Avaya Interaction Center

Agent Script Workflow Reference

Release 7.3.x
May 2015

Notice

While reasonable efforts have been made to ensure that the information in this document is complete and accurate at the time of printing, Avaya assumes no liability for any errors. Avaya reserves the right to make changes and corrections to the information in this document without the obligation to notify any person or organization of such changes.

Documentation disclaimer

"Documentation" means information published by Avaya in varying mediums which may include product information, operating instructions and performance specifications that Avaya may generally make available to users of its products and Hosted Services. Documentation does not include marketing materials. Avaya shall not be responsible for any modifications, additions, or deletions to the original Published version of documentation unless such modifications, additions, or deletions were performed by Avaya. End User agrees to indemnify and hold harmless Avaya, Avaya's agents, servants and employees against all claims, lawsuits, demands and judgments arising out of, or in connection with, subsequent modifications, additions or deletions to this documentation, to the extent made by End User.

Link disclaimer

Avaya is not responsible for the contents or reliability of any linked websites referenced within this site or documentation provided by Avaya. Avaya is not responsible for the accuracy of any information, statement or content provided on these sites and does not necessarily endorse the products, services, or information described or offered within them. Avaya does not guarantee that these links will work all the time and has no control over the availability of the linked pages.

Warranty

Avaya provides a limited warranty on Avaya hardware and software. Refer to your sales agreement to establish the terms of the limited warranty. In addition, Avaya's standard warranty language, as well as information regarding support for this product while under warranty is available to Avaya customers and other parties through the Avaya Support website:

http://support.avaya.com or such successor site as designated by Avaya. Please note that if you acquired the product(s) from an authorized Avaya Channel Partner outside of the United States and Canada, the warranty is provided to you by said Avaya Channel Partner and not by Avaya.

"Hosted Service" means a hosted service subscription that you acquire from either Avaya or an authorized Avaya Channel Partner (as applicable) and which is described further in Hosted SAS or other service description documentation regarding the applicable hosted service. If you purchase a Hosted Service subscription, the foregoing limited warranty may not apply but you may be entitled to support services in connection with the Hosted Service as described further in your service description documents for the applicable Hosted Service. Contact Avaya or Avaya Channel Partner (as applicable) for more information.

Hosted Service

THE FOLLOWING APPLIES IF YOU PURCHASE A HOSTED SERVICE SUBSCRIPTION FROM AVAYA OR AN AVAYA CHANNEL PARTNER (AS APPLICABLE), THE TERMS OF USE FOR HOSTED SERVICES ARE AVAILABLE ON THE AVAYA WEBSITE, HTTP://SUPPORT.AVAYA.COM/ LICENSEINFO UNDER THE LINK "Avaya Terms of Use for Hosted Services" OR SUCH SUCCESSOR SITE AS DESIGNATED BY AVAYA, AND ARE APPLICABLE TO ANYONE WHO ACCESSES OR USES

THE HOSTED SERVICE. BY ACCESSING OR USING THE HOSTED SERVICE, OR AUTHORIZING OTHERS TO DO SO, YOU, ON BEHALF OF YOURSELF AND THE ENTITY FOR WHOM YOU ARE DOING SO (HEREINAFTER REFERRED TO INTERCHANGEABLY AS "YOU" AND "END USER"), AGREE TO THE TERMS OF USE. IF YOU ARE ACCEPTING THE TERMS OF USE ON BEHALF A COMPANY OR OTHER LEGAL ENTITY, YOU REPRESENT THAT YOU HAVE THE AUTHORITY TO

BIND SUCH ENTITY TO THESE TERMS OF USE. IF YOU DO NOT HAVE SUCH AUTHORITY, OR IF YOU DO NOT WISH TO ACCEPT THESE TERMS OF USE, YOU MUST NOT ACCESS OR USE THE HOSTED SERVICE OR AUTHORIZE ANYONE TO ACCESS OR USE THE HOSTED SERVICE. YOUR USE OF THE HOSTED SERVICE SHALL BE LIMITED BY THE NUMBER AND TYPE OF LICENSES PURCHASED UNDER YOUR CONTRACT FOR THE HOSTED SERVICE, PROVIDED, HOWEVER,

THAT FOR CERTAIN HOSTED SERVICES IF APPLICABLE, YOU MAY HAVE THE OPPORTUNITY TO USE FLEX LICENSES, WHICH WILL BE INVOICED ACCORDING TO ACTUAL USAGE ABOVE THE CONTRACT LICENSE LEVEL. CONTACT AVAYA OR AVAYA'S CHANNEL PARTNER FOR MORE INFORMATION ABOUT THE LICENSES FOR THE APPLICABLE HOSTED SERVICE, THE AVAILABILITY OF ANY FLEX LICENSES (IF APPLICABLE), PRICING AND BILLING INFORMATION,

AND OTHER IMPORTANT INFORMATION REGARDING THE HOSTED SERVICE.

Licenses

THE SOFTWARE LICENSE TERMS AVAILABLEON THE AVAYA WEBSITE, HTTP://SUPPORT.AVAYA.COM/LICENSEINFO OR SUCH SUCCESSOR SITE AS DESIGNATED BY AVAYA, ARE APPLICABLE TO ANYONE WHO DOWNLOADS, USES AND/OR INSTALLS AVAYA SOFTWARE, PURCHASED FROM AVAYA INC., ANY AVAYA AFFILIATE, OR AN AVAYA CHANNEL PARTNER (AS APPLICABLE) UNDER A COMMERCIAL AGREEMENT WITH AVAYA OR AN AVAYA CHANNEL PARTNER. UNLESS OTHERWISE AGREED TO BY AVAYA IN WRITING, AVAYA DOES NOT EXTEND THIS LICENSE IF THE SOFTWARE WAS OBTAINED FROM ANYONE OTHER THAN AVAYA, AN AVAYA AFFILIATE OR AN AVAYA CHANNEL PARTNER; AVAYA RESERVES THE RIGHT TO TAKE LEGAL ACTION AGAINST YOU AND ANYONE ELSE USING OR SELLING THE SOFTWARE WITHOUT A LICENSE. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE, OR AUTHORIZING OTHERS TO DO SO, YOU, ON BEHALF OF YOURSELF AND THE ENTITY FOR WHOM YOU ARE INSTALLING, DOWNLOADING OR USING THE SOFTWARE (HEREINAFTER REFERRED TO INTERCHANGEABLY AS "YOU" AND "END USER"), AGREE TO THESE TERMS AND CONDITIONS AND CREATE A BINDING CONTRACT BETWEEN YOU AND AVAYA INC. OR THE APPLICABLE AVAYA AFFILIATE ("AVAYA").

Avaya grants you a license within the scope of the license types described below. Where the order documentation does not expressly identify a license type, the applicable license will be a Designated System License. The applicable number of licenses and units of capacity for which the license is granted will be one (1), unless a different number of licenses or units of capacity is specified in the documentation or other materials available to you. “Designated Processor” means a single stand-alone computing device. “Server” means a Designated Processor that hosts a software application to be accessed by multiple users.

License type(s)

Designated System(s) License (DS). End User may install and use each copy or an Instance of the Software only on a number of Designated Processors up to the number indicated in the order. Avaya may require the Designated Processor(s) to be identified in the order by type, serial number, feature key, Instance, location or other specific designation, or to be provided by End User to Avaya through electronic means established by Avaya specifically for this purpose.

Concurrent User License (CU). End User may install and use the Software on multiple Designated Processors or one or more Servers, so long as only the licensed number of Units are accessing and using the Software at any given time. A "Unit" means the unit on which Avaya, at its sole discretion, bases the pricing of its licenses and can be, without limitation, an agent, port or user, an e-mail or voice mail account in the name of a person or corporate function (e.g., webmaster or helpdesk), or a directory entry in the administrative database utilized by the Software that permits one user to interface with the Software. Units may be linked to a specific, identified Server or an Instance of the Software.

Copyright

Except where expressly stated otherwise, no use should be made of materials on this site, the Documentation, Software, or hardware provided by Avaya. All content on this site, the documentation and the Product provided by Avaya including the selection, arrangement and design of the content is owned either by Avaya or its licensors and is protected by copyright and other intellectual property laws including the sui generis rights relating to the protection of databases. You may not modify, copy, reproduce, republish, upload, post, transmit or distribute in any way any content, in whole or in part, including any code and software unless expressly authorized by Avaya. Unauthorized reproduction, transmission, dissemination, storage, and or use without the express written consent of Avaya can be a criminal, as well as a civil offense under the applicable law.

Virtualization

Each Product has its own ordering code. Note that each instance of a Product must be separately licensed and ordered. "Instance" means one unique copy of the Software. For example, if the end user customer or Business Partner would like to install 2 instances of the same type of Products, then 2 Products of that type must be ordered.

Third party components

"Third Party Components" mean certain software programs or portions thereof included in the Software or Hosted Service may contain software (including open source software) distributed under third party agreements ("Third Party Components"), which contain terms regarding the rights to use certain portions of the Software ("Third Party Terms"). As required, information regarding distributed Linux OS source code (for those Products that have distributed Linux OS source code) and identifying the copyright holders of the Third Party Components and the Third Party Terms that apply is available in the Documentation or on Avaya's website at:

http://support.avaya.com/Copyright or such successor site as designated by Avaya. You agree to the Third Party Terms for any such Third Party Components.

THIS PRODUCT IS LICENSED UNDER THE AVC PATENT PORTFOLIO LICENSE FOR THE PERSONAL USE OF A CONSUMER OR OTHER USES IN WHICH IT DOES NOT RECEIVE REMUNERATION TO (i) ENCODE VIDEO IN COMPLIANCE WITH THE AVC STANDARD ("AVC VIDEO") AND/OR (ii) DECODE AVC VIDEO THAT WAS ENCODED BY A CONSUMER ENGAGED IN A PERSONAL ACTIVITY AND/OR WAS OBTAINED FROM A VIDEO PROVIDER LICENSED TO PROVIDE AVC VIDEO. NO LICENSE IS GRANTED OR SHALL BE IMPLIED FOR ANY OTHER USE. ADDITIONAL INFORMATION MAY BE OBTAINED FROM MPEG LA, L.L.C. SEE HTTP://WWW.MPEGLA.COM.

Note to Service Provider

The Product or Hosted Service may use Third Party Components subject to Third Party Terms that do not allow hosting and require a Service Provider to be independently licensed for such purpose. It is your responsibility to obtain such licensing.

Preventing Toll Fraud

"Toll Fraud" is the unauthorized use of your telecommunications system by an unauthorized party (for example, a person who is not a corporate employee, agent, subcontractor, or is not working on your company's behalf). Be aware that there can be a risk of Toll Fraud associated with your system and that, if Toll Fraud occurs, it can result in substantial additional charges for your telecommunications services.

Avaya Toll Fraud intervention

If you suspect that you are being victimized by Toll Fraud and you need technical assistance or support, call Technical Service Center Toll Fraud Intervention Hotline at +1-800-643-2353 for the United States and Canada. For additional support telephone numbers, see the Avaya Support website: http://support.avaya.com or such successor site as designated by Avaya. Suspected security vulnerabilities with Avaya products should be reported to Avaya by sending mail to: securityalerts@avaya.com.

Trademarks

The trademarks, logos and service marks ("Marks") displayed in this site, the Documentation, Hosted Service(s), and Product(s) provided by Avaya are the registered or unregistered Marks of Avaya, its affiliates, or other third parties. Users are not permitted to use such Marks without prior written consent from Avaya or such third party which may own the Mark. Nothing contained in this site, the Documentation, Hosted Service(s) and Product(s) should be construed as granting, by implication, estoppel, or otherwise,

any license or right in and to the Marks without the express written permission of Avaya or the applicable third party.

Avaya is a registered trademark of Avaya Inc.

All non-Avaya trademarks are the property of their respective owners. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Downloading Documentation

For the most current versions of Documentation, see the Avaya Support website: http://support.avaya.com or such successor site as designated by Avaya.

Contact Avaya Support

See the Avaya Support website: http://support.avaya.com for Product or Hosted Service notices and articles, or to report a problem with your Avaya Product or Hosted Service. For a list of support telephone numbers and contact addresses, go to the Avaya Support website: http://support.avaya.com (or such successor site as designated by Avaya), scroll to the bottom of the page, and select Contact Avaya Support.

Contents

Chapter 1: Introduction

Purpose

This guide provides nformation about the blocks you can use with agent script workflows and other reference information to help you understand and create agent script workflows for Avaya Interaction Center Release 7.3.x.

Intended audience

This guide is intended primarily for those who need to create agent script workflows for Avaya Interaction Center Release 7.3.x.

Related resources

Documentation

See the following related documents at http://support.avaya.com.

Title	Use this document to:	Audience
Avaya Interaction Center and Avaya Operational Analyst Overview and Specification	get information about the new features and enhancements in Avaya Interaction Center. This document describes tested Interaction Center (IC) and Operational Analyst(OA) characteristics and capabilities, including feature descriptions, interoperability, performance specifications, security, and licensing requirements.	Sales Engineers Supervisors Business Partners Customers
IC Installation Planning and Prerequisites	get information about the planning and third-party software required to deploy an Avaya Interaction Center system.	Business partners Customers Implementation engineers
IC Installation and Configuration	get information about how to install and configure an out-of-the-box Avaya Interaction Center.	Sales Engineers Supervisors Business Partners Customers
IC Administration Guide	get information about Avaya Interaction Center (Avaya IC). This guide describes domain and server administration using Avaya IC Manager.	Sales Engineers Supervisors Administrators Business Partners Customers
IC Client SDK Programmer Guide	get information about the Client Software Development Kit (Client SDK) for Avaya Interaction Center.	Sales Engineers Business Partners Implementation engineers Customers

Title	Use this document to:	Audience
IC Client and Server Programmer Design Guide	get information about Avaya Interaction Center (Avaya IC). The purpose of this guide is also to provide an overview of the Avaya IC ORB Toolkit, and a list of components in the Avaya IC product set.	Sales Engineers Business Partners Implementation engineers Customers
Using Config Accelerator	get information about how to use Config Accelerator for simplifying and accelerating the configuration process of Avaya Interaction Center.	Supervisors Administrators Business Partners Customers
IC Database Designer Application Reference	get information about Avaya Interaction Center (IC). This guide describes the prerequisites for installing and configuring Avaya IC.	Supervisors Administrators Business Partners Administrators
Agent Web Client Customization	get information about how to customize Avaya Agent Web Client.	Supervisors Administrators Business Partners Administrators
IC Workflow Designer User Guide	get information about Avaya Workflow Designer and how to use Workflow Designer to create and customize workflows for Avaya Interaction Center.	Supervisors Administrators Business Partners Administrators
IC Media Workflow Reference	get information about the blocks you can use with media workflows and other reference information to help you understand and customize media workflows for Avaya Interaction Center.	Supervisors Administrators Business Partners Administrators
Agent Script Workflow Reference	get information about the blocks you can use with agent script workflows and other reference information to help you understand and create agent script workflows for Avaya Interaction Center.	Supervisors Administrators Business Partners Administrators Implementation engineers Customers

Title	Use this document to:	Audience
IC Workflow API Reference	get information about the extensions and methods specific to the Application Programming Interface (API) for workflow blocks in Avaya Interaction Center.	Supervisors Administrators Business Partners Administrators Implementation engineers
Agent Data Unit Server Programmer Guide	get information about configuring and managing the Agent Data Unit (ADU) server, which is responsible for tracking the state of agents at the contact center.	Supervisors Administrators Business Partners Administrators Implementation engineers
Core Services Programmer Guide	get information for programming and changing the configuration of your IC Core Services.	Supervisors Administrators Business Partners Administrators Implementation engineers
Electronic Data Unit Server Programmer Guide	get detailed information about the Electronic Data Unit (EDU) server, which was previously named the Voice Data Unit (VDU) server, the EDU server configuration, the alarms, and the event monitoring.	Supervisors Administrators Business Partners Administrators
IC Telephony Connectors Programmer Guide	get information on the Avaya Telephony Connector server.	Supervisors Administrators Business Partners Administrators
IC Integration with VP / IR	get information about Avaya Interaction Center (IC) integration with Avaya Aura® Experience Portal (EP) / Avaya Voice Portal (VP) / Avaya Interactive Response (IR).	Supervisors Administrators Business Partners Administrators

Finding documents on the Avaya Support website

Use this procedure to find product documentation on the Avaya Support website.

1. Use a browser to navigate to the Avaya Support website at http://support.avaya.com.
2. At the top of the screen, enter your username and password and click **Login**.
3. Click **Documents**.
4. In the **Enter your Product Here** search box, type the product name and then select the product from the drop-down list.
5. If there is more than one release, select the appropriate release number from the **Choose Release** drop-down list.
6. Use the **Content Type** filter on the left to select the type of document you are looking for, or click Select All to see a list of all available documents.

   For example, if you are looking for user guides, select **User Guides** in the **Content Type** filter. Only documents in the selected category will appear in the list of documents.
7. Click **Enter**.

Training

The following courses are available on the Avaya Learning website at www.avaya-learning.com. In the **Search** field, enter the course code, and click **Go to search** for the course.

Course Code	Course Title
ATC01175WEN	IC and OA Overview
ATC01176IEN	Interaction Center Administration and Configuration
AUCC100010695	IC-Siebel Integration
ATC100011017	IC-Siebel Integration, Installation and Troubleshooting

Viewing Avaya Mentor videos

Avaya Mentor videos provide technical content on how to install, configure, and troubleshoot Avaya products.

Videos are available on the Avaya Support website, listed under the video document type, and on the Avaya-run channel on YouTube.

- To find videos on the Avaya Support website, go to http://support.avaya.com and perform one of the following actions:
  - In Search, type Avaya Mentor Videos to see a list of the available videos.

  - In Search, type the product name. On the Search Results page, select Video in the Content Type column on the left.
- To find the Avaya Mentor videos on YouTube, go to www.youtube.com/AvayaMentor and perform one of the following actions:
  - Enter a key word or key words in the Search Channel to search for a specific product or topic.
  - Scroll down Playlists, and click the name of a topic to see the available list of videos posted on the website.

Note:
Videos are not available for all products.

Support

Go to the Avaya Support website at http://support.avaya.com for the most up-to-date documentation, product notices, and knowledge articles. You can also search for release notes, downloads, and resolutions to issues. Use the online service request system to create a service request. Chat with live agents to get answers to questions, or request an agent to connect you to a support team if an issue requires additional expertise.

Chapter 2: Overview

With the blocks from the Prompter palette in Workflow Designer, you can create agent script workflows that help to improve the quality of agent interaction with customers. Before working with agent script workflows, you should be familiar with the information contained in *Avaya Workflow Designer User Guide*.

Agent script workflows represent predetermined scripts that agents can run in their Prompter pane. Agents frequently refer to these predetermined scripts to assist them in handling requests from customers, whether those requests arrive by phone, email, web chat, or voice chat.

When you implement Prompter in your contact center:

- Agents have immediate online access to agent scripts
- Agents can collect customer information through surveys and checklists
- Agents can automatically save the information to a database

Note:
Some agent script workflows that were created in a previous release of Workflow Designer may not function in this release. To use a workflow created in a previous release, delete the Stop block, then add a new Stop block from the catalog.

This section provides an overview of agent script workflows, including the sample workflow provided with Avaya IC. This section includes the following topics:

Process of an agent script workflow

An agent script workflow is a three step process:

1. Start script:
   a. Avaya Agent calls the script from the database.
   b. The Prompting Engine inserts the required database information in the script.
   c. The first HTML page displays in the Prompter pane of Avaya Agent.
2. Collect data:
   a. The agent steps through the script.
   b. After the agent completes each step in the script, Prompter displays the HTML page for the next step in the script.
   c. Depending on the answers to questions and other actions that the agent performs, Prompter accesses the database, displays the appropriate HTML page, updates tables, and saves data.
3. End script:
   a. The agent completes all steps in the script.
   b. The script displays the HTML page that represents the final step in the script.
   c. If necessary, Prompter updates the database.
   d. Depending on the workflow design, the Prompter pane and any Web browser that it invoked can remain on the desktop and reset itself to re-start the agent script in preparation for a new customer contact.

Types of agent script workflows

Avaya IC can use the following types of agent script workflows:

Prompting script workflows: These workflows use Prompter blocks from the Prompter palette. The script displays a set of survey questions or a qualification process. For example, the agent could ask the contact questions such as "Are you interested in today's special?" or "Are you a registered voter?" For more information, see Prompter objects on page 40.

Application script workflows: These workflows use App blocks from the Prompter palette. The script guides an agent through a series of steps to assist them with a task, such as taking an order from a customer or entering a complaint. For more information, see Database objects on page 52.

Combination agent script workflows: These workflows use both Prompter blocks and App blocks from the Prompter palette. The script guides an agent through a combined set of tasks that includes steps from both prompting workflows and application workflows.

Planning an agent script workflow

When you create and integrate workflows, most of the necessary work is done during the planning phase.

You need to ask the right questions and determine which components are available on an agent desktop. For example, if the agents use Avaya Agent, you can use the Prompter pane in Avaya Agent. Agents can also run the workflows in a Web browser. Your decision affects how you integrate the workflow with the application.

Depending on your system and your needs, you may need to answer a combination of general and specific questions before you can start.

The following are some very general questions that can get you started:

- When and why will the agent run the workflow?
- What information does the contact center need to:
  - Get from the customer?
  - Give to the customer?
- Does the agent need to persuade the customer to do something? For example, purchase a product or join a club?
- What information does the agent need to help persuade the customer?
- What needs to happen after the agent completes the script in the workflow?

Answers to question like these help you determine the following:

Flow of the script: How will the script branch if the customer makes a choice or requests a specific type of information?

Purpose of any information gathered: What needs to happen to information that the agent obtains from the customer. For example, do you need to add the information to an Avaya IC database?

Blocks required by the script: Which blocks do you need to use to perform each step. Can you use the standard blocks available on the palettes in Workflow Designer, or do you need to create one or more custom blocks? For example, you can create a block to parse through the survey history maintained by Prompter and extract the answers entered by the agent on behalf of the customer.

For more detailed information about creating custom blocks, see *Avaya Workflow Designer User Guide*.

Database connections

All agent script workflows typically require access to a database, so the agent can retrieve and save database information while following the script. You specify the name of the database network accessed by your workflow in the Start block.

You can create database networks (modules) with Database Designer. If you are unfamiliar with the networks in your database, review *IC Database Designer Application Reference* before you design a Prompter workflow.

To create an agent script workflow that uses database objects, you should be very familiar with the data model of your Avaya IC databases, including the CCQ database.

Limitations of variables in agent script workflows

The Workflow server only transmits variables of type String, Long, and Float (specifically `SVstring`, `SVlong`, and `SVfloat`) with the HTML pages in agent script workflows. This limitation prevents the Workflow server from flooding the agent desktop applications with large collections of unneeded data.

If you need to get a `SeqCouple` to an agent desktop application, convert the `SeqCouple` to a string using `.tostring` and include that string in a global string variable. For more information, see *Avaya IC Media Workflow Reference*.

Integrating a workflow with a Prompter client control

Agents can launch and run agent script workflows in the following:

- Prompter pane of Avaya Agent
- Web browser

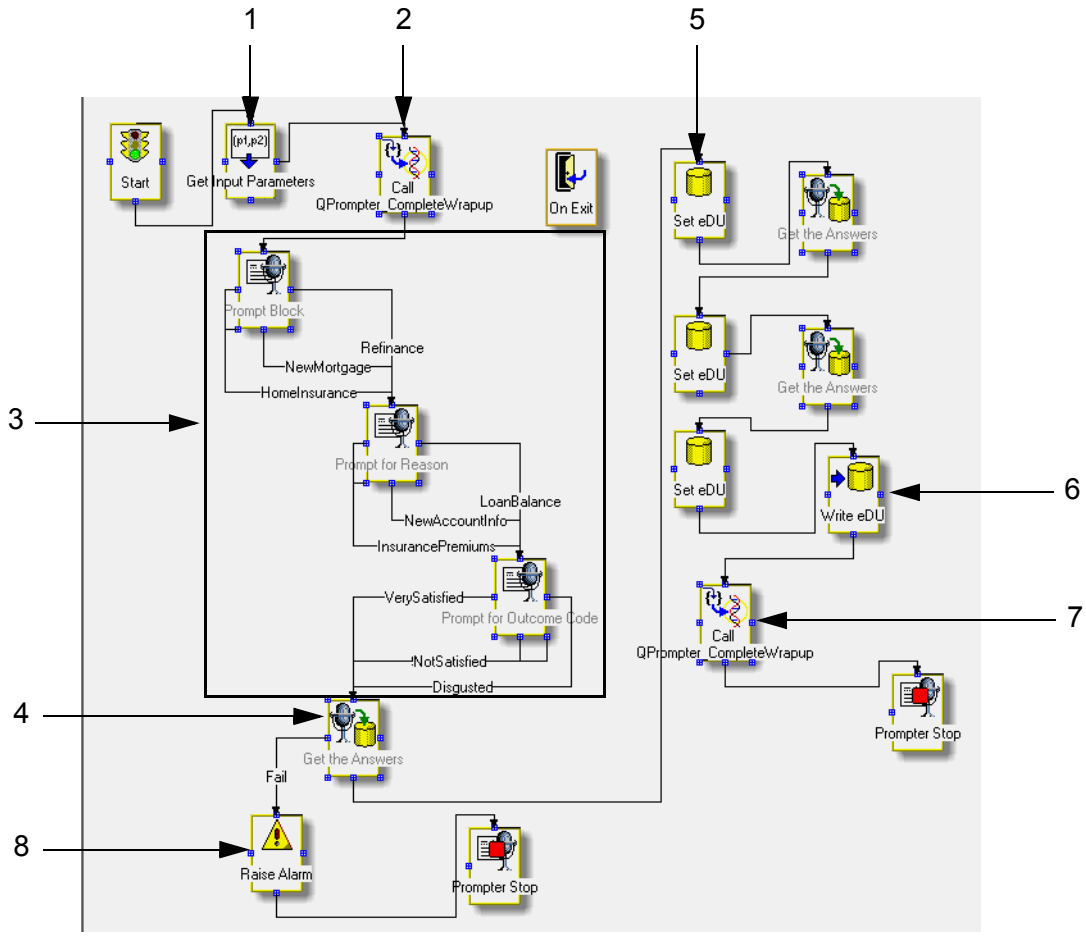
For more information, see *Avaya Agent Integration*.

Sample Wrapup Workflow

Avaya IC installs the Sample Wrapup workflow in the following directory:

```
IC_INSTALL_DIR\IC73\design\IC\Flows\Avaya\Prompter\sample_wrapup.qfd
```

The Sample Wrapup workflow shows an example of a wrapup flow that could be used by the contact center of a bank or an insurance company. The workflow includes a series of questions that an agent can ask a customer at the end of a voice contact or chat contact to enter information about the contact and the reaction of the customer to the contact.



The Sample Wrapup workflow includes the steps described in the following table:

Step	Block Names	Description
	Start	Starts the workflow.
1	Get Input Parameters	Retrieves the values specified in the Parameter_N fields on the **Basic** tab and assigns them to the associated variables in the Target_N fields. By default, this block retrieves and assigns the following values from the EDU: • EDUID is assigned to the variable `$vduid`. • Task ID is assigned to the variable `$taskid`. • Media type is assigned to the variable `$MediaType`. To customize this block, you can add values and variables to fields on the **Basic** tab.
2	Call Prompter Complete Wrapup	Sets the name and specified parameters for an IC Script that runs at the end of the workflow.
3	Prompter blocks	Displays an HTML page in the Prompter pane. Each Prompter block represents one step in the agent script. For example, you can use a Prompter block to display a page that contains a question for the agent to ask the customer and a series of checkboxes for the agent to complete with the answer, such as: • A category that relates to the customer contact, such as auto insurance or a new mortgage. • A more specific reason for the customer contact, such as finding out information about insurance premiums or checking the current balance on a loan. • An outcome code that describes the customer's opinion of the contact, such as Very Satisfied or Not Satisfied.
4	Get the Answers	Returns the answer that an agent enters for the question specified on the Basic tab as a string. If the answer is returned, proceeds to Step 5, Set EDU. If the answer is not returned, proceeds to Step 8, Raise Alarm. You must include one Get the Answers block for each Prompter block. You need to follow each Get the Answers block with a Set EDU block.
5	Set EDU	Sets an EDU element to the value returned in the related Get the Answers block, then assigns these values in a script variable. You must include one Set EDU block for each Get the Answers block.

Step	Block Names	Description
6	Write EDU	Writes the values assigned to the script variable by the Set EDU blocks as a sequence of couples to the EDU for the current contact. You need only one Write EDU block in the workflow, even if you have multiple Set EDU blocks.
7	Call Prompter Complete Wrapup	Sets the name and specified parameters for the `wrapupFlowDone` IC Script that runs at the end of the workflow.
8	Raise Alarm	Raises an alarm in the Alarm server. The alarm contains the information that you specify in the following properties on the **Basic** tab: • AlarmName - The name of the alarm, such as *Prompter Wrapup* • Description - The text to be included in the alarm, such as *Could not get answers to survey* • Priority - The priority to be given to the alarm, such as *info* or *Emergency*
	Prompter Stop	Marks the exit point for the workflow. After the Prompter Stop block, the Workflow server executes the On Exit block.

Chapter 3: Standard blocks

This section describes the standard blocks installed with Workflow Designer that are typically used in agent script workflows. These standard blocks are available on the Prompter palette.

This section only describes blocks on the Prompter palette. For information about blocks on other palettes, see *Avaya IC Media Workflow Reference* and *Avaya Workflow Designer User Guide*.

This section includes the following topics:

Tip:
All agent script workflows also require a Start block. You can find the Start block in the General palette. For more information about the Start block, see *Avaya IC Media Workflow Reference*.

Add Record

Description

Adds a record object to a database table, but does not commit the record to the database. This block is effective only when followed by a Commit Record block.

To use the Add Record block, complete the Table value in the Basic properties of the block by entering the name of the table where the block adds a record object.

Alarms

Generates no alarms.

Connections

Accepts the following connections:

- Input - 1 or more
- Output - 1

App Block

Description

Performs application scripting. See Database objects on page 52 for more information.

Each App Block displays the HTML page for the step in the agent script that this block represents. To create the HTML page, select the **Page** field on the **Basic** tab of the block properties sheet. This selection opens the DHTML Editor, which you use to create the page. For more information, see DHTML Editor on page 26.

Alarms

Generates no alarms.

Connections

Accepts the following connections:

- Input - 1 or more
- Output - 1

App Stop

Description

Terminates an Application workflow in a workflow that includes at least one App Block. This block must be the last block in a script that contains Application blocks. If a workflow contains a combination of Application and Prompter blocks, you must include an App Stop block after the last App Block.

An App Stop Block displays the HTML page for the step in the agent script that this block represents. Typically, this HTML page contains text that informs the agent that the workflow is complete. To create the HTML page, select the **Page** field on the **Basic** tab of the block properties sheet. This selection opens the DHTML Editor, which you use to create the page. For more information, see DHTML Editor on page 26.

App Stop invokes no external methods.

If the workflow contains a combination of App blocks and Prompter blocks, and you want to display a summary of the information gathered by the agent during the agent script, you must also use a Prompter Stop block.

Alarms

Generates no alarms.

Connections

Accepts the following connections:

- Input - 1 or more
- Output - 1

Commit Record

Description

Commits a record object from a database table to the database. This block is typically effective only when preceded by an Add Record block.

To use the Commit Record block, complete the Table value in the Basic properties of the block by entering the name of the table where the block commits the record object.

Alarms

Generates no alarms.

Connections

Accepts the following connections:

- Input - 1 or more
- Output - 1

On Exit

Description

Executes when the workflow ends, whether the workflow completes normally or is interrupted. This block has no connections to other blocks. This block can terminate the EDU and provide extra debugging capability when needed.

In debug mode, this block prints the name of the block that executed just prior to On Exit. This feature is useful for debugging workflows.

A workflow may contain an On Exit block. This block has no anchorsnaps, and is not connected to another block in the workflow. The On Exit block provides a common exit point, and contains code for cleanup and error handling. Whenever a workflow exits, whether through a Stop block, or because of an error, the Workflow server executes the On Exit block if one exists in the workflow.

The On Exit block typically:

- Logs the name of the last block run in the workflow for debugging purposes.
- In debug mode, prints a log entry that includes the name of the last block that ran (from the `lastBlock` Script variable). You can use this entry to trace an unexpected exit, even if the block that exited leaves no debug trail.

- Auto-defines the Script variable `vduTouched`, which other EDU-related blocks can set to `true` or `false`.
- Executes the `VDU.Terminate` method if a block in the workflow touches the EDU and if you set the value of the `vduTerminate` property on the **Advanced** tab to `on`.

Tip:
Although all EDU-related blocks can auto-define `vduTouched` and terminate the EDU, use the On Exit block to perform these functions to prevent EDU records from being orphaned if workflows exit unexpectedly.

Alarms

Generates no alarms.

Connections

Has no connections to any other blocks.

Prompter Block

Description

Creates information, and question and answer scripts. See Prompter objects on page 40 for more information.

Each Prompter block displays the HTML page for the step in the agent script that this block represents. To create the HTML page, select the **Page** field on the **Basic** tab of the block properties sheet. This selection opens the DHTML Editor, which you use to create the page. For more information, see DHTML Editor on page 26.

Alarms

Generates no alarms.

Connections

Accepts the following connections:

- Input - 1 or more
- Output - 1

Prompter Stop

Description

Terminates a Prompter workflow in a workflow that includes at least one Prompter Block. This block must be the last block in a script that contains Prompter blocks. If a workflow contains a combination of Prompter and App blocks, you must include a Prompter Stop block after the last Prompter Block.

You can use a Prompter Stop block to display a summary of the information collected by the agent in the agent script.

A Prompter Stop Block displays the HTML page for the step in the agent script that this block represents. Typically, this HTML page contains text that informs the agent that the workflow is complete. To create the HTML page, select the **Page** field on the **Basic** tab of the block properties sheet. This selection opens the DHTML Editor, which you use to create the page. For more information, see DHTML Editor on page 26.

Prompter Stop invokes no external methods.

Alarms

Generates no alarms.

Connections

Accepts the following connections:

- Input - 1 or more
- Output - 1

Chapter 4: DHTML Editor

The DHTML Editor provides a visual editing environment that you can use to quickly design, format, and add functionality to the HTML pages that an agent reads and completes during an agent script. You do not need to write HTML code to use the DHTML Editor.

For each Prompter or App block in your agent script workflow, use the DHTML Editor to format an HTML page that:

- Prompts agents to fill out fields and answer questions through the script
- Finds and updates information in database fields

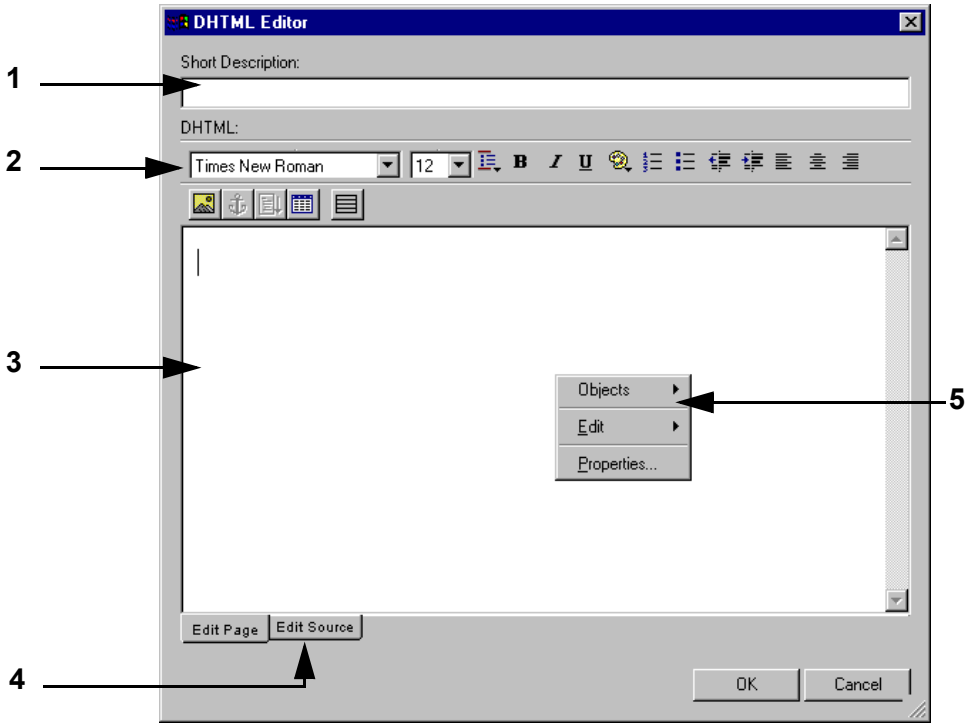
The DHTML Editor opens when you select the DHTML property of a Prompter block or an App block. Each Prompter block and App block contains one DHTML property, `Page`. The Page property is on the **Basic** tab.

Tip:
The Page property defines the HTML page that Prompter displays to the agent when the workflow is run. If you are unfamiliar with properties and with the basic principles of creating a workflow, see *Avaya Workflow Designer User Guide*.

This section describes the DHTML Editor and provides some basic instructions for using the DHTML Editor. The topics in this section include:

DHTML Editor components

The following figure shows the components of the DHTML Editor.



As marked in the previous figure, the DHTML Editor includes the components described in the following sections.

1. Short Description text box on page 28.
2. DHTML Formatting toolbar on page 28.
3. Edit Page work area on page 29
4. Edit Source work area on page 29
5. Right-click menu options on page 30

The components of the DHTML Editor also include keyboard shortcuts. For more information, see Keyboard shortcuts on page 30.

Short Description text box

Use the **Short Description** text box to create a brief description of the HTML page. Workflow Designer displays this description in the Page property for the block. This description enables you to easily identify which HTML page the block represents.

DHTML Formatting toolbar

Use the buttons in the **DHTML Formatting** toolbar to apply formatting to paragraphs, to text, or to the page. Many of the buttons available on this toolbar are similar to features in other Windows applications.

The following table describes the buttons that are unique to the DHTML Editor.

Button	Button name	Purpose
	Paragraph format	Assigns the HTML paragraph format chosen from the drop-down list to the selected text.
	Color palette	Applies the color chosen from the drop-down list to the selected text.
	Insert image	Inserts the selected image into the HTML page. The image must be in a location that the workflow can access.
	Hyperlink	Turns the selected text into a hyperlink to the specified location.
	Show borders	Displays borders around all appropriate objects on the page.

Edit Page work area

WARNING:
Create the design elements before you add Prompter objects or database objects. The code for these objects may not function correctly if you make changes after you add them.

Use the **Edit Page** work area to add any of the following elements to the HTML page:

- Text
- Images
- Hyperlinks
- Prompter objects
- Database objects

The **Edit Page** work area is a visual editing environment. As you add elements to the HTML page, this work area shows you what the page will look like to an agent.

If your HTML page includes design elements, such as tables, use the **Edit Source** work area to create those elements. For more information, see Edit Source work area on page 29.

Edit Source work area

Use the **Edit Source** work area to add design elements to the HTML page. The **Edit Source** work area is a text editor environment that you can use to enter HTML code.

Tip:
If you add design elements in the **Edit Source** work area, return to the **Edit Page** work area before you continue. If you select **Cancel** in the **Edit Source** area, the DHTML Editor closes without saving and all changes are lost.

Even if you are proficient with HTML, do not use the **Edit Source** view to change the HTML source code for Prompter objects or Database objects. If HTML source code for a block contains an error, the workflow compiles, but the error only surfaces when the workflow is run. This results in very difficult troubleshooting to locate the error.

Right-click menu options

The right-click menu of the DHTML Editor provides options that you can use to create or modify elements on a page. The following table describes the right-click menu options.

Menu option	Description
Objects	Lets you create one of the following objects on a page: • Prompter object for a Prompting script • Database object for an Application script After you create one type of object, the other object type is no longer available for the page.
Edit	Lets you add or delete rows in a table. This option is available only after you create a table.
Properties	Displays the properties of the page in eXtensible Markup Language (XML) format.

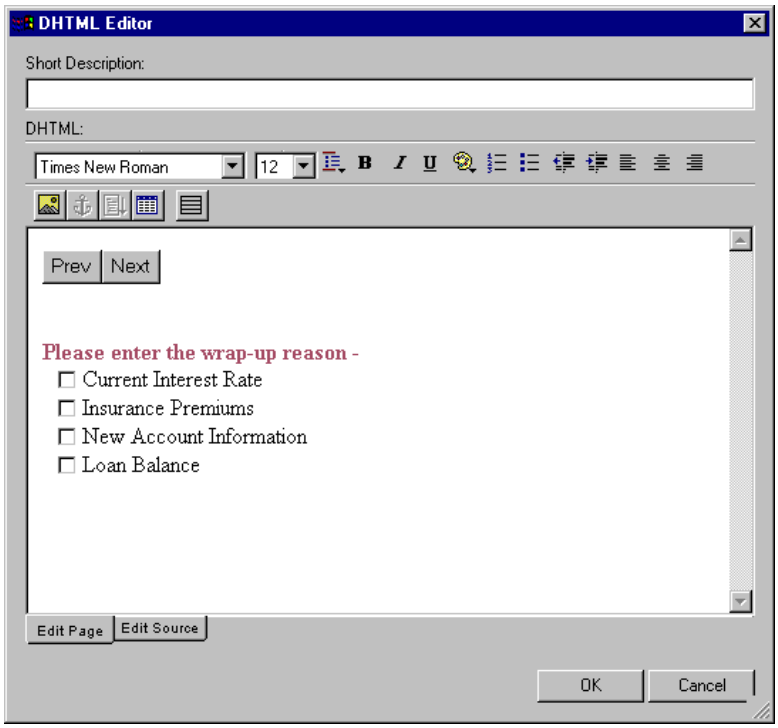
Keyboard shortcuts

The following table describes the standard keyboard shortcuts that you can use in the DHTML Editor.

Shortcut	Description
Ctrl+X	Cut
Ctrl+C	Copy
Ctrl+V	Paste
Ctrl+Z	Undo
Ctrl+Y	Redo

Creating an HTML page

The following figure shows an HTML page with a Prompter object.



To create an HTML page in the DHTML Editor:

1. Select a Prompter or an App block in your workflow.
2. Select the **Basic** tab in the Property sheet to display the Page property.
3. Select the **Ellipsis (…)** button next to the Page property to open the DHTML Editor.
4. In the **Short Description** text box, enter a description for the HTML page.

   This description is displayed on the **Basic** tab, so you can easily identify the HTML page.
5. Enter the questions and answers or database queries for the script in the main section of the DHTML Editor using one of the following methods:
   - Use the Prompter Wizard to guide you through creating a page for a prompting script. See Prompter objects on page 40 for more information.
   - Use the Database Wizard to guide you through creating a page for an application script. See Database objects on page 52 for more information.
   - Type the text in the main section of the **Edit Page** tab. When you create a new HTML page, the DHTML Editor inserts the cursor at the first available line. When you edit an existing page, the DHTML Editor inserts the cursor after the last line of text.

Chapter 5: Workflow basics

Each workflow for an agent script can include either of the following:

- A series of Prompter blocks
- A series of Application blocks
- A combination of Prompter blocks and Application blocks

Each Prompter block or Application block represents a single HTML page in the agent script. Depending on the purpose of the script, each HTML page can contain one question with answer fields, or one piece of information that the agent needs to give the customer.

After you design the script and know the order of the questions and whether the answer given to a question determines which step comes next, you can create the script workflow in Workflow Designer.

Tip:
Before creating an agent script workflow, you must be familiar with the data model of your application, including the table sets in focus modules, database tables, and the fields in the tables. For more information, see the Data Model Reference for your application. If your application has been customized with Database Designer, see your application designer for information on situations of which you should be aware.

To create an agent script workflow, complete the steps in the following sections:

1. Block properties on page 33
2. Creating a project on page 33
3. Adding the blocks on page 34
4. Connecting the blocks on page 35
5. Specifying properties for a block on page 36
6. Configuring DHTML pages for non-English languages on page 37
7. Saving the workflow on page 38

Block properties

The Property sheet contains the properties of a block. The following table describes the information provided by the four tabs in the Property sheet.

Tab	Description
General	Contains the description of the block.
Basic	Contains properties that must be set for each block, including the Page property for Prompter blocks and Application blocks.
Advanced	Contains properties designed for use only by block designers and advanced workflow designers.
Expert	Contains information about the graphical aspects of the block, such as its color and shading, and information about the snap points that connect the block to other blocks.

The properties in the tabs depend on the type of block. You need to review the properties in each tab and modify them if necessary. For more information about block properties, see the *Avaya Workflow Designer User Guide*.

Each property has a property type. The property type is a specific type of information that can be stored in that property.

Some value properties can hold a literal text string such as `demoString`, or a symbol. A symbol is a definition of a variable that shares information between blocks. Advanced users can learn more about symbols in *Avaya Workflow Designer User Guide*.

Creating a project

A project contains a set of related workflows and information about how those workflows are built and stored. Each agent script workflow must be part of a project. If you do not create a project for your script, you cannot open the DHTML Editor.

By default, Workflow Designer saves projects in the same folder as the workflows.

When you open a read-only project, Workflow Designer lets you make it writable. For example, if you maintain your project (`.qfd` and `.prj`) files in a source-code control system, the system frequently makes the files read-only. When you re-open the project, Workflow Designer lets you change the permissions so you can save any changes.

Important:
Project names cannot contain spaces.

To create a project:

1. Select **Project** > **New**.
2. Enter the project name.
3. Select **OK**.

   Your project displays in the **Project** bar.

Adding the blocks

Each block represents a question in the agent script. You place the blocks in the same order that the questions occur in the agent script.

To create a new workflow:

1. Select **File** > **New Flow**.

   A new workflow is made available in the Work Area.
2. In the **General** palette:
   a. Select the Start block.
   b. Drag it to the Work Area.
   c. Drop the block in the Work Area.
3. Drag all needed Prompter, Application, or other blocks from the Prompter palette to the Work Area and drop them below the Start block in the order that the corresponding questions should occur in the workflow.

   For information about the standard Workflow Designer blocks and the associated properties, see "Standard Blocks" in *Avaya Workflow Designer User Guide*.
4. Drag one or more Stop blocks to the Work Area:
   a. If you used Prompter blocks, drop a Prompter Stop block after the last Prompter block in the workflow.
   b. If you used Application blocks, drop an App Stop block after the last Application block in the workflow.
   c. For all workflows, drop the Stop block from the General palette below the last block in the workflow.
   d. If desired, drop an On Exit block at any location in the workflow.

Connecting the blocks

A workflow must indicate the order in which the server executes the blocks. You set the order by using connectors between the blocks. The direction that you draw the connector indicates the order.

The blocks in your workflow have blue squares around their edges. These squares are snap points. They are also known as anchor snaps or connection points. Snap points indicate where you can connect one block to another block.

On the drawing toolbar, you can select one of the following types of connectors:

Tool icon	Description
 Connector	Connects blocks with straight lines and 90 degree bends. If you use this tool, Workflow Designer displays labels in the center of the connector.
 Line	Connects blocks with diagonal or straight lines. This tool does not create lines with angles.

To connect the blocks in your workflow:

1. From the Drawing bar, select a connector tool.

   If the Drawing bar is not displayed, select **View** > **Toolbars** > **Drawing bar**.

2. With the connector tool selected:

   a. Select the blue snap point of the Start block

   b. Drag your mouse down to a blue snap point on the Demo block and release

   Insert the connector in the direction you want the workflow to progress. If you drag the connector up to a preceding block in the workflow, your workflow moves backwards.

Note:

If your agent script workflow includes an On Exit block, do not connect that block to the other blocks in the workflow.

Specifying properties for a block

This section include the following topics that describe the properties that you must set in the blocks of the workflow:

1. Setting the properties in the Start block on page 36.
2. Setting the Page property on page 36.

Setting the properties in the Start block

The network property in the Start block determines which database the workflow uses and how the workflow connects to that database. You must set this property for your agent script workflow to function.

To set the network property in the Start block:

1. Select the Start block to display the Property Sheet.
2. In the Property Sheet, select the **Basic** tab.
3. In the `DBNetwork` property, enter the name of the focus in the `ccq.adl` file that contains the table set used by this workflow.

   For example, enter `q_emailanalysis` to specify that the workflow uses the `q_emailanalysis` focus of the Interaction Center application. You can find the Interaction Center application in the `ccq.adl` file.
4. In the `ToolkitTimeout` property, if desired, change the default timeout of 30000 seconds (5 minutes).

   This default setting lets an agent wait for up to 5 minutes before navigating to the next page in the agent script workflow.
5. Save the workflow.

Setting the Page property

The Page property in each Prompter block and Application block includes the code for the HTML page that Prompter displays to the agent. You must set the Page property for each of the following blocks in the workflow:

Prompter blocks: In the Page property on the **Basic** tab, create the HTML page for this block as described in Prompter objects on page 40.

Application blocks: In the Page property on the **Basic** tab, create the HTML page for this block as described in Database objects on page 52.

Configuring DHTML pages for non-English languages

If you create an agent script workflow for a language other than English, you may need to configure the charset property for each page. If you do not configure the charset property in the DHTML, an agent may see garbled characters or boxes in the agent script.

Tip:
Perform this step for all DHTML pages in Prompter blocks and Application blocks in an agent script workflow that will run in a language other than English.

To configure DHTML pages for non-English languages:

1. Select a Prompter or Application block in the workflow to display the Property Sheet.
2. In the Property Sheet, select the **Basic** tab.
3. Select the **Page** property.
4. In the DHTML editor, select the **Edit Source** tab.
5. Add the following line directly under the **<HEAD>** line in the DHTML code:

   ```
   <META http-equiv="Content-Type" content="text/html;
     charset=language_charset">
   ```

   where ***language_charset*** is the character set for the language of the agent script workflow. For example, if you want to create an agent script workflow in Russian, enter **windows-1251** for the charset value, as follows:

   ```
   <META http-equiv="Content-Type" content="text/html;
     charset=windows-1251">
   ```

6. Save the workflow.

Saving answers to the database

By default, agent script workflows save customer answers in the `qw_response` table. If you select a different database table in the Prompter Script wizard when you create the questions for your prompting script, the workflow runs but does not save the answer in the database.

You can, however, save answers to any database table by including Add Record and Commit Record blocks from the Prompter palette in your Prompting script. For more information about these blocks, see Standard blocks on page 20.

Note:
When answers are saved to the database, the questions are also saved. Weighted scores cannot be saved to the database.

To save answers in a database table other than `qw_response`:

1. Before the Prompter block where you want to save answers to the database, insert an Add Record block into your workflow.
2. Enter the name of the table where you want to save the answer in the `Table` property of the Add Record block.
3. Insert a Prompter block after the Add record block in your workflow.
4. Use the DHTML Editor to configure the question for this step in the Prompting script. See Configuring Prompter questions on page 41 for more information. You must add the name of the table and table field where you want to save the answer.
5. Connect the Add Record block down to the Prompter block.
6. Insert a Commit Record block after the Prompter block in your workflow.
7. Enter the name of the table where you want to save the answer in the `Table` property of the Commit Record block.
8. Connect the Prompter block down to the Add Record block.

Saving the workflow

Use alphanumeric characters in workflow names. If you use a non-alphanumeric character, such as a space or a punctuation mark, Workflow Designer converts the character to an underscore when you compile the workflow. For example, `my flow` and `my.flow` would both become `my_flow`.

To save the workflow:

1. Select **File** > **Save**.
2. Enter the workflow name in the field.
3. Select **Save**.

   Workflow Designer automatically adds the extension `.qfd`.
4. Select **Yes** when Workflow Designer prompts you to insert the workflow into the flowset.

   Workflow Designer saves your workflow in the current project. You can expand the project in the Project bar and see the workflow.

After saving your agent script workflow, perform the following procedures that are found in *Avaya Workflow Designer User Guide*:

1. Verify the workflow.
2. Build the workflow.
3. Load the workflow in the Workflow server.

Chapter 6: Prompter objects

A Prompter object allows you to include a set of questions and answers in an agent script workflow. You create a Prompter object in each Prompter block in a workflow.

After you design the script and know the order of the questions and whether the answer given to a question determines which step comes next, you can create Prompter objects in the code for the Page property of a Prompter block. For each step in the script, the HTML pages that form the Page property display one or more of the following:

- Information for the agent to give to the customer.
- Questions for the agent to ask the customer.
- If desired, a series of pre-determined answers for the customer to select.

To create a Prompter objects in the HTML page of a Prompter block, perform the steps in the following sections:

1. Setting the block text and Page property on page 40.
2. Configuring Prompter questions on page 41.

Note:
If you add text to agent script workflows with the DHTML Editor, an agent who views the workflow through a browser must use a supported browser. For information on supported browsers, see *IC Installation Planning and Prerequisites*.

Setting the block text and Page property

The Page property is a short description in the **Basic** tab that identifies the HTML page. The block text is the title that displays on the block in the Work Area to identify the purpose of the block in the workflow.

To set the block text and Page property for the HTML page that represents a step in the script:

1. Select a Prompter block in your workflow.
2. Add the block text:
   a. Select the **Expert** tab in the Property sheet.
   b. Select **Text** and enter a description of the block's purpose. For example, Prompt for User Name.
3. Select the **Basic** tab in the block's Property sheet to display the Page property.

4. Select the **Ellipsis (…)** button next to the Page property to bring up the DHTML Editor.

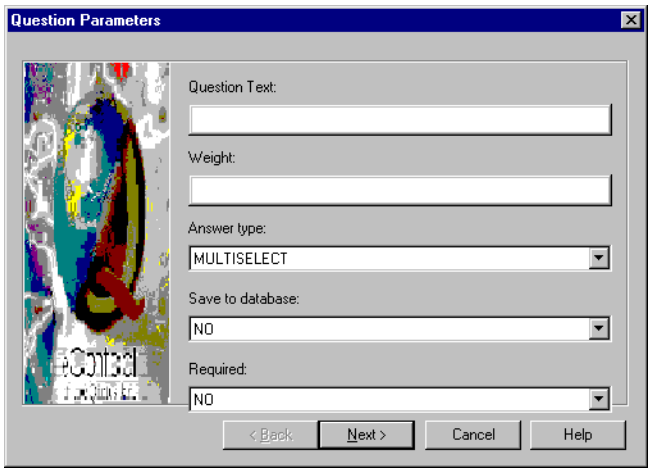
   If your workflow is not part of an open project, Workflow Designer prompts you to add the workflow to a project before the DHTML Editor opens.

5. In the **Short Description** field, enter a description for the HTML page.

Tip:
If your agent script workflow will run in a language other than English, you must configure the DHTML page with the appropriate character set. For more information, see Configuring DHTML pages for non-English languages on page 37.

Configuring Prompter questions

After you set the Page property, the Question Parameters dialog box is displayed.



How you specify the question parameters depends on which answer type you select for the question in the HTML page:

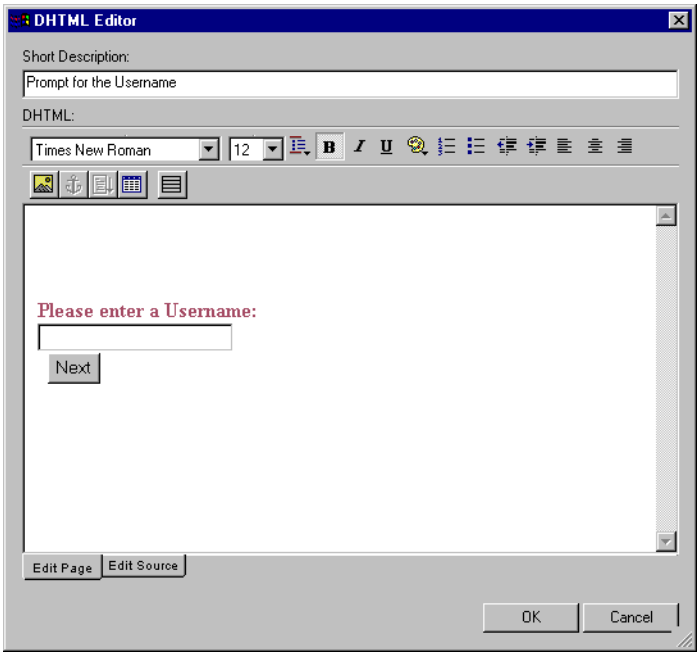
- Creating a combo box question - Agent selects an answer from a predetermined set of answers in a drop-down list.
- Creating a date question - Agent enters a date in a field.
- Creating a long text question - Agent types the answer in a text box with no restriction on the length of the answer.
- Creating a multiselect question - Agent selects one or more answers from a predetermined set of answers that are displayed with check boxes.
- Creating a question with no required answer - Agent cannot enter an answer.

- Creating a radio button question - Agent selects one answer from a predetermined set of answers that are displayed with radio buttons.
- Creating a text box question - Agent types the answer in a text box restricted to a limited number of characters.

If you create multiple questions for a Prompter block, the answers to only one question in the block can cause the workflow to branch to another block.

Note:
You must select a database connection to specify question parameters. Do not name the database connection `default`.

After you complete the question parameters, the DHTML Editor displays the questions and answers seen by the agent:



Note:
If you make changes directly in the HTML source code using the **Edit Source** tab, you must return to the **Edit Page** tab before you are allowed to select **OK** and save those changes.

Creating a combo box question

In combo box format, the answers display in a drop-down list. The agent can select only one answer from the list.

To create an HTML page with the answers in a combo box:

1. Right-click in the DHTML Editor and select **Objects** > **Prompter**.
2. In the **Question Text** field, enter the text of the question.
3. In the **Weight** field, enter the weight to be given to the answer to this question.

   A weight can be between 0 and 1, inclusive of all decimal values. The weight is a relative value that calculates the importance of a question to aid in analyzing customer responses.
4. From the **Answer type** drop-down list, select **Combo box**.
5. From the **Save to Database** drop-down list, select:
   - **Yes** - If you want the answer to be saved in the database when the workflow is completed.
   - **No** - If you do not want the answer saved in the database.
6. From the **Required** drop-down list, select:
   - **Yes** - To require the agent to enter an answer for the question before continuing the workflow.
   - **No** - If the answer is optional.
7. Select **Next**.
8. If you selected **Yes** in the **Save to Database** field:
   a. Select the database table where the workflow saves the answers.
   b. Select a field in the database table that matches the data type of the answer to be saved by the agent.
   c. Select whether to save the answer in the `qw_response` table.
   d. Select **Next**.

   If you do not want to store the answer in a database, Workflow Designer does not display this dialog box.
9. In the **Answers Datasource** dialog box, select the location where the workflow will save the answers:
   - Database (go to Step 10).
   - Text (go to Step 11).

10. If you selected **Database** in Step 9, the **Answers Database Parameters** dialog box is displayed. Select the following to identify where the workflow gets the answers:
    a. The name of the source database table from the **Source Table** drop-down list.
    b. The name of the database field in the selected table from the **Source Field** drop-down list.
11. If you selected **Text** in Step 9, enter the number of possible answers. Workflow Designer opens a series of dialog boxes. In those dialog boxes, repeated for each possible answer:
    a. In the first dialog box, enter the text of the first answer and its score.

       A score is a number between 0 and 100. You can assign a different score for each answer. If you are using answers from a database, you cannot associate a score with your answers.
    b. In the second dialog box, enter the following information to identify the snap point that takes an agent to the next block if the agent selects this answer:
       - Anchor Snap Connection Name.
       - Anchor Snap Connection Point - Select an available anchor snap from the drop-down list.
       - Anchor label - Limited to four characters. This text is displayed on the specified anchor snap in your workflow.

       If you need to add more snaps to the block, see "Creating Snap Points" in *Workflow Designer User Guide*.
12. When you have completed the series of dialog boxes for each answer, select **Finish**.

    If you correctly completed the question and answer screens, you receive a message indicating success and Workflow Designer prompts you to select **Finish**.

Creating a date question

The date answer type is used by an agent to answer a question by entering a date.

To create an HTML page with a date entry field:

1. Right-click in the DHTML Editor and select **Objects** > **Prompter**.
2. In the **Question Text** field, enter the text of the question.
3. In the **Weight** field, enter the weight to be given to the answer to this question.

   A weight can be between 0 and 1, inclusive of all decimal values. The weight is a relative value that calculates the importance of a question to aid in analyzing customer responses.
4. From the **Answer type** drop-down list, select **Date**.

5. From the **Save to Database** drop-down list, select:
   - **Yes** - If you want the answer to be saved in the database when the workflow is completed.
   - **No** - If you do not want the answer saved in the database.
6. From the **Required** drop-down list, select:
   - **Yes** - To require the agent to enter an answer for the question before continuing the workflow.
   - **No** - If the answer is optional.
7. Select **Next**.
8. If you selected **Yes** in the **Save to Database** field, in the next dialog box:
   a. Select the database table where the workflow saves the answer.
   b. Select a field in the database table with data type of `date` or `date time` to match the data type of the answer the agent will save.
   c. Select whether to save the answer in the `qw_response` table.
   d. Select **Next**.

   If you do not want to store the answer in a database, Workflow Designer does not display this dialog box.
9. Select **Finish**.

Creating a long text question

The long text answer type opens a dialog box so that an agent can enter an answer with no restriction on the length.

To create an HTML page with a long text entry field:

1. Right-click in the DHTML Editor and select **Objects** > **Prompter**.
2. In the **Question Text** field, enter the text of the question.
3. In the **Weight** field, enter the weight to be given to the answer to this question.

   A weight can be between 0 and 1, inclusive of all decimal values. The weight is a relative value that calculates the importance of a question to aid in analyzing customer responses.
4. From the **Answer type** drop-down list, select **Long Text**.
5. From the **Save to Database** drop-down list, select:
   - **Yes** - If you want the answer to be saved in the database when the workflow is completed.
   - **No** - If you do not want the answer saved in the database.

6. From the **Required** drop-down list, select:
   - **Yes** - To require the agent to enter an answer for the question before continuing the workflow.
   - **No** - If the answer is optional.
7. Select **Next**.
8. If you selected **Yes** in the **Save to Database** field, in the next dialog box:
   a. Select the database table where the workflow saves the answer.
   b. Select a field in the database table that matches the data type of the answer to be saved by the agent.
   c. Select whether to save the answer in the `qw_response` table.
   d. Select **Next**.

   If you do not want to store the answer in a database, Workflow Designer does not display this dialog box.
9. Select **Finish**.

Creating a multiselect question

In the multiselect answer type, the agent can select one or more answers by selecting the associated check boxes.

To create an HTML page with the answers next to check boxes:

1. Right-click in the DHTML Editor and select **Objects** > **Prompter**.
2. In the **Question Text** field, enter the text of the question.
3. In the **Weight** field, enter the weight to be given to the answer to this question.

   A weight can be between 0 and 1, inclusive of all decimal values. The weight is a relative value that calculates the importance of a question to aid in analyzing customer responses.
4. From the **Answer type** drop-down list, select **Multiselect**.
5. From the **Save to Database** drop-down list, select:
   - **Yes** - If you want the answer to be saved in the database when the workflow is completed.
   - **No** - If you do not want the answer saved in the database.
6. From the **Required** drop-down list, select:
   - **Yes** - Select this option if the agent must supply an answer before the workflow is allowed to continue.
   - **No** - Select this option if answering this question is optional for the agent.

7. Select **Next**.
8. If you selected **Yes** in the **Save to Database** field, in the next dialog box:
   a. Select the database table where the workflow saves the answer.
   b. Select a field in the database table that matches the data type of the answer to be saved by the agent, for example `varchar` or `text`.
   c. Select whether to save the answer in the `qw_response` table.
   d. Select **Next**.

   If you do not want to store the answer in a database, Workflow Designer does not display this dialog box.
9. In the **Answers Datasource** dialog box, select the location where the workflow will save the answers:
   - Database (go to Step 10).
   - Text (go to Step 11).
10. If you selected **Database** in Step 9, the **Answers Database Parameters** dialog box opens. Select the following to identify where the workflow gets the answers:
    a. The name of the source database table from the **Source Table** drop-down list.
    b. The name of the database field in the selected table from the **Source Field** drop-down list.
11. If you selected **Text** in Step 9, enter the number of possible answers. Workflow Designer opens a series of dialog boxes. In those dialog boxes, repeated for each possible answer:
    a. In the first dialog box, enter the text of the first answer and its score.

       A score is a number between 0 and 100. You can assign a different score for each answer. If you are using answers from a database, you cannot associate a score with your answers.
    b. In the second dialog box, enter the following information to identify the snap point that takes an agent to the next block if the agent selects this answer:
       - Anchor Snap Connection Name.
       - Anchor Snap Connection Point - Select an available anchor snap from the drop-down list.
       - Anchor label - Limited to four characters. This text is displayed on the specified anchor snap in your workflow.

       If you need to add more snaps to the block, see "Creating Snap Points" in *Workflow Designer User Guide*.
12. When you have completed the series of dialog boxes for each answer, select **Finish**.

    If you correctly completed the question and answer screens, you receive a message indicating success and Workflow Designer prompts you to select **Finish**.

Creating a question with no required answer

The None answer type lets you provide information needed by the agent to complete the task in the workflow, but that does not require an answer from the agent or contact.

To create an HTML page with no answer field:

1. Right-click in the DHTML Editor and select **Objects** > **Prompter**.
2. In the **Question Text** field, enter the text of the information for the agent.
3. In the **Weight** field, enter 0.

   A weight can be between 0 and 1, inclusive of all decimal values. The weight is a relative value that calculates the importance of a question to aid in analyzing customer responses.
4. From the **Answer type** drop-down list, select **None**.
5. From the **Save to Database** drop-down list, select **No**.
6. From the **Required** drop-down list, select **No**.
7. Select **Next**.
8. Select **Finish**.

   The DHTML Editor displays the text as the agents see it in their browsers.

Creating a radio button question

With the radio answer type, the answers display next to radio buttons. The agent can select only one answer by selecting the associated radio button.

To create an HTML page with the answers next to radio buttons:

1. Right-click in the DHTML Editor and select **Objects** > **Prompter**.
2. In the **Question Text** field, enter the text of the question.
3. In the **Weight** field, enter the weight to be given to the answer to this question.

   A weight can be between 0 and 1, inclusive of all decimal values. The weight is a relative value that calculates the importance of a question to aid in analyzing customer responses.
4. From the **Answer type** drop-down list, select **Radio**.
5. From the **Save to Database** drop-down list, select:
   - **Yes** - If you want the answer to be saved in the database when the workflow is completed.
   - **No** - If you do not want the answer saved in the database.

6. From the **Required** drop-down list, select:
   - **Yes** - To require the agent to enter an answer for the question before continuing the workflow.
   - **No** - If the answer is optional.
7. Select **Next**.
8. If you selected **Yes** in the Save to Database field, in the next dialog box:
   a. Select the database table where the workflow saves the answer.
   b. Select a field in the database table that matches the data type of the answer to be saved by the agent, for example `varchar` or `text`.
   c. Select whether to save the answer in the `qw_response` table.
   d. Select **Next**.

   If you do not want to store the answer in a database, Workflow Designer does not display this dialog box.
9. In the **Answers Datasource** dialog box, select the location where the workflow will save the answers:
   - Database (go to Step 10).
   - Text (go to Step 11).
10. If you selected **Database** in Step 9, the **Answers Database Parameters** dialog box, opens. Select the following to identify where the workflow gets the answers:
    a. The name of the source database table from the **Source Table** drop-down list.
    b. The name of the database field in the selected table from the **Source Field** drop-down list.
11. If you selected **Text** in Step 9, enter the number of possible answers.
12. Workflow Designer opens a series of dialog boxes. In those dialog boxes, repeated for each possible answer, perform the following steps:
    a. In the first dialog box, enter the text of the first answer and its score.

       A score is a number between 0 and 100. You can assign a different score for each answer. If you are using answers from a database, you cannot associate a score with your answers.

b. In the second dialog box, enter the following information to identify the snap point that takes an agent to the next block if the agent selects this answer:

- Anchor Snap Connection Name.
- Anchor Snap Connection Point - Select an available anchor snap from the drop-down list.
- Anchor label - Limited to four characters. This text is displayed on the specified anchor snap in your workflow.

If you need to add more snaps to the block, see "Creating Snap Points" in *Workflow Designer User Guide*.

13. When you have completed the series of dialog boxes for each answer, select **Finish**.

If you correctly completed the question and answer screens, you receive a message indicating success and Workflow Designer prompts you to select **Finish**.

Creating a text box question

The text box answer type opens a dialog box that lets an agent type in an answer that is restricted to a limited number of characters. If you save the answer to the database, the text box is limited to the number of characters in the field.

To create an HTML page with a text box entry field:

1. Right-click in the DHTML Editor and select **Objects** > **Prompter**.
2. In the **Question Text** field, enter the text of the question.
3. In the **Weight** field, enter the weight to be given to the answer to this question.

   A weight can be between 0 and 1, inclusive of all decimal values. The weight is a relative value that calculates the importance of a question to aid in analyzing customer responses.
4. From the **Answer type** drop-down list, select **Text Box**.
5. From the **Save to Database** drop-down list, select:
   - **Yes** - If you want the answer to be saved in the database when the workflow is completed.
   - **No** - If you do not want the answer saved in the database.
6. From the **Required** drop-down list, select:
   - **Yes** - To require the agent to enter an answer for the question before continuing the workflow.
   - **No** - If the answer is optional.
7. Select **Next**.

8. If you selected **Yes** in the **Save to Database** field, in the next dialog box:
   a. Select the database table where the workflow saves the answer.
   b. Select a field in the database table that matches the data type of the answer to be saved by the agent.
   c. Select whether to save the answer in the `qw_response` table.
   d. Select **Next**.
9. If you selected **No** in the **Save to Database** field, in the next dialog box select one of the following formats for the agent to input data into the text box:
   - **Email** - `x@y.z` (for example, `johndoe@anyisp.com`).
   - **Literal** - Only the following characters: (_) (underscore), lower and uppercase alphabetical characters.
   - **None** - Any alphanumeric or standard keyboard character.
   - **Numeric** - Only numbers.
   - **Phone - US** - Any of the following telephone formats: (111) 222-333, 111 222 3333, 111-222-3333, or 1-111-222-3333.
10. Select **Finish**.

Chapter 7: Database objects

A database object allows an agent to access information in a database table as they work through an application script workflow. Each database object represents a step in an application task, such as taking an order or entering a complaint from a customer.

Application tasks typically require an agent to access information in a database:

- To add new data. For example, an agent can add a new order for a customer.
- To modify existing data. For example, an agent can change a customer's address.

To access database information, an application script includes database objects, such as:

- A form that Prompter fills with information from the database
- A browser that displays the results of a search of the database

After you design the script and know which steps use database objects, you can set the Page property and create the database objects for the Application blocks.

Note:

You cannot create Prompter questions and answers in HTML pages attached to Application blocks. A workflow can contain both Application and Prompter scripts, but they must display on different HTML pages.

This section describes how to set the block text and page property for an application block, and how to create a database object to display on the HTML page of an application script. This section includes the following topics that describe steps you perform to create an HTML page for an application script:

1. Setting the block text and Page property on page 53.
2. .Creating database objects on page 53

Setting the block text and Page property

The Page property in the **Basic** tab includes a short description that identifies the HTML page. The block text is the title that displays on the block in the Work Area to identify the purpose of the block in the workflow.

To set the block text and Page property for the HTML page that represents a step in the script:

1. Select a Prompter block in your workflow.
2. Add the block text:
   a. Select the **Expert** tab in the Property sheet.
   b. Select **Text** and enter a description of the block's purpose. For example, Contact Status.
3. Select the **Basic** tab in the Property sheet to display the Page property.
4. Select the **Ellipsis (…)** button next to the Page property to open the DHTML Editor.

   If your workflow is not part of an open project, Workflow Designer prompts you to add it to a project before the DHTML Editor opens.
5. In the **Short Description** text box, enter a description for the HTML page.

Tip:

If your agent script workflow will run in a language other than English, you must configure the DHTML page with the appropriate character set. For more information, see Configuring DHTML pages for non-English languages on page 37.

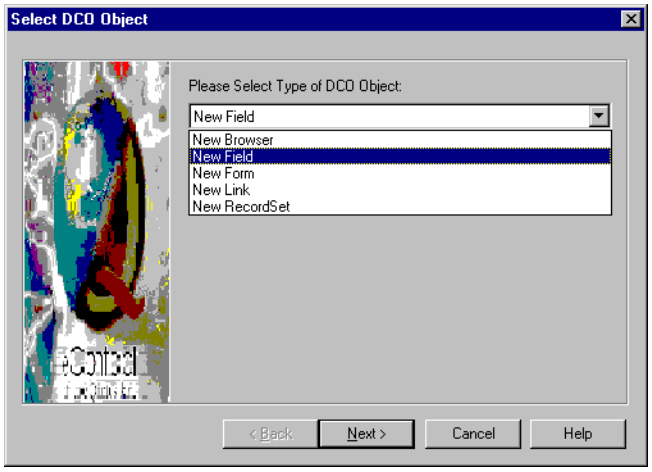
Creating database objects

The Database Wizard assists you in creating the following database objects in agent scripts:

- Creating a browser object - Displays the results of a database search in a scrolling list box
- Creating a field object - Displays the contents of a field from a database table in a text box
- Creating a form object - Lets an agent collect and search for information in an application page
- Creating a link object - Causes an agent script to branch to another block in the workflow
- Creating a record set object - Lets an agent view or update records in the database

Note:
To create a database object, you must be familiar with the structure and content of your database and with the Database Designer description of your database.

The database objects are displayed in a drop-down list in the **Select DCO Object** dialog box.



Note:
If you make changes directly in the HTML source code using the **Edit Source** tab, you must return to the **Edit Page** tab before you can select **OK** to save your changes. If you select **Cancel** in the **Edit Source** tab, the DHTML Editor closes without saving and any changes are then lost.

For more information about database objects, and the structure and data model of your Avaya agent application, see *IC Database Designer Application Reference* and the Data Model Reference for your application.

Creating a browser object

A browser object is a scrolling list box that displays the results of a database search. Before you create a browser object for your script, the browser must be in the ADL file. If the browser does not exist, see *IC Database Designer Application Reference*.

To create a browser object

1. Right-click in the DHTML Editor:
   a. Select **Objects** > **Database**.
   b. Select **New Browser** from the drop-down list.

2. In the **Browser Page** dialog box, do the following actions:
   a. From the **Table Alias** drop-down list. select the alias for the database table that contains the data to be displayed by the browser.

      The database connection determines the list of available tables. See "Specifying Project Settings" in *Avaya Workflow Designer User Guide* for more information.

   b. In the **Browser Mode** field, select one of the following:
      - **None** - If the agent can only view the data.
      - **Select** - If the agent can select a set of records for a detailed view in a Record Set object. For more information, see Creating a record set object on page 59.
      - **Update** - If the agent can modify the data. Browsers let agents modify one field of information for multiple records simultaneously.

   c. Select **Next**.
3. In the second **Browser Page** dialog box, do the following actions:
   a. Select the browser from the **Browser** drop-down list.

      The list contains all browsers that are associated with the table alias that you selected in the previous dialog box.

   b. Select the relation set associated with the browser from the **Relation Set** drop-down list.

      A relation set is a group of relations. A relation is an association between two tables. The selected relation set constrains the information to be displayed in this browser, by restricting the search to the database tables associated with the selected relation set.

   c. If you selected **Update** mode in the previous dialog box, enter the name of the column that agents can update.

   d. Select **Next**.
4. Select **Finish**.

   The DHTML Editor displays the Web page as it is displayed in the Web browser of an agent. You cannot alter the order of columns in the Web page, nor can you split the columns to display in different places within the page. If you move a Web page, you must move all the columns of the Web page.

Creating a field object

Field objects display the contents of a database field in a text box.

To create a field object:

1. Right-click in the DHTML Editor:
   a. Select **Objects** > **Database**.
   b. Select **New Field** from the drop-down list.
2. In the **Field Page** dialog box, do the following actions:
   a. Select the table alias for the table that contains the field from the drop-down list.
   b. Select **Next**.
3. In the **Select Field** dialog box, do the following actions:
   a. Select the field from the drop-down list.
   b. Select **Next**.
4. Select **Finish**.

Creating a form object

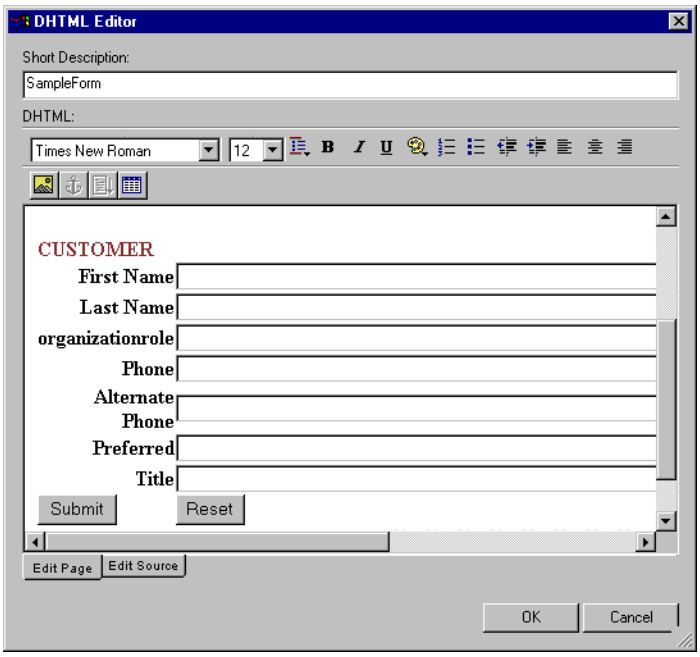
Form objects let agents do the following tasks:

- Collect information - For example, you can create a form that lists products from a database record. You can give each product in the list a check box that an agent can use to indicate a customer response.
- Search for information - The forms display the results of database searches in a browser.

The agent application displays the contents of forms in a table format. Each value is displayed in two connected cells. One cell contains the label and the other contains the value. When you select a cell, you select both the label and the value. You can add or delete rows from the table in the DHTML Editor.

Note:
For form objects, the DHTML Editor does not support drag-and-drop editing. To move items in a table, cut the row, insert a new row, and paste the cut row. The DHTML Editor also inserts an empty row for each `<SPAN>` tag in the HTML. The application does not display the empty rows. Do not delete the extra `<SPAN>` row because this deletes the row associated with the tag.

The following figure shows an example of a form displayed in the DHTML editor.



To create a form object:

1. Right-click in the DHTML Editor:
   a. Select **Objects** > **Database**.
   b. Select **New Form** from the drop-down list.
2. In the first **Form Parameters** dialog box, do the following actions:
   a. Select a table alias from the **Table Alias** drop-down list.

      The table alias determines which table of database information is available on the form.
   b. From the **Form Mode** list, select one of the following:
      - **Query** - Creates a form that searches for a match to text that an agent enters into a field
      - **Update** - Create a form that lets an agent enter or update information
   c. Select **Next**.

3. In the second **Form Parameters** dialog box, do the following actions:
   a. From the **Target Browser** list, select an available browser from the drop-down list.

      The table alias determines which browsers are available. For more information about how to create browsers, see *IC Database Designer Application Reference.*

   b. From the **Relation Set** list, select a relation set.

      A relation is an association between two tables. A relation set is a group of relations. The selected relation set constrains the information to be displayed in this browser, by restricting the search to the database tables associated with the selected relation set.

   c. Select **Next**.
4. In the **Form Fields** dialog box, do the following actions:
   a. Select the fields that you want to include in the form.

      Each field that you select adds a field on your form where an agent can enter text. Select a highlighted field to remove the field from the form.

   b. Select **Next**.
5. Select **Finish**.

Creating a link object

Link objects let agents move the agent script to another block in the workflow.

To create a link object:

1. Right-click in the DHTML Editor:
   a. Select **Objects** > **Database**.
   b. Select **New Link** from the drop-down list.
2. In the first **Link Parameters** dialog box, do the following actions:
   a. Enter the name of an IC Script that the application runs before branching in the IC Script field. This field is optional.
   b. From the **Action** list, select the action to be performed after branching:
      - **Change** - Links a Record Set page to an Update Form page, and lets agents change records
      - **New** - Links to an Update Form page, and lets agents create new records
      - **None** - Specifies that no pre-set action be performed
   c. Select **Next**.

3. In the second **Link Parameters** dialog box, do the following actions:
   a. In the **Anchor Snap Connection Name** field, enter the name of the anchor snap on the block where this link takes the agent.

      For more information about anchor snaps, see "Connecting Blocks" in *Avaya Workflow Designer User Guide*.

   b. From the **Anchor Snap Connection Point** drop-down list, select the location of the anchor snap on the current block that this link uses.

      The drop-down list includes all unused connection points on the block. If you have already assigned a connection point to an anchor snap, that connection point is not in the list.

   c. In the **Anchor Label** field, enter the anchor name displayed on the anchor snap.

      The label can be a maximum of four characters.

   d. Enter the text that the agent sees in the script in the **Link Text** field.

      For example, you can enter `Click here to enter a new customer record`.

   e. Select **Next**.
4. Select **Finish**.

Creating a record set object

Record set objects let agents view and update records in the database.

To create a record set object:

1. Right-click in the DHTML Editor:
   a. Select **Objects** > **Database**.
   b. Select **New Record Set** from the drop-down list.
2. In the **Record Set Page** dialog box:
   a. Select the table alias containing the records to be displayed from the drop-down list.
   b. Select **Next**.
3. Select **Finish**.

The DHTML Editor displays all of the fields which make up a complete record. The agent can use the buttons in the HTML page to display the previous or next record. You can move or delete entries, but other editing features of the DHTML Editor are not available for Record sets.

Chapter 8: Testing and monitoring workflows

This section provides information and procedures to help you test your workflows and ensure that workflow operations are occurring as expected.

This section contains the following topics:

Testing agent script workflows

You can test an agent script workflow in any supported browser. If the workflow includes alternative paths, depending upon the answer to a question, follow every possible path in the workflow to make sure that the script works as designed and the HTML pages display properly.

For information on supported browsers, see *IC Installation Planning and Prerequisites*.

To run a Prompter agent script workflow from an HTTP prompt in a browser:

1. In Internet Explorer, enter the following syntax in the Address or Location box:

   `http://<machine>:<httpConnectorServer_httpport>/<projectname>_shell.htm`

   For example, `http://dev1:9503/listsurvey_shell.htm`

   where:

Variable	Description
`<machine>`	The name of the machine on which the workflow resides.
`<httpport>`	The port number.
`<projectname>`	The name of the project containing this workflow.

2. Select the workflow name to run the agent script workflow.

Monitoring workflow performance

You can trace the execution of the workflow in the Workflow server and monitor how long the server takes to execute the workflow. Problems can arise if the execution of the workflow takes too long or if the workflow sends events too rapidly to an assigned client.

CAUTION:
If a workflow sends events faster than the agent desktop application can process them, the application loses the connection to the Workflow server. If this happens, the workflow eventually encounters a fatal error as it continues to issue events.

Analyzing the Timer information in the Workflow server log file tells you how long the server took to execute the following:

- Flow
- Block
- Vesp method
- Sleep statement

You should monitor workflow performance in a development environment. If you use the monitor feature with the Workflow Debugger, the breakpoints and other features of the Workflow Debugger cause the workflow to run more slowly and can stop the workflow during execution.

Note:
Do not enable Trace Execution of Flows in a production Workflow server unless you are looking for a performance problem. The server logs concise strings to minimize the impact, but the logging of timing data inevitably slows down the execution of a workflow.

The Workflow server logs two kinds of timing entries: Start and End. Start and End tags for a given thread pair and nest, unless an error occurs and the workflow stops.

Start entries use the form `//TIMER:Sx tid @info` where:

- `x` indicates what is being timed:
  - F - workflow
  - B - block
  - M - vesp method
  - S - sleep statement
- `tid` is a unique integer for each existing thread. Timer entries for various threads can overlap. When they overlap, the `tid` becomes important because more than one thread runs at a time.
- `info` depends on the kind of timer designated for `x`:
  - flow - The workflow name (flow.qfd)

- block - The block name (Start)
- vesp method - the first few characters of the method request (`[DS.GetFewRecords("","type=srv")`
- sleep statement - number of ms requested (120)

  For example, the start entry of a workflow could be similar to the following example:

```
//TIMER:SF 11769532 @t4.code
```

End entries use the form `//TIMER:Ex tid timems {@info}` where:

- `x` and `tid` have the same meaning as in Start entries
- `time` is a floating point number of ms
- `@info` is optional and gives additional information:
  - flow - the workflow name
  - block - the block name

The following is an example from the log of a workflow with one block that performs a vesp request, a sleep, then another vesp request:

```
//TIMER:SF 11708256 @t4.code
//TIMER:SB 11708256 @Start
//TIMER:SM 11708256 @[DS.GetFewRecords("","type=srv",
//TIMER:EM 11708256     78.00ms
//TIMER:SS 11708256 @120
//TIMER:ES 11708256    125.00ms
//TIMER:SM 11708256 @[DS.GetFewRecords("","type=per",
//TIMER:EM 11708256     93.00ms
//TIMER:EB 11708256    609.00ms @Start
//TIMER:EF 11708256    609.00ms @t4.code
```

The granularity of the timers varies by platform. You cannot retrieve precise values under the best of circumstances. However, in the above example, the two vesp requests consumed 78+93=172 ms (of 609 ms), and the sleep statement consumed another 125 ms. These three requests account for half of the running time of the block.

Note:
The timer does not include workflow loading time. Workflows are loaded only once, regardless of how often they are used. Workflows rarely require an extended period of time to be loaded.

To monitor workflow performance:

1. In IC Manager:
   a. Select the **Servers** tab and double-click the Workflow server.
   b. Select the **Workflow** tab in the **Workflow server settings** dialog box.
   c. Check **Trace Execution of Flows**.

d. Select **OK**.

2. Run the workflow.
3. After the workflow has stopped running, open the Workflow server log and review the entries under `//TIMER:`.

   The Workflow server log is located with all other Avaya logs in the following directory: *IC_INSTALL_DIR*`\IC73\logs`

Index

E

F

H

I

L

M

N

O

P

Q

R

S

T

U

V

W