



# Oracle Knowledge AnswerFlow Implementation and User's Guide

---

*A Guide to Creating and Implementing AnswerFlow Guided Processes*

---

Release 8.5

Document Number OKAF-UG85-01

December, 2012

---

## COPYRIGHT INFORMATION

Copyright © 2002, 2012, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

### U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are “commercial computer software” or “commercial technical data” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. Other names may be trademarks of their respective owners.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

---

# Contents

---

<b>About This Guide</b>	<b>1</b>
In This Guide	1
Examples of Product Screens and Text	1
Operating System Variations in Examples and Procedures	1
References to World Wide Web Resources	2
<b>Introduction to AnswerFlow</b>	<b>3</b>
The AnswerFlow Application	3
The AnswerFlow Editor	3
The AnswerFlow Datastore	3
The AnswerFlow Runtime User Interface	4
AnswerFlow Processes	4
Context Variables	4
Service Calls	4
Pages and Page Elements	5
Page Elements	5
Logical Nodes	5
Sub-Processes	5
<b>Configuring and Administering AnswerFlow</b>	<b>6</b>
Installing AnswerFlow	6
Managing the AnswerFlow Application	6
Starting and Stopping AnswerFlow	6
Managing AnswerFlow Users	7
Managing AnswerFlow Data	7
Exporting AnswerFlow Data	7
Modifying the Archive Name	8
Importing AnswerFlow Data	8
<b>Creating AnswerFlow Processes</b>	<b>10</b>
Logging onto the AnswerFlow Editor	10
The AnswerFlow Editor	11
Creating Process Components	11
Defining Context Variable Values	12
Defining Service Calls	12
Adding Page Elements and Logical Nodes to a Process	14
Page Elements	14
Logical Nodes	14
Adding a Display Form Page Element to a Process	15
Defining Form Control Parameters	17
Adding a Display IM Document Page Element to a Process	17
Adding a Display HTML Page Element to a Process	18
Adding a Branch Logical Node to a Process	19

Adding a Service Call Logical Node to a Process .....	20
Adding a Set Variable Values Logical Node to a Process .....	21
Adding a Process Call Logical Node to a Process .....	21
AnswerFlow Process Review View .....	22
AnswerFlow Process Build View .....	23
<b>Deploying AnswerFlow Processes .....</b>	<b>24</b>
Integrating AnswerFlow Processes into a Production User Interface .....	24
Using the Sample User Interface .....	24
Configuring Access to Information Manager .....	25
Compiling the Sample User Interface .....	26
Deploying the Sample User Interface .....	26
Viewing the Sample User Interface .....	26
Associating Processes with User Actions .....	27

# About This Guide

This guide provides instructions and supporting information for implementing and using AnswerFlow within an Oracle Knowledge application. This guide is intended for application developers, business analysts, and systems administrators who need to understand how to create and implement AnswerFlow processes.

## In This Guide

The Oracle Knowledge AnswerFlow Implementation and User's Guide is divided into the following sections:

<b>Chapter 1, "Introduction to AnswerFlow"</b>	This section describes AnswerFlow and its application components, and provides an overview of AnswerFlow processes.
<b>Chapter 2, "Configuring and Administering AnswerFlow"</b>	This section describes how to manage the AnswerFlow application and data.
<b>Chapter 3, "Creating AnswerFlow Processes"</b>	This section describes how to create and edit AnswerFlow processes, sub-processes, and their various components.
<b>Chapter 4, "Deploying AnswerFlow Processes"</b>	This section describes how to deploy processes to the runtime environment and the sample user interface.

## Examples of Product Screens and Text

The product screens, screen text, and file contents depicted in the documentation are examples. We attempt to convey the product's appearance and functionality as accurately as possible; however, the actual product contents and displays may differ from the published examples.

## Operating System Variations in Examples and Procedures

We generally use Linux screen displays and naming conventions in our examples and procedure descriptions. We include other operating system-specific procedures or steps as noted in section headings, or within topics, as appropriate.

We present command syntax, program output, and screen displays:

- in Linux format first
- in other Unix-specific variants only when necessary for proper operation or to clarify functional differences
- in Windows format only when necessary for clarity

## References to World Wide Web Resources

For your convenience, we refer to Uniform Resource Locators (URLs) for resources published on the World Wide Web when appropriate. We attempt to provide accurate information; however, these resources are controlled by their respective owners and are therefore subject to change at any time.

# Introduction to AnswerFlow

Oracle Knowledge AnswerFlow is a guided knowledge delivery application that enables you to provide precise and dynamic automated answers and user assistance for complex customer questions.

AnswerFlow enables you to create and deploy automated interactive processes that guide users toward appropriate answers or solutions in cases where:

- **Answers are conditional**, and can vary based on factors such as account status, location, or specific product or model
- **Diagnosis is complex**, and identifying the best response among many possible answers involves asking detailed questions and eliminating alternatives
- **Answers exist in various forms**, and delivering the right one can involve serving an Information Manager article, executing an Intelligent Search question, or making calls to internal or external systems

AnswerFlow's editor enables designers, analysts, and developers to create and refine dynamic service processes to meet complex customer needs.

## The AnswerFlow Application

AnswerFlow includes the following components:

- the Editor, which you use to create and manage AnswerFlow processes, context variables, and service calls
- the datastore, which stores the AnswerFlow objects
- the Runtime user interface (RuntimeUI), which you can use to test processes in development, and which provides a basis for integrating AnswerFlow processes with your production application.

## The AnswerFlow Editor

The Editor is a web-based user interface that is installed and configured as part of the standard installation process. You use the editor to create and manage AnswerFlow processes and their various components as described in “Creating AnswerFlow Processes ” on page 10.

**Important!** If you are using the AnswerFlow Editor with the Internet Explorer Release 7 or Release 8 browser, you must install the Google Chrome Frame. See *Installing and Configuring Oracle Knowledge* for more information.

## The AnswerFlow Datastore

The AnswerFlow installation process creates a dedicated datastore that contains sample AnswerFlow objects, such as processes, context variables, and service calls. The AnswerFlow Editor creates and stores

objects in JSON (JavaScript Object Notation) format, and the application uses SVN to manage the data. The AnswerFlow RuntimeUI accesses these objects and presents them in a sample web user interface. You can move AnswerFlow data between Editor and RuntimeUI instances in development, staging, and production environments using the processes described in “Configuring and Administering AnswerFlow” on page 6.

## The AnswerFlow Runtime User Interface

The AnswerFlow installation process installs and configures the AnswerFlow RuntimeUI. You can customize the Runtime UI to conform to the look and feel of your end-user facing application using the processes described in “Deploying AnswerFlow Processes” on page 24.

## AnswerFlow Processes

An AnswerFlow process is a set of steps that you create to guide end-users to specific goals, such as troubleshooting complex issues or enrolling in programs or services. Processes can collect and validate user input, make calls to sub-processes and available services, and deliver content based on the data and built-in business logic.

AnswerFlow processes comprise various components that you create and manage independently, and which can be reused among multiple processes. Process components can include:

- context variables
- service calls
- page elements
- logical nodes
- sub-processes

## Context Variables

Context variables represent contextual information that a user submits in a process. For example, to assist a customer, it might be necessary to get the customer’s account number. A context variable would be designed to prompt for this information and then to use this information appropriately. You can define context variables while creating a new process or you can define some basic context variables independent of a process. All defined context variables are stored and available to all users when creating a process.

## Service Calls

Service calls are Java calls that perform an action within, or retrieve data from, external resources, such as web services, file servers, and database servers. For example, you could use a service call to retrieve a customer’s equipment model number within your customer database so that the process will display the most relevant solutions. Service Calls are one type of Logical Node, as described in “Logical Nodes” on page 5.



## Pages and Page Elements

Page Elements (or Display Elements) are a means to display text within a process to provide explanation and guidance to end users. They also provide a means to solicit information from users that processes can then use to determine the best solution. Pages are containers of one or more Page Elements.

## Page Elements

Page elements are forms or documents that you display to users within a process. Page Elements include:

- **Display Forms**, which provide forms to capture user input. You define the form contents by specifying various form controls, such as password fields and radio button options.
- **Display IM Documents**, which display Information Manager articles. You specify one or more Information Manager document IDs to display.
- **Display HTML**, which display formatted HTML. You specify the HTML to display as a constant or variable.

## Logical Nodes

Logical Nodes map the logical flow of a process and also create Service Calls, which retrieve information from external systems. Types of Logical Nodes include:

- **Branch**, which splits the process into divergent paths based on conditional logic, usually by evaluating a user response
- **Service Call**, as described in “Service Calls” on page 4
- **Process Call**, which references a previously defined process (sub-process), which then becomes part of the larger process

## Sub-Processes

Sub-processes are processes that you create and use as modules within a process. When you edit a sub-process that is used in one or more larger processes, your changes will affect all the processes that use the sub-process.

# Configuring and Administering AnswerFlow

This section describes how to configure and administer AnswerFlow in a typical enterprise environment, including managing the application, administering users and user security, and managing data within and between development, staging, and production environments.

## Installing AnswerFlow

You install AnswerFlow using the process described in the *Oracle Knowledge Installation Guide*. You must install AnswerFlow on any instance on which you want to:

- create and manage processes and their related data
- present processes as answers to users' questions

When installation is complete, you can begin working with AnswerFlow by logging onto the AnswerFlow editor at the URL that was configured during installation.

## Managing the AnswerFlow Application

You can manage various aspects of the AnswerFlow application, including starting and stopping the application, user authorization, and storing, exporting, and importing AnswerFlow data.

## Starting and Stopping AnswerFlow

You can start and stop an AnswerFlow instance using scripts that start and stop the application server for the instance.

**Note:** You must be within the ICE environment to use the start and stop commands. See *Installing and Configuring Oracle Knowledge* for more information about the ICE environment.

To start and stop the AnswerFlow instance,

- open a terminal window and navigate to:

```
<answerflow_installation_directory>/AnswerFlow/instances/<instance_name>
```

where:

<instance\_name> is the name of the instance, for example Editor

- execute the command to start the ICE environment

```
./setenv.sh.
```

- start, stop, and restart the instance by executing the appropriate command:

```
inquiraaaf.sh start
```

```
inquiraaaf.sh stop
```

```
inquiraaaf.sh restart
```

## Managing AnswerFlow Users

The installation process prompts you to configure an instance of Information Manager that AnswerFlow will use to authenticate users and retrieve articles for use within AnswerFlow processes. All defined Information Manager repository users are authorized to use the AnswerFlow editor.

See the *Oracle Knowledge Information Manager Administration Guide*, Chapter 6, “Managing Users” for more information.

## Managing AnswerFlow Data

AnswerFlow manages the data that you create, update, and delete using the web-based editor user interface in a dedicated datastore. The datastore is local to the instance; as you develop, test, and publish processes, you need to make the data available to the various environments in which you perform these tasks.

You make AnswerFlow processes available to other AnswerFlow instances within development, staging, and production environments by:

- exporting the data from an instance as described in “Exporting AnswerFlow Data” on page 7
- importing the data into an instance as described in “Importing AnswerFlow Data” on page 8

You can export data from, and import data into, both Editor and RuntimeUI instances.

## Exporting AnswerFlow Data

You can export AnswerFlow data from Editor and RuntimeUI instances. The export process exports the entire datastore as a compressed (.zip) archive. To export data:

- open a terminal window and navigate to

```
<answerflow_installation_directory>/AnswerFlow/instances/<instance_name>
```

where:

<instance\_name> is the name of the instance, for example Editor

- execute the command to start the ICE environment

```
./setenv.sh
```

To export data:

- execute the command:

```
exportData.sh
```

The export process copies the data into a .zip archive and places the archive in the following location:

```
<answerflow_installation_directory>/AnswerFlow/data/exports
```

The export process names the archive based on the instance type (Editor or RuntimeUI) and the date and time at which the archive was created. For example, if you export data from an Editor on Sept. 14, 2012, at 6:46:46 PM, the script will name the archive:

```
exportedEditorData-2012_09_14-18_46_46.zip
```

## Modifying the Archive Name

You can modify the name of the export data archive by specifying additional text as an argument to the `exportData.sh` command, for example:

```
exportData.sh MyData
```

**Note:** The argument appends the modification to the base name that the script assigns to the archive; it does not replace the base archive name. For example, if you specify the argument with the text `MyData` and export data from an Editor on Sept. 14, 2012, at 6:46:46 PM, the script will name the archive:

```
exportedEditorData-2012_09_14-18_46_46-MyData.zip
```

You can use the custom text argument when you export data from an Editor or a RuntimeUI instance.

## Importing AnswerFlow Data

You can import AnswerFlow data that has been exported from an instance into either an Editor or a RuntimeUI instance.

When you import data, the import process backs up the existing datastore to a separate location, then deletes the existing data from the datastore before copying in and expanding the newly imported archive. When the import process is complete, the datastore will contain only the newly imported data.

**Important!** The newly imported data will be available the next time the server starts after the import process completes. If the server is running during the import process, you will need to restart the server to make the imported data available to the application.

To import AnswerFlow data:

- open a terminal window and navigate to

```
<answerflow_installation_directory>/AnswerFlow/instances/<instance_name>
```

where:

`<instance_name>` is the name of the instance, for example `RuntimeUI`

- execute the command to start the ICE environment:

```
./setenv.sh.
```

- execute the command:

```
importData.sh <data_file_name>.zip
```

where:

`<data_file_name>` is the name of a previously exported AnswerFlow data archive.

The import process automatically:

- creates a backup of the target instance's existing datastore at the location:

```
<answerflow_installation_directory>/AnswerFlow/instances/<instance_name>
```

- copies and expands the archive into the instance's datastore in the folder:

```
<answerflow_installation_directory>/AnswerFlow/data/<target_instance_name>/SVN/  
Repositories/PM
```

**Note:** You can specify not to create the backup copy using the `--no-backup` argument. You must specify `--no-backup` as the first argument to the import command.

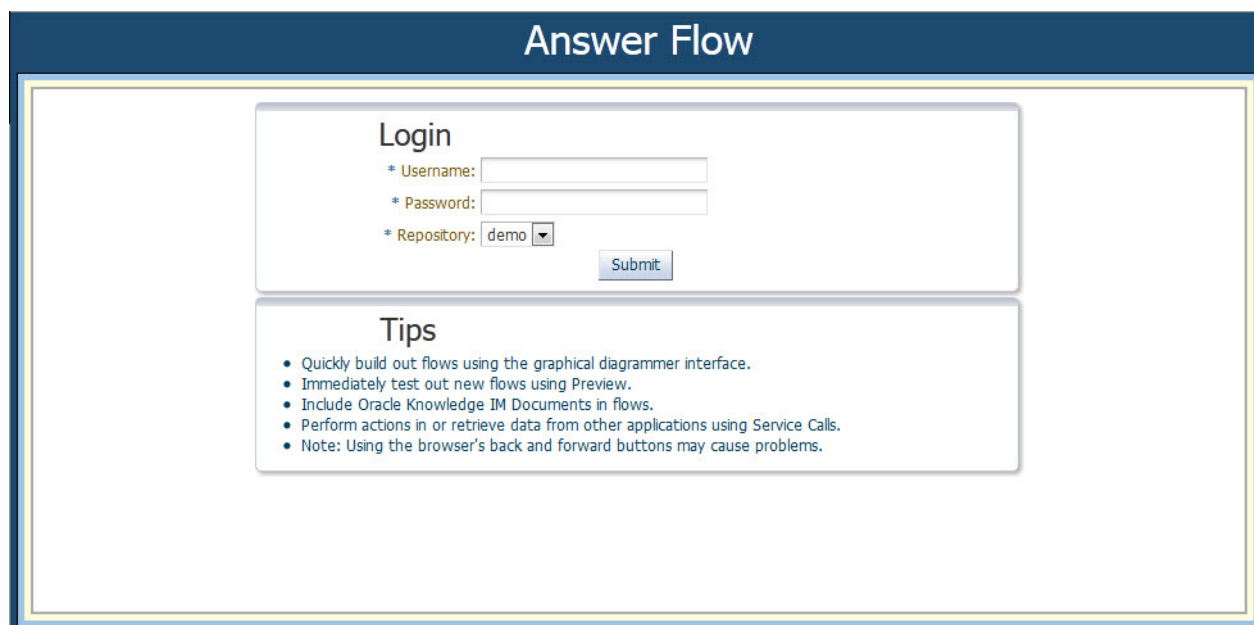
# Creating AnswerFlow Processes

This section describes how to create and edit AnswerFlow processes, sub-processes, and their various components. You create and edit AnswerFlow processes by:

- logging onto the AnswerFlow editor
- creating process components
- creating the process
- testing and validating the process

## Logging onto the AnswerFlow Editor

You log onto the AnswerFlow Editor at the URL provided by your system administrator:



**Answer Flow**

**Login**

\* Username:

\* Password:

\* Repository:

**Tips**

- Quickly build out flows using the graphical diagrammer interface.
- Immediately test out new flows using Preview.
- Include Oracle Knowledge IM Documents in flows.
- Perform actions in or retrieve data from other applications using Service Calls.
- Note: Using the browser's back and forward buttons may cause problems.

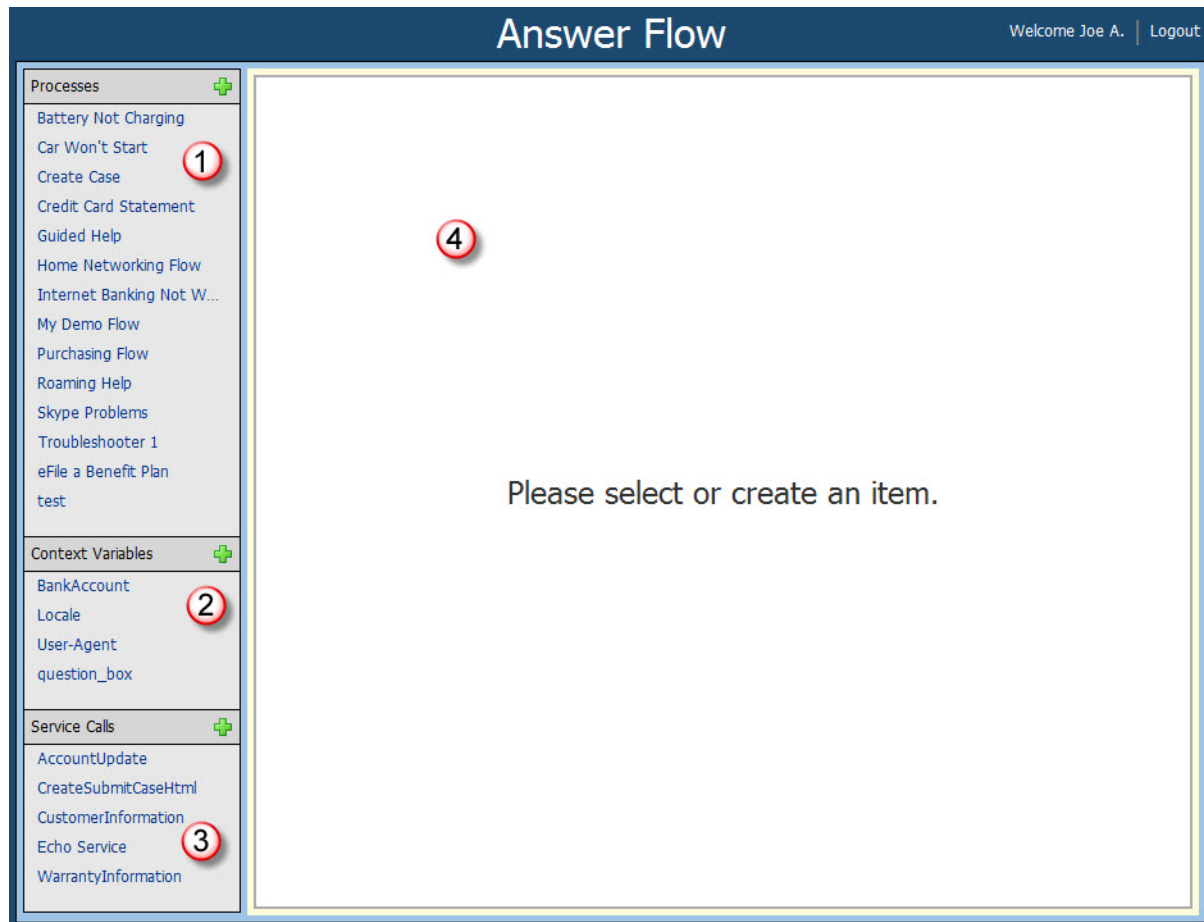
The AnswerFlow Editor login screen prompts you for login information.

- 1 Enter your login information.
  - 2 Select the Information Manager repository that AnswerFlow will use to authenticate users and retrieve articles for use within processes.
- Note:** You configure the Information Manager repository during the installation process.
- 3 Select **Submit**.

The AnswerFlow editor displays the initial screen.

## The AnswerFlow Editor

When you log onto the AnswerFlow editor, the editor displays the initial screen:



The initial screen displays a blank editor canvas, and lists the currently defined processes, context variables, and service calls in the left column.

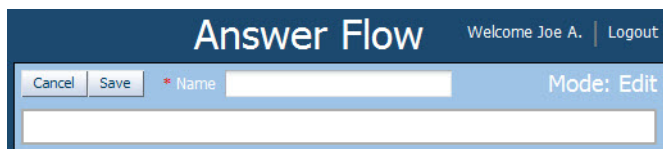
## Creating Process Components

You can use the editor to create the various process components that you will include in processes. You can reuse these components in multiple processes. You can also create processes as modules for reuse as subprocesses within larger processes.

## Defining Context Variable Values

Context variables represent contextual information that a user submits in a process. For example, to assist a customer, it might be necessary to get the customer's account number. A context variable would be designed to prompt for this information and then to use this information appropriately. You can define context variables while creating a new process or you can define some basic context variables independent of a process. All defined context variables are stored and available to all users when creating a process.

Context variables represent contextual information that a user submits in a process. Select **+** to add a context variable or select a context variable from the list to edit or delete that variable.

The screenshot shows a web interface titled "Answer Flow". In the top right corner, it says "Welcome Joe A." and "Logout". Below the title bar, there are "Cancel" and "Save" buttons on the left, a text input field labeled "Name" with a red asterisk indicating it is required, and "Mode: Edit" on the right. Below the "Name" field is a large, empty text area.

Field	Description
Name	Enter a unique, descriptive name to identify the variable.

As you define and use context variables, you can click Details to view which processes and nodes reference the variable.

## Defining Service Calls

A service call allows a process to contact a third-party service—web services, file servers, database servers—wherever data is stored that you want to bring into a service process. For example, you might want to retrieve a customer network equipment before helping them solve a networking issue. This information is stored in a database that lists the equipment originally supplied to the customer. The service call could connect to the database, retrieve the information, and then display it on the screen for the customer to confirm.

You can define service calls while creating a new process or you can define some basic service calls independent of a process. All defined service calls are stored and available to all users when creating a process.



Field	Description
<b>Name</b>	Enter a unique, descriptive name to identify the service call.
<b>Return Type</b>	Select the return type, either Bean (boolean) or Map, to designate the type of information that the service call returns.
<b>Class Name</b>	Enter the Java class name.
<b>Method</b>	Enter the method used to retrieve the information. For example, <code>getCustomerInformation</code> .
<b>Parameters and Return Parameters</b>	
<b>Name</b>	Enter the parameter name.
<b>Type</b>	Select the parameter type from the dropdown list. Selections are: <ul style="list-style-type: none"> <li>• Object</li> <li>• string</li> <li>• boolean</li> <li>• byte</li> <li>• char</li> <li>• double</li> <li>• float</li> <li>• int</li> <li>• long</li> <li>• short</li> </ul>
<b>Actions</b>	Select <b>X</b> to delete the parameter.

# Adding Page Elements and Logical Nodes to a Process

Page elements and logical nodes are basic components that you use to define how a process will display to users:

- **Pages** contain the various Page Elements that display within a defined process step.
- **Page Elements** provide text to guide users through the process and also provide input channels to allow users to enter data that AnswerFlow can then use to reach a solution. Page Elements are also called Display Elements because they display things to the end-user.
- **Logical Nodes** map the logical flow of a process and also create Service calls allowing AnswerFlow to retrieve information available only on external systems, such as a database.

## Page Elements

Page elements are forms or documents that you display to users within processes. You add page elements to the pages that define the steps within processes using the right hand side Configuration Details panel. Page Elements include:

- **Display Forms**, which provide forms to capture user input. You define the form contents by specifying various form controls, such as password fields and radio button options.
- **Display IM Documents**, which display Information Manager articles. You specify one or more Information Manager document IDs to display.
- **Display HTML**, which display formatted HTML. You specify the HTML to display as a constant or variable.

To add a Page Element:

- 1 Drag a Page Element from the left side onto the process build area.  
AnswerFlow automatically connects it to the last node placed on the page; you can change the page element location by selecting and dragging the connector to the node you want to connect it with.
- 2 Specify the various fields in the Configuration Details panel to define the page element.

## Logical Nodes

Logical Nodes map the logical flow of a process and also create Service Calls, which retrieve information from external systems. Types of Logical Nodes include:

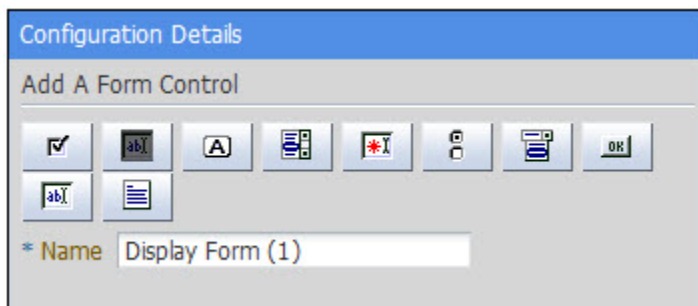
- **Branch**, which splits the process into divergent paths based on conditional logic, usually by evaluating a user response
- **Service Call**, as described in “Service Calls” on page 4
- **Process Call**, which references a previously defined process (sub-process), which then becomes part of the larger process

To add a Logical Node:

- 1 Drag a Logical Node from the left side onto the process build area.  
AnswerFlow automatically connects it to the last node placed on the page; you can change the location by selecting and dragging the connector to the node you want to connect it with.
- 2 Complete the configuration details to define the page element purpose in the process.

## Adding a Display Form Page Element to a Process

You can define a form to collect various types of information from users within a process using the Display Form page element. To display a form, it must be added to the process and then Form Controls, specific elements that request user input, must be defined.



To add a Display Form page element:

- 1 Drag the Display Form page element onto the Process Build area.  
AnswerFlow automatically connects it to the previous node.
- 2 In the Configuration Details area, select Form Controls to display on the form on the page.
- 3 Enter the necessary parameters for each desired Form Control.

When you add a Form Control, it appears in the Local Variables section of the AnswerFlow screen. Select **Is Input?** or **Is Output?** as appropriate.

**Configuration Details**

Add A Form Control

\* Name: Display Form (1)

**Check Box**

**General Properties**

☒ Label:

Default Value: ☐ Constant ☐ Variable

Options: ☐ Constant ☐ Variable

Input Variable: Check Box Variable (1)

Class:

**Local Variables**

Name	Initial Value	Is Input?	Is Output?	Delete
Check Bo		<input type="checkbox"/>	<input type="checkbox"/>	

Field	Description
Name	Enter a unique name to identify the form.

### Form Control Description

#### Check Box



Adds a check box to the form. Use for simple yes/no responses.

#### Hidden



Adds a hidden variable to the form. This places a hidden field on the page.

#### Label



Adds a label to the form. Use to label a screen object.

#### Multi-Select



Provides the ability to select multiple options. Use if users are allowed to make multiple selections from a menu. This is different from a radio button, which only allows one option to be chosen.

#### Password



Requires password submission.

#### Radio Button



Add a radio button to the form. Use for two distinct choices.

## Form Control Description (*continued*)

### Select



Adds a selection variable to the form. Use when users are presented with a menu and must make one selection.

### Submit



Adds a submit button to the form.

### Text



Adds a field where customers can enter information. Use for small fields such as Name or Customer Number.

### Text Area



Adds an area where more lengthy text can be entered.

## Defining Form Control Parameters

You define Form Control parameters when you add the form control to the process. Depending upon the Form Control added, there will be different parameters that must be defined. Not all parameters appear for every Form Control. The following table lists all of the parameters available for all of the Form controls.

Parameter	Description
<b>Label</b>	This is the label that appears on the page in the User Interface.
<b>Constant/Variable</b>	Select whether this value is a constant or variable value. Depending upon which form control is selected, this might be defined for the Label, the Default Value, additional options (see "Options" below), or a combination of these parameters.
<b>Default Value</b>	A default value that populates the field when the page is accessed. This is optional.
<b>Options</b>	Enter additional options. This is used for Check Box, Multi-Select, Radio Button, and Select only. This provides user input options for various types of selections or menus.
<b>Input Variable</b>	AnswerFlow automatically generates this variable. This corresponds to a variable in the Local Variables section of the page. <b>Tip:</b> Rename the variable as you define the element to provide a more meaningful name. For example, if The field label is User Name, you might rename the variable username.
<b>Class Name</b>	Used only for the Label form control.
<b>Required</b>	Selecting this box makes the element a required field in the User Interface.
<b>Tooltip</b>	Enter a tip for the kind of data to enter if the label does not provide adequate description. For example, a page might have two fields, one labeled Customer Number and another labeled Account Number. A tooltip might be, "This is your five digit number provided for returning customers." for the first field and , "This is your eight digit account number if you purchase for your business or other organization." for the second field. The tip is revealed when the user places the cursor over the field.
<b>Ghost Text</b>	Similar to the tooltip, this is text that appears greyed-out in the field in the UI. For example, for a Customer Name field you might designate how the entry should be made with the ghost text of "LAST,FIRST" in the field.
<b>Image URL</b>	Enter the URL for an image to use for the submit button. This is used for Submit only.

## Adding a Display IM Document Page Element to a Process

AnswerFlow can point the user to a specific Information Manager document at any point in a process. This is useful if you create a troubleshooting process and at a certain point, you have determined what the issue is

and can identify a solution. Once the AnswerFlow process has guided the customer through troubleshooting and eliminated other possible solutions, it can present the solution to the customer.

**Important!** You cannot add Information Manager articles that belong to channels having a schema attribute of type **List** to a Display IM Documents display element within a Process. The **List** attribute specifies a defined data list, which can be in the form of check boxes, drop-down menu, radio button, etc. If you do try to add an article that belongs to this type of channel, the editor will display the following error message:

```
A connection to the server has failed.
```

This error message can also result from various other types of errors.

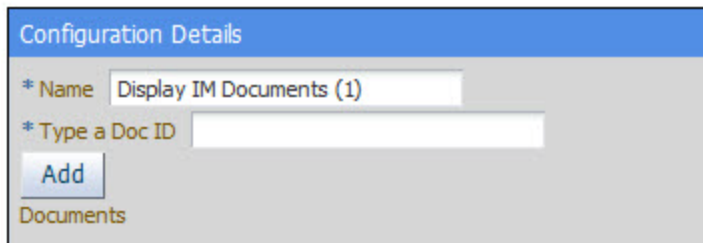
The error condition prevents you from adding another article until you reset the display element input field; to reset the display element and add an article from a supported channel type:

- de-select the display element, or change focus by selecting a different editor function
- re-select the element
- add an article that resides in a supported channel type

See the Content Channels section of the Information Manager Administration Guide for more information about Channel Attribute Types and Data Lists.

To have AnswerFlow point directly to an Information Manager document:

- 1 Drag the Display IM Documents page element onto the Process Build area.  
AnswerFlow automatically connects it to the previous node.

The image shows a 'Configuration Details' dialog box. It has a blue header bar with the text 'Configuration Details'. Below the header, there are two input fields. The first is labeled '\* Name' and contains the text 'Display IM Documents (1)'. The second is labeled '\* Type a Doc ID' and is empty. Below these fields is a blue button labeled 'Add'. At the bottom of the dialog, the word 'Documents' is displayed in a smaller font.

- 2 In the Configuration Details area, provide a Name for this node, perhaps the title of the document.
- 3 Enter the document ID in the **Type a Doc ID** field.
- 4 Select **Add**.

## Adding a Display HTML Page Element to a Process

AnswerFlow can display custom HTML at any point in a process. This is useful to display brief content not contained in an Information Manager document.

To have AnswerFlow display custom HTML:

- 1 Drag the Display HTML page element onto the Process Build area.  
AnswerFlow automatically connects it to the previous node.

Configuration Details

\* Name

☒ Constant ☐ Variable

- 2 In the Configuration Details area, provide a Name for this node.
- 3 Select whether this is a constant or variable element.
- 4 In the white text area, enter the HTML code to use.

## Adding a Branch Logical Node to a Process

Branch nodes provide the ability to branch a process based on one or more conditions, such as a user response, that the system evaluates.

To add a Branch node to a process:

- 1 Drag the Branch node onto the Process Build area.  
AnswerFlow automatically connects it to the previous node.

Configuration Details


\* Name

Conditions

Priority: 1

Condition:

Destination:

- 2 In the Configuration Details area, provide a Name for this node.
- 3 Select **Add** to create the first condition.
- 4 Select  to edit the condition.  
The Condition Builder appears on the screen.

Condition Builder

☐ Constant ☒ Variable

equals (ignore case) ☐ Constant ☒ Variable

OR

☐ Constant ☒ Variable

equals (ignore case) ☐ Constant ☒ Variable

- 5 Select whether this is a constant or variable element.

- 6 Select the input type from the dropdown menu.
- 7 Select the condition from the dropdown menu.
- 8 Enter the input variable.
- 9 Select **+AND** or **+OR** to add additional conditions.

**Note:** As you add conditions, AnswerFlow updates the Configuration Details area.

- 10 Select **Close** when completed.

## Adding a Service Call Logical Node to a Process

Use a Service Call when information from an external system, such as a database, is necessary.

To add a Service Call node to a process:

- 1 Drag the Service Call node onto the Process Build area.  
AnswerFlow automatically connects it to the previous node.

The screenshot shows the 'Configuration Details' dialog for a 'Service Call' node. The 'Name' field is set to 'Service Call (1)'. The 'Service Call' dropdown menu is set to 'CustomerInformation'. Below this are two sections: 'Input Parameters' and 'Output Parameters'.

**Input Parameters:**

Name	Value
name	<input type="radio"/> Constant <input type="radio"/> Variable
email	<input checked="" type="radio"/> Constant <input type="radio"/> Variable

**Output Parameters:**

Name	Variable
name	<input type="text"/> ▼
email	<input type="text"/> ▼
address	<input type="text"/> ▼
phone	<input type="text"/> ▼
registered	<input type="text"/> ▼

- 2 In the Configuration Details area, provide a Name for this node.
- 3 Select the Service Call from the dropdown menu.
- 4 Complete the Input Parameters and Output Parameters sections.



## Adding a Set Variable Values Logical Node to a Process

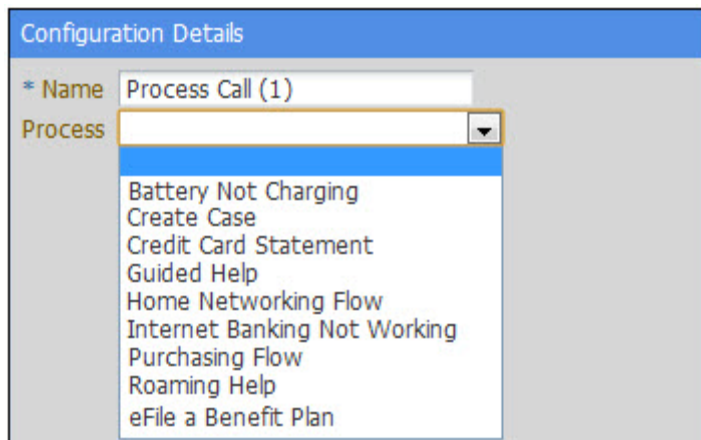
To add a Set Variable Values node to a process:

- 1 Drag the Set Variable Values node onto the Process Build area.  
AnswerFlow automatically connects it to the previous node.
- 2 In the Configuration Details area, provide a Name for this node.
- 3 Select **Add** to create a new variable.
- 4 Enter the variable value parameters.

## Adding a Process Call Logical Node to a Process

Use the Process Call logical node to reus a previously defined process flow within a new, larger process.

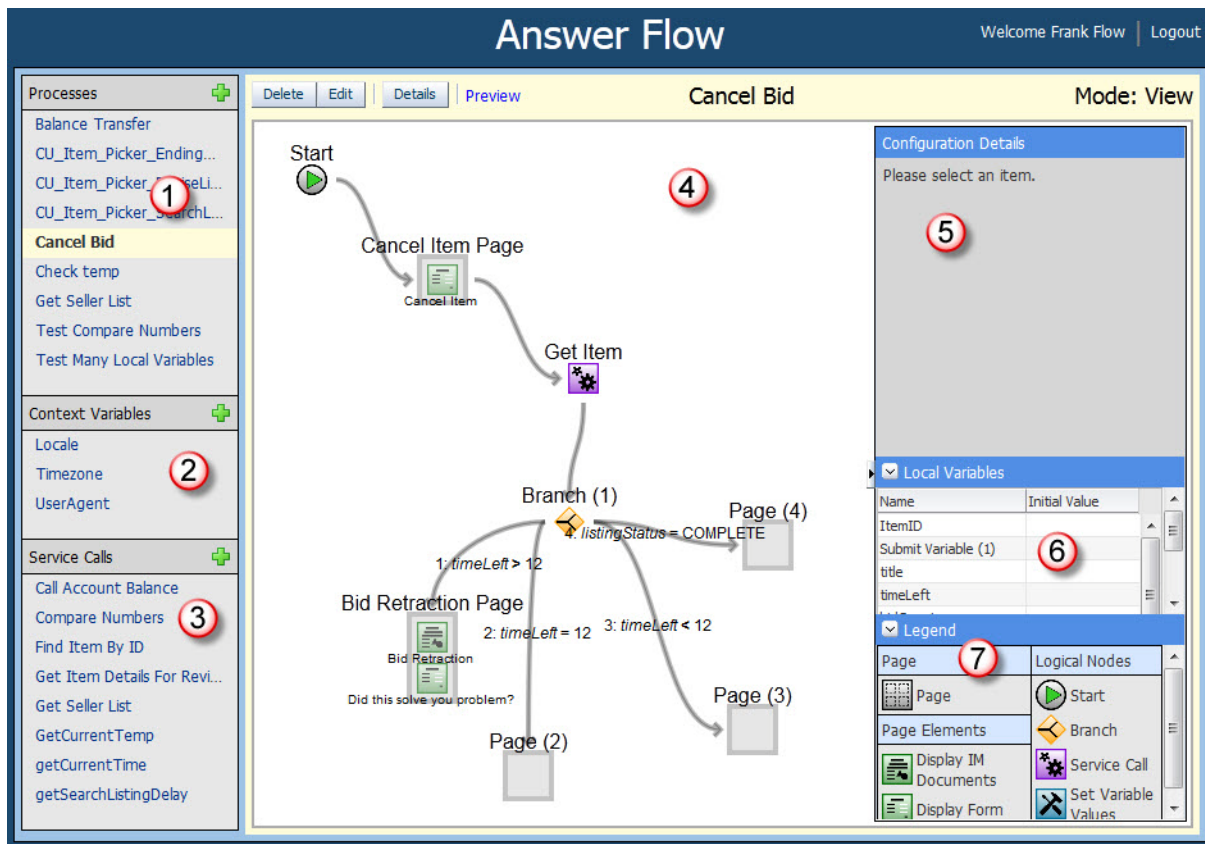
- 1 Drag the Process Call node onto the Process Build area.  
AnswerFlow automatically connects it to the previous node.



- 2 In the Configuration Details area, provide a Name for this node.
- 3 Select the Process from the dropdown menu.  
AnswerFlow displays the Input Parameters and Output Parameters of the selected process.

## AnswerFlow Process Review View

When you log in to AnswerFlow, you see a window similar to the following graphic.

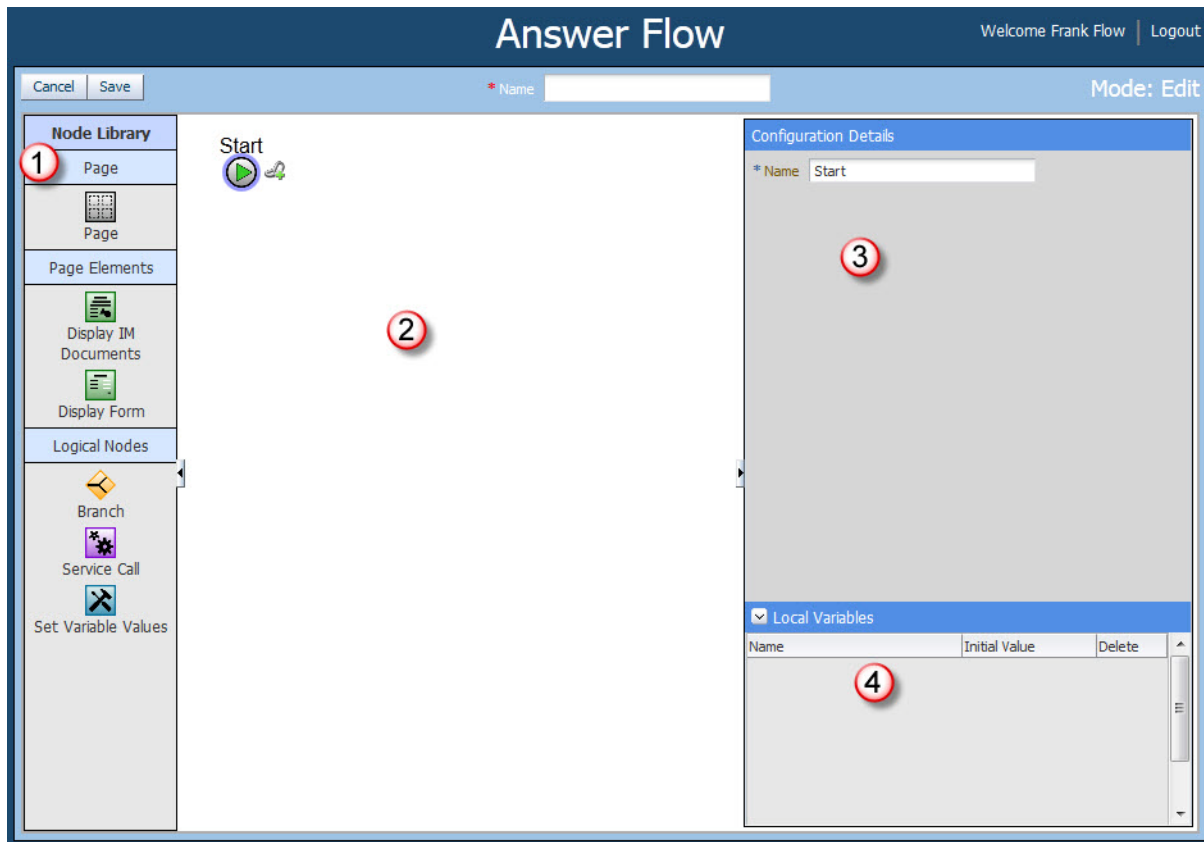


The process build screen is divided into seven main areas:

- 1 **Processes** – Lists all of the available Processes. Users create, edit, and delete processes, and select between those processes that have been defined. select **+** to add a process or select a process from the list to edit or delete that process. A process is a guided help flow that collects user input, make service calls and deliver a document based on the data and built-in business logic.
- 2 **Context Variables** – Lists defined Context Variables available when building a process. Select **+** to add a context variable or select a context variable from the list to edit or delete that variable.
- 3 **Service Calls** – Lists defined Service Calls available when building a process. Select **+** to add a service call or select a service call from the list to edit or delete that service call. Service calls are Java calls that perform an action or collect data from an external system.
- 4 **Process Build Area** – The Process Build Area provides a workspace to build, view, or edit a process. Select **Delete** or **Edit** to delete or edit a selected process. Select a process node and then **Details** to view details about individual nodes.
- 5 **Configuration Details** – Displays Configuration Details about a selected node. Select a process node and then view configuration details about individual nodes.
- 6 **Local Variables** – Provides a list of variables used in the process displayed in the workspace.
- 7 **Legend** – Provides visual guides to interpret each individual component of a process.

## AnswerFlow Process Build View

When you select **+** to add a process, you see a window similar to the following graphic.



The Process Workspace is how users create and edit processes. Users define the components that are contained within a process, the order in which those components are executed, and any branching for conditional components. The Process Workspace comprises four primary sections:

- 1 **Node Library** – The Node Library contains the components used to build processes. It includes Page Elements and Logical Nodes.
- 2 **Process Build Workspace** – The main workspace provides a review area for a selected process. The Main Workspace is where users' define the components that are contained within a process, the order in which those components are executed, and any branching for conditional components.
- 3 **Configuration Details** – As described in "AnswerFlow Process Review View" on page 22.
- 4 **Local Variables** – As described in "AnswerFlow Process Review View" on page 22.

# Deploying AnswerFlow Processes

You can deploy processes to make them available as answers to users' questions in a runtime environment by:

- exporting and importing AnswerFlow data among development and production instances as appropriate, as described in “Exporting AnswerFlow Data” on page 7 and “Importing AnswerFlow Data” on page 8
- integrating AnswerFlow processes into a production user interface
- associating processes with user actions (questions, etc.) using Intelligent Search Managed Answers

## Integrating AnswerFlow Processes into a Production User Interface

The AnswerFlow installation process installs a sample user interface that you can use as the basis for integrating AnswerFlow processes into a production user interface. You can deploy the sample user interface to an existing server using the steps in this section.

**Note:** You can contact Oracle Consulting Services for assistance in integrating AnswerFlow processes with a production user interface.

## Using the Sample User Interface

The AnswerFlow installation process installs a sample user interface that you can use as the basis for developing and integrating user interface elements to present AnswerFlow processes within your production application. The sample user interface is located in the RuntimeUI instance directory:

Directory	Description
sample/docs/SampleUI/lib	contains jars for the SampleUI project
sample/docs/SampleUI/src	contains source code for the SampleUI project

You can deploy the sample user interface to an existing RuntimeUI instance running on WebLogic Server.

**Important!** You must deploy the sample user interface from the ICE environment.

You deploy the sample user interface by:

- configuring the sample user interface authentication for Information Manager
- compiling the sample user interface
- deploying the sample user interface

## Configuring Access to Information Manager

You must configure the sample user interface to access the Information Manager instance that AnswerFlow is configured to use, as described in *Installing and Configuring Oracle Knowledge*.

You configure the sample user interface to access the Information Manager instance by modifying the authorization settings in the sample user name service (`SampleUserNameService.java`). The sample user name service returns the user credentials and repository information that the AnswerFlow application requires in order to access the configured Information Manager instance.

You configure access to Information Manager by coding the various methods in the file `SampleUserNameService.java`. To modify the user name service:

- edit the file `SampleUserNameService.java`, located at:

```
${ANSWERFLOW_INSTALL_DIRECTORY}/AnswerFlow/Sample/SampleUI/src/main/java/com/inquirator/
pmeditor/security/SampleUserNameService.java
```

- extend the following methods as appropriate for your environment, so that:

Method	Description
<code>getCurrentUserName</code>	returns the name of a user that has permission to access the configured Information Manager instance. This method accepts credentials as a parameter.
<code>getUserName</code>	returns the name of a user that has permission to access the configured Information Manager instance.
<code>getCurrentPassword</code>	returns the user's password. This method accepts credentials as a parameter.
<code>getPassword</code>	returns the user's password.
<code>getCurrentRepository</code>	returns the reference key of the IM repository that contains the documents. This method accepts credentials as a parameter.
<code>getRepositoryReferenceKey</code>	returns the reference key of the IM repository that contains the documents.

**Note:** For validation and demonstration purposes, you can edit these methods to return static (hard-coded) credentials for a specific Information Manager user, as follows:

Method	Description
<code>getCurrentUserName</code>	specify <code>return "IM_UserName"</code> <i>where:</i> IM_UserName is the actual name of the authorized user.
<code>getUserName</code>	specify <code>return "IM_UserName"</code> <i>where:</i> IM_UserName is the actual name of the authorized user.
<code>getCurrentPassword</code>	specify <code>return "IM_Password"</code> <i>where:</i> IM_Password is the user's actual password.
<code>getPassword</code>	specify <code>return "IM_Password"</code> <i>where:</i> IM_Password is the user's actual password.

Method	Description
getCurrentRepository	specify return "REPOSITORY_KEY" <i>where:</i> REPOSITORY_KEY is the repository reference key.
getRepositoryReferenceKey	specify return "REPOSITORY_KEY" <i>where:</i> REPOSITORY_KEY is the repository reference key.

## Compiling the Sample User Interface

You can build the sample user interface prior to deployment using a packaged ANT utility. You can compile the sample user interface without deploying it.

To compile the sample user interface:

- open a terminal window and change to the AnswerFlow runtime user interface directory:

```
cd ${ANSWERFLOW_INSTALL_DIRECTORY}/AnswerFlow/instances/RuntimeUI
```

- start the ICE environment:

```
./setenv.sh
```

- change to the sample user interface directory:

```
cd ../../Sample/SampleUI
```

- execute the sample user interface build process:

```
iceant
```

The build process creates the `SampleUI.war` file.

## Deploying the Sample User Interface

To deploy the sample user interface:

- from the ICE environment in the AnswerFlow sample user interface directory, as in “Compiling the Sample User Interface” on page 26, execute the deployment script and specify:
  - the location of the sample user interface source code, for example:

```
deploy.sh -s ../../Sample/SampleUI -w ../../Sample/SampleUI/target/SampleUI.war
```

- the location of the web archive (.war) that the deployment script will create, for example:

```
deploy.sh -s ../../Sample/SampleUI -w ../../Sample/SampleUI/target/SampleUI.war
```

## Viewing the Sample User Interface

You can view the sample user interface by

- starting the RuntimeUI server from the ICE environment
- navigating a browser to the sample user interface

To start the server:

- execute the following command from an ICE prompt:

```
inquiraaaf.sh start
```

- navigate to:

```
http://${RUNTIMEUI_SERVER_LISTEN_ADDRESS}:${RUNTIMEUI_SERVER_LISTEN_PORT}/SampleUI/  
dashboard
```

where:

`{RUNTIMEUI_SERVER_LISTEN_ADDRESS}` is the listen address specified at installation, for example, `localhost`, and `{RUNTIMEUI_SERVER_LISTEN_PORT}` is the listen port specified at installation, for example, `8233`:

```
http://localhost:8233/SampleUI/dashboard
```

## Associating Processes with User Actions

You can make AnswerFlow processes available as answers to users' questions or responses to other user actions by configuring processes as responses within Managed Answers. Managed Answers are a type of Intelligent Search Dictionary Rule that you configure to return a specific answer in response to a specific question or type of question.

See the *Intelligent Search Language Tuning Guide* for more information on configuring Managed Answers. Contact Oracle Consulting Services for assistance in associating AnswerFlow processes with Managed Answers.