Manager so that ICM software routes e-mail messages that are processed by E-Mail Manager.

For more information about configuring ICM software and E-Mail Manager in an integrated environment, see the following documents:

- *Cisco ICM Software Configuration Guide*

- *Cisco E-Mail Manager Installation and Configuration Guide*

- *Cisco E-Mail Manager Administration Guide*

- *Cisco ICM 5.0 Multichannel Software Implementation Map*

- *Cisco ICM 5.0 Multichannel Software Overview*

Overview of E-mail Routing through ICM Software

When Cisco E-Mail Manager (CeM) is in an integrated environment, the CeM administrator can choose to defer message assignment to ICM software. The administrator determines if skill groups can initiate message routing through ICM software when the skill groups are created. The administrator then defines rules to assign messages to those ICM Routing skill groups.

In this situation:

1. A message enters CeM.

2. A rule assigns the message to an ICM Routing skill group.

3. CeM sends a request for assignment to ICM software.

   Note: In ICM Integrated with CeM, for an invoked route request the allowed responses are: Return an Agent, Return a Label, or Failure (the script could not be found or something

is wrong). CeM deals with this by placing the email message into the externalRoutingError system queue.

4. The Router runs a routing script to determine which CeM agent to assign the message to. The routing script can also determine that the message should be assigned to a local CeM routing skill group.

> **Note:**
>
> • To assign a message directly to an agent, the ICM script must use the Queue to Agent node to route e-mail messages, not the Agent node.
>
> • While ICM software can assign a message directly to a local CeM routing skill group, you would not typically design a script to do this without first trying to queue the message to a skill group or directly to an agent. If the message, based on its content, is meant to be assigned to a local CeM routing skill group, you could make that assignment directly through CeM rules, and not use ICM software.

5. Based on the routing script's determination, the ICM software instructs CeM to assign the message to a particular agent or CeM routing skill group.

6. The message is placed in the agent's or skill group's queue.

7. If the message is assigned to an agent by ICM software, it is presented to the agent in push mode; if the message is assigned to an CeM routing skill group, it is placed in that skill group's queue.

Example - Queuing E-mail to a Skill Group

The following script example shows how an e-mail message can be queued to a skill group:

Figure 131: Queuing email to skill group
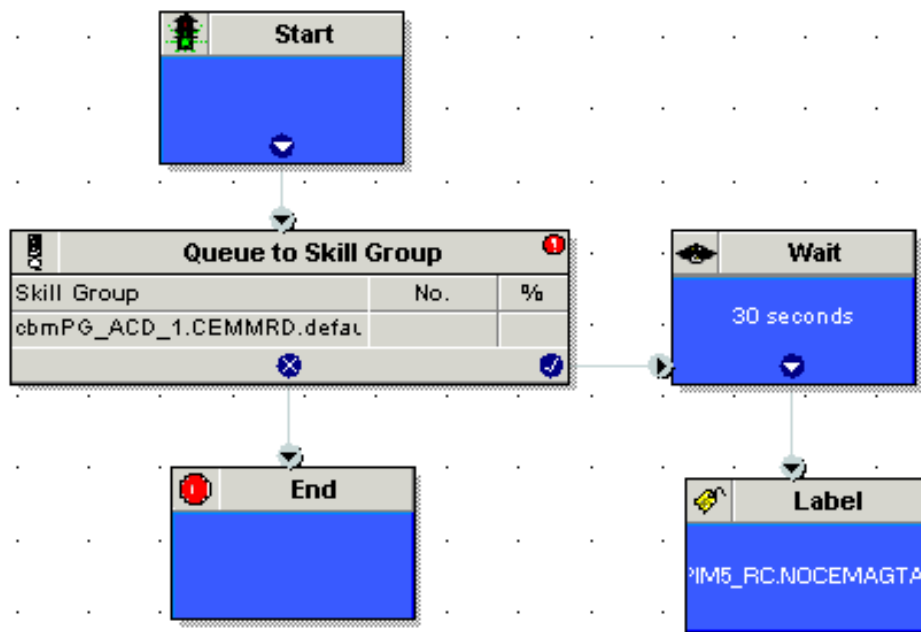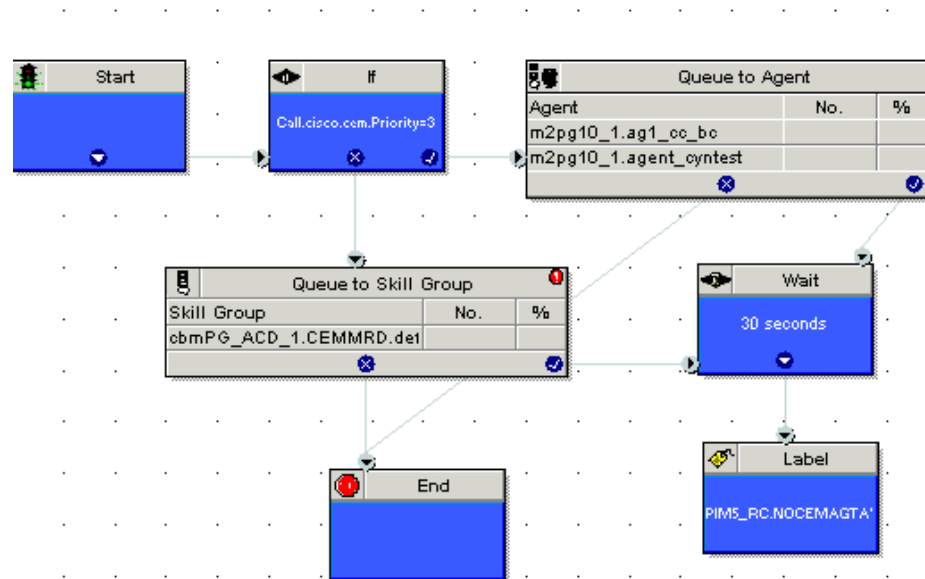

In this example:

1. The script is executed when a e-mail message is assigned by E-Mail Manager to an ICM Routing skill group. This script is scheduled to run for messages assigned to that skill group. For more information on call types and scheduling e-mail routing scripts, see Example - How ICM Software Determines the Call Type for an E-Mail Request.

2. The script queues the request to the skill group. At this time, ICM software attempts to find an available agent who is a member of that skill group. When the agent is found, ICM software instructs E-Mail Manager to push the e-mail message to that agent, so the agent can respond to the customer.

3. If no agent is found within 30 seconds, as defined in the Wait node, the script uses the Label node to return a label associated with a E-Mail Manager local skill group. E-Mail Manager then places the message in that local skill group. The message is not pushed to an agent; rather, it waits in the local skill group queue for an agent to pick it.

Example - Routing a Message Based on Priority

The following script example shows how an e-mail message can be queued directly to an agent or to a skill group, depending on the message's priority:

Figure 132: Routing based on priority

In this example:

1. The script is executed when a e-mail message is assigned by E-Mail Manager to an ICM Routing skill group. This script is scheduled to run for messages in that skill group. For more information on call types and scheduling e-mail routing scripts, see Example - How ICM Software Determines the Call Type for an E-Mail Contact.

2. The script tests the message's priority, the value of the cisco.cem.Priority variable. See Data for E-Mail Requests for more information.

3. If the message is marked "Urgent" (cisco.cem.Priority value of 3), control passes to the Queue to Agent node. That node lists two agents who are e-mail supervisors and who handle urgent messages. E-Mail Manager then pushes the message to the first of these agents who is logged in.

4. If the message is not urgent, control passes to the Queue to Skill Group node. ICM software attempts to find an available agent who is a member of that skill group. When the agent is found, ICM software instructs E-Mail Manager to push the e-mail message to that agent.

5. If no agent is found within 30 seconds, in either the Queue to Agent node or Queue to Skill Group node, as defined in the Wait node, the script uses the Label node to return a label associated with a E-Mail Manager local skill group. E-Mail Manager then places the message in that local skill group. The message is not pushed to an agent; rather, it waits in the local skill group queue for an agent to pick it.

Example Universal Queue Scripts

You can design scripts to route contacts from different media in a Universal Queue environment.

Following are example scripts for use when ICM is part of an Universal Queue environment:

- Selecting Agents from Skill Group

- Categorizing by Media Routing Domain

- Queuing to Agents

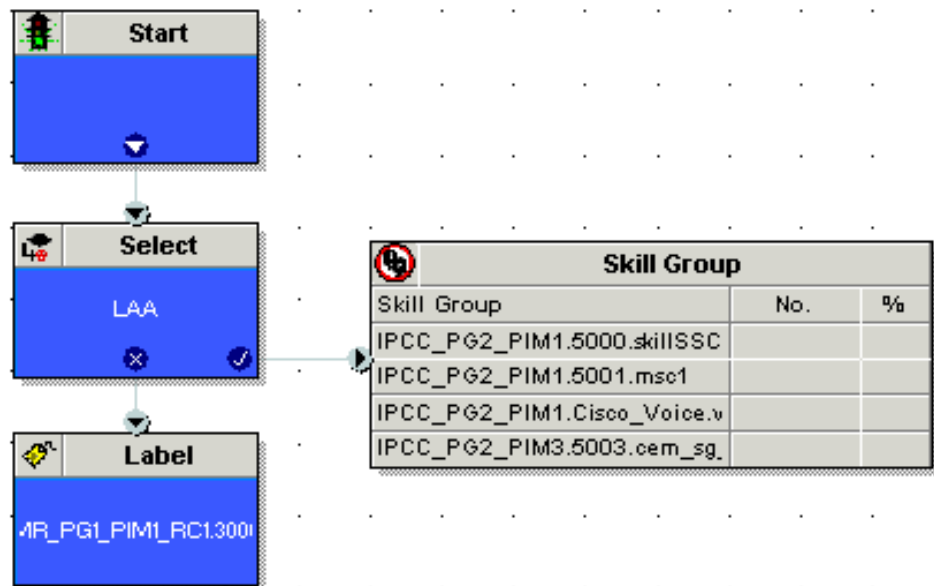These scripts are only examples; your company's needs may differ.

For more information about configuring ICM software and Universal Queue, see the following documents:

- *Cisco ICM Software Configuration Guide*

- *Cisco ICM Software: IP Contact Center Installation and Configuration Guide*

- *Cisco ICM Software: IP Contact Center Administration Guide*

- *Cisco ICM Software: IP Contact Center Laboratory Guide*

- *Cisco ICM 5.0 Multichannel Software Implementation Map*

- *Cisco ICM 5.0 Multichannel Software Overview*

Selecting Agents from Skill Groups

The following script example shows how contacts of from different channels can be routed to the Longest Available Agents in skill groups that are specific to the different channels:

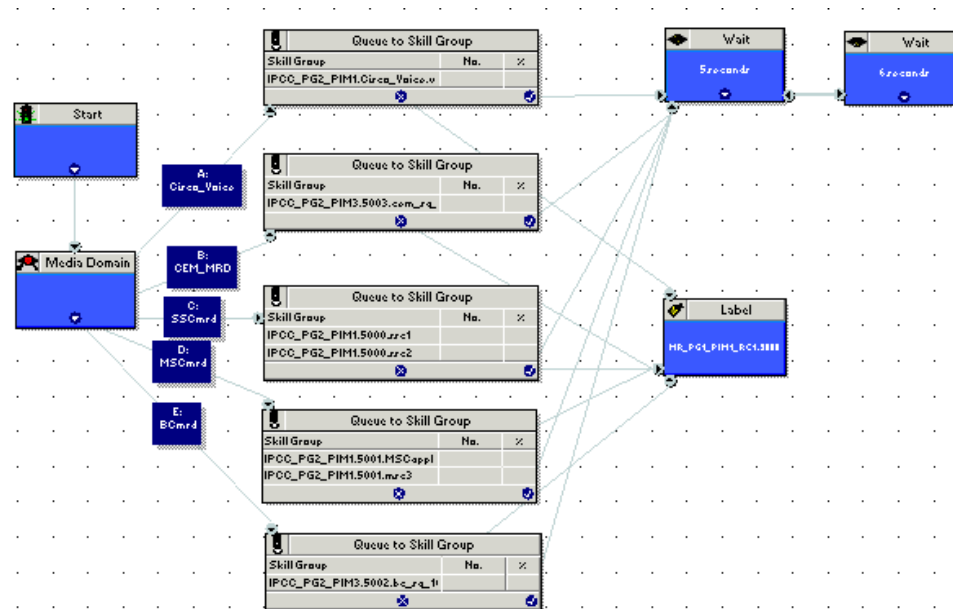*Figure 133: Selecting Agents from Skill Groups*

You would schedule this script to run for Call Types associated with contacts from the different channels. The script then selects the Longest Available Agent from the skill group in the Media Routing Domain for that channel. The agents may be logged in to different Media Routing Domains and working with contacts from different channels; the Router determines an agent's availability across channels.

Categorizing by Media Routing Domain

The following script example shows how contacts can be categorized by Media Routing Domain, then queued to skill groups specific to that Media Routing Domain:
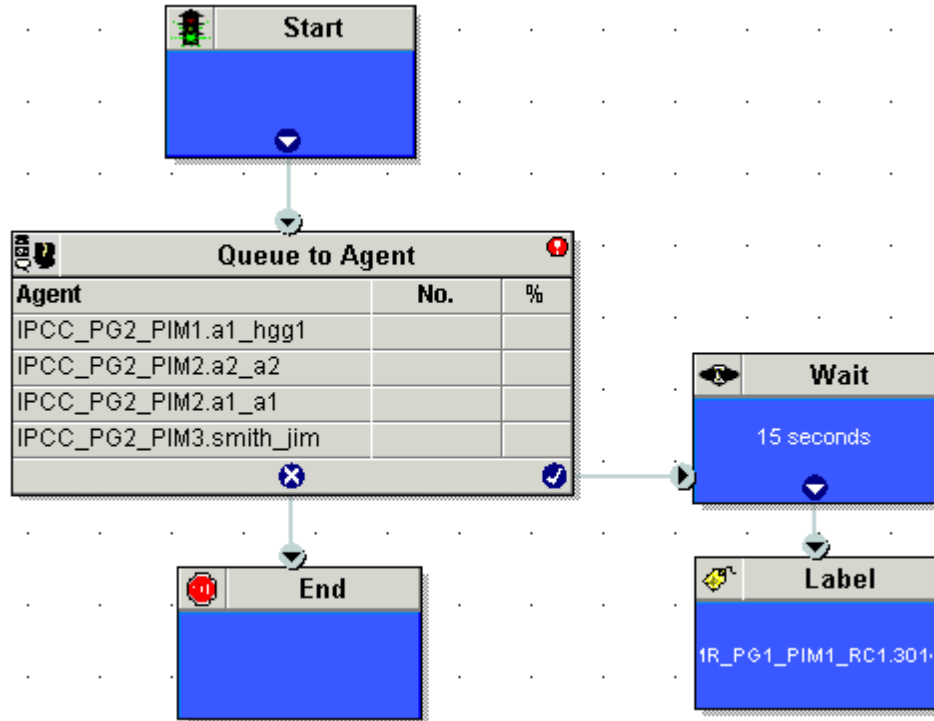
Figure 134: Categorizing by MRD

You would schedule this script to run for Call Types associated with contacts from the different channels. The script then uses the Media Routing Domain node to detect the MRD of the contact and branches to a Queue to Skill Group node that specifies skill groups specific to that MRD.

Queuing to Agents

The following script example shows how contacts from different channels can be queued to agents:

Figure 135: Queuing to Agents

You would schedule this script to run for Call Types associated with contacts from the different channels. In the Queue to Agent node, each row defined for an agent also contains a Media Routing Domain selection. The script queues the contact to the agent with the selected MRD that matches the MRD of the contact.

Example IPCC Enterprise Scripts

Following are example scripts for use when ICM is part of an IPCC Enterprise environment.

- Redirection on Ring No Answer

- Agent Transfer

- Supervisor Assist Script

These scripts are only examples; your company's needs may differ.

For more information about configuring ICM software as part of an IPCC environment, see the following documents:

- *Cisco ICM Software: IP Contact Center Installation and Configuration Guide*

- *Cisco ICM Software: IP Contact Center Administration Guide*

- *Cisco ICM Software: IP Contact Center Laboratory Guide*

> **Note:** Additional IPCC example scripts are available in the *Cisco IP Contact Center Enterprise Edition Reporting Guide*, available from **http://www.cisco.com/**.

Redirection on Ring No Answer

When configuring ICM software in an IPCC Enterprise environment, you configure Agent Desk Settings. When creating agents, you then associate each agent with one of the Agent Desk Settings you created.

One attribute of Agent Desk Settings is the **Ring no answer dialed number:**

Figure 136: Reroute on Ring No Answer

In order for calls to be routed when an agent does not answer, you must create and schedule a script for the Call Type mapped to the dialed number selected for the agent's Desktop Settings.

For example, you may schedule a simple script to run when agents do not answer that tries to select the longest available agent from a set of skill groups, and if that fails, requalifies the call to a new Call Type to have it rerouted:

Figure 137: Reroute on Ring No Answer Script
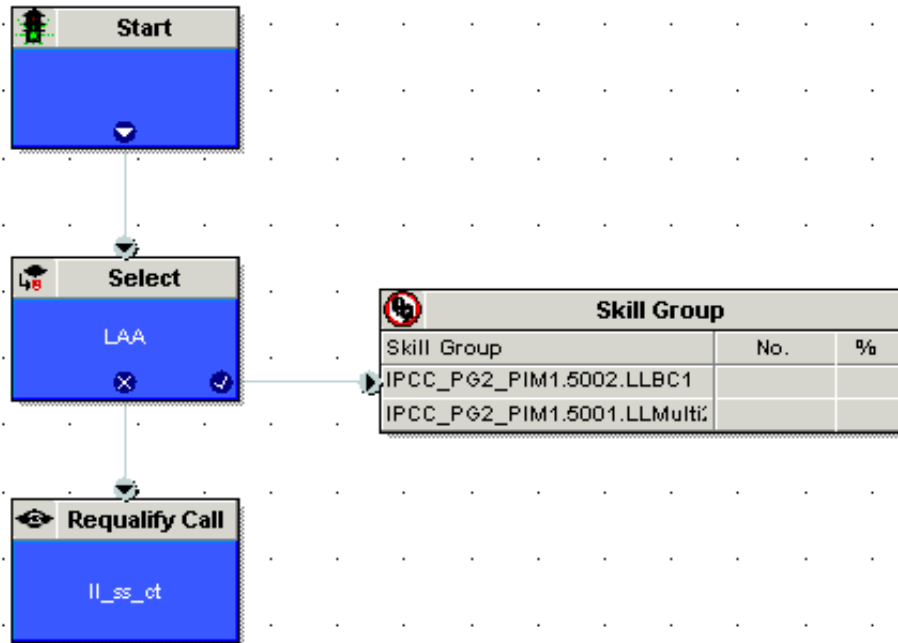
Agent Transfer

When configuring Call Types and dialed numbers in an IPCC environment, you typically may have a dialed number of *Routing_client*.9999 for internal calls from agent to agent. You would create a call type associated with the dialed number, and schedule a script for calls of this call type. The script routes internal calls, which also allows you to track and report on such calls.

For example, you may schedule a simple script to run for internal calls that tries to route directly to the agent using the Agent to Agent node, which selects the agent by peripheral and the expression Call.CallerEnteredDigits:
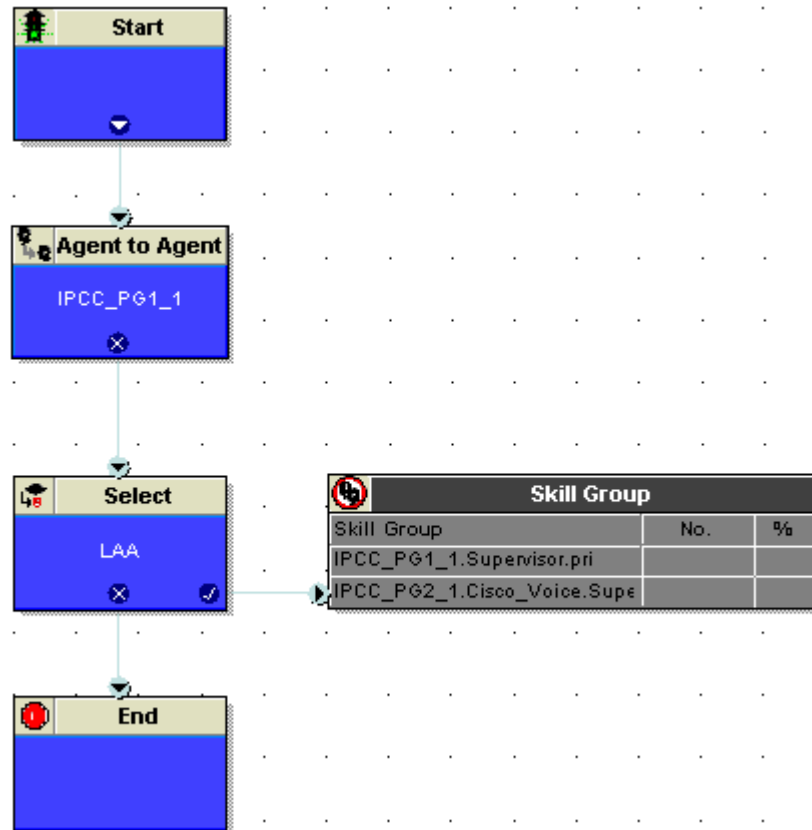
Figure 138: Agent to Agent Transfer

If the node fails, then the script tries to select the longest available agent from the set of supervisor skill groups:

Figure 139: Agent to Agent Node Script

Supervisor Assist

The following is an example of a supervisor assist script.

You must have completed the following for this script to run properly:

- Configure a supervisor Dialed Number (same as any DN, but associated with the Supervisor script).

- Configure a call type (mapped to the supervisor DN associated with Script).

Figure 140: Example Supervisor Assist Script
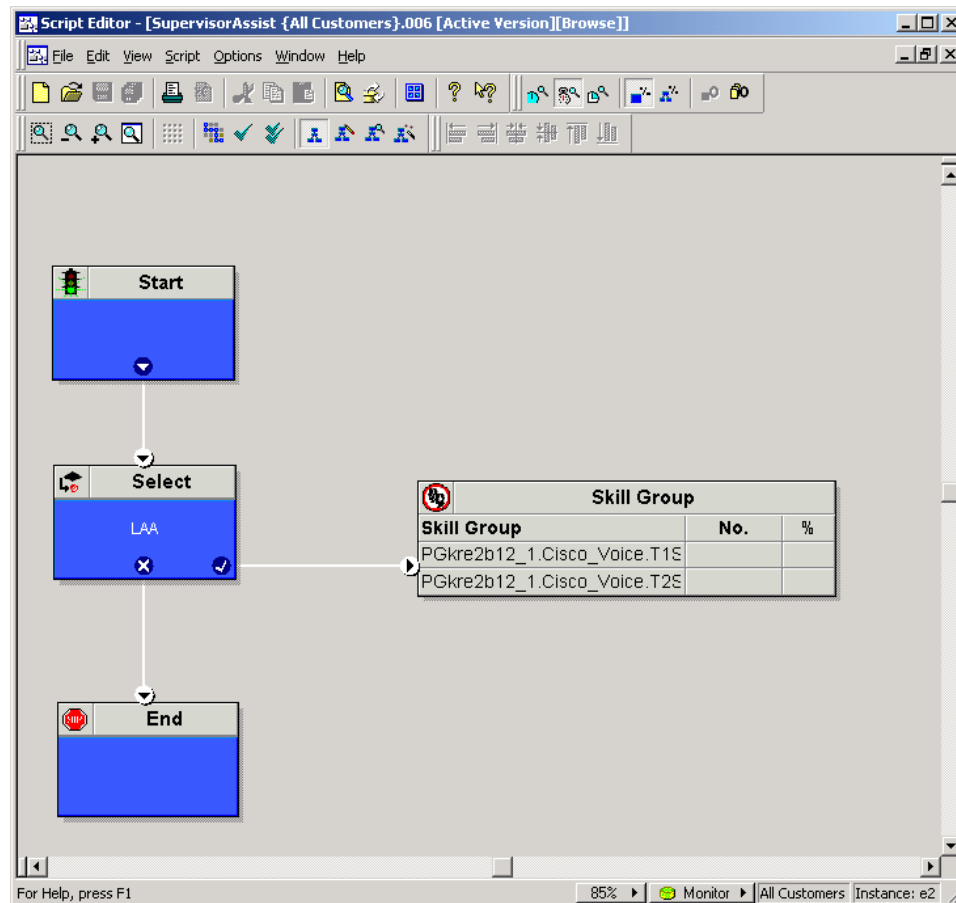
Additional Example Outbound Option Scripts

Following are example scripts for use when ICM is part of an Outbound environment:

- Setting the OutboundControl Variable and Skill Group Reservation Percentage

- Using the Dialed Number for the MR Routing Client and Routing through a Select Node to a Skill Group

- Transfer to IVR Using Outbound Option with IP IVR

- Transfer to IVR Using Outbound Option with CVP/ISN

These scripts are only examples, your needs may differ.

Setting the OutboundControl Variable and Skill Group Reservation Percentage

An administrative script is used to control the setting the OutboundControl variable and the skill group reservation percentage. These values are used by the Outbound Option Dialer to

determine which mode each skill group uses. One script can be used to control all Outbound Option skill groups, or multiple scripts can control multiple Outbound Option skill groups.

This administrative script is comprised of Start, Set, If , and End nodes. Use the Set node to set skill group variables (OutboundControl and OutboundPercent).

Set the OutboundControl variable by entering it in the Value field of the Set Properties window:

- INBOUND: Indicates that this skill group is disabled for outbound use and only takes inbound calls.

- PREDICTIVE_ONLY: Dials several customers per agent. After reaching a live contact, the Predictive Dialer transfers the customer to a live agent along with a screen pop to the agent's desk. The predictive algorithm is designed to calculate the number of lines to dial per available agent to keep agent wait time to a minimum.

- PREDICTIVE_BLENDED: Agents receive inbound calls, but could be used for an outbound call when available.

- PREVIEW_ONLY: Reserves an agent prior to initiating an outbound call and presents the agent with a screen pop. The agent might then:

  Accept the call:

  – Dials the customer and transfers the call to the agent.

  Skip the call:

  – Agent receives another customer call.

  Skip-Close the call:

  – Skips the current preview call and closes the record so it will not be called again.

  Reject the call:

  – Releases the agent. At this point, the system might deliver the agent another preview outbound call or a new inbound call.

  Reject-Close the call:

  – Rejects the current preview call and closes the record so it will not be called again.

- PREVIEW_BLENDED: Agents receive inbound calls, but could be used for an outbound preview call when available.

- PREVIEW_DIRECT_ONLY: Agents can only place outbound calls and hear ring tones, such as phone ringing or busy signal.

- PREVIEW_DIRECT_BLENDED: Agents can receive inbound calls, place outbound calls, and hear ring tones, such as phone ringing or busy signal.

- PROGRESSIVE_ONLY: Similar to PREDICTIVE_ONLY; however, lines to dial per agent is not calculated—users configure a fixed number of lines that will always be dialed per available agent.

- PROGRESSIVE_BLENDED: Similar to PREDICTIVE_BLENDED, but a fixed number of lines will always be dialed per available agent.

Note: If the administrative script (where the OutboundControl variable or reservation percentage is set) is running, but the modes/percentages are not being updated at the Dialer, do the following: – Make sure that the skill group being controlled is the base skill group, and not the primary or secondary skill groups. Although agents may be logged into just the primary or secondary skill group, the outbound control variable must always be set on the base skill group. – Verify that the outbound control variable mode is spelled correctly.

Set the OutboundPercent variable in the same administrative script. Select the OutboundPercent variable in the Set Properties window and enter the agent percentage in the Value field. This variable controls the percentage of agents, which are logged into a particular skill group, which should be used for outbound dialing. For example, if there are 100 agents logged into skill group N, and the OutboundPercent variable is set to 50%, 50 agents would be allocated for outbound dialing.

Note: This variable does not allocate specific agents for outbound dialing, just a total percentage.

The following diagram displays a very simple administrative script where both the OutboundControl variable and the outbound percentage are set for a skill group. A script in a production call center would typically be more complex, perhaps changing these variables due to time of day or service level.

Figure 141: Setting the OutboundControl Variable and Skill Group Reservation Percentage
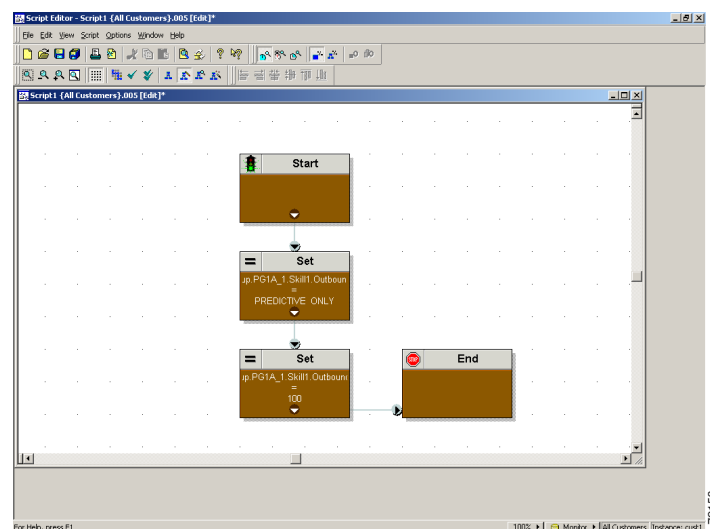
Using the Dialed Number for the MR Routing Client and Routing through a Select Node to a Skill Group

The following diagram displays a sample routing script that uses the dialed number for the MR routing client and routes through a Select node to a previously configured skill group. Add additional DN nodes to route to agents in additional skill groups as you must maintain a 1:1 ratio of dialed numbers to skill groups.

Warning: The outbound percentage in the Set node must be set to a value other than zero (0) or it appears to the node that there are no agents assigned to the Outbound Option and no outbound calls will be made.

Warning: Do not use Select Route nodes, multiple skill groups, or sevices.

Note: Translation routes are not used in the IPCC System PG, so routing scripts using this PG do not need to use this object.

Configuring Queue to Agent Node

- Right-click the Queue to Agent node and select **Properties**.

- Click **Change** in the "Queue to agent type" section.

- Click **Lookup agent reference by expression**, then click **OK**.

- Enter the agent expression Call.PreferredAgentID.

- Make sure the Peripheral column is left blank.

- Select the enterprise skill group.

- Select the enterprise route.

- Click **OK** to save the Queue to Agent node.

Figure 142: Queue to Agent node Properties
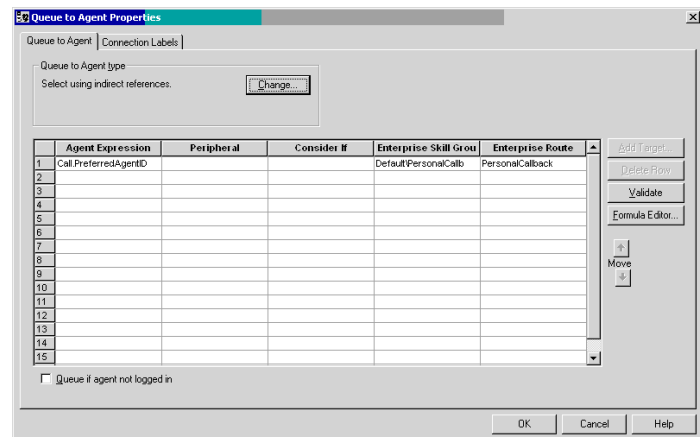
> **Note:** Lines connecting objects cannot appear on top of objects and therefore, partially display under the objects; for example, the line connecting the "X" (output terminal failure) on the Select object to the End object runs partially under the Select object.

Figure 143: Using the Dialed Number for the MR Routing Client and Routing through a Select Node to a Skill Group
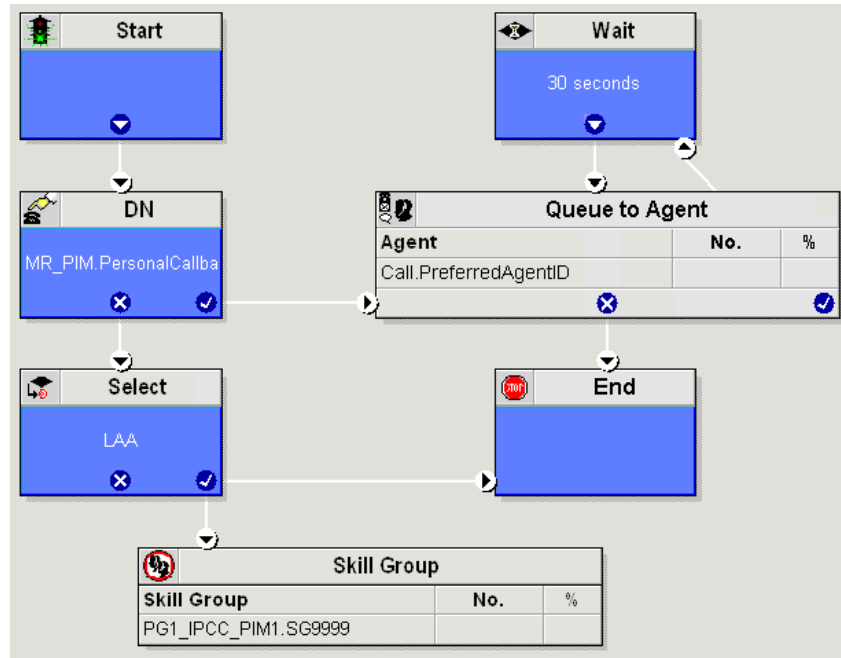
Transfer to IVR Using Outbound Option with IP IVR

The following diagram displays a routing script for a transfer to IVR campaign using Outbound Option with IP IVR.

Figure 144: Transfer to IVR Using Outbound Option with IP IVR
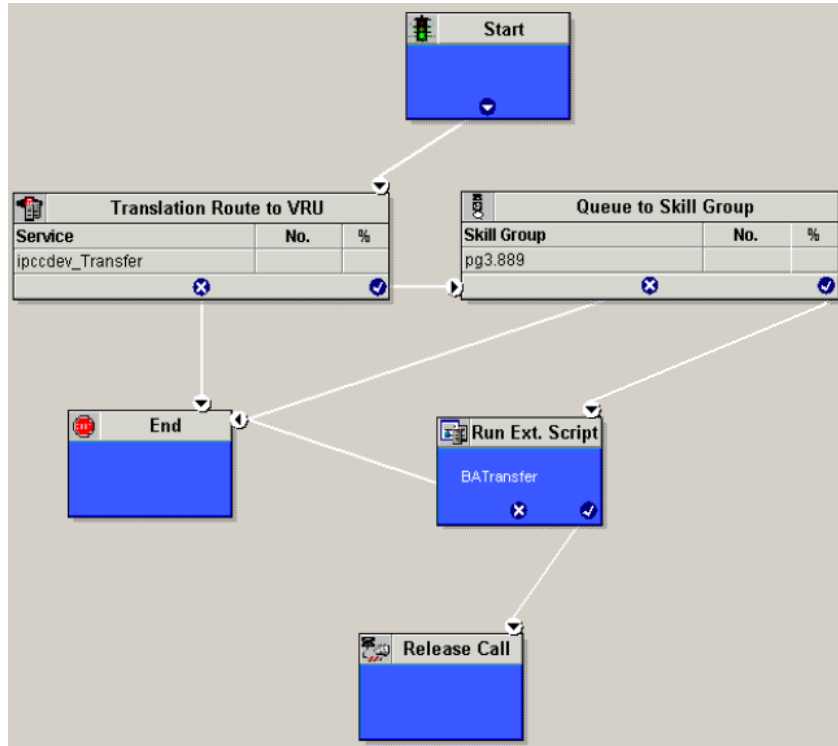
Transfer to IVR Using Outbound Option with CVP/ISN

The following diagram displays a routing script for a transfer to IVR campaign using Outbound Option with CVP/ISN.

Figure 145: Transfer to IVR Using Outbound Option with CVP/ISN
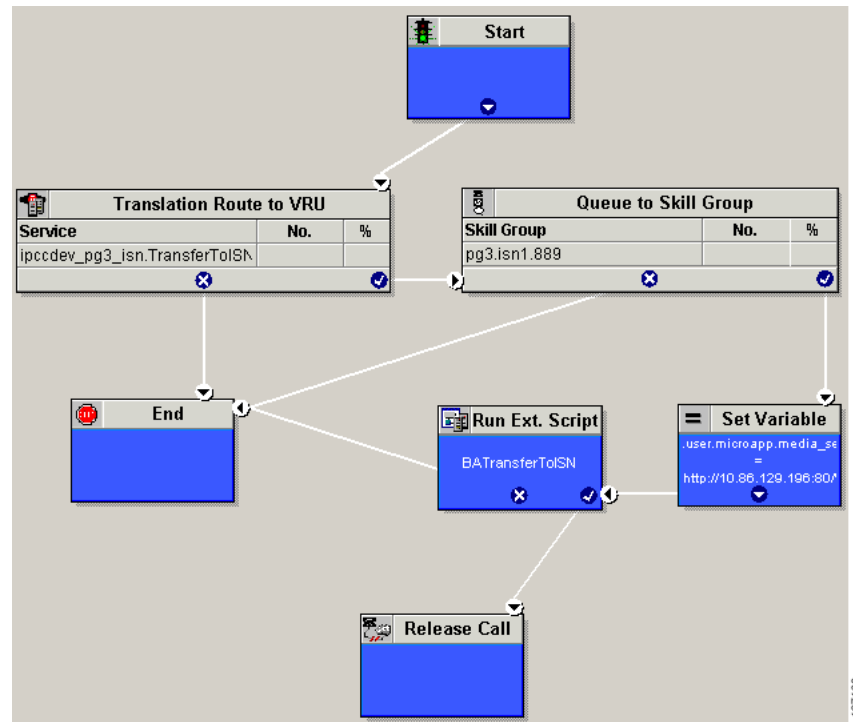
Estimated Wait Time (EWT) Queueing

The Script Editor's built-in Minimum Expected Delay (MED) calculation does not apply to IPCC. Instead, you can use a formula on an IF node to determine the MED between two skillgroups.

When to use EWT Queing

Look ahead skillgroup checking of a secondary choice skillgroup should be done if there are:

- no available agents in the primary or first choice skill group

- the call has already been sent to a IVR queue point

- the call has been queued to the primary or first choice skill group

- an adequate amount of time has elapsed (customer and call dependent)

After the above conditions have been met, if there are no agents available in the subsequent or secondary choice skillgroup, then Minimum Expected Delay (MED) of the subsequent skillgroup should be calculated.

Example EWT/MED Script Formula

The example formula below can be used in an IF node to determine the EWT/MED for a call that can be routed to either a primary or secondary skill group. The call is routed to the skill group with the lowest MED.

```
EntSkill.Default\EnterpriseSkillgroupSec.Avail ||

(EntSkill.Default\EnterpriseSkillgroupPri.LoggedOn>0)

&&

(EntSkill.Default\EnterpriseSkillgroupSec.LoggedOn>0)

&&

((SkillGroup.CCM_PG_1.CCM_SG.sec.RouterCallsQNow+1)*

(ValidValue(EntSkill.Default\EnterpriseSkillgroupSec.
AvgHandledCallsTimeTo5,<constant>))

/EntSkill.Default\EnterpriseSkillgroupSec.LoggedOn)

<

((SkillGroup.CCM_PG_1.CCM_SG.pri.RouterCallsQNow+1)*

(ValidValue(EntSkill.Default\EnterpriseSkillgroupPri.
AvgHandledCallsTimeTo5,<constant>))

/EntSkill.Default\EnterpriseSkillgroupPri.LoggedOn)
```

EWT/MED Script Explained

If there are agents available, then no there is no need to calculate the MED and the IF node should return TRUE value. If there are no agents available, then the MED of the secondary skillgroup should be calculated.

The formula first checks to make sure agents are logged into both the primary (already queued to) and secondary (not queued to) skillgroups. If there are agents logged into both skillgroups, then the minimum expected delay of the secondary skillgroup is compared to the minimum expected delay of the primary skillgroup to see which is less.

If the secondary skillgroup MED is less, then the call should be queued at the new secondary skillgroup. (The call can either be kept in the initial skillgroup or taken out of the initial skillgroup when it is queued to the secondary skillgroup.)

However, if the secondary skillgroup's MED is not less than the primary skillgroup's MED, then the call should remain queued at the initial skillgroup only. +1 is used to add this new call into the algorthium.

Since the `RouterCallsQNow` variable is only applicable at a Skillgroup and not an Enterprise Skillgroup level, then the aggregate of all the skillgroups within the Enterprise Skillgroup `RouterCallsQNow` field must be multiplied by the Enterprise SkillGroup `AvgHandledCallsTime`.

The `<constant>` value is site dependent. It should be a value that represents the average Handle time of a call. This constant will only be used if it is at the start of the day and the proper `AvgHandledCallsTime` has not been calculated yet.

Chapter 14

Hosted Scripting Considerations

This section covers scripting consideration to use in an IP Contact Center Hosted-Edition system. For more details on scripting in an ICM/NAM environment see the NAM Installation Guide.

This chapter contains the following topics:

NAM Script Configuration

Scripting on the NAM only requires one Dialed Number script per Customer Instance. However, your design may include more.

The NAM routing script sends the dialed number to the correct ISN Media Server for treatment, then ultimately to the ICM Gateway of the CICM instance for the particular Dialed Number.

Figure 146: Example NAM Routing Script
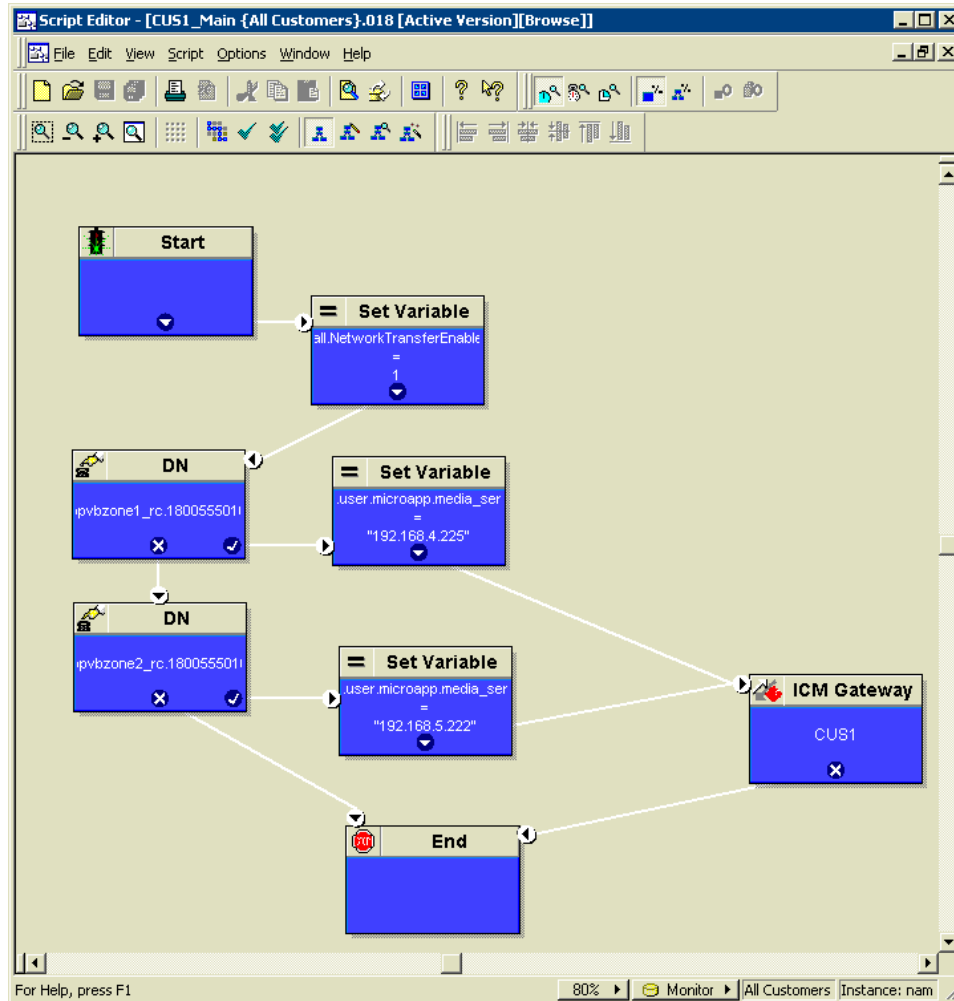
The first script node after the Start should be a "set variable" node that contains the following:

- Object Type — Call

- Object — (no selection)

- Variable — NetworkTransferEnabled

- Value — 1

Figure 147: NetworkTransferEnabled Set Variable Node
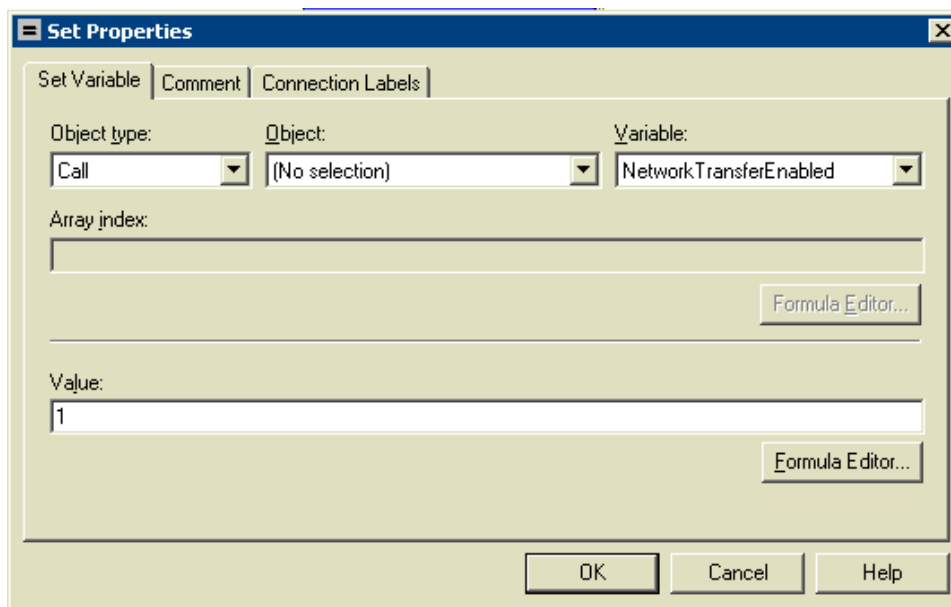
After you have set the variable for NetworkTransferEnabled you can create a Dialed Number (DN) Node to route to a particular CICM Instance Routing Client.

If the Dialed Number node is true, then you typically send the call to the ISN Media Server using another Set Variable node as described below.

The first script node after the star should be a "set variable" node that contains the following:

- Object Type — Call

- Object — (no selection)

- Variable — user.microapp.media_server

- Value — the IP address of the ISN Media Server, enclosed in quotes, for example "192.168.10.11"

Figure 148: ECC Variable user.microapp.media_server Set Variable Node
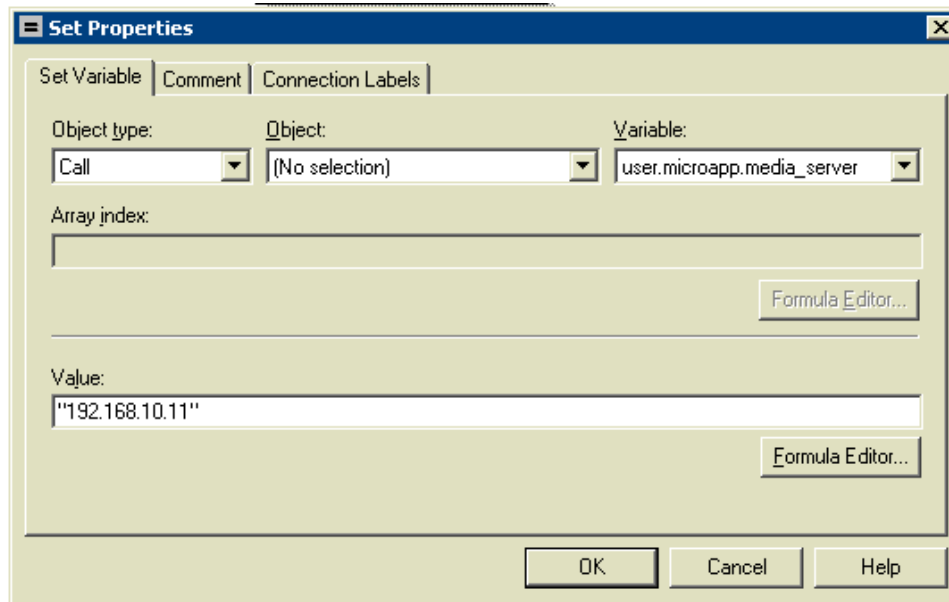
At this point, scripting is conducted on the ISN. Consult the *Cisco Internet Service Node (ISN) Configuration and Administration Guide, Appendix B* for details on scripting in ISN.

Next, the call is sent to the ICM gateway of a particular customer instance.

If multiple DNs are used, then you can send the call to a different ISN Media Server based on the different DN.

Finally, the call is sent to the ICM gateway of a particular customer instance. Add an ICM Gateway Node and select the customer to which to route the call.

Figure 149: ICM Gateway Node
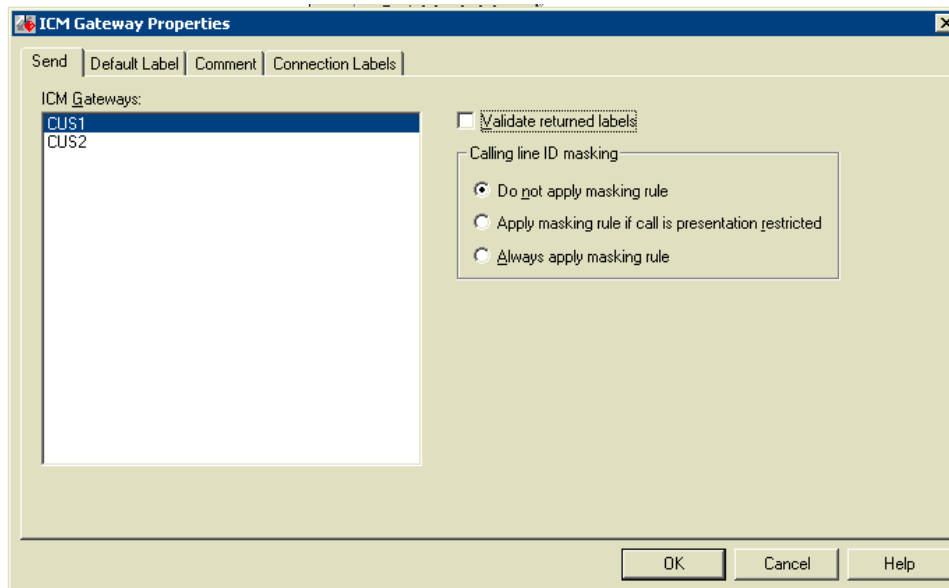
Note: Do NOT select the Validate returned labels checkbox. If you do so, you will have to provision the NAM with all of the labels that exist on every customer instance.

CICM Scripting Considerations

Note:

- See the *IPCC Administration Guide for Cisco IPCC Enterprise Edition, Cisco IPCC Enterprise Edition Release 7.0(0)* for general scripting requirements for a CICM Instance.

- Consult the *Cisco Customer Voice Portal (CVP Configuration and Administration Guide, Appendix B* for CVP additional scripting considerations.

Create CICM VRU Scripts

VRU Scripts differ from ICM routing scripts. A configured VRU Script runs only when ICM instructs it to do so from an ICM routing script. A VRU Script on ICM is the configured record for the VRU script that resides on the ISN. A VRU Script executes to collect digits, play hold music, or perform many other common IVR functions.

Step 1 From the Configuration Manager, select **Targets > Network VRU Script > Network VRU Script List**.

The Network VRU Script List dialog box appears.

Step 2 On the Attributes tab, enter the following configuration information for the BasicQ script:

- **If:** Network VRU

  Then: Enter isnvru.

- **If:** VRU Script Name

  Then: Enter script name (for example, BasicQ).

- **If:** Name

  Then: Enter the script file name (for example, BasicQ.aef).

- **If:** Timeout (seconds)

  Then: Enter 180.

- **If:** Configuration param

  Then: Leave blank.

- **If:** Customer

> **Then:** Select the same ICM customer you selected for Call Type from the drop-down list.

Step 3   Select the Interruptible check box.

Step 4   Click **Save** and then click **Close**.

CICM VRU Script Considerations

Use the CICM Script Editor SendToVRU node to connect the call to the Network VRU.

For more information about creating scripts, see *ISN Configuration and Administration - Chapter 2, "Using NAM/ICM with the ISN IVR Solution"*.

Note: A RunVRU Script or Queue node is an "implicit" SendToVRU node, although error handling is easier if the explicit "SendToVRU" node is used.

RONA and ISN

When using IPCC with ISN, the ICM Router Requery function is used to take the call away from the non-answering agent and requeue it for service.

There are two places in which configuration must occur:

- In the Agent Desk Settings - the **Ring no answer dialed number** field must be blank, and a **ring no answer time** must be set.

- In the example script Scripting for RONA:

  – The Queue node for the skill group that selects the first agent must have Target Requery enabled.

  – Raise the priority of the call so that it goes to the front of the queue.

Scripting for RONA

When scripting for RONA, you should Enable Requery on the node in the script that selects the first agent. Depending on the type of node used, the Requery mechanism selects a new target from the available agents or requires additional scripting. The IPCC/ICM Scripting and Media Routing Guide describes how Requery works for the different nodes.

Figure 150: Changing the Queue Type to Enable Requery
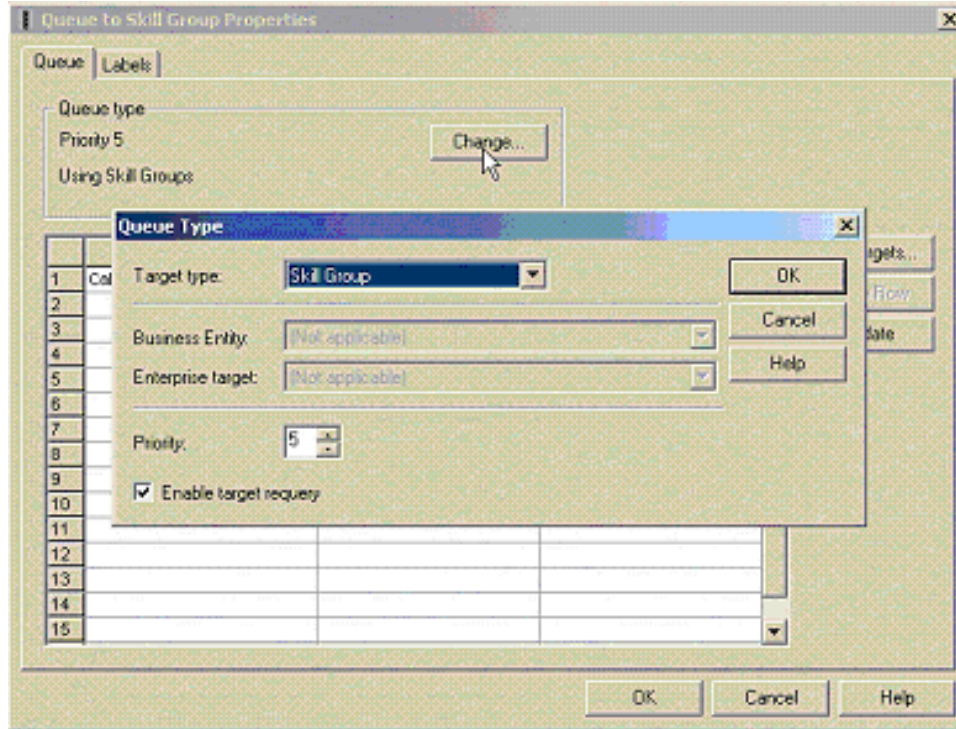
Figure 151: RONA Scripting in IPCC Hosted Edition
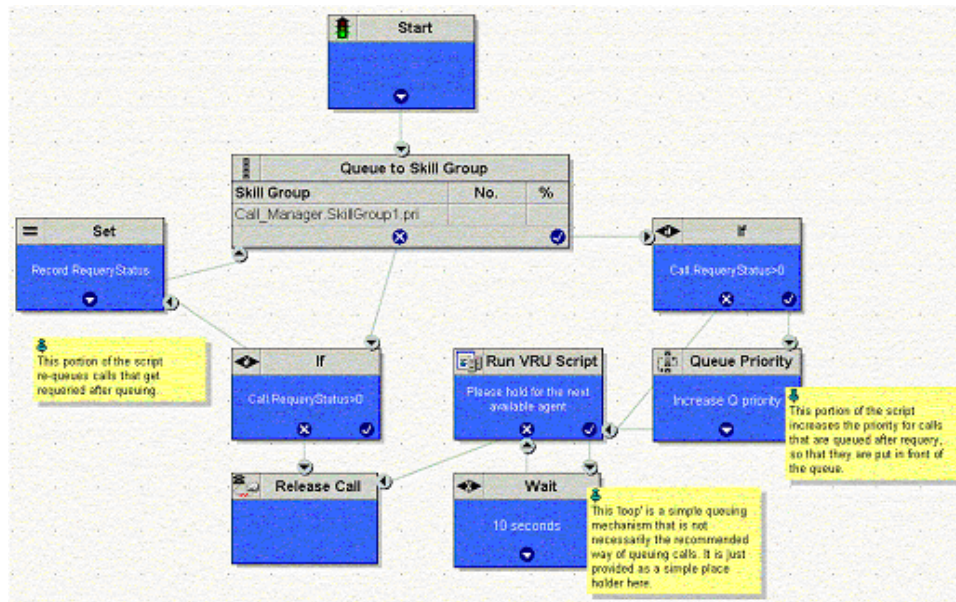
If there is an available agent, the Queue node selects the longest available agent from the skill groups configured. If there is no available agent, the script then queues the call with a priority set in the node and continues down the success exit of the node.

When an agent becomes available the ICM always selects the longest queued call from the ones with the highest priority. The RONA mechanism works as follows:

1. The Queue node selects an agent.

2. If the agent does not answer the call, then the script exits through the failure terminal of the Queue node.

3. The If node tests the RequeryStatus variable. If it has value of greater than zero, this is a requery call and the script re-queues the call.

4. In the Scripting for RONA example above, it also sets a flag using a call variable for reporting purposes.

5. Assuming that there are no agents available, the Queue node immediately exits through the success terminal.

---

6. **If:** If this is a required call.

   Then: It increases the Queue Priority of the call so that it is handled before any other calls in queue.

---

7. It then enters the normal wait loop with RunScripts.

Chapter 15

Troubleshooting

Calls not successfully routed to agent

Symptom:

In Monitor mode, Script Editor shows calls coming in and being routed; however, no calls are actually reaching agents.

Message:

No Message

Cause:

If a label is misconfigured or missing for a phone to which an agent is assigned, when this agent is assigned a call, additional calls are not processed and successfully routed to other agents. This can apply to calls routed through the Agent, Skill Group, Enterprise Skill Group, and Queue nodes

Action:

Ensure that labels are properly configured for all agent phones

Duplicate email routing requests

Symptom:

For every request E-Mail Manager sends to the ICM Router, it appears to the Router as if two requests are sent. This affects reporting, system lode, and counts displayed when viewing scripts in Monitor mode.

Message:

No Message

Cause:

The script processing requests from E-Mail Manager contains an erroneous entry for a label and no agents are available to be assigned the new request.

Action:

Ensure that labels are entered correctly in the script.

Glossary

administrative script

An administrative script is an object you create through ICM software that performs background processing, such as setting values for variable or collecting data from external databases. Data set by an administrative script can be used by routing scripts. Administrative scripts run on a schedule you determine.

call control variables

Call control variables provide information about the current contact that is being routed by the script. Call control variables include information about where the route request came from, contact classification data, and data to be passed to the peripheral that receives the contact.

caller-entered digits

Caller-entered digits are numbers entered by the caller in response to prompts. For example, a caller may enter a number to indicate the type of service needed.

calling line ID (CLID)

The calling line ID (CLID) is a string that represents the telephone number from where the call originated. The CLID is sometimes referred to as the ANI (Automatic Number Identification).

calling line ID (CLID)

The calling line ID (CLID) is a string that represents the telephone number from where the call originated. The CLID is sometimes referred to as the ANI (Automatic Number Identification).

call type

A call type is the first-level category of a contact determined by data about the contact. You associate a script with a call type. When a contact of a certain call type is received, the associated script runs on that contact.

categorization

Categorization is the process of classifying a contact based on certain data associated with the contact. Through categorization, a script can determine how to best process a contact.

dialed number (DN)

A dialed number (DN) is a string that represents the telephone number dialed by the caller; for example, 8005551212 might be a dialed number.

expanded call context (ECC) variables

Expanded Call Context (ECC) variables store values associated with the contact.

expanded call context variables

Expanded Call Context (ECC) variables store values associated with the contact.

formula

A formula consists of one or more expressions that ICM software evaluates to produce a value that it can use for subsequent script processing.

media classes

Media classes are the types of contacts that ICM software can route in an integrated contact center system. Supported media classes include: Voice for Cisco ICM Software; Single Session Chat for Cisco Collaboration Server; Multi-Session Chat for Cisco Collaboration Server; Blended Collaboration for Cisco Collaboration Server and Cisco Media Blender; E-mail for Cisco E-Mail Manager.

media routing domain

A Media Routing Domain (MRD) is a collection of skill groups and services associated with a specific communication channel. For example, E-Mail Manager uses the ICM software uses a MRD to route a task to an agent who is associated with a skill group and a particular channel. MRDs are defined in ICM configuration and have unique IDs across the enterprise.

network target

A network target is an end point on the network to which a script can send a contact.

network VRU

A Network VRU supports ICM software's service control interface. An ICM routing script can divert a call to a Network VRU and instruct the VRU to perform specific processing before ICM software determines the final destination for the call.

route

A value returned by a routing script that maps to a target at a peripheral; that is, a service, skill group, agent, or translation route to a label.

routing script

A routing script is an object you create through ICM software that: (1) Examines information about a contact, whether the contact is made through a phone call or other medium, and uses that information to classify the contact. (2) Analyzes the state of the contact center and determines that best way to process the contact. (3) Directs the contact to the most appropriate destination, such as an agent, a skill group, or an announcement.

routing target

A routing target is an entity to which ICM software can route a contact. The routing target receives the contact and processes it accordingly.

service

A particular type of processing that the customer requires. For example, a customer may require help installing software, and thus the Technical Support service. A service is associated with a peripheral.

skill group

A collection of agents at a single contact center who share a common set of competencies.

skill target

A skill target is an entity at a peripheral or in the enterprise to which ICM software can route a contact.

target

A target is an entity to which ICM software can route a contact. The routing target receives the contact and processes it accordingly.

target requery

Target Requery is a script node feature that addresses routing failures due to transient failures in the network (such as network congestion). If the determined destination for a contact is available but not reachable, Target Requery attempts to find a different valid destination.

target set

A target set is a list of possible targets. During script processing, the actual target is chosen from the set by the preceding node on the script branch, a Select node or Distribute Node.

translation route

A translation route is a target at a peripheral that does not map to a specific service, skill group, or agent. When a contact arrives with the trunk group and DNIS that correspond to a translation route, the Peripheral Gateway (PG) is responsible for determining the ultimate target.

universal queue

Universal Queue is the term used to describe ICM software's ability to route requests from different channels to agents who work with customer contacts in multiple media. With Universal

Queue, ICM software treats requests from different channels as part of a single queue. Routing scripts can send requests to agents based on business rules regardless of the channel from which the request came. For example, ICM software can route phone, single session chat, and e-mail message requests to an agent who works with all these channels, based on the agent's skills and current tasks.

user variables

User variables are variables you create to serve as temporary storage for values you can test with an If Node. For example, you could create a user variable called usertemp to serve as a temporary storage area for a string value used by an If node.

variable

A variable is a named object that holds a value. You use variables in formulas to select targets and help in call tracking.

VRU

A VRU, or Voice Response Unit, is a telecommunications device, also called an Interactive Voice Response Unit (IVR), that plays recorded announcements and responds to caller-entered touch-tone digits. A VRU can also be equipped with Automatic Speech Recognition (ASR) or Text-to-Speech (TTS) capabilities.

VRU MicroApps

A VRU, or Voice Response Unit, is a telecommunications device, also called an Interactive Voice Response Unit (IVR), that plays recorded announcements and responds to caller-entered touch-tone digits. A VRU can also be equipped with Automatic Speech Recognition (ASR) or Text-to-Speech (TTS) capabilities.

Index