Informatica® ActiveVOS
9. 2. 6

Central and ActiveVOS Screenflow

This product includes software licensed under the terms at http://www.tcl.tk/software/tcltk/license.html, http://www.bosrup.com/web/overlib/?License, http://www.stlport.org/doc/ license.html, http://asm.ow2.org/license.html, http://www.cryptix.org/LICENSE.TXT, http://hsqldb.org/web/hsqlLicense.html, http://httpunit.sourceforge.net/doc/ license.html, http://jung.sourceforge.net/license.txt , http://www.gzip.org/zlib/zlib_license.html, http://www.openldap.org/software/release/license.html, http://www.libssh2.org, http://slf4j.org/license.html, http://www.sente.ch/software/OpenSourceLicense.html, http://fusesource.com/downloads/license-agreements/fuse-message-broker-v-5-3- license-agreement; http://antlr.org/license.html; http://aopalliance.sourceforge.net/; http://www.bouncycastle.org/licence.html; http://www.jgraph.com/jgraphdownload.html; http://www.jcraft.com/jsch/LICENSE.txt; http://jotm.objectweb.org/bsd_license.html; . http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231; http://www.slf4j.org/license.html; http://nanoxml.sourceforge.net/orig/copyright.html; http://www.json.org/license.html; http://forge.ow2.org/projects/javaservice/, http://www.postgresql.org/about/licence.html, http://www.sqlite.org/copyright.html, http://www.tcl.tk/software/tcltk/license.html, http://www.jaxen.org/faq.html, http://www.jdom.org/docs/faq.html, http://www.slf4j.org/license.html; http://www.iodbc.org/dataspace/iodbc/wiki/iODBC/License; http://www.keplerproject.org/md5/license.html; http://www.toedter.com/en/jcalendar/license.html; http://www.edankert.com/bounce/index.html; http://www.net-snmp.org/about/license.html; http://www.openmdx.org/#FAQ; http://www.php.net/license/3_01.txt; http://srp.stanford.edu/license.txt; http://www.schneier.com/blowfish.html; http://www.jmock.org/license.html; http://xsom.java.net; http://benalman.com/about/license/; https://github.com/CreateJS/EaselJS/blob/master/src/easeljs/display/Bitmap.js; http://www.h2database.com/html/license.html#summary; http://jsoncpp.sourceforge.net/LICENSE; http://jdbc.postgresql.org/license.html; http://protobuf.googlecode.com/svn/trunk/src/google/protobuf/descriptor.proto; https://github.com/rantav/hector/blob/master/LICENSE; http://web.mit.edu/Kerberos/krb5-current/doc/mitK5license.html; http://jibx.sourceforge.net/jibx-license.html; https://github.com/lyokato/libgeohash/blob/master/LICENSE; https://github.com/hjiang/jsonxx/blob/master/LICENSE; https://code.google.com/p/lz4/; https://github.com/jedisct1/libsodium/blob/master/LICENSE; http://one-jar.sourceforge.net/index.php?page=documents&file=license; https://github.com/EsotericSoftware/kryo/blob/master/license.txt; http://www.scala-lang.org/license.html; https://github.com/tinkerpop/blueprints/blob/master/LICENSE.txt; http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html; https://aws.amazon.com/asl/; https://github.com/twbs/bootstrap/blob/master/LICENSE; https://sourceforge.net/p/xmlunit/code/HEAD/tree/trunk/LICENSE.txt; https://github.com/documentcloud/underscore-contrib/blob/master/LICENSE, and https://github.com/apache/hbase/blob/master/LICENSE.txt.

This product includes software licensed under the Academic Free License (http://www.opensource.org/licenses/afl-3.0.php), the Common Development and Distribution License (http://www.opensource.org/licenses/cddl1.php) the Common Public License (http://www.opensource.org/licenses/cpl1.0.php), the Sun Binary Code License Agreement Supplemental License Terms, the BSD License (http:// www.opensource.org/licenses/bsd-license.php), the new BSD License (http://opensource.org/licenses/BSD-3-Clause), the MIT License (http://www.opensource.org/licenses/mit-license.php), the Artistic License (http://www.opensource.org/licenses/artistic-license-1.0) and the Initial Developer's Public License Version 1.0 (http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/).

This product includes software copyright © 2003-2006 Joe Walnes, 2006-2007 XStream Committers. All rights reserved. Permissions and limitations regarding this software are subject to terms available at http://xstream.codehaus.org/license.html. This product includes software developed by the Indiana University Extreme! Lab. For further information please visit http://www.extreme.indiana.edu/.

This product includes software Copyright (c) 2013 Frank Balluffi and Markus Moeller. All rights reserved. Permissions and limitations regarding this software are subject to terms of the MIT license.

See patents at https://www.informatica.com/legal/patents.html.

DISCLAIMER: Informatica LLC provides this documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of noninfringement, merchantability, or use for a particular purpose. Informatica LLC does not warrant that this software or documentation is error free. The information provided in this software or documentation may include technical inaccuracies or typographical errors. The information in this software and documentation is subject to change at any time without notice.

NOTICES

This Informatica product (the "Software") includes certain drivers (the "DataDirect Drivers") from DataDirect Technologies, an operating company of Progress Software Corporation ("DataDirect") which are subject to the following terms and conditions:

1. THE DATADIRECT DRIVERS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.
2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS.

Publication Date: 2024-12-12

Table of Contents

Chapter 4: For Administrators.

Preface

This module contains information about Process Central and guide creation.

CHAPTER 1

Process Central

Welcome to ActiveVOS Process Central, home for managing your work items, including the following:

- Tasks are part of an automated business process that requires your input.
- Requests are forms you submit to start an automated business process.
- Reports show interesting data for active processes and tasks.
- Screenflow (License required). Guides are simplified, automated business processes that you can create, publish, and run easily. The *Guides* tab requires a separate license for Screenflow and only appears if you have a license installed.

ActiveVOS Process Central may look different from user to user because it can be customized for your organization and for you personally.

Using ActiveVOS Central

Before running ActiveVOS Central, be sure your ActiveVOS Administrator has completed configuration and database setup by following the instructions in the *Server Installation, Configuration, and Deployment* topics located elsewhere in this help.

Logging in to ActiveVOS Central

The Log In page appears for you to type in your Username and Password. Select **Log In** to continue.

Here are some pointers:

- You need a valid username and password to log in.
- Before you open ActiveVOS Central, ActiveVOS server must be running. Start Central by typing the following into your browser: http://localhost:8080/activevos-central
- Change the port number if 8080 is already in use on the computer.

Session Time Out

If you are logged into ActiveVOS Central, but are not actively using it, the session may expire. The default time-out setting is 30 minutes, but can be different in your environment. If a time-out occurs, the Log In page appears.

Note: Be sure to save your task response after you enter it so that the changes are not lost if your session times out.

What is an ActiveVOS Task?

A task is a unit of work that has arrived in ActiveVOS Central.. The task contains information for your review and requires a response. When you submit a response, the task is complete. Tasks are most likely sent to you and other users who share your area of expertise. Each of you can claim a task to work on. When you do this, the task is removed from other users' list.

Note: This pane is not available in some versions of Process Central.

The task you work on is a step in an automated, running business process. The process is waiting for you or someone else to send back data so that the process can continue to its next step.

For details on who can claim and manage tasks, see .

To get started on your task list, see .

Navigating in the Tasks View

This topic discusses the items found within Tasks View.

Roles

The Roles list contains different types of users. The default types are:

- **User**: Potential owners of a task, who can claim and complete a task or assign it to another user.
- **Administrator**: Same as potential owners, but can also see a list of all actual owners.
- **Stakeholder**: Same as administrator.
- **Initiator**: Same as user, but can skip a task if it was defined as "can be skipped."

By default, all Role types are shown. Your list may contain roles defined for your enterprise.

Notifications, Open Tasks, Closed Tasks

Tasks are automatically organized into three groups:

- **Notifications**: There is no action required other than to read and remove the notification.
- **Open Tasks:** Includes unclaimed and claimed tasks. Claimed tasks are those that you selected to work on.
- **Closed Tasks**: Includes completed, exited, obsolete, and error tasks.

Show

The Show list allows you to select a group of tasks to display. By default, you can display All, Claimed, or Unclaimed tasks. Your list may contain custom selections.

Refresh

Refresh your task list periodically to view new and changed tasks in the list. For details, see *Refreshing Your Task List*.

RSS

For details, see *Using an RSS Feed*.

Task Listing

Each task displayed in the task list has the following default information. Your list may contain custom columns.

- **Id**: Assigned when the task is created.
- **Status icon**: Indicates the current task status, such as Unclaimed or Completed.

- **Priority**: A priority is number beginning at zero, which is the highest priority. Priority zero and one have flag icons. Lower priorities (higher numbers) do not have an icon.

- **Name**: Descriptive name for the task.

- **Subject**: Short description of the task.

- **Owner**: Person who claimed the task, if it was claimed.

- **Created**: Date the task arrived in Process Central.

Notice that you can see more details in the *Additional Information* section.

Search

Search on the currently displayed list of tasks. For tips on basic searching, see *Using Process Central*.

You can search for a task by **Task Id**, **Search By** attribute, or any string in any displayed column.

If the task definition defines a Search By expression, you can type in the expression result. For example, if an expression evaluates to a product Id or a sales amount, you can type in the value.

Refreshing Your Task List

It is a good idea to refresh your task list periodically by selecting the **Refresh** button. While you are logged in, the following can occur:

- New tasks arrive.

- The priority of a current task changes.

- Another user claimed the task.

- A task is removed from your list due to an automated deadline action.

- Another user released a task back to the pool.

Note: If a task is removed from your list or if its priority changes, look for a related email. Automated actions, such as an email sent to you, are often associated with a change in a task's status.

About Users and Administrators

Depending on your assigned roles, you may be a task user (or task initiator), a task administrator (or stakeholder), or both. Your role is assigned when a task was set up, and can vary from task to task.

You may also have a custom role that is shown in the Roles list.

Administrators can manage the tasks in which they were named as an administrator. In addition to working on tasks, an administrator can view a list of tasks and task owners.

A task initiator can skip a task. If the task was defined with a "can be skipped" option, a **Skip** button is displayed.

Claiming and Performing Other Actions for a Task

A task goes through stages as different actions occur. Here is a list of actions you can take before completing a task, and what happens when you do:

- **Claim**
  You can claim a task to start it. When you claim a task, it is not available for other users to work on and they cannot see it. When you claim a task, all of the task work options are enabled, as described in "Responses, Comments, and Attachments" on page 11.

  Note: If you are the only person that a task is sent to, the task is automatically marked Claimed, and you can begin working on it.

- **Assign**
  For details, see "Assigning a Task to Another User" on page 13.

- **Release**
  If you claim a task, you can release it, meaning you will not finish working on it. The task returns to unclaimed status and any user in the group can claim it. The task reappears in other users' task list when they refresh their view.

- **Set Priority**
  You can change the priority of a task after you claim it. The highest priority is zero. You can assign any non-negative number (0, 1, 2, and so forth) to lower or raise the priority. The Set Priority feature is in the Additional Information section at the bottom of a task in the Detail tab. Select the link on Priority to open the **Set Task Priority** dialog. Add a number, zero or higher, and click **OK**.

- **Save**
  See "Responses, Comments, and Attachments" on page 11.

- **Complete**
  See "Completing a Task" on page 12.

- **Skip**
  Skip the task, meaning it will not be completed. If the task's definition does not have an *Is Skippable* value, this option is not available.

Responses, Comments, and Attachments

A task has information you need to complete it. This section has general information about your tasks.

Note: You must provide a response in each field marked with an asterisk. These fields are required.

Saving Your Response

After you enter a response, select **Save**. Saving your response prevents its loss in case your session times out.

Using Additional Information

The *Additional Information* section of a Task has four tabs:

- Comments
- Attachment
- History--see "Viewing Task History" on page 13
- Detail--see "Viewing Task Details" on page 14

Comments

You can view and add comments about a task for other users and administrators to see. The comments accompany a task.

You can edit or delete comments that you add, but not comments that others have added. Administrators can edit or delete any comments associated with the task. Once the task is completed, comments cannot be edited or deleted.

To add a comment:

1. Scroll to the bottom of a task to see the Additional Information section.
2. On the Comments tab, select **Add**.
3. Type text into the **Add Comment** window.
4. Click **OK**.

To edit a comment:

1. On the Comments tab, select a comment that you added.
2. Select **Edit**.
3. Add to or revise text in the **Edit Comment** window.
4. Click **OK**.

To delete a comment:

Delete a comment that you added by selecting the comment and pressing **Delete**.

Attachments

Attachments are documents that support a task that you or others may need to use.

To view an attachment:

1. Scroll to the bottom of a task to see the Additional Information section.
2. On the Attachments tab, if it exists, attachments are listed by name and type.
3. Select the attachment to display the **Open** dialog.
4. Select an option to open the file or save the file to your computer.

To add an attachment:

1. On the Attachments tab, select **Add**.
2. Select the file you are adding.
3. Press **Save**.

To delete an attachment:

To remove an attachment from the task, select it and then press **Delete**. This action does not affect files on your computer.

Completing a Task

Completing a task sends a response back to the business process.

To complete a task:

1. On the task form, supply the response. A field marked with an asterisk is a required field.
2. If needed, add or remove attachments and add comments.
3. Press **Save**.
4. Press **Complete**.

The task is marked Completed and is no longer displayed in your list of Open tasks. If desired, you can view a list of Closed tasks.

Note: Closed tasks are deleted by an automatic process. This is set up and controlled by your Process Server administrator.

Assigning a Task to Another User

A task's potential owners, actual owner, and business administrators can assign a task to another user, making that person the actual owner of the task. It also adds that person to the list of potential owners if not already on the list. A task can be assigned when it is in an active state (Unclaimed or Claimed), and for the new owner, it is in the Claimed state. If the old and new owners are already logged in, they must select **Refresh** to see the change to their task list.

To assign a task:

1. Select a task from the list.
2. Select **Assign**.
3. In the **Assign Task** dialog, select a user. The user does not need to be among the potential owners group, but must be a named in the Identity Service (member directory).

The task is now claimed by the new owner.

Note: Sometimes a task is automatically reassigned, as described in .

Deadlines: What Happens When a Task is Overdue

A task can have several deadlines and associated actions, including:

- **Start deadlines**, which describe when a potential owner must start working on the task. The deadlines are enabled as soon as the task is created and are disabled once a potential owner starts the task.
- **Completion deadlines**, which describe when an owner must complete the task. The deadlines are enabled as soon as the task is created and are disabled once an owner completes the task.
- **Expiration date**, which is enabled as soon as the task is created and is disabled once an owner completes a task

Based on start and completion deadlines, a task can trigger escalation actions. For example, there might be the following type of actions, each with its own start deadline:

- If this task is not started within three hours after it is created, send an email reminder to the task owner.
- If this task is not started within six hours after it is created, raise the priority level, and send an email reminder to the task administrator.
- If this task was claimed but was not started within three hours after being created, release the task from the current owner and return it to the pool of potential owners.

Similar types of escalation actions could also exist for completion deadlines.

Because there may be escalation actions associated with the tasks you are working on, you should refresh your task list, as described in .

Reading and Removing Notifications

A notification requires no action except to read it. When you finish with it, you can remove the notification.

Viewing Task History

In the Additional Information section at the bottom of a task form, you can see the History tab.

The History tab displays the various stages a task goes through. You can see when the task first arrived and what has happened to it since then. The History tab shows who made changes and who owns the task.

Viewing Task Details

In the Additional Information section at the bottom of a task form, you can see the Detail tab

The Detail tab, which is in the Additional Information section at the bottom of the task form, displays the same information as in the task listing, as well as a few additional details. Here are the fields within it:

- **Task name**: The name of the task
- **Subject**: Brief text indicating the kind of task
- **Description**: An explanation of the detail
- **Created**: The date and time the task arrived
- **Modified**: The date and time the task was last modified; for example, it shows when a comment or attachment was added
- **Escalated**: The date and time that an escalation action occurred; for example, an email sent to an administrator
- **Owner**: If the task is claimed, the name of the person who claimed it
- **Status**: Claimed, Unclaimed, Completed
- **Priority**: A non-negative number indicating the priority of working on a task--the highest priority is zero
- **Start By**: The date and time by which the task must be claimed, if specified
- **Complete By date**: The date and time by which the task must be completed, if specified
- **Expires**: The date and time that the task is no longer available to work on, if specified

For examples of various *Start By and other deadlines, see* .

Using an RSS Feed

You can use built-in RSS feeds to access the data within the Process Central. You set up an RSS subscription by selecting the **RSS** button in the listing area. Type in the credentials you used to log into Process Central. (The subscription is for a RSS 2.0 feed.) For example, if you want to view a list of open tasks as RSS, you can subscribe on that page.

You can also view task details by using the Process Server task feed service and a feed URL can be set up with optional parameters.

ATOM Example

With the Process Server running, type the following into a Web browser:

```
http://[host]:[port]/active-bpel/services/REST/
      AeB4PTaskFeed?filter=open&role=administrator&maxTasks=10
```

Provide the host and port for your installation of Process Server.

This URL returns feed data for a maximum of 10 open administrator tasks from the Process Central authenticated on your server.

RSS Example

"Open Tasks" is the default setting, so there is no need to add any of the optional query parameters except for the format (rss):

```
http://[host]:[port]/active-bpel/services/
        REST/AeB4PTaskFeed?format=rss
```

The following table describes the parameters you can use in a URL.

Parameter	Allowed Values
details	true or false (default) If true, feed content includes additional text such as status, priority, and owner. Otherwise, only the feed item subject is returned. The default value is false.
filter	notifications, open (default), closed, unclaimed, claimed, completed, error, obsolete, failed
format	atom, rss. Indicates the preferred feed format - ATOM 1.0 or RSS 2.0. The default is atom.
maxTasks	integer_value; default is 20.
role	user (default), administrator, initiator, stakeholder.
searchBy	"Search By" expression that is part of the task definition
taskIndexOffset	integer_value. Start at task `integer_value` and display `integer_value + maxTasks`; default is 0; for example, Display tasks 15 through 35.

Note: When you enter user credentials for an RSS feed, they remain cached until the browser is restarted. So if you logout of Process Central and sign back in with different credentials, the credentials match those you previously entered for the RSS feed.

Vacation Delegation Settings

Select the **Settings** link next to the **Log Out** link in the header of Process Central to open the **Vacation Delegation Settings** dialog.

You can select a delegate to work on a task while you are on vacation (or for any other reason). You can specify different delegates for different time periods, but the time periods cannot overlap. This feature gives you a way to specify a task assignment for all tasks, instead of on individual tasks that have an Assign feature.

A delegate can only work on claimed or auto-claimed tasks from an actual owner.

A delegated person can work on your tasks after logging into Process Central.

Tips for selecting delegates:

- Select **Add Delegate** to open the **Delegate Details** dialog.
- Enter the name of another user in your identity service and a **Start Date**.
- An **End Date** is not required, but if you leave it blank, you cannot add additional delegates.
- Add additional delegates for different time periods.
- If you picked the wrong times, select **Remove** and re-add the delegate.
- **Search** text matches the exact name of a delegate.

Making Requests

A request is a form you submit to start an automated business process. (The form is usually designed in Process Developer and deployed to an Process Server.)

The form is based on information needed for the starting activity of a business process. If a field is required, it is marked with an asterisk.

You may have a personal list of request forms, and they may differ from those of other users.

To view a list of requests:

1. Select **Requests** from the left-hand navigation area of Process Central.
2. If available, select a request folder.
3. Single-click a request to preview it; double-click to open it in a new tab.

Use **Search** to locate a request in the current list. You can type in a word from the request name or description. For example, to find all requests with the word "human", you can type `human` in the **Search** box.

To submit a request:

- Select a request from the list of requests.
- Expand the viewing area to see the entire form. The **Send Request** (or similar) button is at the bottom of the form.
- Enter or select data for all fields displayed with an asterisk, and optionally fill in the remaining fields. The form may have a field for adding an attachment.
- Select **Send Request**.
  A response can be returned to you immediately, depending on business process executed.
- Close the Request tab, or if you are using the request in the preview area, clear out the data in the form by selecting **Reset Form**.

Submitting a Request

To submit a request:

- Select a request from the list of requests.
- Expand the viewing area to see the entire form. The **Send Request** (or similar) button is at the bottom of the form.
- Enter or select data for all fields displayed with an asterisk, and optionally fill in the remaining fields. The form may have a field for adding an attachment.
- Select **Send Request**.
  A response can be returned to you immediately, depending on business process executed.
- Close the *Request* tab, or if you are using the request in the preview area, clear out the data in the form by selecting **Reset Form**.

Viewing Reports

Process Central Reports provide charts and graphs about processes and tasks.

- To view a report, select **Reports** in the left-hand navigation area, and select a report category.
  A list of reports displays in the report listing area. Select a report to see if a preview is available, or double-click to open the report in a new tab.

- Use **Search** to locate a report in the current list. You can type in a word from the report name or description. For example, to find all reports covering a 24-hour period, you can type 24 in the **Search** box.

When a report is open, use the Report toolbar as follows:

- To refresh the data in the report, select **Run Report** from the toolbar.

- In the dialog, click **OK**.

You can also export the report to a spreadsheet or print the report.

CHAPTER 2

Guides Overview

A guide is a set of screens that prompts users to review, enter, or confirm data. For example, a step might display account details or prompt the user to confirm the status of a sales call. Behind the scenes, steps interact with your application by extracting and storing data. Guides run within mobile apps or on traditional platforms such as a PC or a Mac.

The following image shows the guide design process:
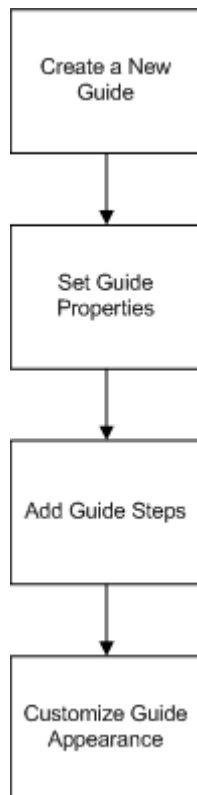
You can create guides without technical expertise or formal training.

You want to automate the process a sales manager follows to create a new account or update an existing account in Salesforce.

To create the guide, you perform the following general steps:

1. Create the screens that contain questions and possible answers. As the options are defined, the sequence of steps from the script appear as a flowchart in the canvas.
The following image depicts the guide:
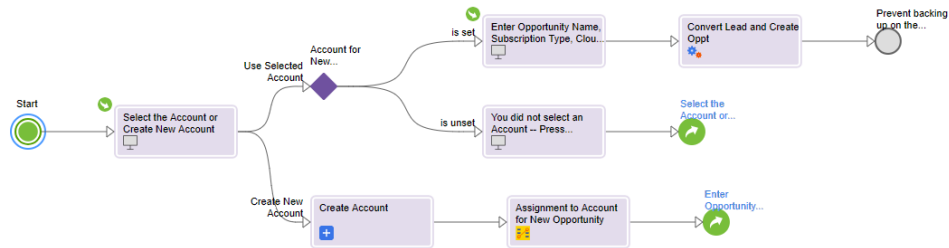
2. Insert optional automated actions. For example, call a process that converts a lead and creates a new opportunity. When you run the guide, if the guide takes the appropriate path, the process is called.

3. Publish the guide so that users can access it for sales activities.

Guide Aea on Your User's Pages

The guides that you create should be placed on an object's page. For example, if a salesman is creating a lead, you want a lead handling guide to appear on the page where the user is working with the lead's information.

Note: In Salesforce, the pages are actually definitions of kinds of objects; for example, a lead or an opportunity. An individual lead--for example, for a person named "John Smith"--is also called an object.

After a Salesforce administrator alters page definitions, the user automatically sees a Guide area on a page. For example, if an administrator added a Guide area to the Lead's page design, users will see it whenever they look at a lead.

When the user clicks on a link in this area, a new tab having the guide's title appears within the guide area.
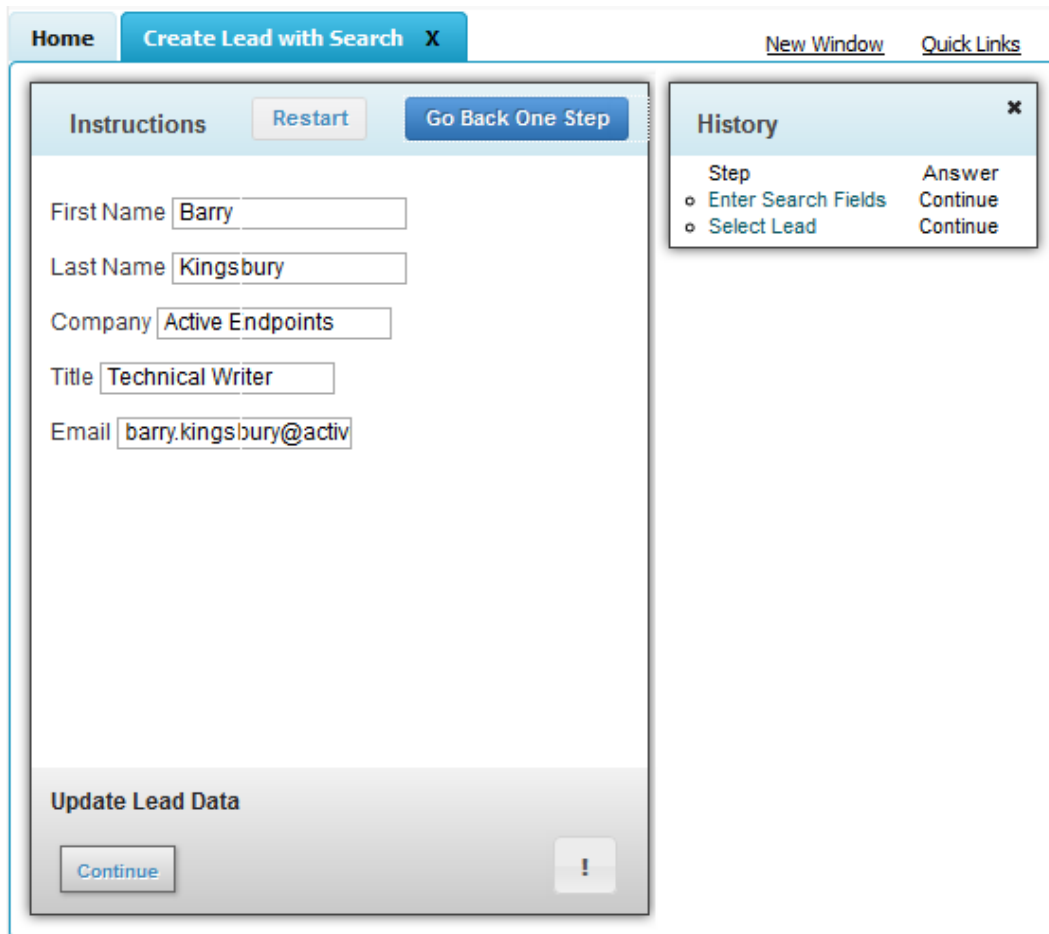
This figure shows the *Create Lead with Search* guide and a history of the steps the user performed. If the user clicks on a step within the history, Process Designer unwinds what it has done and puts the user back at that step. This also "undoes" the work that was done since that step executed. So, for example, if the user picks a step in the history that precedes steps that created new Salesforce objects, these newly-created objects are deleted.

The way in which steps display within a browser differs, depending upon the device. The previous figure shows how a step would look running on a PC or a Mac. Here's how a guide would look running on an iPhone:
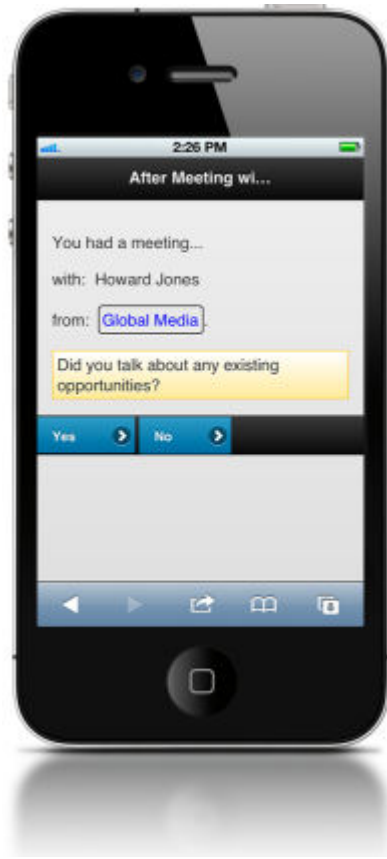
Using the DesignHome

The Design Home page is provides access to all of the objects in your system. From here, you can search for existing objects or create new guides, processes, process objects, service connectors and connections.

Each object opens on a separate page or "tab" to the right of Design Home.

Guides List Area

All of your objects appear in this list within the Design Home page, grouped by object type and by user-defined tags, if any:
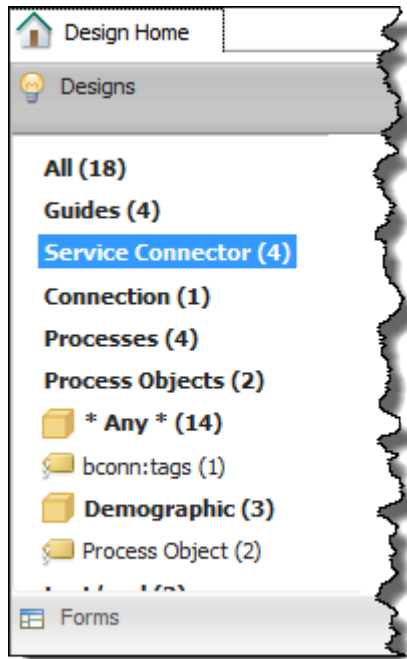
Tags create categories that act like folders to group objects in this List area. The number in parenthesis is the number of objects that have the same tag.

The tag categories act like an index. If you click on one of them, only objects in that category display. You can define Tags in the **Properties** for each object you create. You can enter more than one tag for each object.

Tags With Special Meaning

- *** Any ***: The guide is not associated with an object. In contrast to Home, it doesn't appear on a page.
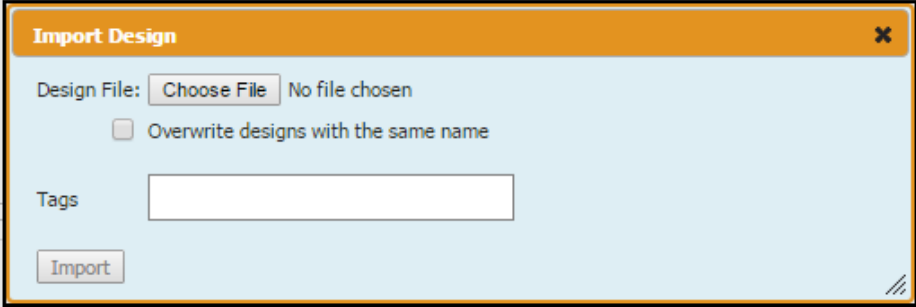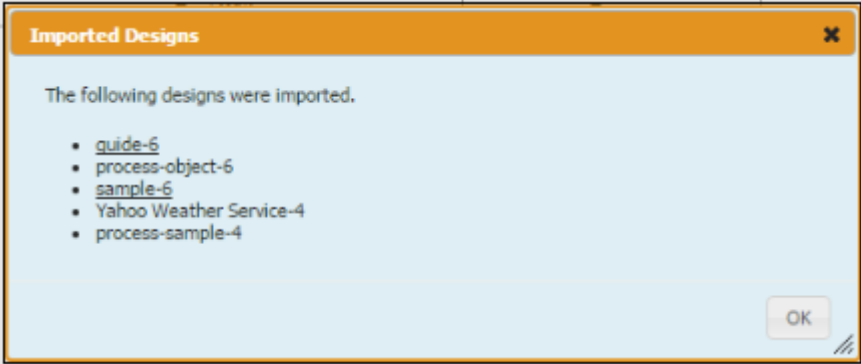
Guide Home Toolbar

Use the Design Home toolbar to manage your objects. Here is the toolbar:
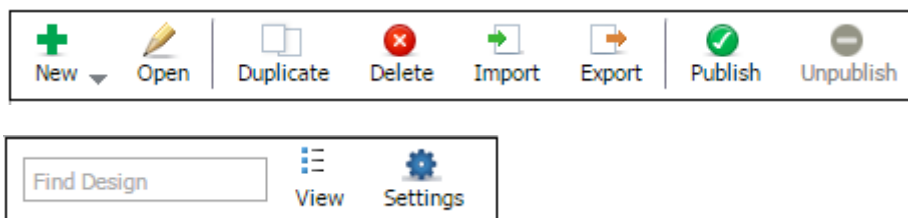
The following table describes each button:
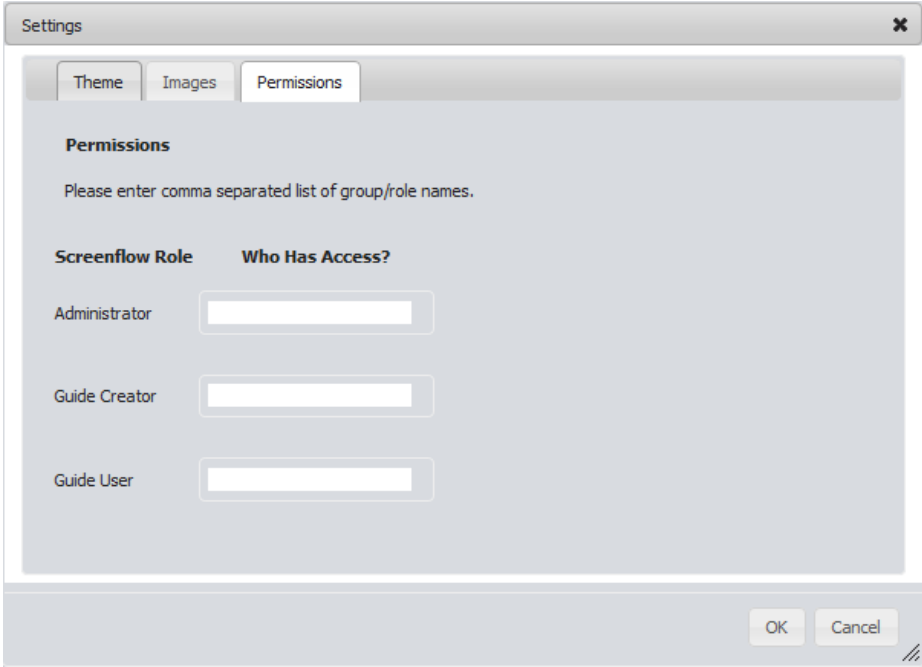
Button	Description
	Opens a list of objects that you can create. When you click an object, it opens in a new tab.
	Opens a selected object. **Note:** You can also open an object by double-clicking on it.
	Creates a copy of the selected object. Process Designer adds a numeric suffix to the object's title; for example, the suffix could be '(1)'. **Note:** To rename an object, you must unpublish it, change the name, and then publish it again.
	Deletes the selected objects. Click **Delete** and then click **Yes** to indicate that you are sure that you want to delete the object. **Note:** You cannot undo deleting an object. Also, when you delete an object, you unpublish the object.
	Writes objects to your computer. Click the Export button, select one or more guides in the list, and then click **Export**. If you select more than one guide, Ctrl+ left click on all but the first. If you select more than one, they will be in a ZIP file. If you select only one guide, the exported file will be an XML file with an additional two or three letters that describe the object being exported. For example, if you export a connection object, the extension will be conn.xml. When you click Export, the following dialog box appears:

Note:
- If the you export a guide that contains images, the images are exported as well. The extension of an exported guide that contains images is .gd.zip.
- When you export more than one object in a single export operation, they are collectively saved in a single ZIP file.

Button	Description
Import	Imports previously exported objects. When you click the **Import** button, you see the following dialog box: **Import Design** ✖ Design File: [Choose File] No file chosen ☐ Overwrite designs with the same name Tags [ ] [Import] Use the following steps Follow the below steps to import objects: 1. Click **Browse** to find either a single object or a ZIP file with more than one object. 2. Select **Overwrite designs with the same name** if you want to replace an existing object with an imported object with the same name. 3. You can enter tags that describe the imported files in the **Tags** field. You must use only ASCII characters when you enter tags. Other types of characters, like UTF-8, are currently not supported. 4. Click **Import** to import the objects. After you click Import, the following dialog box appears: **Imported Designs** ✖ The following designs were imported. • guide-6 • process-object-6 • sample-6 • Yahoo Weather Service-4 • process-sample-4 [OK] You can click on the objects in this dialog box to see an expanded view of each item in the background. **Note:** - A ZIP file could include images associated with a step. If you select the **Overwrite designs with the same name**, all existing images are replaced by imported images with the same name. - When you import a ZIP file containing more than one object, they are unpacked individually.
Publish	Click **Publish** to make the object ready for use, and visible to other users. **Note:** You do not need to publish Process Objects. They are available as soon as you save them. If you republish an object that is being used by another user, all running instances are migrated to the newly published object. If, however, all running instances are not migrated, the old definition is used To stop some objects, like guides, from being used with an older definition, click**Restart** on a running guide to use the new definition. If you want other users to use only your newly published guide, click**Terminate Running Guides** at the bottom of the Details area. This stops all instances of that guide. **Note:** If you publish a process that uses Salesforce, and the process is invoked by its URL, you must have a published Salesforce connection to invokes a process.
Unpublish	Click **Unpublish** to make an object unavailable and no longer visible to users. If the object is being used by others when you unpublish it, it remains available to them until the process stops running. **Note:** If you urgently need to stop other users from using a running guide (but not a running process) click **Terminate Running** button at the bottom of the Details area.

Button	Description
Find Design	Use the **Find Design** field to search for text that matches what you enter. This text can be part of an object's name or an object's description. After you type three letters, you see a list of objects containing this text. As you type additional letters, irrelevant objects no longer appear.

Button	Description
⚙ Settings	**Note:** Settings only apply to guides created within Guide Designer. Click **Settings** to view a dialog box where you can select a theme or upload an image. Following is a screen shot of the Theme tab: When you select a theme, the way the guide page displays changes. You can edit the way a theme appears by altering the CSS (Cascading Style Sheet) that it uses. For more information, see Customizing Themes. If you select **Mobile Theme** for a guide, your users will only see the guide when they access it from a mobile phone browser. Use the **Images** tab to upload images that you can place in steps. The following is a screenshot of the **Images** tab

Button	Description

The fields within this dialog box are:
- Search area: Use the top-left field to find an image using text that you enter in the **Title** or **Description** fields. Use the right text box to type a tag that was used to label an image. As you type a tag's text, you will see tags that match what you are typing.
- **URL**: The URL for the imported image.
- **Title**: A title for the image.
- **Description**: Text that describes the image.
- **Tags**: If you have many images, associate each image with a tag. For the image in the above figure, the tag is 'flowers'.
- **Browse**: Click this button to select an image on your computer. After you select and confirm your choice, the image appears here.

To delete an image, place your cursor over the image, and then click the small red "x" that appears on it.

Some applications may display a **Permissions** tab, as shown below:

Button	Description
	The fields within Permissions dialog box are: - Administrator: Administrators have complete access to guides. They can create guides, run guides, and use the **Permissions** tab. Only administrators can see and use the **Terminate Running Guides**. This button appears in the upper-right panel of the Guides pane. - Guide Creator: Guide Creators can create and run guides. However, they do not see the **Permissions** tab. - Guide User: Guide Users can only run guides. **Note:** The **Administrator** field is mandatory. The Guide Creator and Guide User fields are optional. Also, you cannot enter a user name here.
View List Icon	Select either **List** or **Icon** from this list to change how information about objects is displayed. Most figures in this area of the page display in the Icon view. The List view, of which a portion is seen here, displays information about objects in a more compact format.

Details Area

The **Details** area contains information about a guide. The items at the top of this area are those that were entered into a Properties dialog. The rest are modification and publication information added by Process Designer. Similar information is displayed for other kinds of objects when you select them in the File area. The following figure shows the Details area from within Process Designer for a guide. Other objects have less information, and only a guide will have a usage area.

Note: If you are using Process Designer from within Salesforce, you will see less information.

In some cases, you will see a **Run Guide** button following the guide's name. After you press this button, Process Designer process launches the guide in a new tab. If this was created in Guide Designer and the theme was Smart phones, the guide displays using the mobile theme.

Notice the *Embed Code* text area. This is an address that you can embed in pages so that the guide can be accessed outside of Process Central. For example, this could be the address that a mobile phone user taps to use a mobile guide.

The **Terminate Running** button stops this guide from executing. Because more than one user could be using this guide, this ends all of them. If you did not publish a new version of this guide, Process Designer restarts the guide the next time a user opens it.

Note: Only users with admin permissions can terminate a running guide.

If you had republished a guide that is being used, Process Designer switches people using it to the new definition if it can. If they should be using the newly published guide, press this button.

Customizing Themes

Built-in themes determine how a guide looks on desktop/tablet devices and smart phones. When a guide displays, it uses the theme set in the guide properties.

You can also customize guide themes and add images.

To customize a guide theme, select **Settings** in the toolbar:
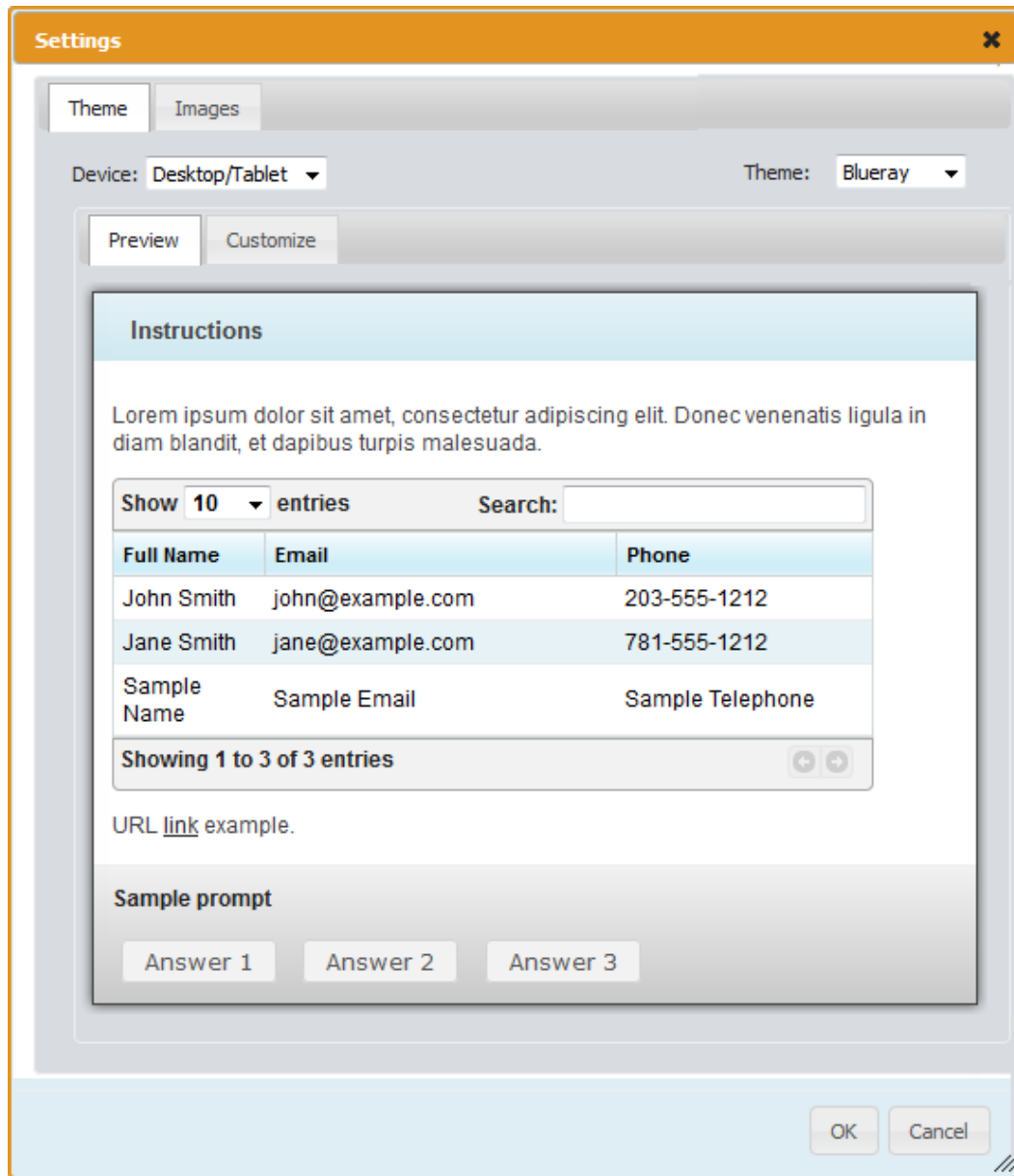
Here, you can make and preview theme changes.

Click the **Theme** tab to display options to customize and preview.

Select the target device from the **Device** list to display the available themes for that device. For example, if you select **Desktop/Tablet**, the available themes include Blueray, Redmond, and Vader.

Customize Options

Click the **Customize** tab to modify details for a built-in theme.
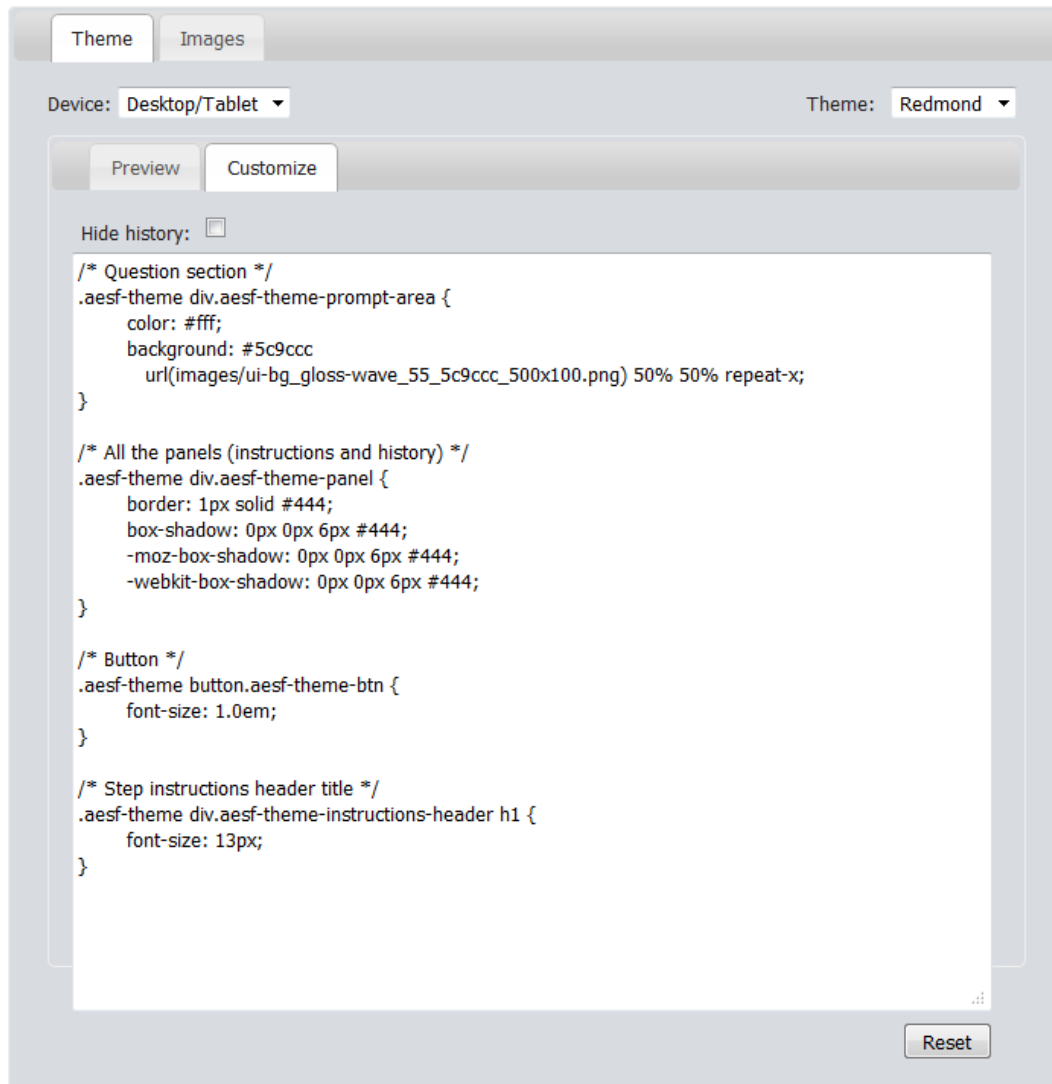
Here, you can modify the CSS class details for each part of the guide, step, or paragraph or character style in the theme. For example, to change the color of the step's header text to blue, you would set the following:

```
aesf-theme div.aesf-theme-instructions-header h1 {
    color:blue;
}
```

You can use any CSS property normally available for the element type.

See Settings for information on adding images.

Other Commands

You can also use these options:

- **Hide History**: Check to hide the history pane when a guide launches. (The user can still display this panel during guide execution.)
- **Reset**: Click to revert to the original theme settings.

Usage Area

Note: The Usage area and its buttons are only available for guides.

Pressing the buttons within the **Usage** area tells Screenflow to display summary information about the selected guide. Information appears in its own tab.
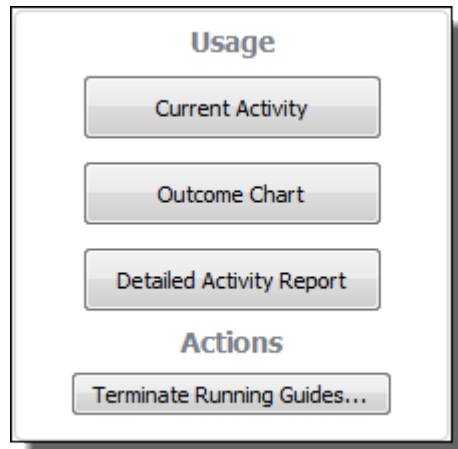
Note: If logging was turned off for a guide, information is either limited or will not exist. This is set using the *Guide Execution Logging* setting within the Guide Properties Advanced tab.

Current Activity

Note: The Usage area and its buttons are only available for guides. Also, only users with admin permissions can view this report.

Selecting the **Current Activity** button tells Process Designer to display a *Current Activity* tab for the selected guide. The information displayed shows the interactions that occurred today.

The table displays:

- **User**: The user who ran the guide.
- **Object**: The object (often a person) upon which the guide was invoked. Click on the link to go to it.
- **Current Screen**: The step being executed when this report is created.
- **Started**: The date and time at which the guide began executing.

Note: If logging was turned off for a guide, information is either limited or will not exist. This is set using the *Execution Logging* setting within the Guide Properties Advanced tab.

Outcome Chart

Note: The Usage area and its buttons are only available for guides. Also, only users with admin permissions can view this report.

Selecting the **Outcomes** button tells Screenflow to display a pie chart showing the different outcomes that occurred.

Note: An "outcome" is a value (always text) that indicates the result of executing a guide to its final step. These outcomes were set in the guide properties.
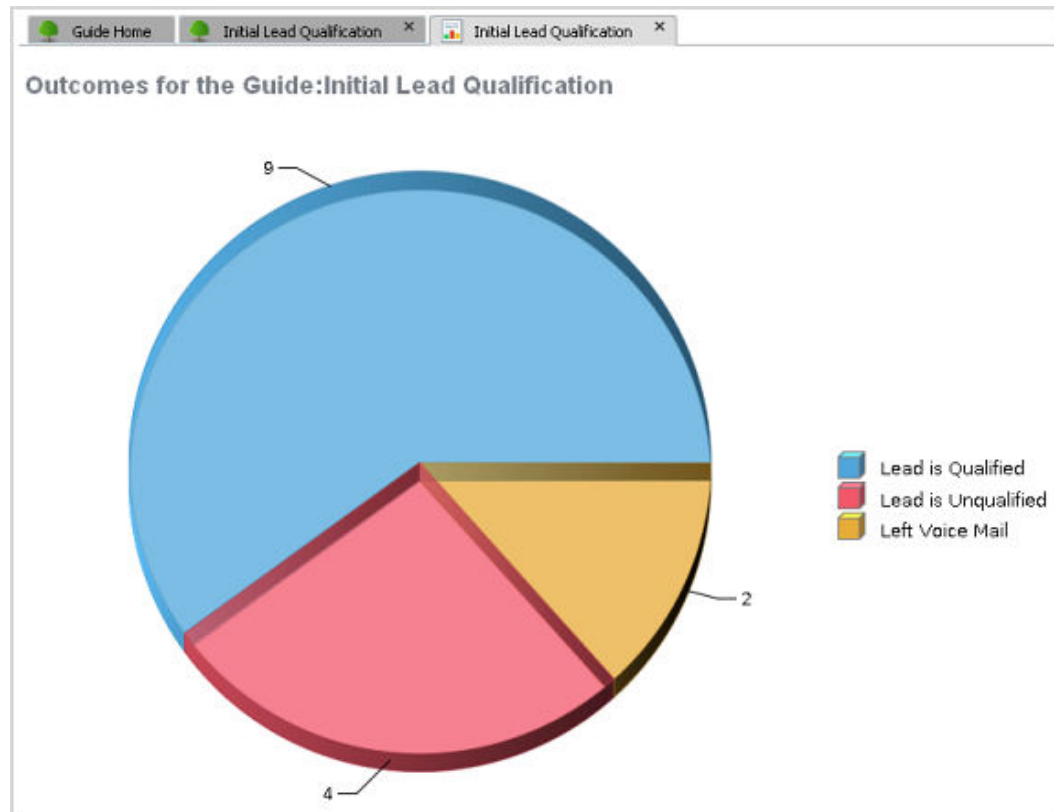
The number of outcomes shown in the chart is the total for all interactions, not just today's.

Note: If logging was turned off for a guide, information is either limited or will not exist. This is set using the *Guide Execution Logging* setting within the Advanced guide properties.

Detailed Activity Report

Note: The Usage area and its buttons are only available for guides. Also, only users with admin permissions can view this report.

Selecting the **Detailed Activity Report** button tells Screenflow to a display a comprehensive activity report. It has more information that is shown in the **Current Activity** report.

The table displays:

- **User Name**: The user who ran the guide.

- **Object**: The object (often a person) that launched the guide. This is shown as a link that you can use to go to the object's Screenflow's page.

- **Screen**: The last step executed within the guide. If you click on an item's link, Screenflow displays a log, showing activities.

- **Start Date**: The date and time at which the guide began executing.

- **Duration**: (not in Current Activity report) The amount of time it took to execute a guide's steps.

- **Outcome**: (not in Current Activity report) The outcome set when the guide completed executing.

The top of the report contains buttons and controls on both the left and right. The right side buttons let you move between pages. The left side is discussed in the following sections.

Note: If logging was turned off for a guide, information is either limited or will not exist. This is set using the *Guide Execution Logging* setting within the Guide Properties Advanced tab.

Toggle Table of Contents

> **Note:** These buttons, the Detailed Activity Report, and the Usage area are only available for guides. Also, only users with admin permissions can see this information.
>
> Pressing the **Toggle table of contents** buttons tells Screenflow whether it should display a table of contents with the report.

Export Data

> **Note:** These buttons, the Detailed Activity Report, and the Usage area are only available for guides. Also, only users with admin permissions can see this information.
>
> Pressing the **Export data** button tells Screenflow to display a dialog box that lets you define the way in which it exports a report's data.

> The following snapshot shows the controls within this dialog box:
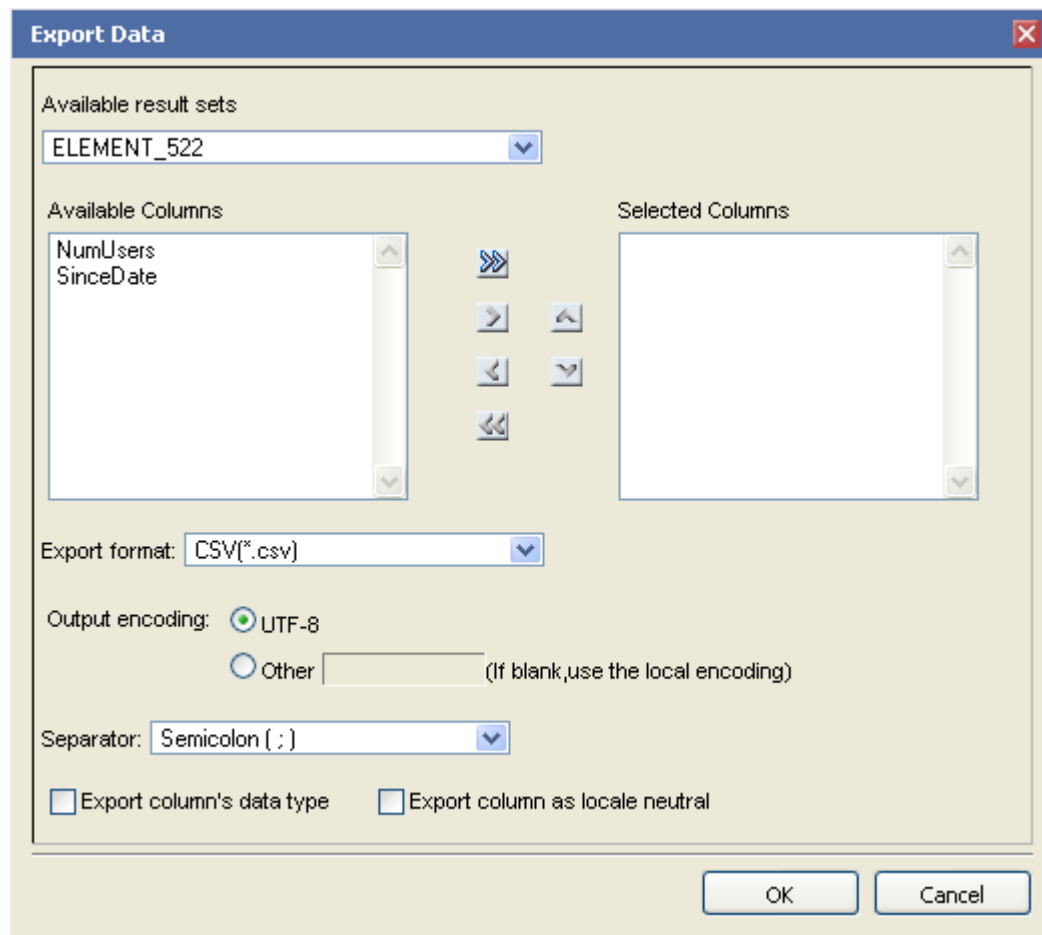
> The only control that you should change is **Separator**. This command names the character that separates columns within a record in the exported data. Choose one of the following: *Comma*, *Semicolon*, *Colon*, *Vertical Line*, or *Tab*.

Export Report

Note: These buttons, the Detailed Activity Report, and the Usage area are only available for guides. Also, only users with admin permissions can see this information.

Pressing the **Export report** button tells Screenflow to display a dialog box that lets you define a report's output format. Your report is not paginated or otherwise changed by Screenflow. Instead, the data it will write will be compatible with one of these applications.

Here is this dialog box:
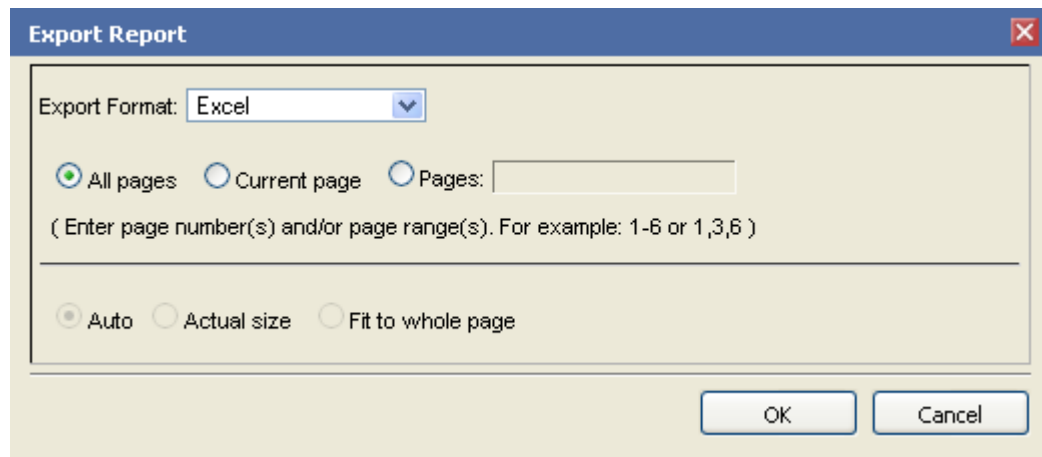
Items within this dialog box are:

- *Export Format*: Choose the export format. Screenflow can export the report in the following formats: *Excel*, *PDF*, *PostScript*, *PowerPoint*, and *Word*.

- *Pages*: Sets the pages to be written. Your choices are *All pages*, *Current page*, and *Pages*. If *Pages* is selected, you can enter page numbers and page ranges; for example, 1-6, 9, 15, 21-24.

- *Size adjustment buttons*: If PDF, PostScript, or PowerPoint is selected, you can set the relationship between the report's actual page size and its print size. Your choices are *Auto* (which will make the report smaller if it will not fit on the page), *Actual Size* (no size changes are made), and *Fit to whole page*, which can either make the document larger or smaller so that it fits on one page.

Print Report

Note: These buttons, the Detailed Activity Report, and the Usage area are only available for guides. Also, only users with admin permissions can see this information.

Pressing the **Print Report** button tells Screenflow to create a file in the format you specify, display it in its application (which is either a browser or Adobe Acrobat), and invoke the application's print dialog box so that you can print it.

The following illustration shows the controls within this dialog box:
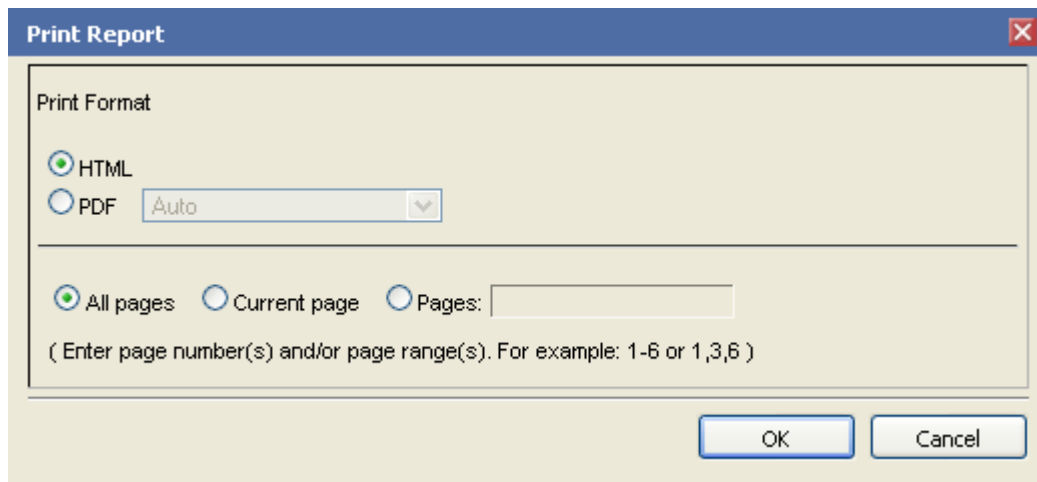
The controls within this dialog are:

- **Print Format**: Choose either *HTML* or *PDF*. If you choose PDF, additional options let you set how the page's image fits on a page, as follows: *Auto*, *Actual Size*, and *Fit to whole page*. (These options are discussed in "Export Report" on page 36.)

- **Pages**: Sets the pages to be written. Your choices are: *All pages*, *Current page*, and *Pages*. If *Pages* is selected, you can enter page numbers and page ranges; for example, 1-6, 9, 15, 21-24.

Creating Guides and Processes

You can create guides or processes while working in Process Designer or while working in the Salesforce Guide Designer, if you have licensed the Salesforce Managed Package.

1. From Informatica Cloud, select **Design** > **Services & Processes** to start Process Designer. From Salesforce, select **Guide Designer**.
2. From the Design Home page:

- Double-click the guide name or click **Open** in the toolbar to edit an existing guide or process.
- Click **New** and select Guide or Process to create a new object.

You can complete the guide or process properties described in the following sections.

Note that:

- Each guide or process has a **Start** step to identify its starting point.
- For each subsequent step, you can specify properties using the **Property Sheet** accessible on the right side of the canvas. Open the **Step Type Palette** on the left side of the canvas to drag specific step types into the guide or process.
- Each guide or process also contains an **End** step.

Navigating the Tree

Most trees contain more steps than can fit on your screen. They are almost always too wide and often too high. To help you focus on the steps you are creating and editing, the guide designer focuses on the step you're currently working on, only showing a few steps before and after it. It also mostly limits what it is displaying to the branch containing this step and perhaps a few steps in nearby branches.

Using Your Keyboard

You can use the following keys and shortcuts:

Key	Description
Up Arrow	Moves to a step immediately above the current step
Down Arrow	Moves to a step immediately below the current step
Left Arrow	Move to the step immediately to the left of the current step
Right Arrow	Moves to the step immediately to the right of the current step
Insert	Inserts a new step in front of the selected step
Delete	Deletes the selected step or steps and its subtree
Ctrl+X	Cuts (removes) the current step or step and its subtree
Ctrl+C	Copies the current step or step and its subtree
Ctrl+V	Pastes the step or subtree previously cut or copied
Ctrl+Z	Undoes the previous action
Ctrl+Y	Restores the action that was undone by the **Undo** command

Using Your Mouse

Changing Which Steps are Displayed

The space where you view the process might be too small to see all of process's steps. You can get to other part of the process by clicking your left mouse in white space within the window, then dragging the window's contents.

Opening a Step

To open the **Step** Information dialog, place your cursor over a step and double click. The editor then displays the **Step** Information dialog. When you are done working on a step, you can select one of the following buttons at the bottom of the dialog:

- **OK**: Applies the changes and closes the step editor.
- **Cancel**: Cancels the changes and closes the step editor.
- **Apply**: Applies the changes but does not close the step editor. You can select another step to work on without closing the editor.

Other Ways to Use Your Mouse

Your mouse also lets you perform the following actions:

Zooming

Use the wheel on your mouse to zoom in and out, making what you see either larger or smaller.

Inserting

Double-click on the arrow between steps to insert an empty step between the two.

Rearranging Branches

When the editor adds branches to the right of a step--for example, after you've inserted a Decision step-- you can rearrange the branches by selecting a branch and move it up or down.

Moving a Step to a New Place in the Guide

Click on a step, then drag it to it's new location.

Moving a Step and the Steps that Follow

To move a step and the steps that follow to a new place in the guide, click and hold the arrow preceding the steps to be moved and drag the arrow into its new position.

Jump Steps

While most steps have lines leading away from them, a Jump Step does not. After the jump is defined, you will see a link to the right of the step.

Here's an example:
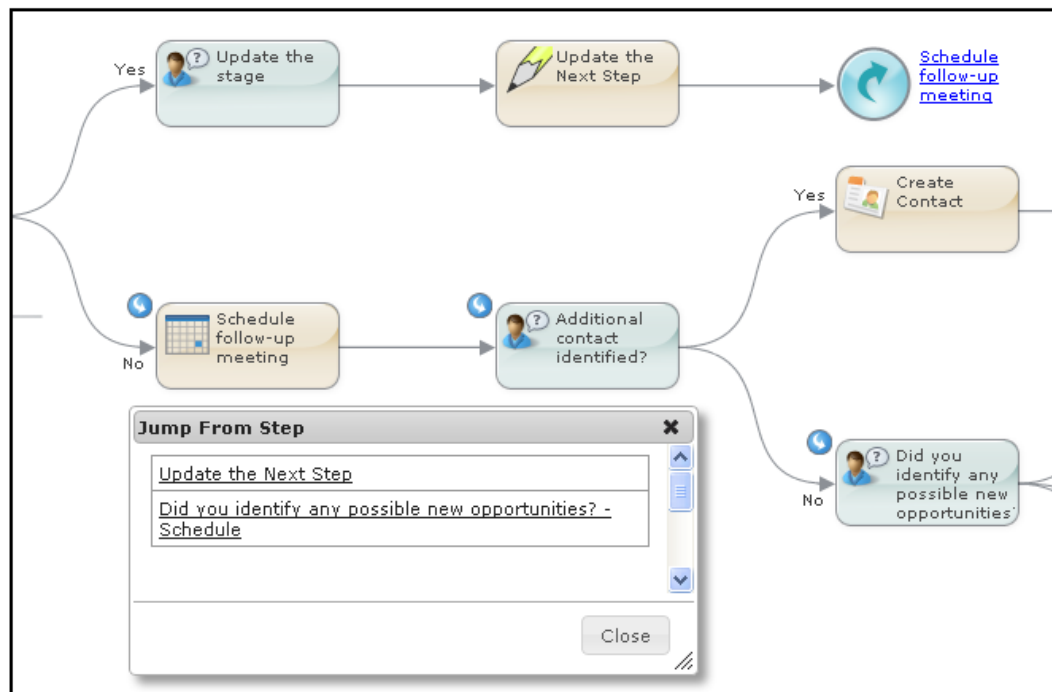
If you click on the link in the upper-right corner, the designer selects the step being jumped to. In this figure, this is step in the lower-left corner.

Three of the steps in this figure have small arrows similar to a Jump Step in their upper left corner. This lets you know that another step is jumping to it. If you click on a small arrow, the editor displays a dialog that names the steps jumping to this step. If you click on one of these "jump from" steps, the editor focuses on that step.

Embedded Steps

Embedded steps, let you go directly to the embedded guide. Similarly, the guide that was embedded can place you in its embedding guide. In the following figure, the embedding guide has a "+" button within it. If you press that button, the embedded guide is displayed.

In the embedded guide, notice the inward pointing arrow at the start step. If you click this button, it displays a dialog containing the guides that embed this guide. Clicking on the link within that dialog takes you to that guide.
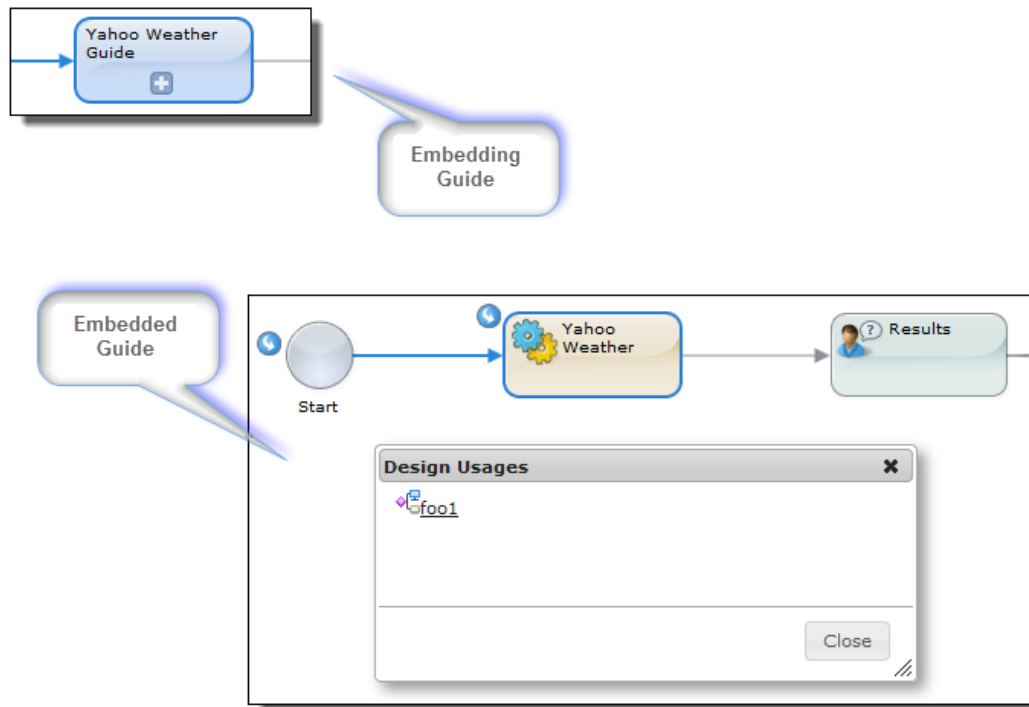
Setting Guide Properties

All guides share some common characteristics. These include having a name, the type of object that the guide applies to (for example, lead, campaign, opportunity, and so on), possible outcomes (for example, "Lead is qualified"), and the way it looks when displayed.

You use will use the Guide Properties dialog to set these and other characteristics.

General Tab

You can specify the following General properties for guides:

Property	Description
Name	A descriptive name to identify the guide in Guide Designer and the name that appears when it is available for use in other objects such as subprocess steps. **Note:** - Guide names cannot begin with a number. If you enter a number as the first character, Guide Designer adds an underscore to the name. For example, "123" becomes "_123". - To change the name of a published process, you must first unpublish the guide. Then, change the name and republish the guide.
Location	The location of the project or folder you want the process to reside in. To select a location for the guide, browse to the appropriate project or folder, or use the default location.
Description	A description of the guide.

Outcomes Tab

When a user finishes the last step in the guide, you may want to note how it ended. These endings are called 'outcomes'.

Use the **Start** > **Outcomes** to define possible outcomes.

In the following image, six outcomes are defined for a guide:
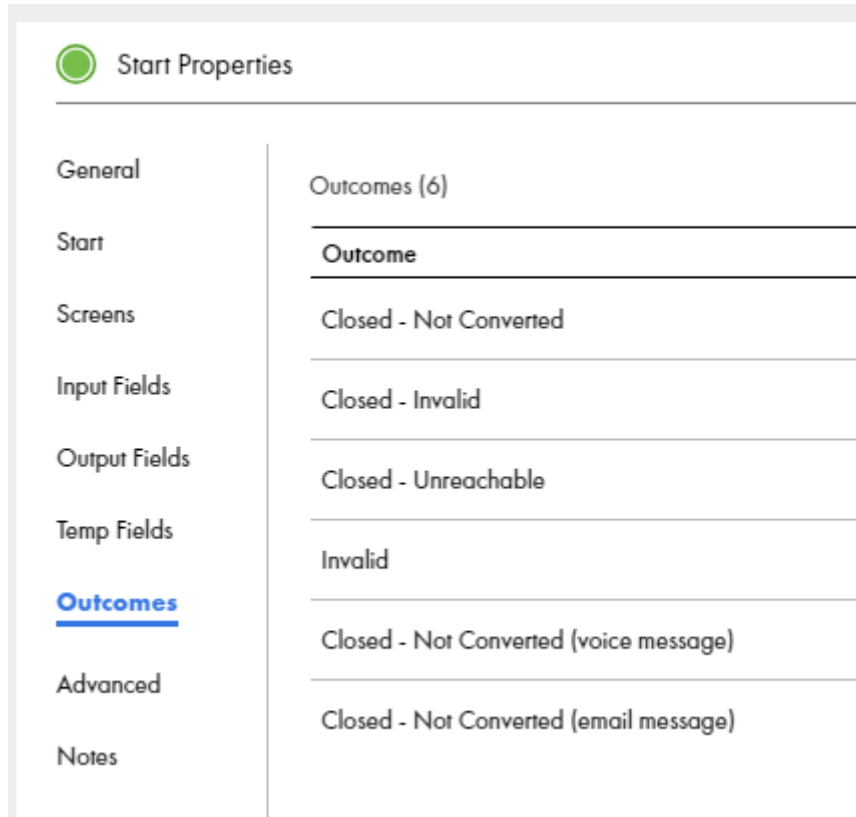
The following image shows the canvas view of a guide with part of a guide with four outcomes:
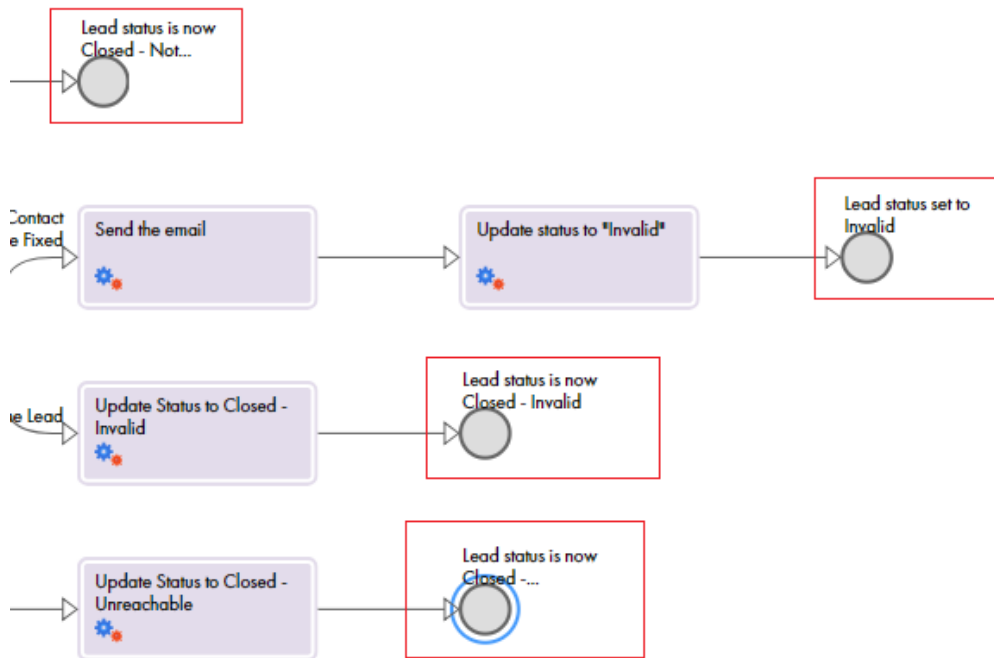
You can also add an outcome while editing an End/Milestone step by selecting **Add new outcomes** in the list of available outcomes.

Fields Tab

The fields available for a guide or process are those contained in the object(s) to which the guide or process applies. This is specified in the Applies To property (on the General tab).

You define a name and type for each of the following field categories, for use in a guide or process:

Field Categories	Description
Input	The Input field values are available in all steps of the process or guide.
	An embedded guide may need to access information from the embedding process or guide. Creating input fields is also a way to define this information so that it can be passed to the embedding guide.
	If the field is required for the guide or process to execute, check **Required**.
Output	The Output field values are set when the process executes and are then available in the steps that follow. Define output fields only for use in an embedded process or guide. These output fields do not appear in lists until the values are returned from an embedded process or guide.
Temporary	The Temporary field values be used in all of the current guide's steps. However, a temporary field's value is not available to an embedded guide or process.

Advanced Properties

Use the Advanced tab to set the following properties:

Allow Restart

Determines whether users can restart this guide from the first step. If you enable this option, when the guide restarts, it undoes previously executed steps.

Execution Logging

If checked, Application Integration logs details about the guide steps and execution, so it is available for reports.

Rollback Screens

If checked, your application can undo changes made to the database when a user clicks **Go Back One Step** or clicks a step in the History panel. If it is disabled, changes are not undone.

If an embedded guide enables roll back, then the undo is applicable only to the scope of the embedded guide, unless the parent guide also enables roll back.

Always Start New

If checked, opening a guide always creates a new instance of the guide.

Feedback email list

If you want users to send feedback, add one or more email addresses (separated by commas) to receive feedback. Users can then click the feedback option on the guide to display the following dialog:

The dialog shows the guide's name and the step being executed.

Show Guide Only When (Salesforce only)

You can specify criteria that determine when this guide displays in the "Guides" area on a user's Salesforce page.

The following image shows the **Advanced** tab:

Creating and Editing Steps

The following sections contain information on creating and editing the different kinds of steps.

Step Common Features

The individual steps you can add to a guide or process determine the execution path and options.

Every step has a title, which displays in the canvas as you design a guide or process.

For screen steps, the title is the prompt. Choose a title that helps the user understand the purpose of the step.

The way in which you set the data used in a step is by setting its source to a content, field, screen, query, or formula.

As you create a step, press **Preview** to review how Process Designer displays the current step.

Guide Screen Editor

The screen editor lets you add instructions and fields to a screen that displays in a guide.

The toolbar contains commonly used editing commands. If you press the ▽ button (**Show/Hide More Tools**) at the right edge of the toolbar, Process Designer displays additional options.

The **Insert Read-Only Field** and **Insert Field for Update** lists are described in the Field Picklists topic.
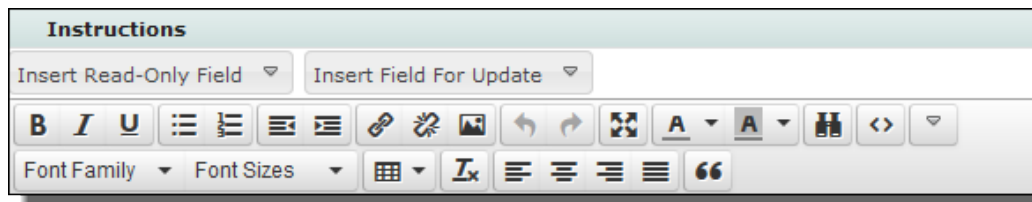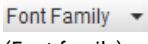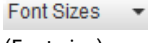
The screen editing options are:
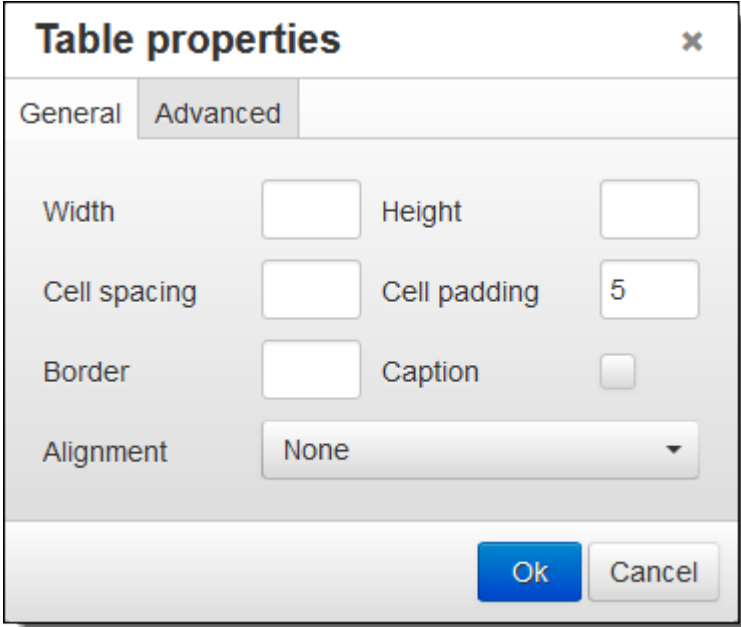
Button	Description
B (Bold)	Sets selected text to **bold** or indicates that the text you are about to type will be bold. If you press this button when bold text is selected, the bolding is removed. (Keyboard: Ctrl+B)
I (Italic)	Sets selected text to *italic* or indicates that the text you are about to type will be italic. If you press this button when italic text is selected, the italic is removed. (Keyboard: Ctrl+I)
U (Underline)	<u>Underlines</u> selected text or indicates that the text you are about to type will be underlined. If you press this button when underlined text is selected, the underlining is removed. (Keyboard: Ctrl+U)
(Bullet list)	If the current paragraph does not have a bullet, adds a bullet and changes the paragraph's indent. If the paragraph has a bullet, this command removes the bullet and removes a level of indenting. When you press Enter within or at the end of a paragraph, the next paragraph also has a bullet. To end the bullet list, either type two returns or reselect this icon.
(Numbered list)	If the current paragraph does not have a number, adds a number and changes the paragraph's indent. If the paragraph has a number, this command removes the number and removes a level of indenting. If the preceding paragraph is part of a numbered list, selecting this command adds it to that list. When you press Enter within or at the end of a paragraph, the next paragraph also has a number. To end the numbered list, either type two returns or reselect this icon.
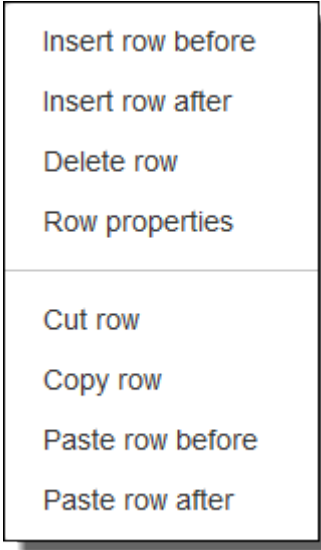
Button	Description
(Decrease indent)	If the current paragraph or selected paragraphs are indented, it removes a level of indenting.
(Increase indent)	Adds a level of indenting to the current paragraph or to selected paragraphs.
(Insert/edit link)	Changes the selected text into a link. After you select this button, Process Designer displays the following dialog: **Insert link** ✕ Url [ ] Text to display [ ] Title [ ] Target [None ▼] Ok Cancel The files in this dialog are: - **URL**: Where you type a Web address. - **Text tp display**: If you wish text to display instead of the URL, type that text here. - **Title**: Text that appears when a user moves the cursor over the link. (This text may not appear in all browsers.) - **Target**: Select New Window from this picklist to display the link in a new window. If you do not change this from None, the URL appears in the current window. If you press this button while your cursor is anywhere in a link, you can edit the link's information. If you forget to select text, the link will not display. However, it will be created. Correct this by pressing the **<>** button to locate the link. If you know HTML, you can edit the link directly. If you do not, just delete it. (An empty link's text is similar to "`<a href=""><a>`".)
(Remove link)	Removes a previously created link. Your cursor can be anywhere within the link. The text in the "Text to display" field remains after you select this button.

Button	Description
(Insert/ edit image)	If an image is not selected, press this button to insert a new image. To change an existing image or its properties, select the image before pressing this button. After pressing this button, Process Designer displays the following dialog:

The images you see in the top area were previously imported into Process Designer using the **Settings** button located on the Design Home page. See the Process Central help for information on settings..You can also upload an image for use in the current screen step. An image uploaded here is not available for use in other steps within this guide or in any other guide.

You can also insert any image available on the web into the step. Do this by dragging it from the web site directly into the screen editor at the position at which you want to inset it.

The fields within this dialog are:
- *Search area*: Use the left text box to search for an image using text that you typed either in the *Title* or *Description* fields. Use the right text box to type a tag that was used to label an image. As you type a tag's text, you will see tags that match what you are typing.
- **URL**: If the is accessible using a URL, enter it here.
- **Title**: A title for the image.
- **Description**: Text that describes the image. Notice the text that was added to this dialog.
- **Width x Height**: Use these fields to change the width and height of an image. You can either enter the size in pixels or enter a percent of the original size. The values you type can be less than or more than what the image's original dimensions were. If you only type a width and you entered pixels, the size is changed proportionally. For example, if you change the display width to 235 in the previous example, the display height will be 93 or 94 depending upon your browser. If you entered a percent value, that percent is also applied to the height.
- **Original W x H**: The images width and height as it is stored.

Button	Description
	- **Float**: Use the options within this picklist to indicate the image's position in relation to where you inserted it. Your choices are None (it is displayed in line), Left (the image floats to the left), and Right (the image floats to the right). - **Browse**: Press this button to display a dialog that lets you select an image using your computer's file explorer. After you select and confirm your choice, the image will appear here.
(Undo)	Eliminates the previously executed action. (Keyboard: Ctrl+Z)
(Redo)	If you have previously selected **Undo**, pressing this restores the change. (Keyboard: Ctrl+Y)
(Fullscreen)	Enlarges the size of the step so that it completely fills the area beneath the Process Designer tabs. If you have already enlarged it, pressing this button restores the step to its original size. **Note**: You cannot insert fields into the step when the editor is in full screen mode.
(Text color)	Pressing this button displays a pallette of colors that can be used when displaying text. If no text is selected, pressing this button and selecting a color changes the color of the text you type. If you move the cursor away from this text, the color is what exists at the cursor. Pressing this button and selecting a color while text is selected changes the color of the selected text.
(Background color)	Pressing this button displays a pallette of colors that can be used for coloring the background behind text. If no text is selected, pressing this button changes the background color of the text you type. If you move the cursor away from this text, the background color is what exists at the cursor. Pressing this button and selecting a color while text is selected changes the background color of the selected text.
(Find and replace)	Pressing this button tells Process Designer to display a dialog into which you can type the text you wish to locate in the current screen. Use the Process Designer's Find Step command to search for text in all of the guide's steps. If you enter text in the Replace with field and press the Replace button, the selected text is replaced. If you select Replace all, all the text that can be found is replaced.

Button	Description
<> (Source code)	If text is being displayed, pressing this button tells Process Designer to display the HTML that is being used to format the text in a different browser window, which is the HTML source editor. This button is not a toggle. After you finish making changes in the source editor, press its **OK** button.
▽ (Show/hide more tools)	Adds a second row of screen editor buttons. If this second row is showing, pressing this button removes it.
Font Family ▼ (Font family)	Pressing this button displays a picklist of the fonts that can be used to display text. If no text is selected, pressing this button and selecting a font family changes the font of the text you type. If you move the cursor away from this text, the font will be what exists at the cursor. Pressing this button and selecting a font while text is selected changes the font of the selected text.
Font Sizes ▼ (Font size)	Pressing this button displays a picklist of the font sizes that can be used to display text. If no text is selected, pressing this button and selecting a size changes the size of the text you type. If you move the cursor away from this text, the font size is what exists at the cursor. Pressing this button and selecting a size while text is selected changes the size of the selected text.
▦ ▼ (Table)	Pressing this button tells Process Designer to display a picklist of table commands. Select Insert Table to display a grid of boxes. Use your mouse to select the number of rows and columns that will appear within the table. If the cursor is within a table, you can select other commands in this picklist.

Button	Description
table properties	Selecting Table Properties within the picklist displays a dialog box with two tabs.

Use this dialog to display the height and width of the table. (You should probably set the width as a percent -- use the percent symbol.) The height is seldom set. Cell spacing sets the space between table cell. Cell padding is the amount of space between a cell's contents and the cells edge. If you want a border, enter a number to indicate its width. If you want a caption to a table (this is not usual given the small size of the step's canvas), select this button. A small area above the table will appear. Finally, use the Alignment picklist to select None, Left, Center, or Right. None and Center are the most commonly used.

The Advanced tab lets you set the table's style, and border and background colors.

Select the **Delete Table** command to remove the table from the canvas.

| *cell properties* | If you select Cell from the Table picklist, a second picklist displays. Use this picklist to define cell properties, merge two cells together, or split cells that were merged.

When defining cell properties, you can specify the cells width and height (normally, you wouldn't select a height), its type (Cell or Header), the Scope (None, Row, Column, Row group, or Column group) of what is changed, and the horizontal and vertical alignment of a cell's contents. |

Button	Description
row properties	If you select Row from the Table picklist, a second picklist displays. Insert row before Insert row after Delete row Row properties Cut row Copy row Paste row before Paste row after Row properties are similar to cell properties. Other commands in this picklist are self-explanatory
I_x (Clear formatting)	If you've changed the way in which text displays, pressing this button tells the screen editor to remove your changes.
(Align left)	Aligns currently selected paragraphs or the paragraph containing the cursor to the left margin.
(Align center)	Centers currently selected paragraphs or the paragraph containing the cursor within the left and right margins.
(Align right)	Aligns current selected paragraphs or the paragraph containing the cursor to the right margin.
(Justify)	Adds spaces between words to lines in the currently selected paragraphs or the paragraph containing the cursor so that the text is justified; that is, all text is aligned to the left and right margins.
66 (Blockquote)	Changes the indent of selected paragraphs or the paragraph containing the cursor. If the paragraphs are already within a blockquote, this restores the original margin.

Service Call Step

Guides need not be self-contained. Most guides embed externally created sets of actions, as follows:

- **Service Calls**. To make a call to specific actions from within the current guide.
- **Embedded Guides**. To place an embedded guide within the current guide.
- **Process Calls**. To place a process or subprocess within the current guide.

Note: Service calls are often specific to the object selected in the Properties dialog, so you may not see it. In addition, there are objects that are unique to one object and only appear when that object is selected. For example, if the guide is being applied to a Salesforce lead, you will see a Create Lead service call action.

You define Service Call steps for tasks that can be automatically executed. You can define them within Process Designer or access those available in services and systems integrated with Process Designer, such as Salesforce.

You can see both built-in actions and those added for applications and services categorized within the Action list. For example, the Create category shown here references several actions, including Create Account:
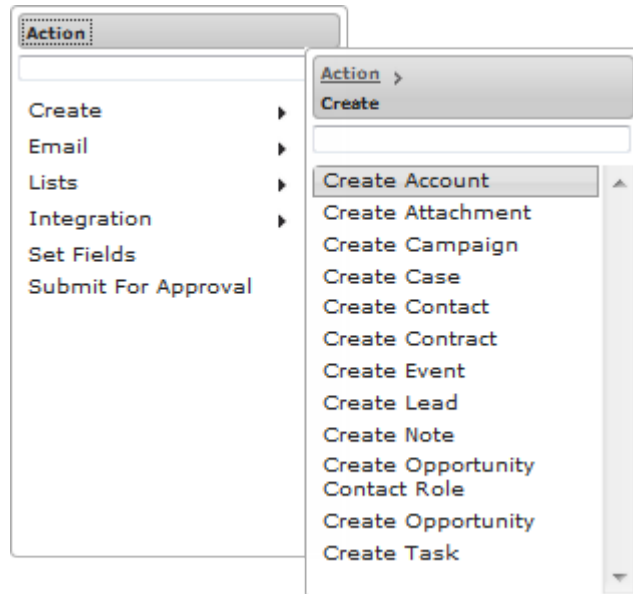
The actions listed are specific to your organization. However, you can usually find the built-in actions in the Advanced, Lists, and Integration categories. Some of the built-in actions are available only if you are licensed to use the Salesforce connector.

Drill down to see available actions using the right arrows.

The Action list also contains a list of Processes if you have published a process in Process Designer, similar to how service connectors are accessible in a process. The list shows the name of each process available to the Service Call step. If the process also exposes events, the event names appear in the list.

Tabs

May step types include three basic tabs, Information, Input, and Input Screen. The following image shows the tabs in a Service Call step:
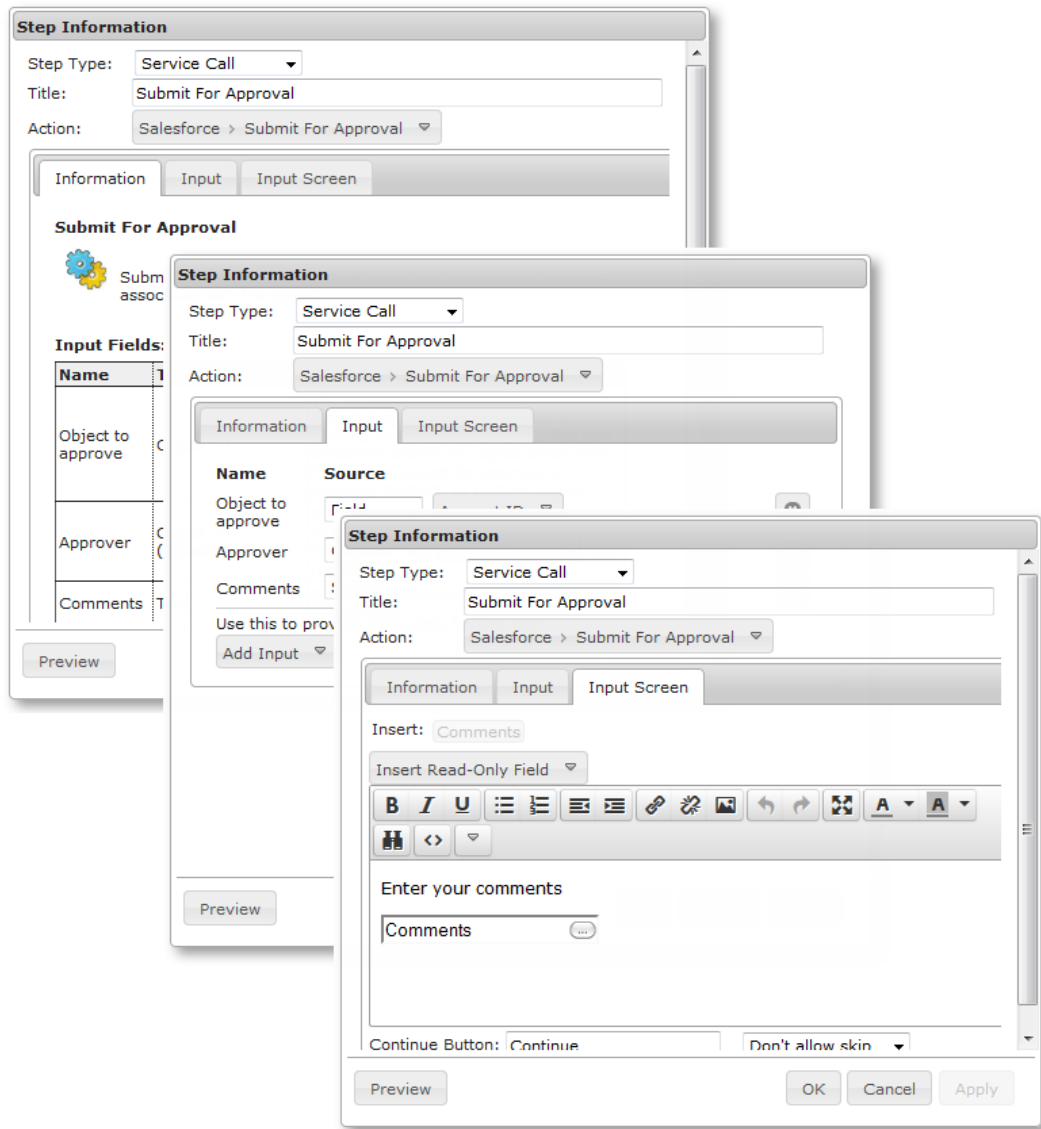
Information Tab

The *Information* tab has a description of the Service Call step and a description of the input and output fields for the step. The descriptions in this tab can help you when you are using the object's input or output fields. Because this information is automatically generated, only input and output field names and types appear in this tab. There are no descriptions for them.

Note: The information tab, depending upon the object the step is operating upon, lets you choose to Run on a single object or Run on each object in list until either the list ends or until an outcome occurs.

Selecting "Run on a single object" tells Process designer that it will only be applying a service call or a subprocess on one object. It the other option is selected, the step operates on each object in an object list. If this option is selected, it is indicated in the canvas with a looping arrow button. Clicking on this buttons opens the invoked step.
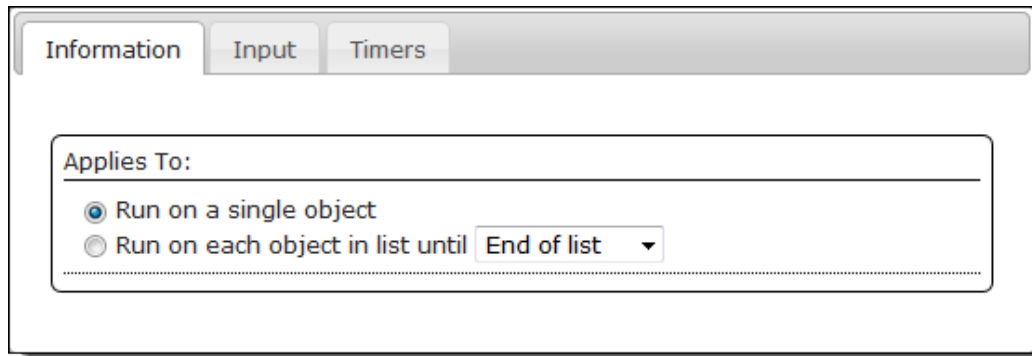
Input Tab

The *Input* tab shows the names of input fields that are most often used when using this kind of step. For Service Call steps that create objects, the input fields shown are the fields that are most frequently needed when the object is created. (If you do not need a field and it is optional, you can delete it.) You can choose additional (optional) input fields from the list. For example:
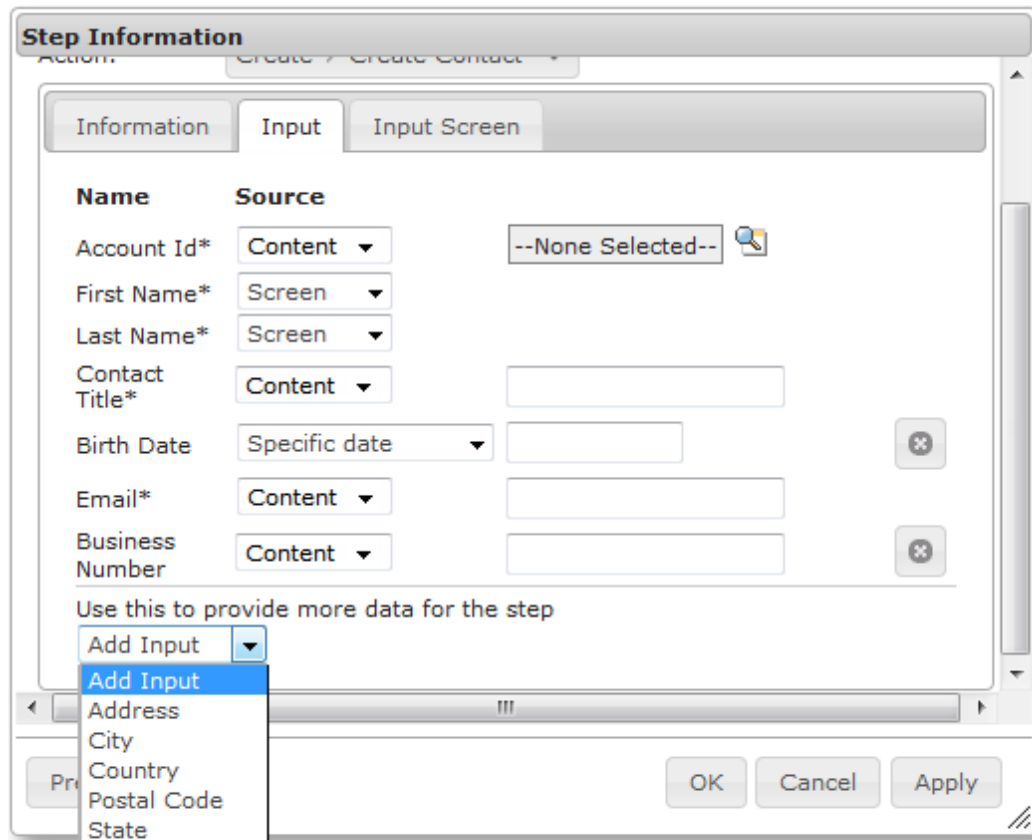
Use the delete icon to remove a field. This removes input fields that you do not want to pass to the Service Call step. The names of required fields are displayed with an asterisk. These fields cannot be deleted.

Use the **Add Input** list to add additional input fields (or re-add input fields that were accidentally deleted). The source for the field's data is obtained by selecting **Constant**, **Field**, **Screen**, **Query**, or **Formula** in the list following the field's name. (Not all of these always appear.)

The fields shown in the Add Input list are more than just those than are shown in the Information tab as it contains all fields within the currently selected object and includes the fields that were declared within the properties.

After a step executes, all steps that follow can access output fields created in the step upstream. This enables you to use the screen editor to insert them into the step's canvas.

Input Screen Tab

Note: The Input Screen tab is only available for guides.

If you select **Screen** as the source of the input field source, you must place that field on a screen using the *Input Screen* tab. For example:
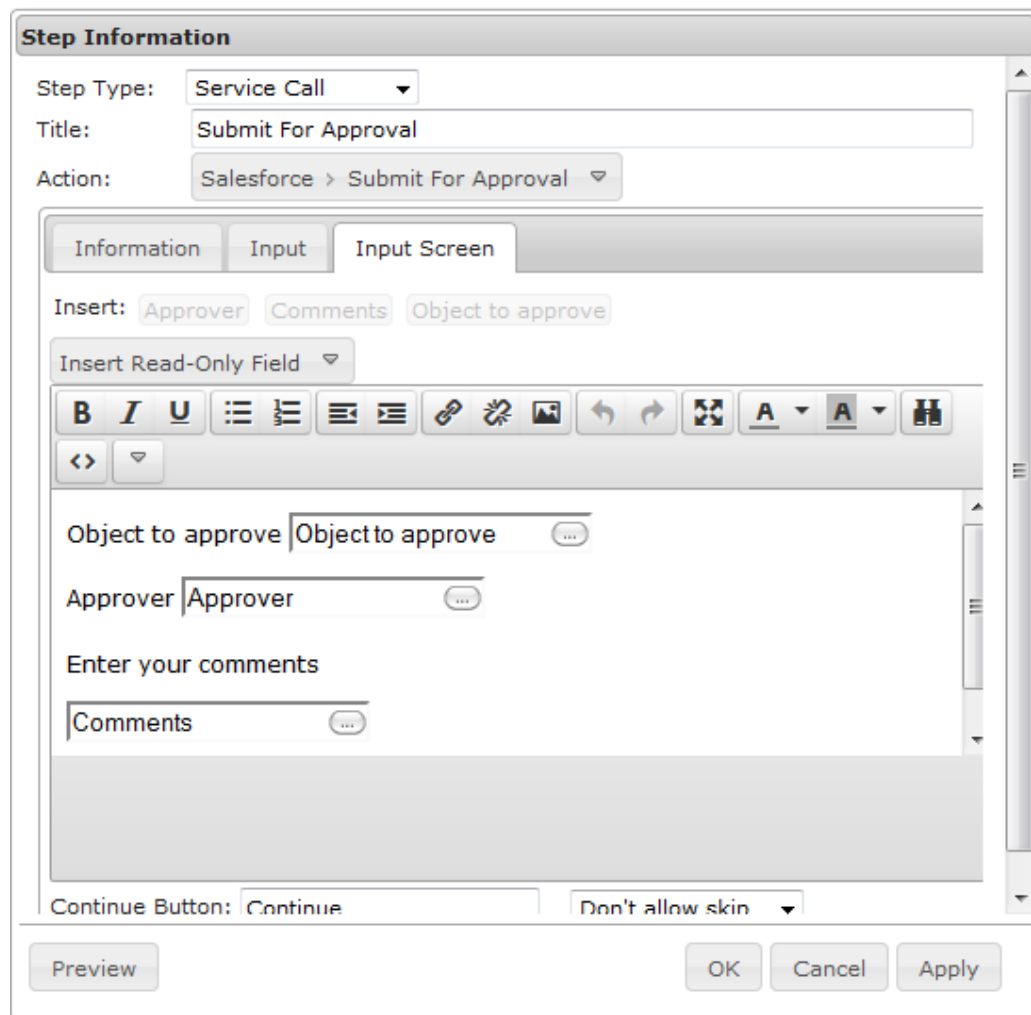
When users see this step, they will see three places in which they can enter information.

To add an input field to the screen editor, click on it. The editor adds input field names as a label followed by an input area for the type of data being requested; for example, a text entry field or a picklist.

Hint: You can make what you enter more readable if you align text and fields.

Here are a few things you should know:

- Because the input fields are being added to the editing screen, you can add instructions and comments and rearrange the way the screen appears.

- Notice the **Insert Read-Only Field** picklist. Use a read-only field to display information that will help the user or present object information. For example, you can display the name of the user who is running the guide (using the "Current User" field) at the top of a screen that is asking for the information needed for creating a new Event.

- You can double click within any input field placed on the screen to display the **Field Properties** dialog.

- Use the "Don't allow skip/Allow Skip Button" picklist to define whether the guide's user can go to the next step without performing any of the step's activities. The following figure illustrates each of these behaviors.
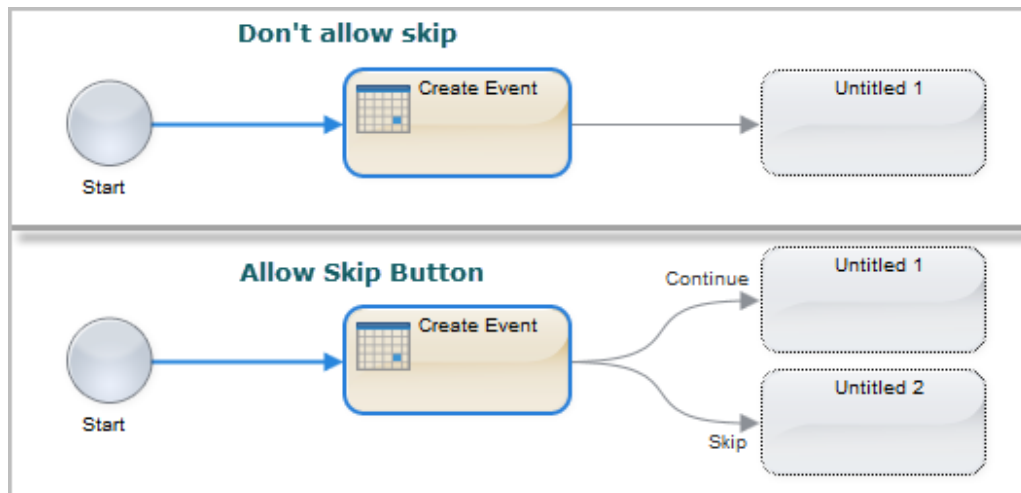
If you select "Don't allow skip", the user must perform the actions indicated within the step. Because this action is required, only one possible action can follow, which is that Guide Designer executes the next step. In contrast, if you select "Allow Skip Button", there are two possible answers: one if the user performs the action and the other when no actions are performed. Because the action is optional, Guide Designer displays a button that the user can press to skip the step. While the default text for the button is "Skip", you can change it by entering text within the data entry area displays that displays after you select "Allow Skip Button."

Events Tab

On the Events tab, you can set timers that determine when the action occurs, and also specify Messages for use with message providers and correlation.

In the previous image, two timers were added. The top shows "After". Here you are specifying a duration; that is, the timer event occurs after the amount of time selected expires. In contrast, the bottom timer event occurs at the time you specify and you can also specify the number of days. Both of these timers are non-interrupting, which means each branch executes separately from other branches. This mode of executing is similar to a Parallel paths step that does not converge to a common step after the branch ends its execution. The difference is that if the timer branch branches do not execute if the non-timer step concludes.
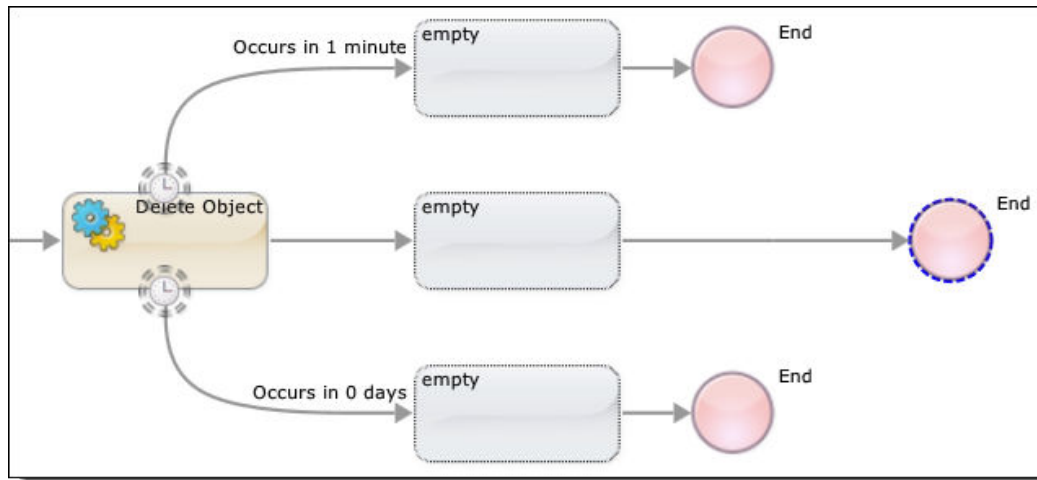
Stated in another way, the timer interrupts the step upon which the timer is set and the timer's path executes.

If you had created interrupting timers, the branches merge back together. Here's an example:
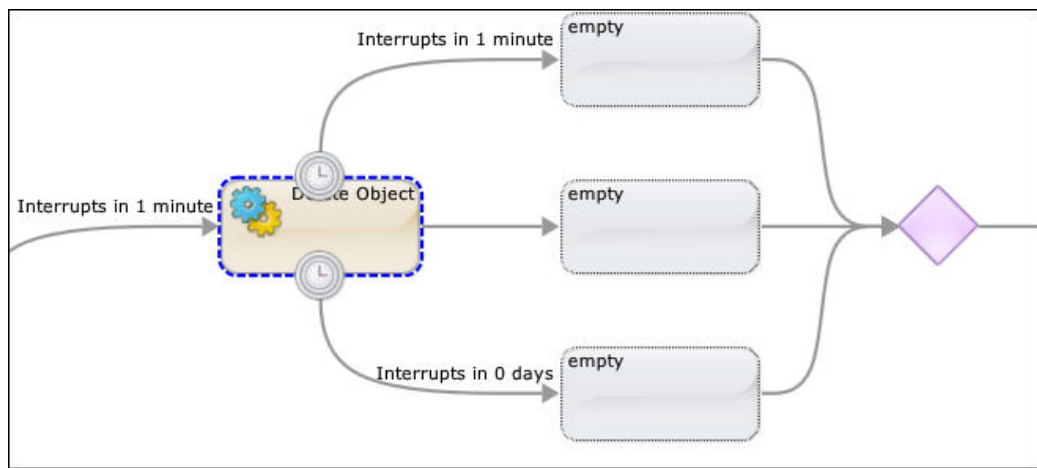
A step can have both interrupting and non-interrupting timers, as the following shows:
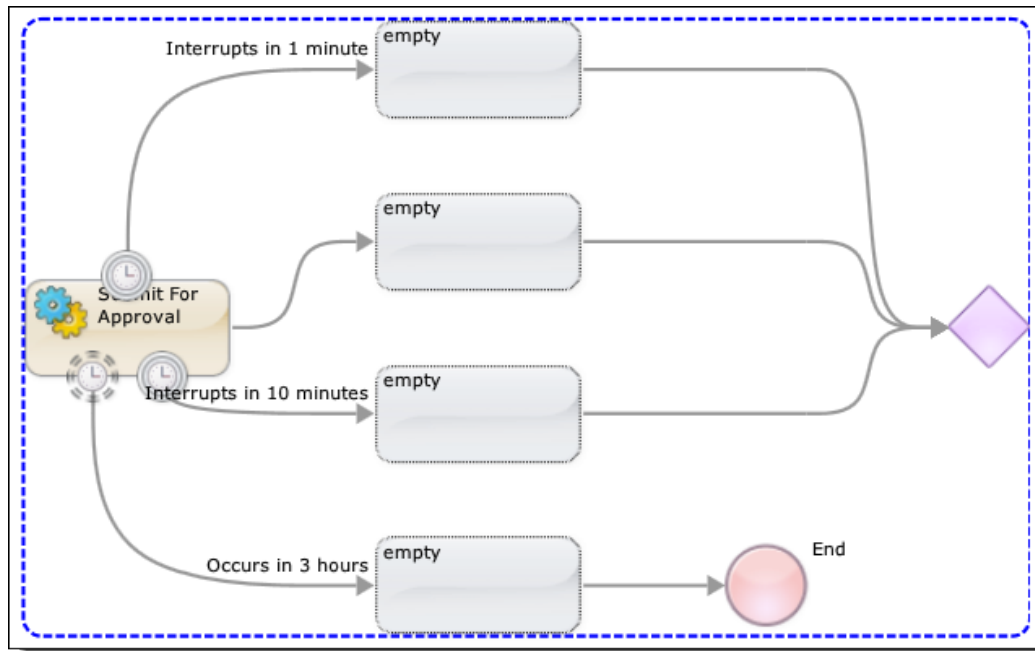
Here are some usage notes:

- To add a timer, press the **Add Timer** button.
- To delete a timer, press the "**x**" button.
- To reorder the branches, press the Reorder button. This displays a dialog box that lets you change the way branches display in the canvas. The order in which the branches display has no effect on the order in which branches execute.

Note: Process Designer cannot interrupt a subprocess invoked in the agent from a process executing in Informatica Cloud. An interruption only occurs if all processes are executing in the agent or in the cloud.

Email Actions

Note: This action is not always available.

If email actions can be performed, you may see a Service Call step that looks something like this:

Step Information

Step Type: Automated Step ⌄

Title: Email Sender

Action: Email Sender ▽

| Information | Input | Input Screen |

Email Sender

✉ This automated step allows you to send email.

Input Fields:

Name	Type
From	Text
Reply To	Text
Recipients	Text
CC Recipients	Text
BCC Recipients	Text
Subject	Text
Body	Rich Text Area
Attachment	Attachment

Output Fields:

Name	Type
Send Email Status	Picklist

Preview   OK   Cancel   Apply

For information on using email, refer to the Sample Orchestration project. After creating the project, review the objects in the `Email and URL Links Demo.gt.xml` guide as well as `EmailSender.bpel` file.

Create Step

Use the Create step to add new object instances. For example, if you work with an Account object, you can use the Create step to add a new account.

The following image shows a Create step for the Account object:

To create new objects:

1. Choose the **Input Fields** tab and select all the fields that you will be adding to the object. Some are required; others are optional.

2. For each field name, set its source to one of the following values: **Content**, **Field**, or **Formula**.

Using Reference Fields

Note that:

*   If you do not include a reference field on the Input Fields tab for the object you are creating, Process Designer uses the value of the current object at runtime.

*   Reference fields for the *Applies To* object are set if they are not set explicitly in the Input tab.

*   When the content of a reference field is set to empty, it is not included in the create statement. (This prevents problems that might occur when explicitly setting a field to empty or null could violate data constraints.)

Output Checkpoint Action

Use the Output Checkpoint action to write a checkpoint. This action only exists when your application is using a WS-Human Task BPEL process. Normally, the changes a guide makes to objects are not visible until the guide concludes. Executing an Output Checkpoint Service Call step makes these changes available to any other process that has access to the information.

Reassign Action

Use the Reassign action to reassign the current object to another user. Typically, only one field, a user name, need be used in this kind of step. However, your application may require more.

Set Fields Action

Note: This action is deprecated. Use an Assignment step instead.

The **Set Fields** action can set the value of any field that is available to the guide. The primary fields (those shown first in the **Add Input** picklist) are those from the **Applies To** object's fields. The picklist may also include:

- Types of objects related to this one
- Field's defined in the *Guide Fields* tab within the **Guide Properties** dialog
- A Service Call step output fields (these are defined in the application)
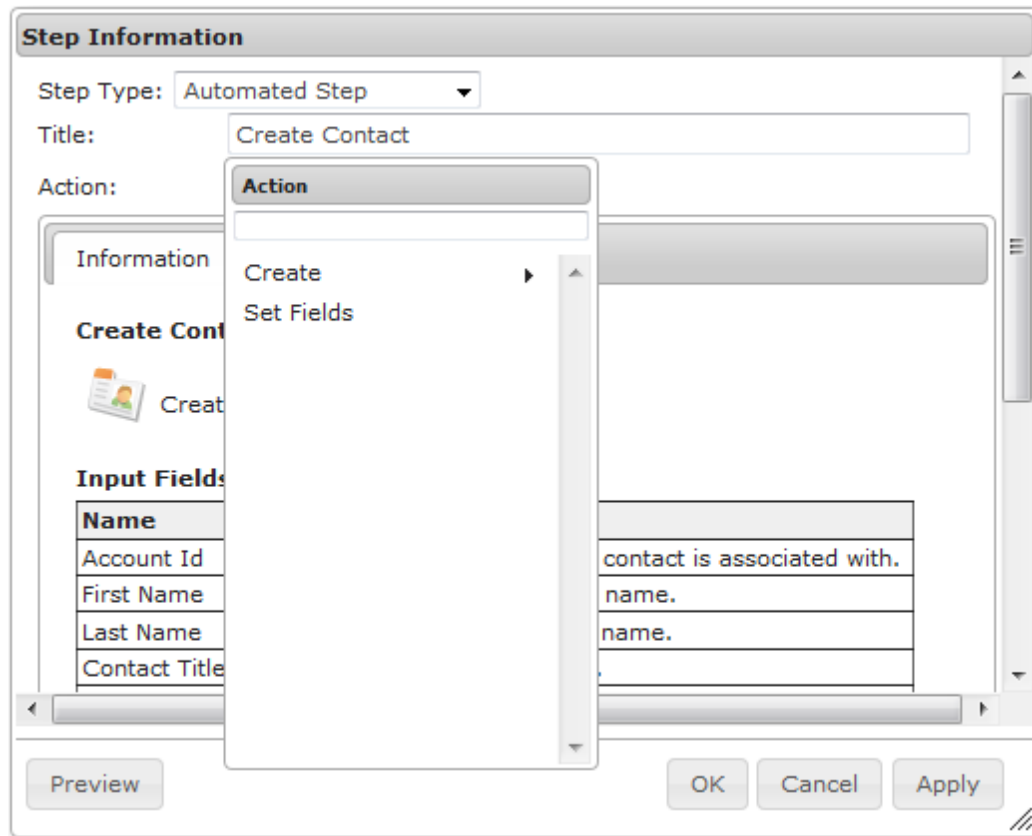
Data Decision Steps

A decision step allows a guide to take different paths depending on the value of some field. Here's an example that shows a Data Decision step having three choices:

Step Information

Step Type: Data Decision ▼

Title: Annual Revenue

Field: Annual Revenue ▽   ⋈ Reorder

Paths:

Path 1: Less or equal ▼ 100000 ⊗

Path 2: Greater or equal ▼ 1000000 ⊗

⊕ Add

Path 3: Otherwise

The first two paths are tests that the step performs acting upon the value of the *Annual Revenue* field. These tests compare the value against the data in the step. If a condition is met, the execution path proceeds on the associated branch.

For each data decision path you add, a branch is added to the canvas. For example, this shows the two conditions and the "Otherwise" path:
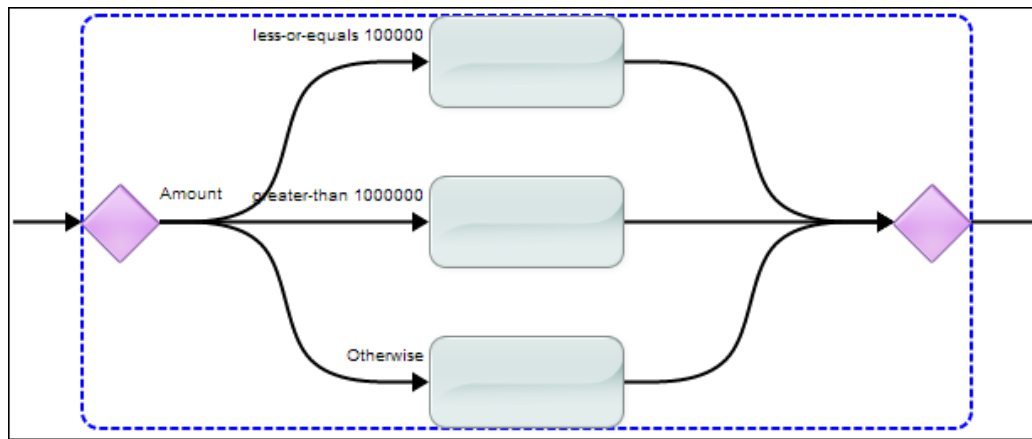
Each branch can have as many steps as you need. After you select the step type, a new empty step is added. Simply delete any unused steps.

You can jump to any of the steps within a Data Decision from another step.

You can rearrange the order in which a path displays in the canvas by dragging it to a new position.

A Data Decision step always has an Otherwise path. This path handles execution if none of the data meets the conditions in your explicit tests.

Evaluating Paths

Conditions in a data decision step are evaluated in a straightforward manner based on the criteria you specify

While there is no limit to the number of paths but you should use as few as possible to minimize complexity.

A Data Decision step can lead to another decision step. For example, the top branch in the next figure executes if the annual revenue is less than or equal to 100,000. The second decision tests if the city is Boston. If the value is less than or equal to 100,000 and the city is not Boston, execution goes the untitled step.
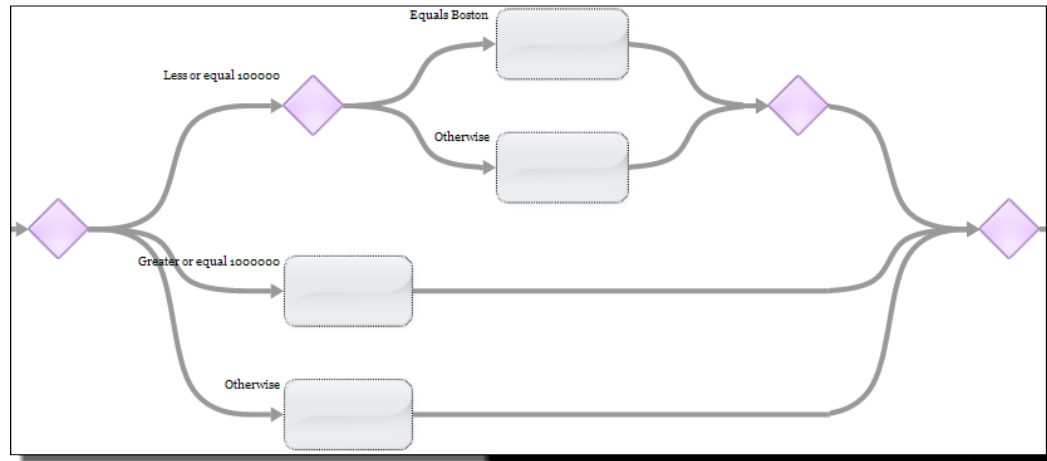
Using this technique, you can support Boolean AND logic by putting the test for the second condition on the true branch of the first condition. The condition to be met in the top branch of the previous figure is "Annual Revenue is less or equal to 100,000 AND the city is Boston".

Similarly, you can support Boolean OR logic by adding a test for the second condition on any of the branches.

Remember that the order in which you declare tests is important. For example, you might have a test for a value less than 100 and another that looks for a value less than 10. If the value is 8, both tests would be true. However, if the less than 100 test executes first, the second condition would never be evaluated.

Data Decision Tests

The tests that you can apply to numeric values are:

- Equals
- Not equal to
- Less than
- Less or equal
- Greater than
- Greater or equal
- Is set
- Is unset

These selections have their usual meaning. However, "is set" and "is unset" can also mean "exists" and "does not exist".

For text, your choices are:

- **Custom Answers: Contains**: Enter one or more strings that test true if they are in the field. If there is more than one string that can be tested for an answer, separate each with a comma. This means that the text you are looking for cannot contain a comma. For example, "Boston, MA" is true if the text contains either "Boston" or "MA". Notice that "Springfield, MA" also matches as "MA" is one of the match conditions. If the field's value is restricted (for example, the answers are predefined), those choices appear in a list as shown here:

- **Custom Answers: equals**: The text in the test completely matches the text in the field.

Compare this with the "Contains" choices. Here, you can choose only one item from a list so each path has its own branch. In contrast, you can choose more than one item if you choose "Contains". (Select multiple items by clicking on the first item, then using Ctrl-click for all others.)

In this example, the user's choices are constrained.

- **Custom Answers: Starts with**: The items all begin with the characters you enter.
- **Custom Answers: Ends with**: The items all end with the characters you enter.

- **Custom Answer: Starts with any of**: Type a comma-separated list of what the step should match. For example, if you select a first name field, you might enter "Bar, Bil". This would match Barry, Barbara, or Bill. The text you type here is case sensitive; that is, if you type "bar", no match occurs if the name is "Barney".
- **Answer per choice**: If the data could be displayed as a list or all possible values for the field are defined in the object (for example, predefined lead sources), choose this option to create one path for each possible choice.

```
Step Information

Step Type:  Data Decision Step  ▼

Field:       Lead Source  ▽

Answers:  ◉ Answer per choice    ◯ Custom Answers

Answer 1:   Advertisement
Answer 2:   Employee Referral
Answer 3:   External Referral
Answer 4:   Partner
Answer 5:   Public Relations
Answer 6:   Seminar - Internal
Answer 7:   Seminar - Partner
Answer 8:   Trade Show
Answer 9:   Web
Answer 10:  Word of mouth
Answer 11:  Other
Answer 12:  otherwise
```

After confirming your settings, a new branch is added for each answer. Continuing with this example, it would create 12 branches.

When the guide executes, users see a step that contains a button for each branch.

If the field that you choose is a multi-select list, the available options display so you can select one or more.

Reordering Tests

To change the order of your paths, select **Reorder**. You can now use the up and down arrows to reorder the paths.

Sometimes, the order in which you test is important and it might not be immediately obvious. For example, assume that address contains the city and state. A test such as "address contains any of Boston" is more specific than "address contains any of MA" and should be the first path.

A second reason might be that you would like the branches to appear in some order on the screen. For example, you may want the most used branch to always be on top.

Embedded Guide Step

Guides need not be self-contained. Most guides, actually, embed externally created sets of actions, as follows:

- **Embedded guides**. Another separate guide is placed within the current guide.
  An Embedded Guide step places one guide within another guide, and that guide runs when a user executes that step. This means that the steps defined in one guide (the embedded guide) are executed within another (the embedding guide). Embedding one guide in another is one of the more powerful and useful features within Process Designer.

- **Process Calls**. The actions defined in a process or subprocess are placed within the current guide. See the help on the Process Call or Subprocess steps for more information.

- **Service Calls**. The actions defined in a service are placed within the current guide. See the help on the Service Call step for more information..

End/Milestone Step

The End/Milestone step marks the end of a guide's execution or a milestone. A milestone generally indicates the end of one phase in a guide's activities. You select the function to be performed by selecting the **Ending Type**, which is either *End of Guide* or *Milestone: End of Current User's Work*.

Note: You can make the text that you enter more readable if you insert text and fields into a table.

End Milestone End of Guide

An **End of Guide** step marks the place where a guide finishes executing. Every branch in a guide concludes with an **End of Guide** step (unless it is a Jump Step). An **End of Guide** step can (and usually does) specify an outcome. For example, here is a filled-in *Possible Outcomes* tab within a **Guide Properties** dialog.
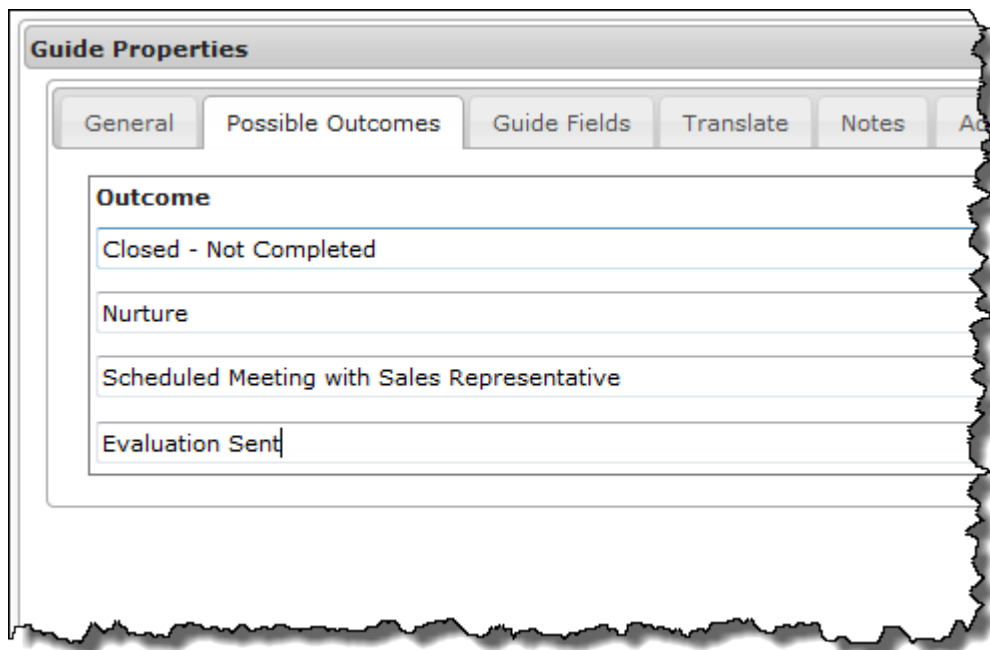
This declares that when this guide concludes, it should have one these four outcomes. (More can be added within this step, as is described later in this topic.)

There can be and usually are more branches than **End of Guide** steps. This occurs when more than one branch has the same outcome. Here is a part of a guide where two end steps have one conclusion, two have another, and just one has a unique outcome:
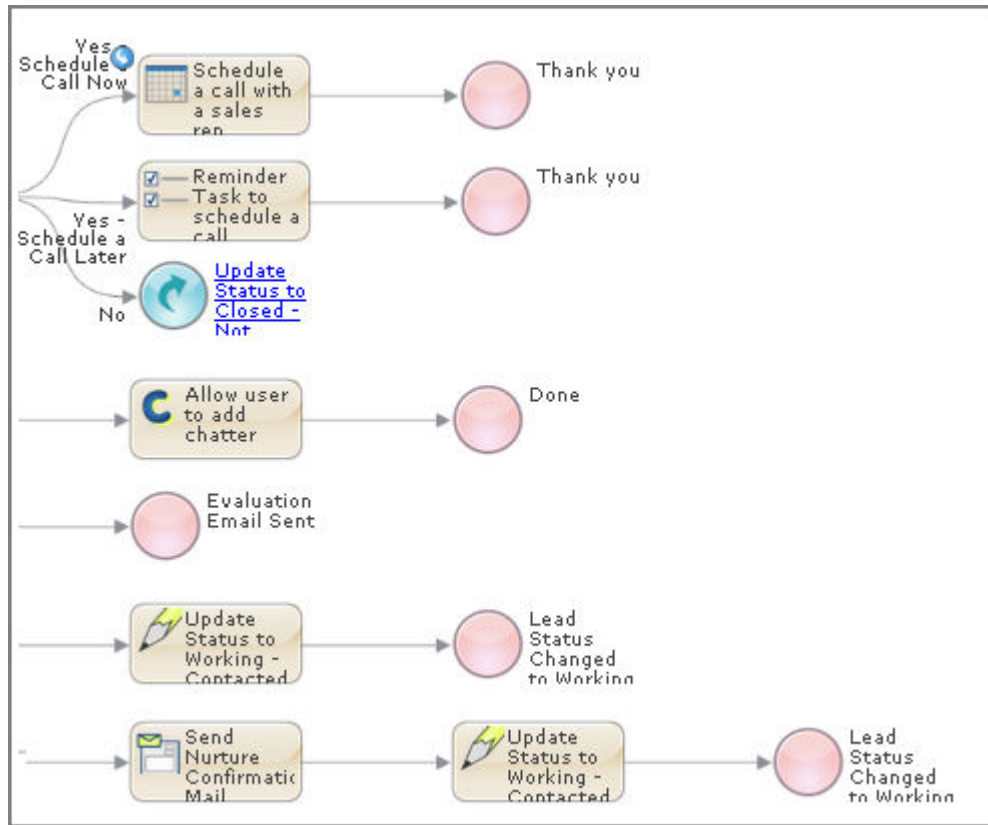
Use the **Outcome** picklist to set the branch's outcome. Here is an example:
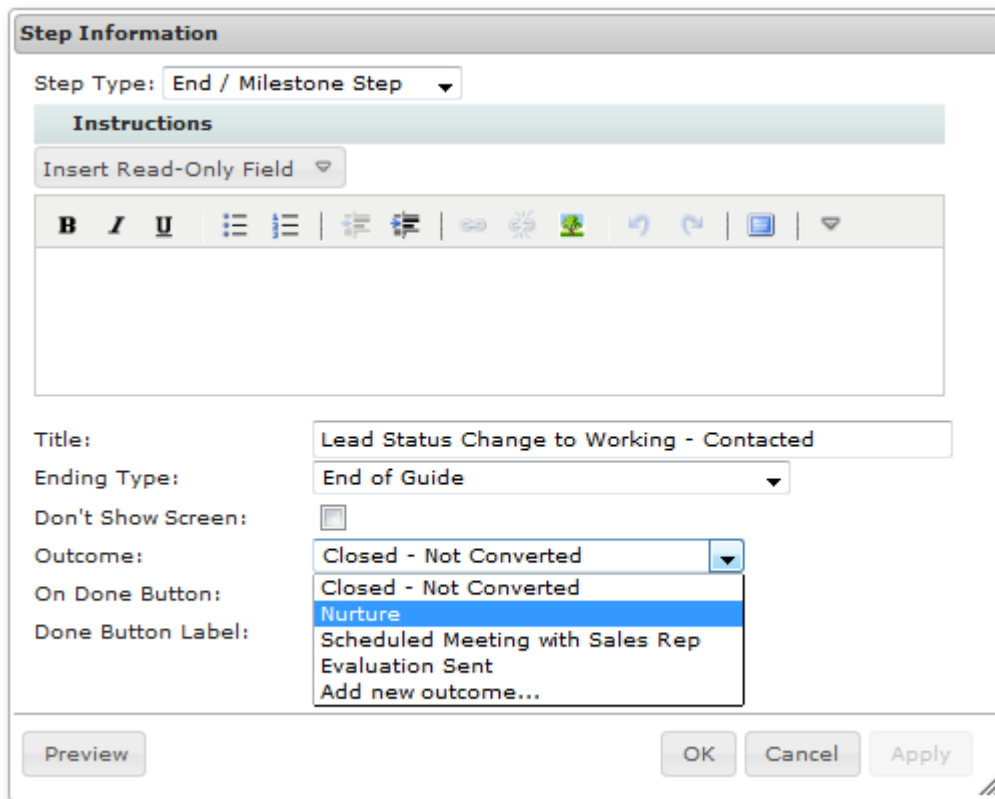
Here, *Nurture* is selected. The **Outcome** picklist contains all of the outcomes you declared within the **Guide Properties** dialog. If the outcome has not yet been added, you can add it by clicking on *Add new outcome* (at the bottom of the picklist). After you create a new outcome, the Guide Designer shows it on remaining outcome picklists and also adds it to the Outcomes tab in the **Guide Properties** dialog.

When the executing guide reaches an End of Guide step, Guide Designer performs the action you set using the **On Done Button** picklist.
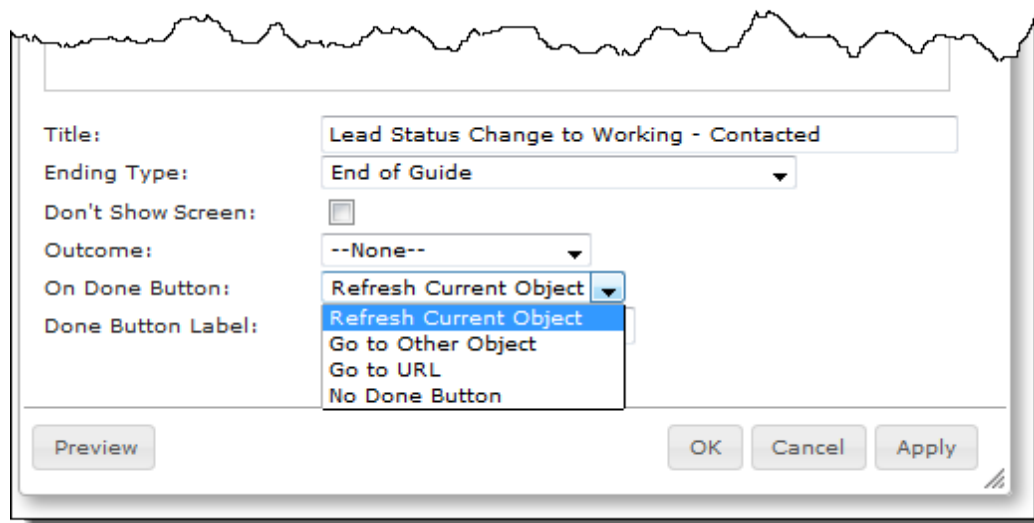
If you want the final action of the guide to be to take the user to another object or to a URL instead of displaying this step, you can bypass it being displayed by checking *Don't Show Screen*. This means that the

last step the user sees is one where there is an interaction. If previous steps are either Service Call or Process Call steps, for example, the user's interaction will appear to end after that step.

Note: A browser's popup blocker must be shut off before this feature is used. On mobile guides, "Refresh Current Object" is ignored.

End Milestone Milestone

A **Milestone** step is a place that has significance. For example, the guide is handed off to another user. In most cases, users running the guide see a screen having a **Done** button. Clicking the button can perform the same actions as those performed when the user presses a **Done** button in an **End of Guide** step (as well as a few additional ones) but the guide isn't really done. When the guide is next run (possibly by someone else other than the original user), it continue from the step that follows the milestone step.

Note: In some cases, you may simply be using a milestone stop to record an event that occured and the guide will automatically continue to the next step.

Here is what a Milestone step looks like:

Milestones are especially useful when a user hands off work to someone else. (If the guide is handed off, the user who will be handing the guide off should let the next user know that something needs to be done. This may be built into your application's workflow rules.) When that user starts the guide, it starts at the step following the milestone. Milestones should also be used if the user normally stops executing the guide at a step to do something else.

After a guide hits a milestone, further activities do not need to occur. However, the guide will be terminated if no further actions occur within 14 days. Non-milestone guides are terminated in 7 days. Once action begins after a milestone is reached, the guide will be terminated if it does not either end or reach another milestone in 7 days.

The fields within this dialog are:

- **Title**: Enter the text that will appear as this step's title.

- **Ending Type**: Choose the kind of activity that will be performed when this step executes, either *End of Guide* or *Milestone: End of Current User's Work*.

- **Don't Show Screen**: If this box is checked, the guide's user will not see this milestone step. Instead, the action indicated in the *On Done Button* area immediately occurs after this step executes.

- **Allow user to go back**: (Milestone only) If this box is checked, the user can go back to steps executed before the milestone. In either case, all steps that occurred within the guide display in the guide's history. However, if the user cannot go back, the user cannot select one of these previously executed steps.

- **On Done Button**: The options in this picklist let you select the action that will occur after this step completes. Two of these actions are unique to milestone steps.

  - **Refresh Current Object**: The page from which you launched the guide is refreshed so that the action performed by the guide is visible on the object's page.

  - **Go to Other Object**: When selected, the screen editor adds a **Click To Select Field** picklist to the right. Select one of the items in it. Many are IDs associated with the current object. This is especially valuable for a guide whose primary purpose is to create a new object (for example, a lead). You might then have the **End Step** specify that the user should be taken to the object after the guide finishes executing.

  - **Go to URL**: When selected, the screen editor adds a **Click To Select Field** picklist to the right. Select a field that is of type URL. When users click **Done** after completing the guide, this URL displays in their browsers.

  - **No Done Button**: When the last step within a branch is reached, it often has a button indicating that there are no more steps for the user to perform. Pressing the **Done** button indicates that one of the activities named in this picklist is performed. This option indicates, as it says, that a **Done** button is not displayed.

    Note: When simulating a guide, a **Done** button still displays. This lets you go to the step following the milestone. Without it, a simulation would always end at a milestone.

  - **Refresh Current Object and Continue**: (Milestone only) After the object upon which this guide is executing is refreshed, execution moves to the step that follows.

  - **Continue**: (Milestone only) notes that the milestone was reached and execution continues with the next step.

- **Done Button Label**: By default, the text for the **Done** button is *Done*. Use this field to change it to something else.

Jump Step

When creating a guide or process, one set of choices might require some activities and a branch would occur. After these activities are completed, the flow of steps should rejoin the actions of another branch.

For example, a guide that leads a sales rep through a conversation with a prospect could take many different branches as the rep talks about features and overcomes objectives. However, at some point in some of these branches, the rep should schedule a meeting. The scheduling step and the steps that follow may be same. Rather than add a scheduling step to each branch that needs one, you could instead jump to a scheduling step that has the scheduling steps.

A **Jump Step** is exactly what it sounds like. In the following illustration, the "Were they interested?" step has two branches, "Yes" and "No". This "Yes" branch leads to a **Jump Step** that tells the process to move to and execute the "Schedule a Meeting?" step.

Unlike embedded steps or service call steps, a jump step changes the execution path.
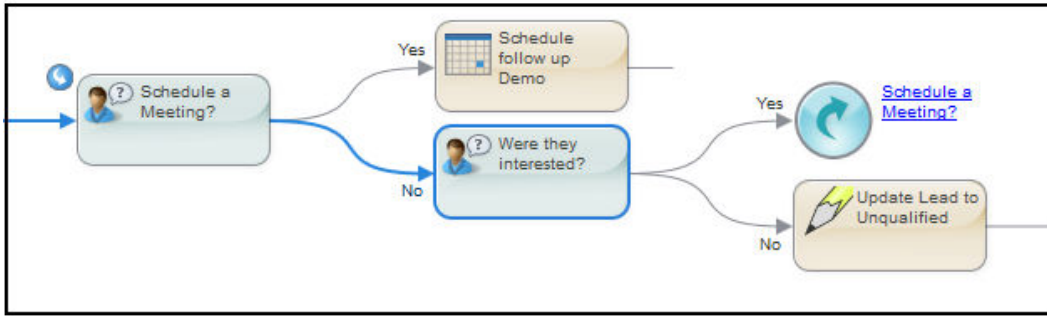
Note: Jumps within a Parallel Path have the limitation that the jump can only occur within the same branch; that is, you cannot jump out of the Parallel Path nor can you jump to another branch. You can jump from one branch to another within a Decision step.

A Jump step is displayed as a circle with a rounded arrow in it. To the right of the circle is the name of the step to which the step will jump. This name is also a link. When you click on it, the step that is the target of the jump is displayed and selected.

The target of the jump has a small rounded arrow at its upper left corner. In the previous figure, notice the arrow points into the step for the target of the jump while the arrow of the jump step points away from it. This little arrow lets you know that the step is also executed from another step in addition to being executed normally within the branch.

If more than one step jumps to the same target step, you can hover over the small arrow to see how many jump steps have this step as their target. If you click on the *jumped to* arrow, the editor displays a dialog showing all of the steps jumping to it.
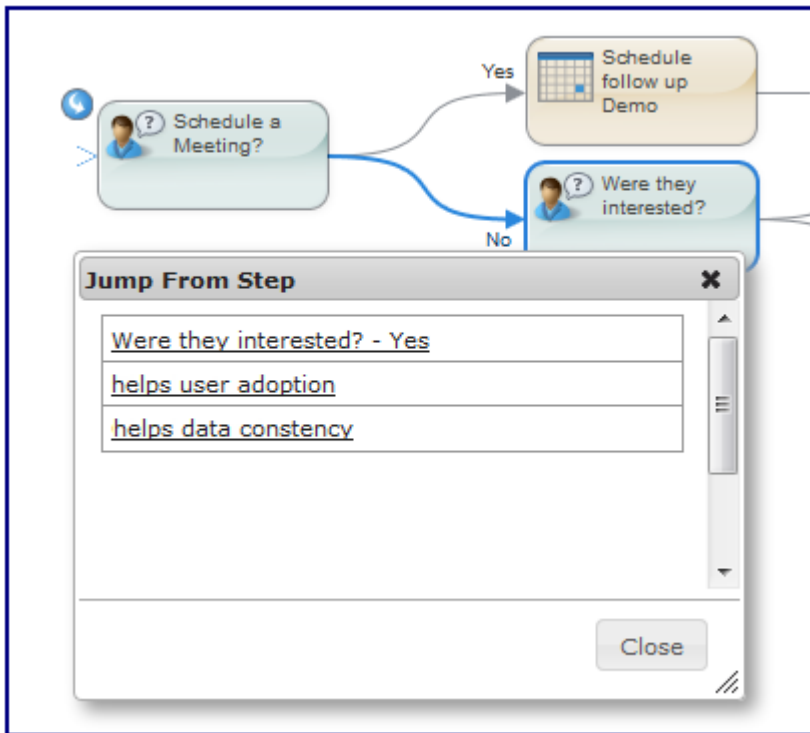
To create a Jump step, simply choose the Jump Step type and, from the To list, select the step to which you want to change the execution path.
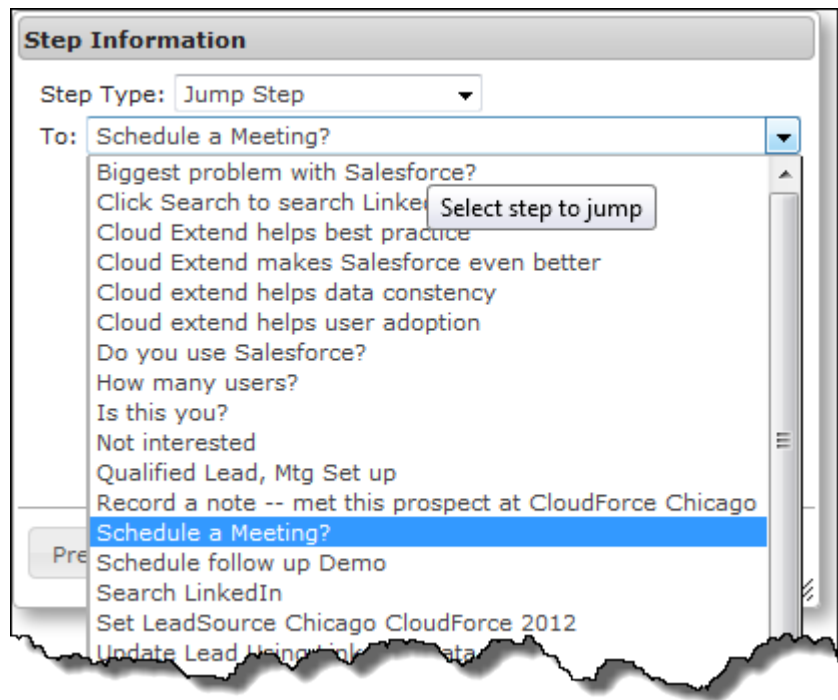
Screen Step

A Screen step has two primary functions:

- Displays instructions telling the user about what is being displayed, what to enter, and so on.
- Updates an object's data based on user input.

Here is a step that presents what a sales rep would say to a prospect:
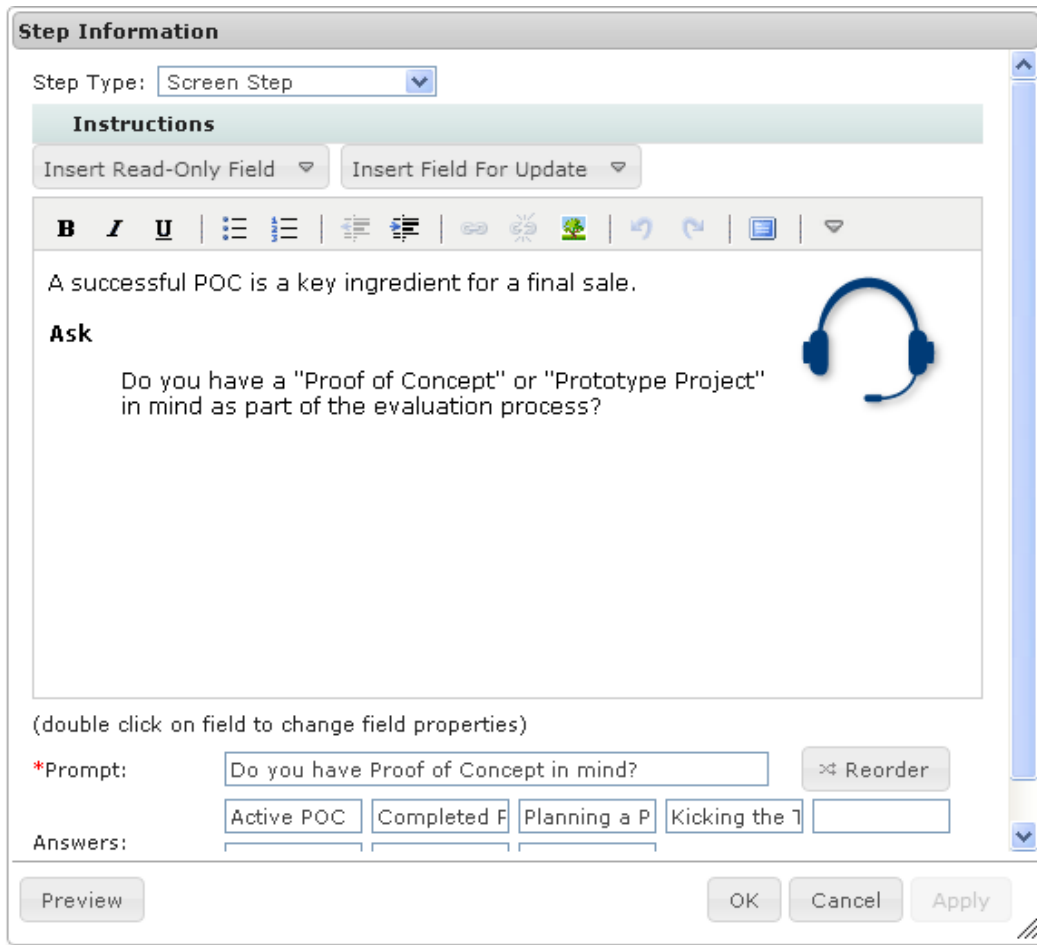
This screen shows the sales rep a question that should be asked. At the bottom of the screen, the rep sees the text entered in the **Prompt** field and a button for each of four possible answers:
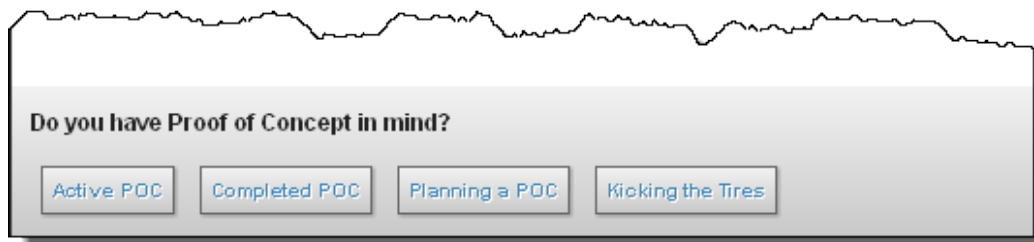
After creating a Screen step, Process Designer creates a branch for each answer. In the following image, the guide has three answers. Initially, the branches converge at the diamond:
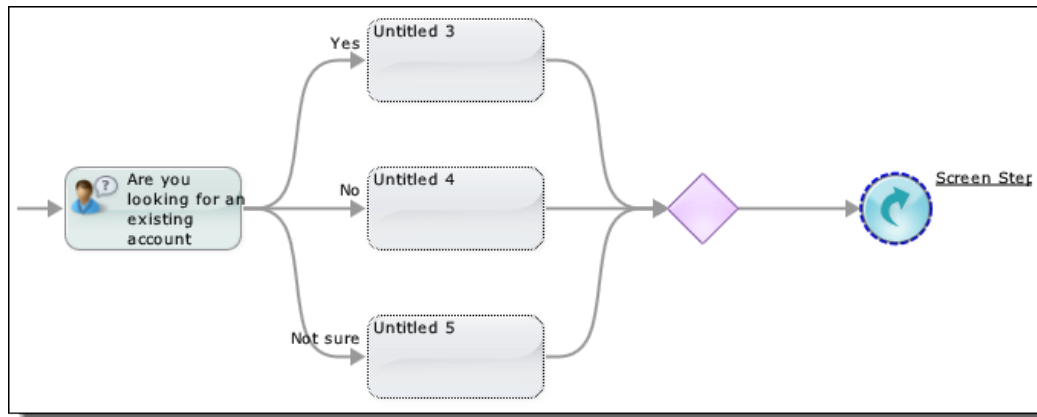
More typical guides take each branch and add steps to it. In the following image, two steps flow into an End step and one proceeds to a Jump step:
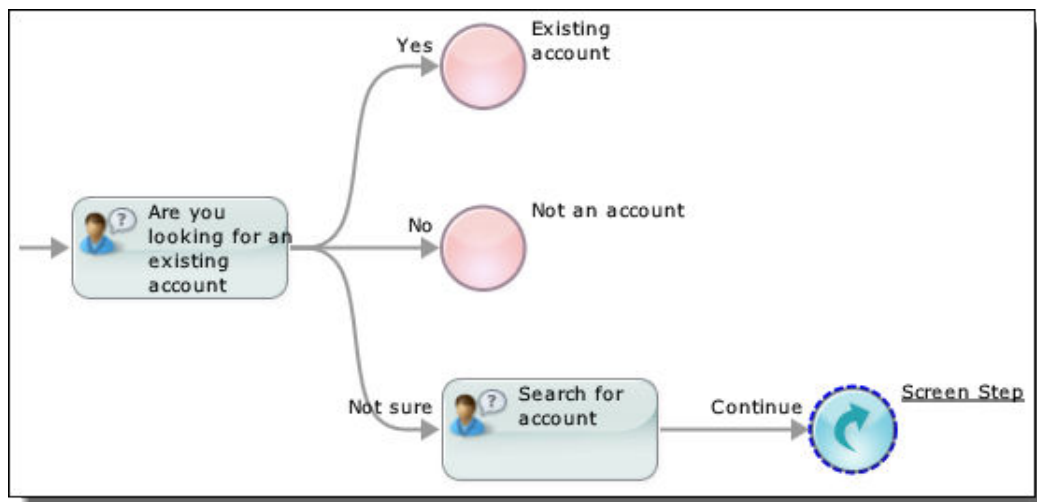
The following figure shows a Screen step with some data added to its canvas. Two lists appear in the editor: **Insert Read-Only Field** and **Insert Field for Update**. Each contains fields that Process Designer can display when it executes the step. An updateable field shows the current value of a field on the screen. However, users can modify this value. When users modify an updateable field, the new value is stored into your application as soon as they click on any of the **Answer** buttons. Here's an example containing seven read-only fields:

Step Information

Step Type: Screen Step

Instructions

Insert Read-Only Field ▽   Insert Field For Update ▽

B *I* U | ☰ ☷ | ⇥ ⇥ | ⚭ ⚮ 🖼 | ↩ ↪ | ▣ |

Picture
First Name   Last Name
Title
Company

Result This Result   out of Num Results

(double click on field to change field properties)

*Prompt: Is this the Contact you were looking for?   ⋈ Reorder

Answers: Yes   No   Next match

Preview   OK   Cancel   Apply
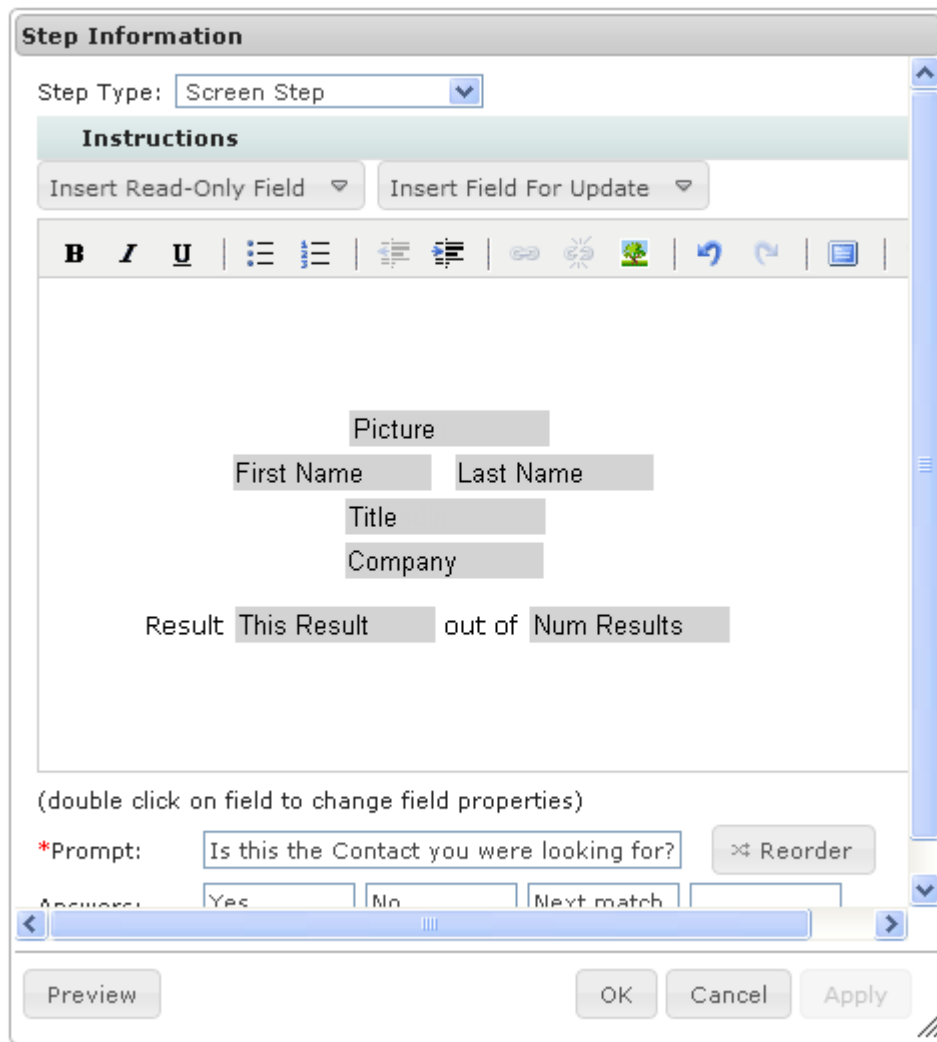
Updateable fields can be displayed as a box into which users type text.

You can click within a field's text box to display and edit the field properties.

You can use the **Reorder** button at the bottom of the dialog to change the order in which the screen editor displays branches. Sometimes, rearranging the order can make the step's action clearer.

Handling Data

Inserting Fields Using Picklists

There are many places where you add or use field names.
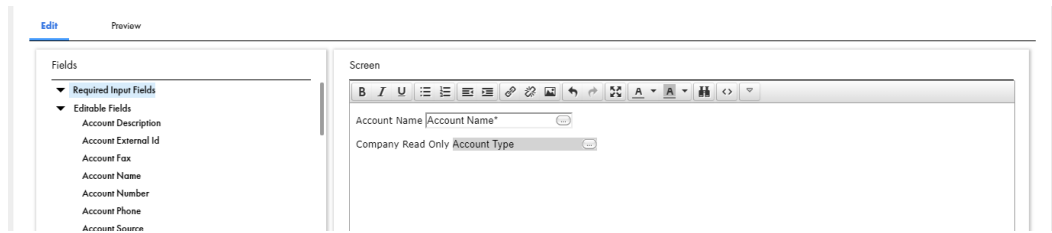
When a user runs a guide, Process Designer replaces the field with the value within the object. For example, if an Account Type field was inserted and the Account Type is "Grade 1", users will see "Grade 1" instead of the field name when they are running the guide.

Three kinds of lists exist in steps:

- **Read-Only Field**: A read-only field is a field whose value cannot be changed. You can tell that it is read-only field because it displays in gray.
  The text in front of the field was added as descriptive text in front of the field. If you didn't add a description, the user might not know what information was being displayed.

  Note: When an object ID is placed in a canvas as a read-only field or put into a column in a table, it displays as a link to the object. If you want to show the object ID's value, insert it into a text field as `{! Id}`.

- **Editable Field**: An editable field lets the user see the current value of the field as well as modify its value.

-

Click the "**...**" button to see a field properties dialog. For more information, see [" Introduction to Data Types and Field Properties" on page 88](#).

Autogenerated Fields

A field can be virtual; that is, the object has a field that is actually a composite key or a composite field. When Guide Designer displays the name of this virtual and autogenerated field, it displays it as `"Object_name"` **+** `"_IID"`.

Entering Fields as Text

You can enter text and the value of a field as the information you type in text boxes such as the one in the following.

To specify a field value, choose Content as the Source and then click the small icon to the right of the text box. From the list of fields that displays, select the field you want to enter. If you know the name of the field, you can also type it in the text box as *{$name_of_field}*.

This example shows the values of four fields from the current object as the Content for the Description field.
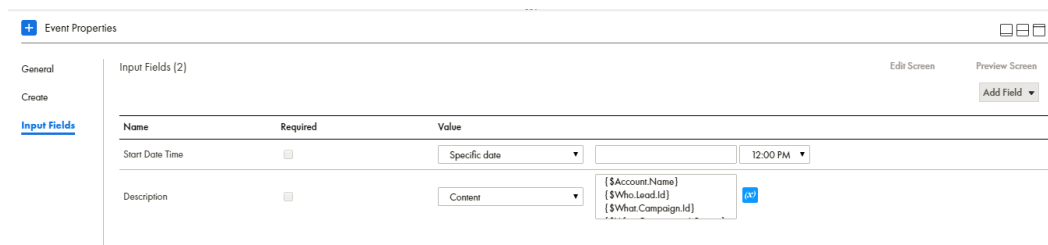
You can add other text to display with the field values. For example, you could specify:

```
The name is: {$name}
```

Related Object Fields

Many of the objects within your application have relationships to other objects. For example, in a CRM application such as Salesforce, cases are related to the account and contact objects. An account is related to opportunities, cases, expense reports, and so on.

Note: You only see related objects and their fields if these relationships exist. If specific objects were created for your application, these relationships may not exist.

When you set the type of object that the guide or process **Applies To**, the fields associated with that object and any related objects are available for your application.

This list shows a Lead object associated with other objects. Those objects are associated with others. In this example, the Converted Account object has a set of Related Objects, including an Owner. The Owner also has related objects:
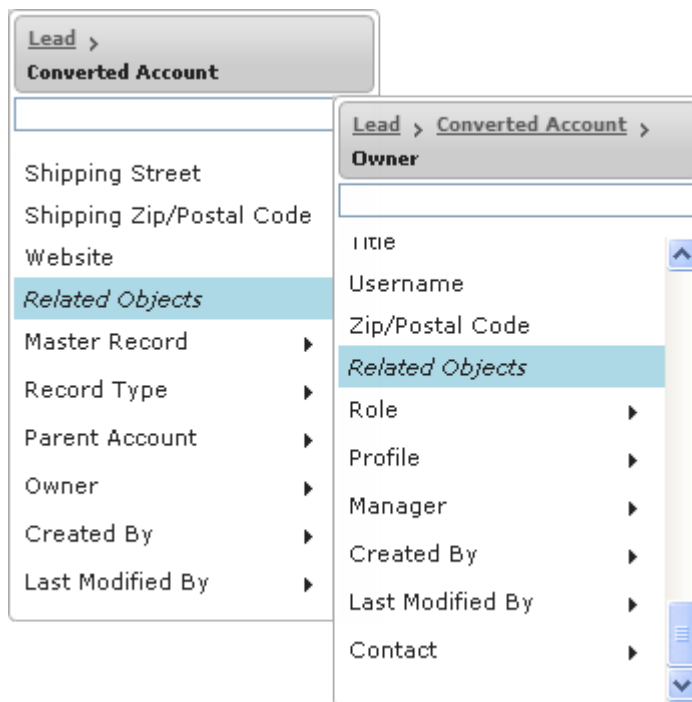
Setting Data in Steps

Some steps that you will create ask users to enter information before the step's actions occur. Examples include Service and Create. In contrast, other steps already have the information they need when they begin executing. If the step has the information it needs, the action occurs automatically and the user doesn't see it. Instead, the user sees the step that follows. Although this topic uses a Create step as an example, there is no difference setting data in other kinds of steps

In the step properties section, you can see information about the fields associated with the action or object. If an entity or object is not selected, no data appears in the Information tab.
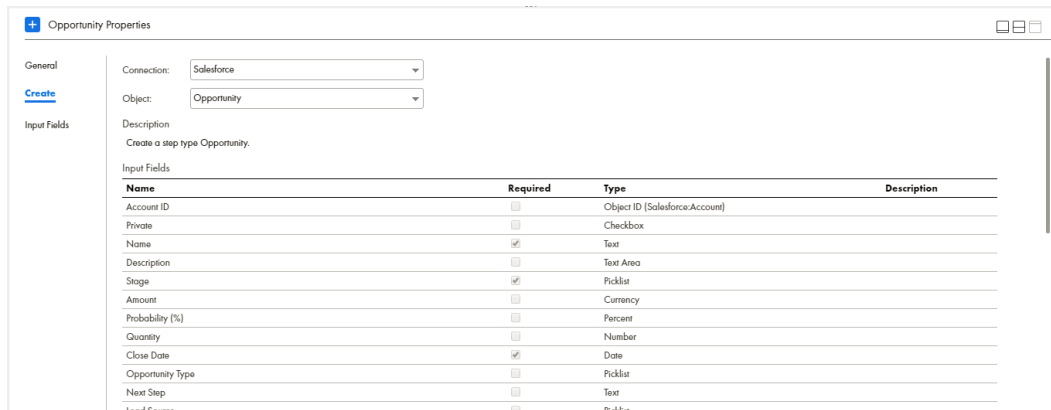
This section lists all of the fields that can supply data when this step creates an opportunity object. The fields that you will actually be using are named and set within the Input tab.

Setting Source Values

You can tell Guide Designer about the source of the values that will be assigned to a field. The following figure shows part of a Create step where fields have different sources:

- **Add**: If the field being entered is of type `ObjectList`, selecting this item will let you append an item to that list.
- **Content**: The values used by the step is what you define here and does not change. For example, the Rating field, which was defined as having one of three values, will be set to one of these three as you are designing this guide. The value you set here is what will be used when this object is created. However, if this were text, the text may also have embedded fields. This example has three different kinds of content fields. The differ because Process Designer knows what kind of data is being set. For example, Converted was defined as a checkbox, Rating as a picklist with predefined values, and Zip/Postal Code is defined as text.
- **Field**: The value comes from a field in an object.
- **Formula**: You will use numeric operators such as:

  - +

  - -

  - *

  - div

  - mod

  You can use built-in and XQuery functions to set a value.
- **Query**: (This is not shown.) You tell Guide Designer which information it should locate. For example, Guide Designer might look for a company whose name is "Acme".

- **Screen:** (Guides only) You must insert an input field within the Input Screen tab's editing area.

Setting Source Values: Content

A Content source is one in which the person designing the guide is deciding the value that is passed as input to a step. For example, you could use Content to set the Description field of a Create step creating an event to "Demonstrate the product" rather than requiring the guide's user to provide the description.

Sometimes, you want to create content that contains text and field values. For example, you could have a "mailing" field that contains a person's name and address. You would add this information by including field strings.

Click the (x) icon to see a list if fields.

After you select a field, Process Designer places a field code into the text box at the cursor's position. You can individually select as many fields as you need; however, you can only select one item at a time.

Here's an example:

You can also add text to this inserted field indicator. For example, In front of `{$Account.name}`, you could type "`Name: `".

When Process Designer executes this step, it replaces each field indicator with the text extracted from the field.

Setting Source Values: Field

If you select Field for the Source, Process Designer displays a collapsed list to the right. You can now select the field whose value Process Designer will write into the object when this step executes. As with other places where you select fields, the bottom of the picklist may show related objects whose fields can be used.

Setting Source Values: Formula

Fields that take numeric input fields can use a formula to set a value. For example, you can use this with the Assignment step to add a number to an existing field. Use numeric operators such as +, -, *, div, and mod to perform an operation upon fields.

You can also use any XQuery function within a formula. (These are defined at http://www.w3.org/TR/xpath-functions-30/.)

Evaluation Notes

Process Designer evaluates formulas as you might expect. Note that:

- `$Field1.Field2.Field3` is pre-processed to de-reference down to the contents of `Field3` and would bind the contents to its value.

- Special characters in field names are removed. Fields that start with numbers are given an "x" prefix.

- If `$Account` is a by-value object, you must use an XPath path expression to get at its contents; for example, `$Account/Owner/Name`.

- Object lists should be sequences containing either ID values (for by-reference object lists) or the XML representation of the records. For example, two object lists can be appended using the following expression: `($a,$b)`.

- Object references within a list are separated by semi-colons; for example, `(a; b; c)`.

Setting Source Values: Screen

If you select Screen for the Source, the end user provides data for this field. This means that you must add a field to the step's Input Screen tab so that the guide's user can type information. In the following example, the source on the Input tab for three fields was set to Screen:
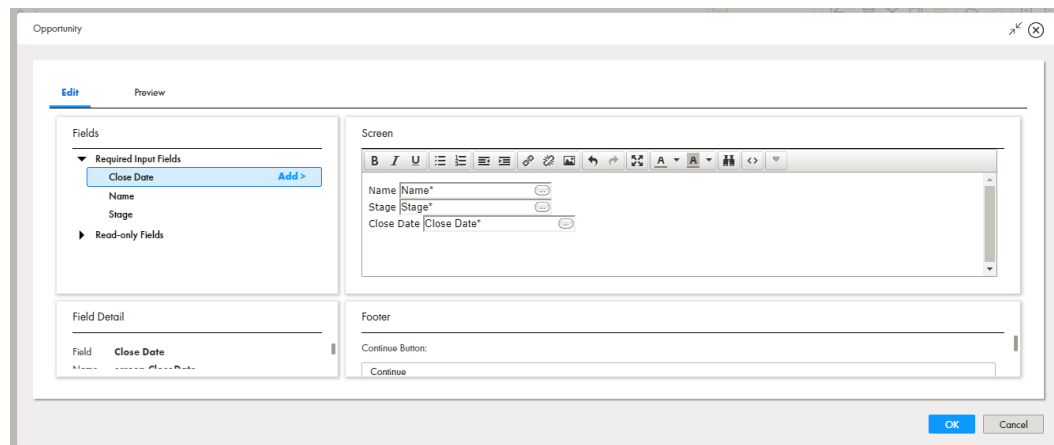
Note: The text you enter will look better if placed in tables.

Like all fields, you can click within it to display a properties dialog.

Embedded Guide Step

To better orchestrate and manage your business processes, you can design guides that embed other guides, similar to how you would create a subprocess that is called from another process.

For example, suppose there are a set of steps associated with entering client information. If there are different ways that your users interact with clients and many need to enter client information, you could group these client information steps into their own guide. Now, whenever a guide needs to enter client information, it just "embeds" the guide. This also means that all embedding guides use the same steps
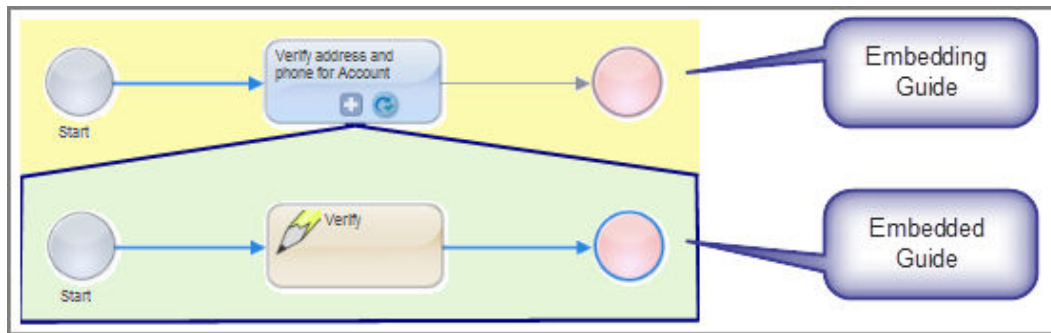
Using an embedded guide has several advantages, including these:

- **Avoids Repetition**: If you place commonly used patterns in a separate guide, you can save time and create other efficiencies if you place the discrete set of steps into a guide that can be embedded and reused in other guides. When you change the embedded guide, you can easily make those changes available to all guides that use it.

- **Clarifies Design**: Embedding can make your design look simpler. Removing steps and placing them into an embedded guide means that the embedding object only shows one step rather than all of the steps. This can simplify the diagram on the canvas. Also, if a problem occurs in the ways steps occur, embedding can make it easier to locate a problem.

- **Clarifies Logic**: If there are many steps that need to operate on other objects, this allows you to create logical chunks of work that each apply to different objects (such as opportunities and leads). This decreases the complexities of working with multiple object types in a single guide.

You handle the data received from a guide and the data that can get sent back to it using the Input and Output fields defined in the guide properties.

Embedding and Embedded Field Relationships

When you publish an embedding guide or process, the embedded object appears within the embedding object, as shown below.

- An embedded guide, like all guides, has a set of outcomes. When you embed a guide, the outcomes from that guide become different branches of the embedding guide. This means that the embedded guide's outcomes are treated as if they were the answers from the previous step. More precisely said, the Guide Designer creates a branch for each of the embedded guide's outcomes.
  For example, suppose the embedded guide has three outcomes: "Yes", "No", and "Maybe". After you embed the guide, the designer adds three branches to the embedding guide, one for each outcome:
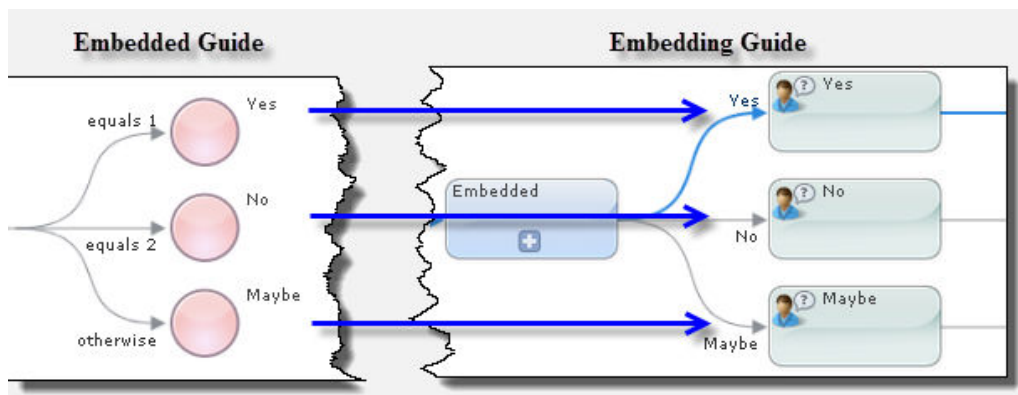
- Many guides receive data from the embedding process and return results, so be sure to define the required fields.

- Notice the relationship of field information in these two dialogs.

   After one process or guide is embedded in the other, the fields within the embedded object are available in the embedding object.
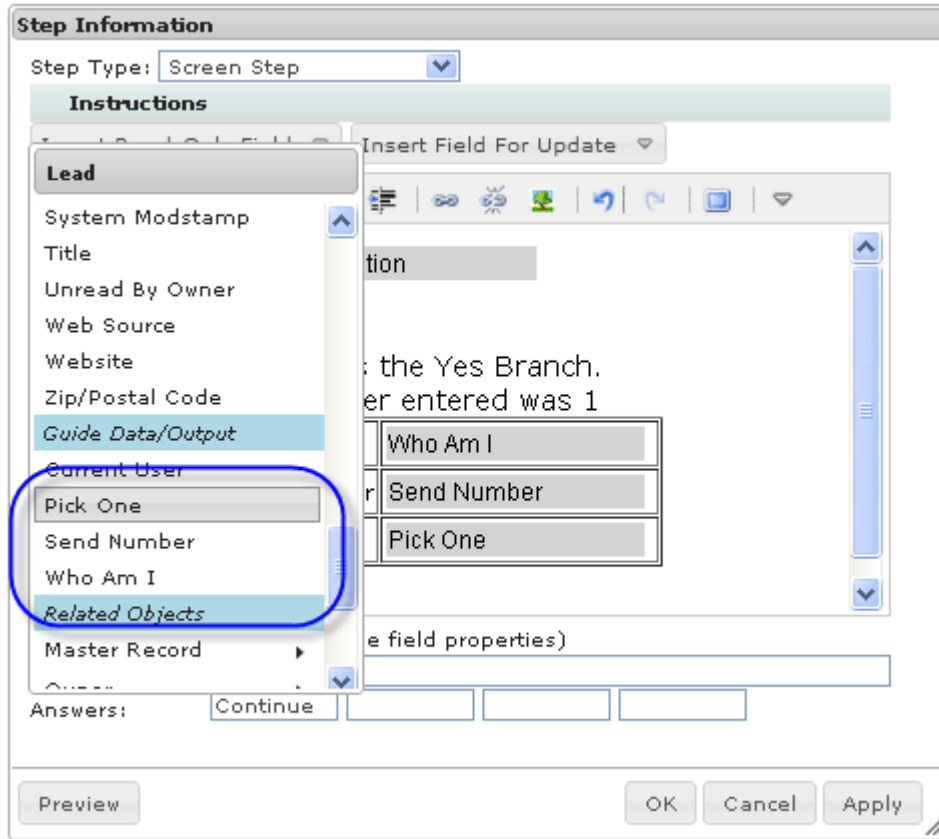
However, these fields only become available in the steps following the embedded step.

Multiple Embedding Guides

Suppose you have three guides whose names are guide 1, guide 2, and guide 3. Also, guide 1 embeds guide 2 and guide 2 embeds guide 3. In this case, guide 1 does not see the fields in guide 3. If you need to access these fields, add output fields to guide 1 that have the same name as the fields in guide 3.

An Embedded Example

The first topic in the following example shows the implementation of a simple guide embedded within the guide discussed in the second topic. The actions performed by these guides are:

1. Create a **Screen Step** that is just a placeholder so that the example does not immediately jump into the embedded guide. (This is done for this example, A guide could jump directly into an embedded guide.)

2. Execute the embedded guide, passing it a number.

3. Within the embedded guide, enter data.

4. A **Data Decision Step** routes the guide to an **End/Milestone Step** based on the passed in number.

5. The embedding guide displays the values entered in Step 3.

Embedded Guide Example

The following illustration shows the design of this example's embedded guide.
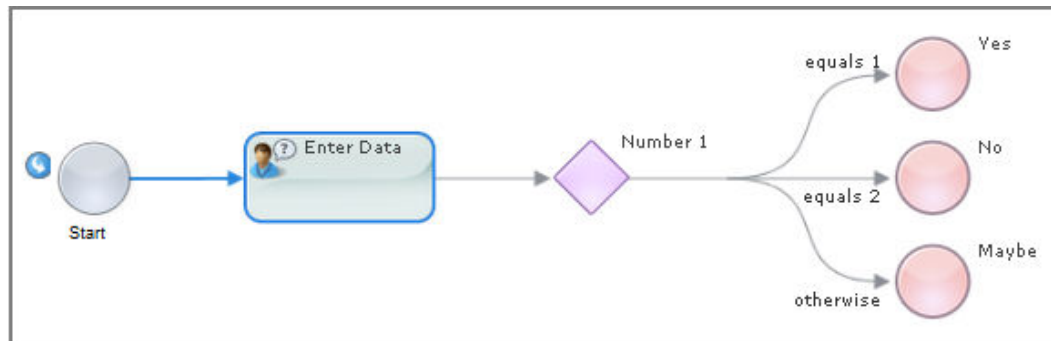
The **Embedded Guide** step within the embedding guide asks the user to enter a number. Based on the value of this number, the guide chooses one of the three branches.

You must define fields for all the values sent to the embedded or embedding guide in the guide properties.

Here, four fields are declared. One is an input field and the others are output fields. These fields are available in all steps within the embedded guide.

Input fields are meant to be read. That is, a guide places data into an input field and this data is read by the embedded guide. If, however, the embedded guide changes the value of an input field, this change is seen by the embedding guide.

An output field does not exist until it is set to a value within an embedded guide. For example, if the fifth step in a guide is an embedded step, output fields are invisible in the first five steps. However, they are visible beginning at the sixth step.

For more information on selections you make in the *Type* column see " Introduction to Data Types and Field Properties" on page 88.

These fields are used in the following Screen Step:
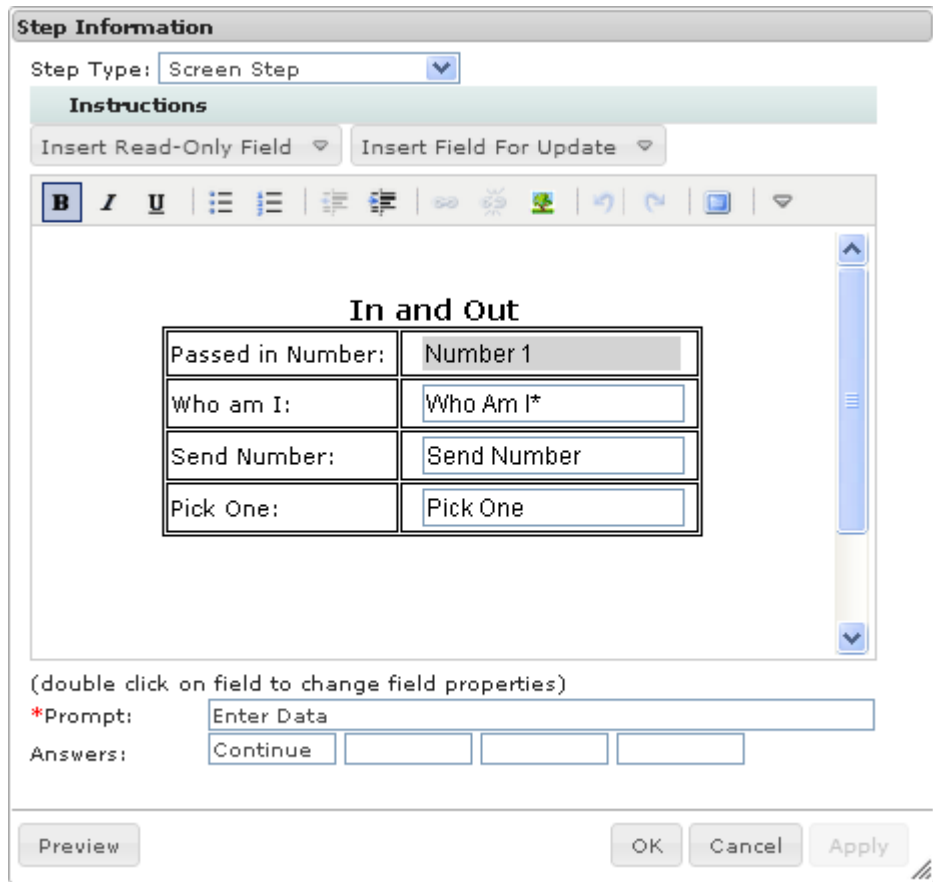
The table contains the fields set in the *Guide Fields* tab. Select the fields to insert in the step from the insert lists.

Here, no object fields display because the **Applies To** field was set to *** Any ***. This means that the embedded guide does not have access to any of the fields within any objects. If this guide needed access to object fields, you will need to select the object in the **Applies To** list or pass them to the guide as input fields.

The step following this data decision step uses the value placed in the **Number 1** field to choose the guide's outcome.

Creating an Embedding Guide

This topic show how to create a small embedding guide; that is, it describes creating a guide into which another guide will be embedded. Here is the definition of this guide:
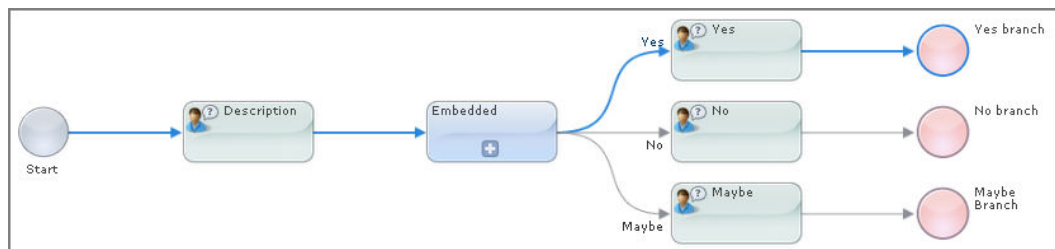
The steps within this guide perform the following actions:

1. Display a step that describes what is about to happen. This step has no content. Instead, it just has text that gives instructions. After users read the instructions, they press a button and move to the Embedded step.

2. The user types a number within the **Embedded** step, the embedded step executes, and then returns.

3. The embedded branch's **End/Milestone Step** dictates the path that this guide takes.

4. The guide ends with its own **End Step/Milestone Step**.

In this example, the embedding guide acquires information from the embedded guide. This means that you will need to know the following about the embedded guide:

- Its input fields
- Its output fields
- Its possible outcomes

The steps displays the following properties:

Look at the sections of this dialog:

- **My Embedded Guide** is the name of this guide. This guide *applies to* Leads, which means this embedded guide has access to Lead information. (Using the controls within the **Applies To** area is discussed later in this topic.)

- The **Input Fields** area shows that you need to add four fields. These four fields appear on all insert fields picklists.
  Note: If the embedding guide passes a field as input, an update to it within this embedded guide also updates the field in the embedding guide once the embedded guide finishes.

- The **Possible Outcomes** area shows you the outcomes that can be assigned to end steps within the embedded guide. Each outcome has its own branch in the embedding step in the same way that each answer within a screen step has its own branch.

- The **Output Fields** area identifies the information that the embedded guide returns to the embedding guide. All steps after the Embedded Guide step (in the embedding guide) will see these fields. Said in a slightly different way, when you access fields in any step before the **Embedded Guide** step, you will not see any output fields. However, they are shown in field picklists and can be used after the embedded step finishes.

In the **Embedded Guide** step, if you select **Screen** as the source of the "Number 1" input field, Process Designer adds a "Number 1" button to the input screen so that you can add it in the Input Screen.
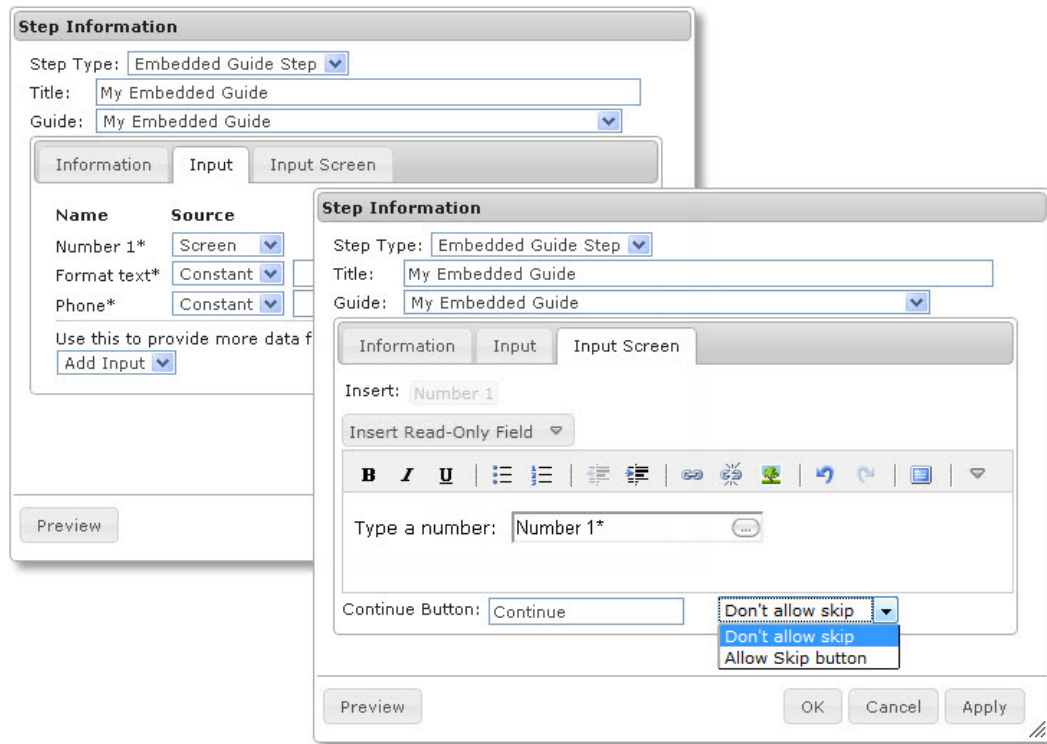
Introduction to Data Types and Field Properties

Process Designer uses information contained within your application database. Each database table contains object types that you can access through your connections to work with the fields needed in a process.

When you define process or step properties, you drill down to view the objects and select associated fields from a list.

For example, in following case, the Salesforce objects give you access to many object types that you can use in the process:
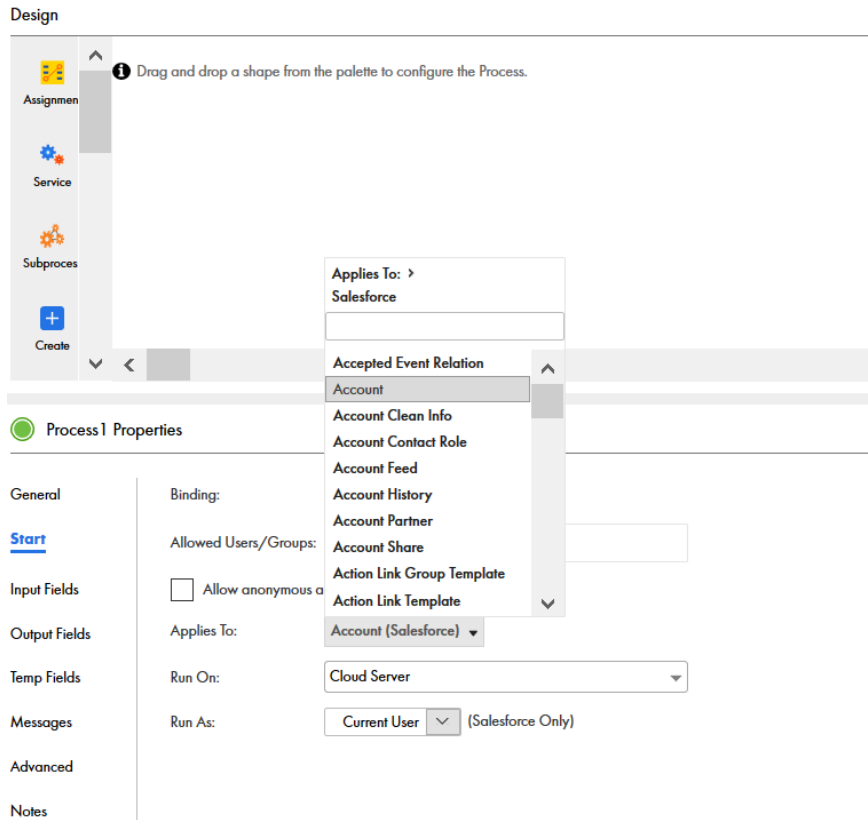
In most cases, a process determines what type of data is being stored based on the application. For other fields, you may need to define them as Simple types and select the appropriate data type. Apart from the data type, you might want to control other field properties. For example, you could specify a default value or hover text to help the user.

To update field properties, simply click in the field to edit.

Read the topics in this section to learn more about the field properties and data types.

Using Fields in Steps

The data types that you name in the property sheet may be numbers, text, time, and so on. These data types determine how the data is handled in a process and displayed in a guide.

The following image shows the Account Type field:

The Default Value and Show List fields appear only when you define the field in a step:

- **Default Value**: Select Content, Field, or Formula.
  - Content displays a control where you can enter a specific value.
  - Field displays a list from which you can select the default value.
  - Formula allows you to set the value using a formula you enter here.

  Note: If you select a default value for a field that is on a screen "for update", the user does not see the default value unless the field's current value was not set before the step displays. If it was set, the user sees this current value. Service Call steps differ as the default value is always shown since there is no existing value.

- **Show List**: For information, see ["Selecting and Displaying Objects" on page 98](#).

Using Fields in Embedded Steps

When a guide is embedded within another guide, the embedded guide can receive data from the first guide and send data back to it. (For more information, see the ["Embedded Guide Step" on page 80](#) topics.) Within the embedded guide, you define this data by selecting the Fields tab. In the Type list, select the data type of this field.
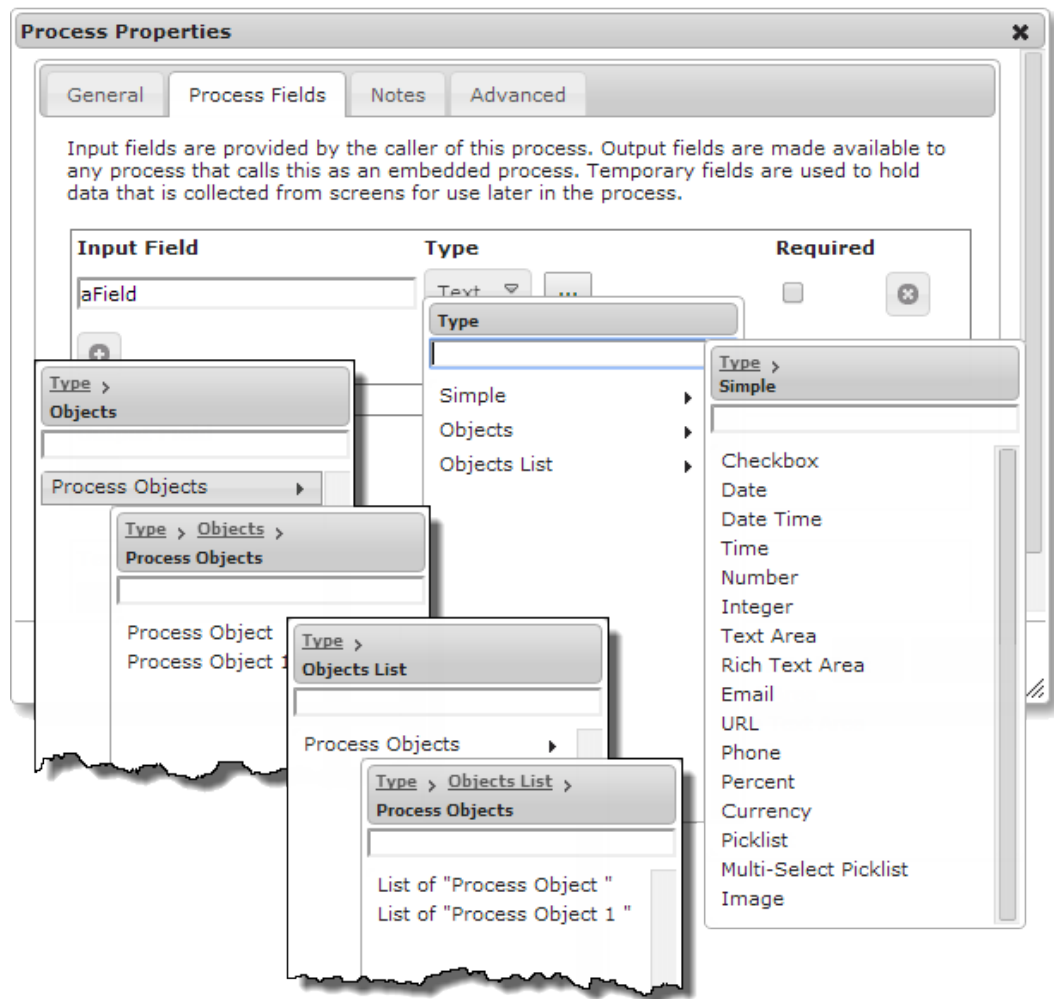
The items in this list are all data types. Each represents the kinds of data that a guide can use. Most of the type, all you need do is select the type. If you need to be more precise, you can fill in additional information using a **Field Properties** dialog, which is displayed by clicking the "**...**" to the right of the item set in the Type column.

Note: If the data type is an object or an object list, you must use the **Field Properties** dialog to specify additional information.

To the left of the Type picklist is the name that you are assigning to the information passed to the guide from the embedding guide. For definitions for each of these items, see "Types of Data and Field Properties" on page 92.

If the value of an *Input field* is changed in the embedded guide, it's value is also changed in the embedding guide. This is because the scope of the input field is both the embedding and embedded guide.

Notice how there are three kinds of data types. The first (simple) are built-in. The other two are ones that you add through services and process objects that you've added.

Types of Data and Field Properties

You can specify Field Properties, derived from API data types and options, in the Informatica Process Designer. Field Properties may be either of the following:

- General data type properties. For example, specified on the **Fields** tab of the Process Properties.
- A field that you can tailor for the context. For example, a pick list on a Screen step.

The following base options are available for many data types in the Field Properties:

- **Default Value** or **Initial Value**: The value for an item when it is initially displayed.
- **Field**: The name of the field from which the **Field Properties** dialog is invoked.
- **Hover text**: Text that a user sees when they place their cursor over the field.
- **Required**: If checked, a value for this field must exist.
- **Show List**: Values to display are determined at runtime based on what is configured here.

The following table explains built in data types:

Note: Depending on where you access the Field Properties, the available data type options may vary.

Data Type	Description

Any	Use the Any data type if you want to declare fields that are process objects without setting the Reference To option. This means that the data type is an object, but you do not need to specify which object it is. If you use the Any object, you cannot drill further into the object.
	
	Some services return large amounts of data, but you may need only a small percentage of it. Use the Any data type to model all the data returned in to just a few fields.
	
	For example, when you use a REST proxy, it passes a received payload as input and then responds to this payload. The REST proxy would do some processing on the request or response, and it would not have to understand the whole message.
	
	A second example would be supporting heterogeneous lists. While simple REST APIs tend not to include heterogeneous lists, SOAP-based APIs do. For example, a SOAP API might have a "query" operation that returns data that depends on the query itself. Here, the static type of the output for the "query" operation would be an objectlist of the any type.
	
	A third example is calling a service that returns records that match a query. Each record can have a unique set of fields, so those could not be known at design time. At runtime, each field would just be turned into a child element of the record element, using a tagname that corresponds to the field name. The converted JSON returned as output might be:
	
	`{`
	`  "record": [`
	`      {"ROWID_OBJECT": "2001", "CREATOR": "admin", ...},`
	`      {"ROWID_OBJECT": "2002", "CREATOR": "admin", ...},`
	`      ...`
	`    ]`
	`}`

Attachment, Attachments	Use Attachment and Attachments to a process to pass through an attachment and extract data like file size and file name using functions on the response. The default Maximum File Size is 5,242,880 bytes and the default Maximum Number of Files is 10. You can change these field properties. **Note:** If you run a process on the cloud server, do not use an attachment whose size is more than the default 5,242,880 bytes. The cloud server cannot process attachments that are greater than 5,242,880 bytes.

Data Type	Description
Checkbox	A checkbox allows the user to make a true/false decision. This decision can alternatively be displayed as Yes/No or No/Yes For example, you could use a checkbox to indicate if an address is a work phone number or a home phone number. As an alternative, a checkbox can be displayed as yes/no values. - **Show as**: a picklist that has the following items: Yes/No and No/Yes. Your choice sets the labels that the user will see.
Currency	A currency field contains a monetary value. In addition to numbers and perhaps a decimal point, you can also use commas where they are needed. Commas are always optional. - **Length/Decimal places**: Specify how many numbers can be entered (the left box). Use the right box to enter the maximum number of digits to the right of the decimal point. If you enter 0, there are no digits to the right of the decimal point. - **Hide currency symbol**: When checked, a currency symbol does not display.
Date	The output date is in UTC, and output date and time are in ISO 8601 format. For example, if you enter an input date of 2016-03-29, and use the currentTime XQuery function to get date and time values, you see the following output: - Date: 2016-03-29Z - dateTime: 2016-03-29T06:00:48.525Z - Time: 06:00:48.525Z The following are sample valid date values: - 2001-10-26 - 2001-10-26+02:00 - 2001-10-26Z - 2001-10-26+00:00 - -2001-10-26 - -2000-04-01 If you do not pass a time zone, Informatica Process designers assumes the time to be in UTC. If you need to format the way in which the date appears, see *Formatting Dates, Times, and Numbers*.
Date Time	When selecting a Date Time field in a process, the user can select the time from a list. - **Default Value**: Either enter a specific date, a field, or an interval, as shown in the figure. - **30 Minute Increments**: If checked, a time picklist displays lists hours and half hours. If it isn't checked, the user can type in any time value (for example, 10:37). The output time is in UTC, and output date and time are in ISO 8601 format. For example, if you enter an input date of 2016-03-29, and use the currentTime XQuery function to get date and time values, you see the following output: - Date: 2016-03-29Z - dateTime: 2016-03-29T06:00:48.525Z - Time: 06:00:48.525Z The following are sample valid dateTime values: - 2001-10-26T21:32:52 - 2001-10-26T21:32:52+02:00 - 2001-10-26T19:32:52Z - 2001-10-26T19:32:52+00:00 - -2001-10-26T21:32:52 - 2001-10-26T21:32:52.12679 If you do not pass a time zone, Informatica Process designers assumes the time to be in UTC. If you need to format the way in which the date and time appears, see *Formatting Dates, Times, and Numbers*.
Email	The content of this field is an email address. - **Display # chars**: The number of characters that are displayed within the process for this address. - **Max # chars**: The largest number of characters that can be entered for this address.

Data Type	Description

Formatted Text	The format is a representation of the kind of character a user can type within the text being entered.
	- **Format**: A pattern for text that the user types. You can use the following characters when defining a format:
	  - **a**: A lowercase letter; that is a through z.
	  - **9**: A number.
	  - *****: Any letter, number, or symbol
	Note: the "-" is not listed. If you can type this character, it is displayed when the user sees the step.

Image	A field that will contain an image. This is most often used when an automated action retrieves an image. - **Show preview**: Displays a preview of the image. - **Max width**: The maximum width of the area in which the image is displayed. - **Max height**: The maximum height of the area in which the image is displayed. - **Display # chars**: The number of characters that are displayed within the process for this address.
Integer	A positive or negative whole number. Use the **Digits** field to enter the maximum amount of numbers the user can type. Numeric values display using the format of the user's locale. If you need to format the way in which the number appears, see *Formatting Dates, Times, and Numbers*.

Multi-Select Picklist	A picklist field from which the user can select one or more rows. To select multiple items, the user clicks each one while pressing the CTRL key.
	- **Height (rows)**: The number of rows, each containing one list item that displays in the process.
	- **Comma Separated List**: The list of items within the list. Separate each item in the list with a comma. Each item specified is available in the list at runtime.
	Usage Notes
	- If you declare the data type of an input field as a Multi-Select Picklist or a Picklist and you do not enter values for the settings, you must use a semi-colon ";" as a separator instead of a comma. Be sure to use a semi-colon for an Equals condition and a comma for a Contains condition.
	- If the available values are not specified in the Properties for a Multi-Select Picklist or a Picklist and you receive a list of values from a search service, you must:
	  1. Specify the search service; and
	  2. Populate the list from a Screen step or Input options in other steps. Note that the list of available values exists only in the current step. To use the same list in other steps, specify the search service again.
	- You can also assign values to an undefined Multi-Select Picklist or Picklist in the Assignment step. In that case, use Formula as the Source and the value(s) of the Picklist or Multi-Select Picklist as the Content. When you enter a formula, be sure to enclose the values in quotes, for example: *"value"* for a Picklist and *"value1;value2"* for a Multi-Select Picklist.

Number	A positive or negative decimal number. Use the **Length/Decimal places** field to specify how many numbers can be entered (the left box). Use the right box to enter the maximum number of digits to the right of the decimal point. If you enter 0, there are no digits to the right of the decimal point. Numeric values display using the format of the user's locale. If you need to format the way in which the number appears, see *Formatting Dates, Times, and Numbers*.
Object ID	The object ID is the ID of the object instance. For example, it identifies one of your leads. (This value is assigned to the object by your application.) The unique parts of this dialog are discussed after this table. . Fields that are used here and in Field Properties dialogs are discussed at the `top` of this topic.

Data Type	Description
Object List	An object list is a set of object IDs. An object ID is the ID of an object instance. For example, it identifies one of your leads. - **Reference To**: The type of object that can be contained in the list. You must enter the object type's official name, not its label. If the list can contain more than one object types, you can enter a comma-separated list of object types. However, you should avoid doing this as there are query and table capabilities that can't be provided for fields unless they only have one **Reference To** type. - **Display Fields**: The columns (they must be manually entered) from the data associated with the object ID. - **Height (rows)**: The number of rows, each containing one list item, that displays in the process. - **Width** (%): The percent of the area in the used to display information. A read-only object list has a limit of 100 rows, even if the query associated with uses a LIMIT clause, asking for more rows to be returned. Other fields are discussed at the `top` of this topic.
Percent	A number expressed as a per cent. - **Precision**: The maximum number of digits in the number. - **Scale**: The maximum number of digits to the right of the decimal point. If you enter 0, there are no digits to the right of the decimal point. Numeric values display using the format of the user's locale. Here are some examples--each of these examples shows a number greater than 100%: - Precision 5, scale 2: 123.45 or 123.4 or 123. or 123 - Precision 5, scale 0: 12345 or 12 or 12,345 If you need to format the way in which the percent appears, see *Formatting Dates, Times, and Numbers*.
Phone	A field into which you can type a phone number. - **Format**: Enter the character that represents a number, which is 9 and any other display characters. For example, 99-999 will display as "_ _ - _ _ _" in the process.
Picklist	A picklist field from which the user can select one row. - **Comma Separated List**: The list of items within the list. Separate each item in the list with a comma. Within the process, each item displays in its own row. **Usage Notes** - If you want to display text to the user and assign a different value to the field, separate the label from the value with an equals ("=") sign. For example, if you type *Red, White, Blue* as the values, the user sees these three strings and the selected string is assigned to the field. However, if you enter *Red=1, White=2, Blue=3*, the user sees the same three strings but the number value for the selected item is assigned to the field. - See also the Usage Notes for Multi-Select Picklists for information on receiving a list of values from a search service.
Rich Text Area	A field that will contain HTML commands as well as text. Enter this text using the HTML editor. - **Height (rows)**: The number of rows within the area the process displays into which the user can type text. - **Width (%)**: The width of the text area relative to the process width. You cannot enter JavaScript into a Rich Text Area.
Text	A control that changes the size and number of characters that the user can type text. - **Max # chars**: The maximum number of characters that can be typed when entering text. - **Display # chars**: The number of characters that are displayed within the process. As the process size is limited, use this field to show how much of what the user types is displayed.

Data Type	Description
Text Area	A control into which the user can enter text. This can display as one or more rows. - **Width (%)**: The width of the text area relative to the process width. - **Height (rows)**: The number of rows within the area the process displays into which the user can type text.
Time	Select the Time field to select time from a picklist. The field properties dialogs that are displayed are: The output time is in UTC, and output data and time are in ISO 8601 format. For example, if you enter an input date of 2016-03-29, and use the currentTime XQuery function to get date and time values, you see the following output: - Date: 2016-03-29Z - dateTime: 2016-03-29T06:00:48.525Z - Time: 06:00:48.525Z The following are sample valid Time values: - 21:32:52 - 21:32:52+02:00 - 19:32:52Z - 19:32:52+00:00 - 21:32:52.12679 If you do not pass a time zone, Informatica Process designers assumes the time to be in UTC. If you check *30 Minute Increments*, you see a time picklist that displays hours and half hours. If it is not checked, you can enter any time value. For example, you can enter 10:37. If you want to format the way in which the time appears, see *Formatting Dates, Times, and Numbers*.

Data Type	Description
URL	A field that will contain a URL. - **For Read-only Display**: Your choices are *Link*, *Button*, or *IFrame*. - **For Read-only Label**: Text that identifies the contents of the field. - **Height**: The height of the area in which the URL is displayed. - **Width**: The width of the area in which the URL is displayed. - **Display # chars**: The number of characters that are displayed within the process. As the process size is limited, use this field to show how much of what the user types is displayed. A URL can be an input, output or temporary process field or even a field from. The field's data is assumed to be a valid URL. To create a button in a screen step: 1. Create a field by using the *Fields* tab in the process properties. 2. Use an Assignment step to assign it data to the URL being linked to. This data, which will be a content, can contain references to other fields such as ID fields by using `{!FieldName}` syntax. Process Designer automatically inserts the data using the button to the right of the text entry area. This is very useful when linking to other pages or other data dependent hyperlinks. 3. On any screen following the Assignment step, place the field within a step by selecting the file from the Read-Only fields picklist. 4. Select the field's ... button to specify its properties. Change **For Read-only Display** to **Button** and enter the text you would like to appear in the button in the **For Read-only Label**.

XML	You can use XML data type with only with an Informatica Process Designer process.
	
	To use the XML data type in a process, you can do either of the following:
	- You can use the XML data type in the assignment step of a process.
	- You can use the XML data type with JMS or AMQP. Here, the payload can be XML data and you can design a process that sends or message or receives a message.
	
	  If you use XML data with JMS or AMQP, wrap the payload within a root element.
	
	To map simple XML data elements to other fields, use formula and XQuery functions.
	
	For example, consider the following XML:
	
	`<o:order xmlns:o="urn:purchasing:system">`
	`    <o:number>123</o:number>`
	`    <o:amount>9213.32</o:amount>`
	`  </o:order>`
	
	You can use this XQuery function to search for the order number:
	
	`$input.myInboundOrder/*:number/text()`
	
	When you query XML data with namespaces, you must use the asterisk (*) character.

Object ID

The object ID is the ID of the object instance. For example, it identifies one of your leads. (This value is assigned to the object by your application.)

- **Reference To**: The type of object that the field refers to. You must enter the object's official name, not its label. If the object can refer to more than one type, you can enter a comma-separated list of object types. However, you should avoid doing this as there are query and table capabilities that can't be provided for fields unless they only have one **Reference To** type.
  A polymorphic relationship is a relationship where the referenced objects can be one of several different object types. If the object ID can refer to more than one object, check the Polymorphic field. A picklist is now available for choosing these objects. For example, while the Account object was selected, objects that could also be referenced are "Accepted Event Relation", "Account Contact", and so on. In this list

- **Display Fields**: The columns (they must be manually entered) from the data associated with the object ID.

When you are using this field in other steps, the information displayed differs. Here's a Process Designer example. When inserting the field, you have two choices.

If you insert the first (PolymorphicID), and click on the "..." in inserted field, the **Field Properties** dialog that displays shows all of the items selected in the Properties dialog:

However, when you select the item from within the Polymorphic section of the Insert Field for Update picklist, what you see are the individual items that were named. You will need to click again to get to the item that you actually want.

Object ID Notes and Comments

When an Object ID is placed in a canvas as a read-only field or put into a column in a table, it displays as a link to the object. If you want to show the Object ID as a number, insert it into a text field as "`{!Id}`". However, when an ID field is included in a result from a **Lookup** dialog, the ID's value displays.

When an object ID is displayed in a column or picklist, Process Designer displays with a meaningful name. For example, `b6f0f0b2-9f38-4771-8365-58eb4f7b7d41` might display as "Acme". However, when an object ID field is included in the results from a Lookup Dialog, it displays using the value of the ID; for example, it might display as `b6f0f0b2-9f38-4771-8365-58eb4f7b7d41`.

If you use a content field to enter an object ID and are using a sandbox, you will need to adjust the ID when you move the process to your production environment. Rather than using a content, you could use a query to load the object ID by name for the object.

Selecting and Displaying Objects

Within a step, you may want your users to select one or more object. You can display this information in a variety of ways:

- A pop-up search dialog for searching for objects by name
- A picklist from which they can select one object
- A multi-select picklist for selecting more than one
- A table of objects, and you can select which of the object's fields it should display in separate column
  If only one object can be selected, Guide Designer adds a column containing radio buttons; otherwise, it adds checkboxes.

Note: If you are just displaying one object, you can display the object's fields as a one-row, multi-column table.

By default, Guide Designer displays a search dialog box when users are searching for information for fields that can be updateable. You can override the default and tell Guide Designer to display object information using either a picklist or table. Picklists and tables are populated using a query that filters the possible objects down to just those that should be presented. For example, Guide Designer could show only the contacts that are associated with an account.

You can customize the fields being shown for each object. If only one field from these objects is displayed, Guide Designer shows this information as a list. However, if you are displaying more than one (for example, a lead's first and last name), Guide Designer shows each within its own column in a table.

Note: Object queries are not available when the field you are updating can reference more than one type of object (that is, the **Reference To** control within the **Field Properties** dialog includes multiple object types). If this is your situation (that is, you cannot use a regular query), you must select and use an advanced query.

Displaying Information

There are actually twelve ways in which Guide Designer can display object information, six for when you working with one object (a reference) and six for when working with a list. The following table presents the ways in which you can display object information, and each group of six is displayed in a different color.

Carefully notice the differences between rows. For example, if only one object can be selected from a list of objects, Guide Designer displays radio buttons. However, if more than one object can be selected, it displays check boxes.

The information in these columns is as follows:

- **How Displayed**: The left column indicates how objects are displayed; the remaining four columns in the row show how this choice is made.

- **How Inserted**: The information is displayed as read-only or it can be displayed so that the user can update it. This is controlled by the way in which the field was inserted into the step.

- **Display Columns**: If you want the information **"Displaying Columns" on page 103** within a table, select the columns within the **Field Properties** dialog. Not making a selection has the same effect as selecting just the name field.
  Note: The **Field Properties** dialog is extensively discussed later in this topic.

- **Show List?**: Have you indicated that a list of possible objects should be shown to users. If yes, you will need to select one of the possibilities in the **Show List** picklist within the **Field Properties** dialog? If you don't make a selection, users can choose any of the object type's objects of the appropriate object type.

- **One Object or List?**: Is the field presented on the screen as type Object ID or Object List?

How Displayed	How Inserted	Display Columns	Show List?	One Object or List?
Hyperlink	Read-only	One	N/A	One object (a reference)
Single row table	Read-only	Multiple	N/A	
Search	Update	One	No	
Search with multi column result	Update	Multiple	No	
Picklist	Update	One	Yes	
Table with radio buttons	Update	Multiple	Yes	
Table: one column	Read-only	One	N/A	Object list
Table: multi column	Read-only	Multiple	N/A	
Multi-select picklist (none selected)	Update	One	No	
Table with checkboxes (none selected)	Update	Multiple	No	
Multi-select picklist (Show List results selected)	Update	One	Yes	
Table with checkboxes (Show List results selected)	Update	Multiple	Yes	

Note: If Guide Designer displays just one item, it displays it as a hyperlink.

Using the Field Properties Dialog

When you define steps within a process, you insert fields to handle input, output, or other variables that you might need to receive from a service or pass to another step in the process. Each of the fields you add has a

set of field properties that you can configure to determine the field source, the formatting, display properties, and list handling.

For some fields, you can also use the Expression Editor to define complex formulas that determine the field values and other attributes.

The topics in this section describe the options in the Field Properties dialog and the kinds of queries that you can write for use with field properties.

Controls in All Many or Some Field Properties Dialog

Most **Field Properties** dialogs have the following controls:

- **Default Value**: Sets a default value for the information being displayed. Your choices are *Content* (object type), *Field* (has the object type), *Formula*, and *Screen* (user specified object).
- **Display Fields**: Names the fields to be displayed in columns and the order in which they display.
- **Field**: The name of the field from which the **Field Properties** dialog was invoked.
- **Height (rows**): Sets the number of rows that will be displayed.
- **Hover text**: Text that is displayed after a user places the cursor over the field.
- **Reference To**: The type of object to which the field applies. You cannot change this value--not in all dialog boxes.
- **Required**: Click this checkbox if this value must either be entered or cannot be deleted and not replaced.
- **Show List**: Defines the way in which objects are located.
- **Width (%)**: Sets the width of the displayed information as a percentage of the width of the process.

Show List: Advanced Query

Selecting **Advanced Query** lets you finely tune the information retrieved for the list.

After the query executes, the default action is to return a list whose values are object IDs; the list also has the name field for display.

The source for each of the query's four parameters can be Content, Field, Formula, or Screen. In almost all cases, the source is Content. The items that you can set are follows:

- *Object Type*: The object type upon which the query executes.
- *Where Clause*: Enter an SQL WHERE clause. The text you enter here does not include the WHERE keyword. Here, Content means you will write the WHERE clause, Field means that the field contains the WHERE clause, etc.
- *Display Field:* Set a value here to return a field other than the current one. Process Designer will retrieve this data, and then display it within a picklist. If you do not specify a field, Process Designer uses the *namefield* (the first one, if there is more than one).
- *Value Field*: Returns a different value than what Process Designer would use by default. If you do not set a value field, the ID of the returned object is used.

Advanced Technique: You could use an advanced query to create a step that displays the query's results within a step. You might want to do this in a different "testing" guide.

Show List: Custom List

Selecting **Custom List** let you enter your own list of values. For example, you might individually type the names of the New England states. When the process displays, the user sees the items you enter in a picklist.

You must separate items from one another using a comma (",").

Notice that you can also set a *Default Value*.

Show List: List Child Objects

Use related (or child) objects in the query. While similar to using related objects within fields, using **List Child Objects** is simpler as you do not need to specify the query's condition.

After you select **List Child Objects,** you also need to specify the child's object type. Your choice for source is automatically set to "Content", which is the only allowable source.

Show List: None

Use this setting to display a search box that allows the user to specify any value for the field. If the field is a reference to an object, not using *Show List* lets the user locate objects of the type named in the *Reference To* area.

You control how to display information about objects using the *Display Columns* item.

Show List: Object Query

Use a query to retrieve object information from all of the objects that are selected by the Where Clause.

After you select **Object Query**, the Process Designer adds the Object Query controls to the **Field Properties** dialog. The *Source* for the *Where Clause* is always Content. While you can enter your condition directly into the text area, it is usually far easier to click the **Add Condition** button.

Note: The difference between a value and what is shown in the display field may not be clear. In many cases, they may be the same. The difference is that the display field is what is shown to the user while the value is what is stored. For example, if a currency value is stored with two places after the decimal, its display could omit these numbers. This distinction is almost always used with ID fields as you want to display something like a Name while using the ID as the actual value that is stored when the user selects it. Also, you may be using code tables where the code is something other than an Object Id. For example, imagine a States object where the code is the two letter abbreviation that is stored as the value in other objects, but you want to display the full name to users when they are selecting State.

After using this dialog, you can edit the information that it added.

Selecting the field from the picklist is suggested as it is sometimes not obvious what the field's internal name is. The text entered here is a standard SQL WHERE clause. Also, this text does not include the WHERE keyword.

If you are creating more than one condition, you can use the standard AND and OR operators. You can also use the NOT operator to invert the meaning of the condition.

Each condition has four parts:

- The name of a field in the object. In this example, the field is one of those contained within the Account object (this is the object named in the *Reference To* area).
- An operator that Process Designer uses when it compares the field value on the left with values on the right.
- The kind of data that will be compared. Your choices are Content, Formula, or Field. If you choose *Field*, you are comparing the contents within the field in the current object with the contents of the field in the type of object being queried.

- The data to which the field on the left is being compared. If the source is Content and you click within this area, a small icon appears to the right. After selecting it, Process Designer displays a picklist from which you can select the name of a field in another object that will be used when making the comparison.

To add an SQL LIMIT clause to a query, enter it in the **Where** clause text box. For example, enter "Limit 200" to limit the number of retrieved rows to 200. Add a space after the WHERE information and then enter the SQL LIMIT. Else, enter the SQL LIMIT information on a new line.("Limit" can be in upper, lower, or mixed case.) By default, 100 rows are returned. If your process requires more than 100, you must add this clause. However, a read-only object list has a limit of 100 rows, even if you set a query associated with it to more than 100.

If you use a JDBC connection, you can use the ORDER BY unction to sort data.

To use the order by function, click **Order By** and select a field. For example, you can sort a list of names by a person's last name.

Note: You cannot use the order by function to make OData-enabled requests to a JDBC connector.

Show List: Picklist for Field

Record types are information that you associate with objects that let you select some of the objects. For example, suppose you have "On the Road" and "In the Office" records types for a lead. Your process can use these values to select what other information it should display. If the record type value is "On the Road", you could have a picklist with values A, B, and C. If the value was "In the Office", picklist values could be C, D, and E.

Use this option when using a "Create" Service step with a dependent picklist. While Object Type and Field Name are automatically filled in, you will need to enter a Controller Value. If the user will fill in the field containing the record type, set the Source to Field and also select the appropriate field that this is dependent on. If it is Content just enter its value.

For example, you have a Create Lead Service step that has a "Business Sector" picklist and an "Industry" picklist dependent on it. When the user selects "Technology" from the "Business Sector" picklist, Process Designer can display values set for that sector in the "Industry" picklist, values such as "Computer Hardware" and "Biotechnology." In order for this to operate correctly on the Create screen, you must edit the field properties for the "Industry" field and change the Controller Value" to Field and select "Business Sector" as the field.

Your options when you select Picklist for Field are:

- *Object Type*: The object type containing the object field. This should already be filled in correctly, so you don't need to change it.
- *Field Name*: The name of the field that defines the picklist values. This should already be filled in correctly, so you don't need to change it.
- *Controller Value*: For dependent picklists to display correctly, fill in this controller value. Setting this value to be content is not a good idea since the ID for a record type would be different when used in sandbox and or in the product. What you should do is use a query to dynamically set the controller value. This query must return just one value.

Show List: Users in Role

Users in Role tells Process Designer to locate all users who have the role that you identify.
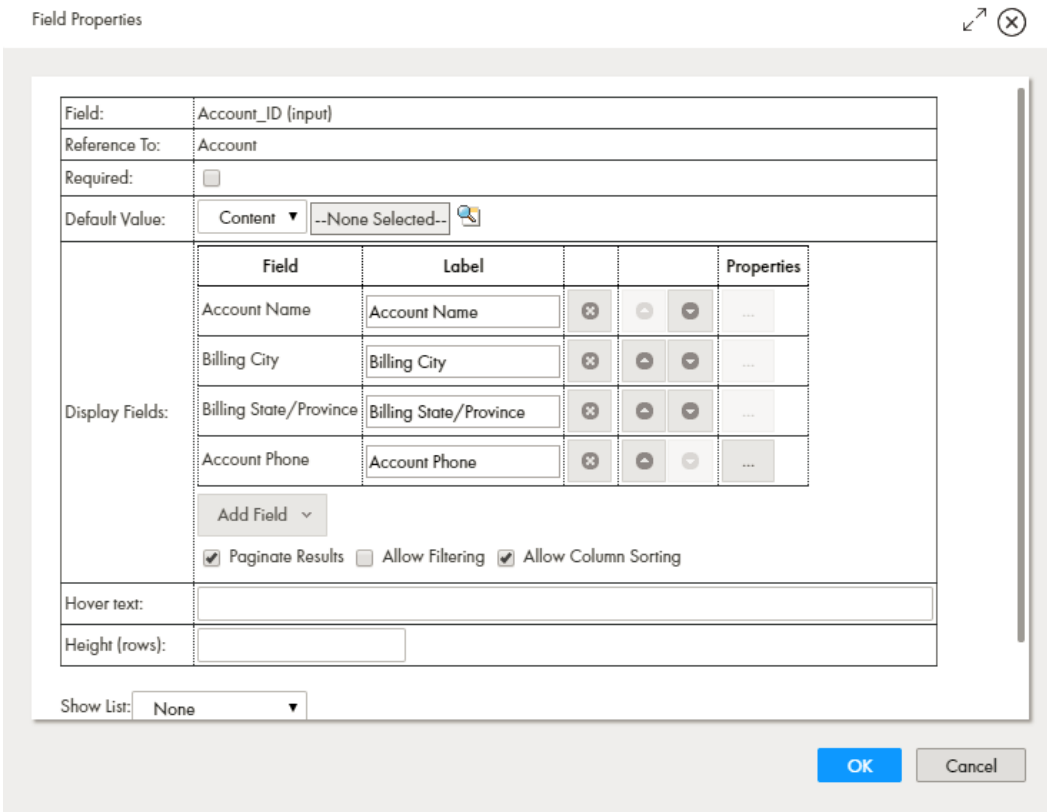
When you choose this option for Show List, you also select the Role name that Process Designer will use to locate users in that role. The source for each of these can be "Content", "Field", "Formula", or "Screen".

Displaying Columns

Use the **Display Fields** field to tell Guide Designer which of the object's fields it should display in a table. If the field can be updated, each row also has either a radio button if users can only select one object or a checkbox if they can select more than one.

Use the **Add Field Column** button to add columns. You can use the arrow buttons to the right of the name to change the order. If you do not want to use the default column heading, enter the text you prefer.

Here's a four column table:

The three check boxes are as follows:

- *Paginate Result*: When selected, the output is paginated.
- *Allow Filtering*: When selected, the user can enter text in a text box to select some of the displayed results.
- *Allow User Column Sorting*: When selected, the user can click on a column heading to sort the results by that column's contents.

Note: When an ID field is included in a result from a Lookup dialog, the ID's value displays.

Navigating in Process Central

Guide Designer is a system for creating guides--guides are a set of screens and these screens are called steps--that lead users through a series of actions for performing a task. These guides run within browsers and these browsers can run on computing platforms such as a PC or a Mac. Each step contains an action. It could contain information that your user needs, a form that needs to be filled out, and so on. Behind the scenes, steps interact with an application by extracting, storing, and manipulating data.

The parts of Central related to guides are:

- Forms: This pane contains all of your forms, some of which may not be guides. Process Central guides are within the Guides Launchpad folder.

- Reports: (Administrators only): Here you will be able to display a report that gives you information about guides that are either executing or have finished executing.

- Guides: While the Forms pane is where people execute guides, the Guides pane is where you design guides. Guides that you design are available to other users.

The following figure shows where these features are located.
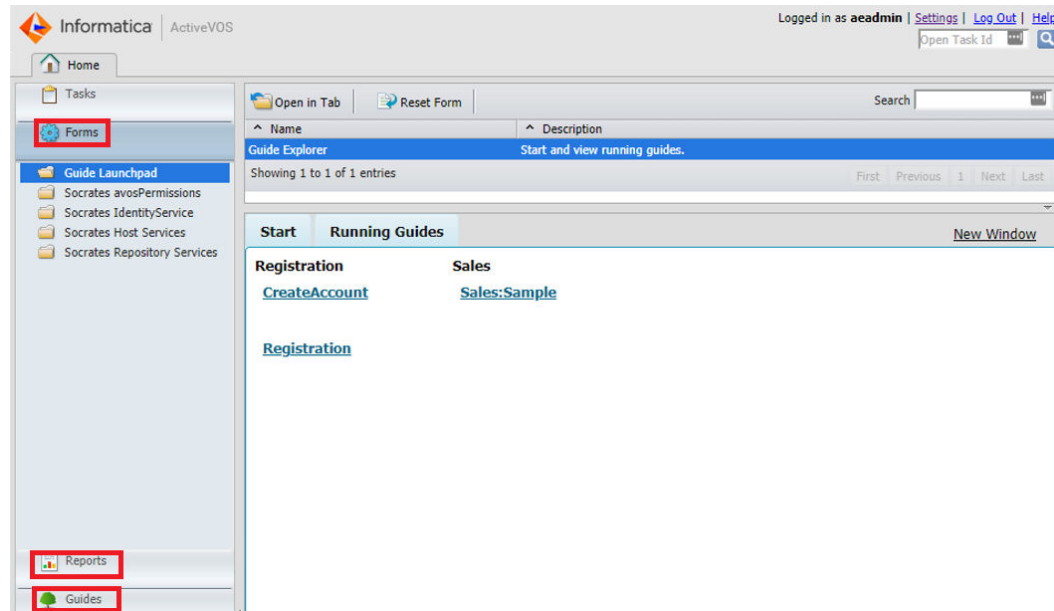
Forms Pane

After you open the Forms pane, click on Guide Launchpad and select Guide Explorer to show the guides available for you to launch. The two tabs that always display are Start and Running Guides.
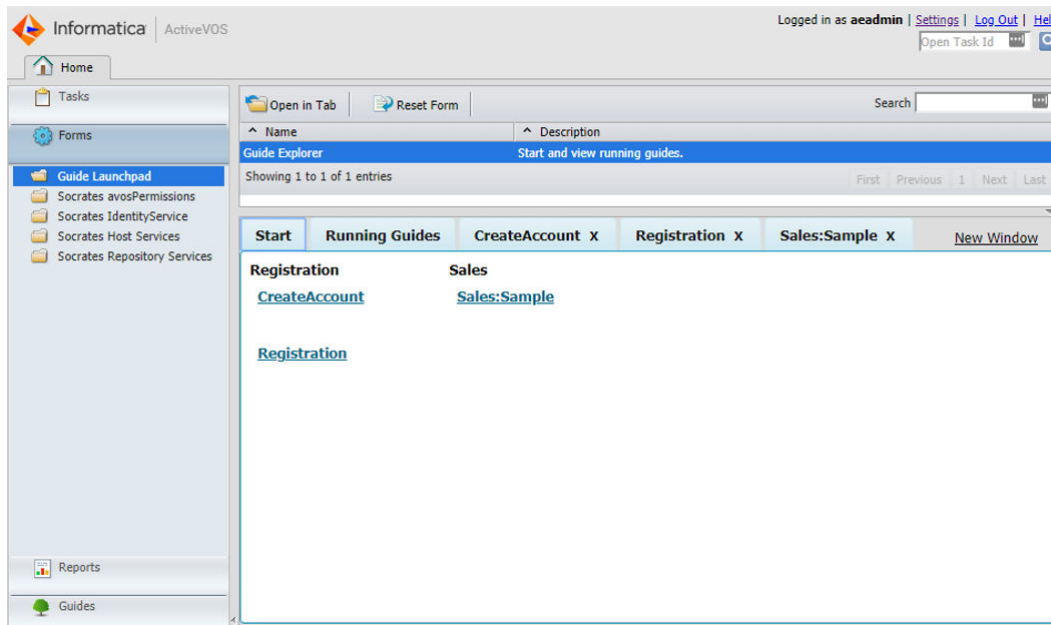
The guides shown in this Start tab are only those that were designed to be run from this tab. Other guides are designed to be run from other forms or from within web pages. Clicking on a link starts the guide's execution.

To start a guide, click on its link. If you click on a link of a guide that is already running, the guide redisplays within the previously created tab. Also, if you click on a tab, the guide within the tab displays. If you exit from Process Central, and log in again, the tabs from running guides do not display. Another way to go to a running guide, and the only way to go to a running guide after restarting Process Central, is to click on the Running Guides tab, and then click the link within the Current Screen column.
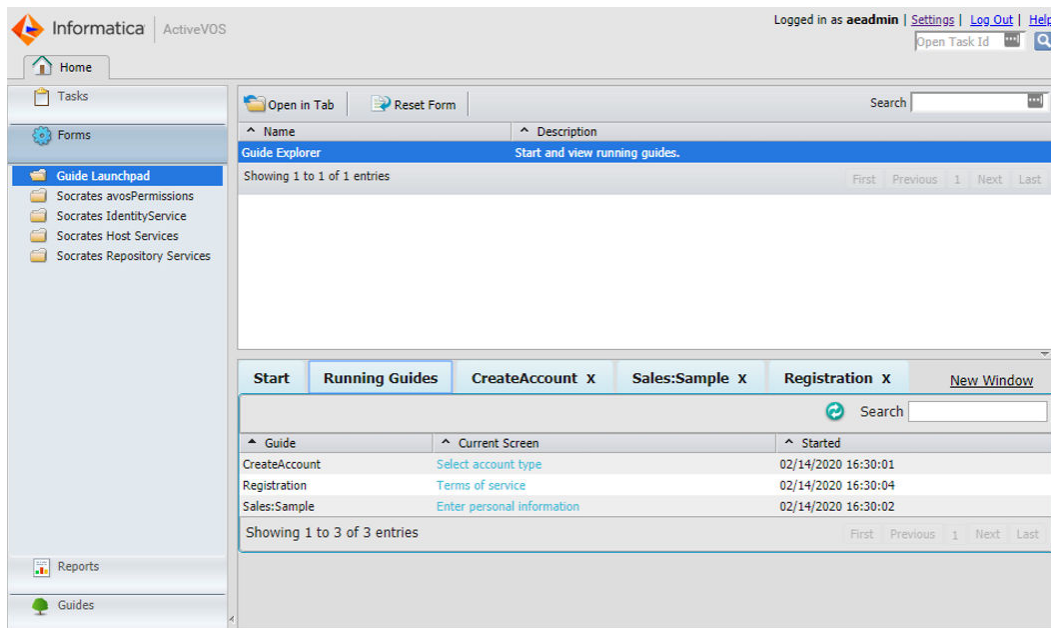
If you do not wish a guide's tab to display, click the 'X" within the tab. While the tab disappears, this does not stop the guide's execution.

Notice that you can sort the information by clicking on a column's header. If you have a great number of guides executing, you can look for a particular guide by typing part of its name in the Search box.

If you have begun executing a guide after displaying this tab, you will need to press the Refresh button (to the left of the Search field).

Designs Pane

When you click on the Designs pane, Process Central information about the guides, processes, process objects, connectors, and service connectors that you have created..
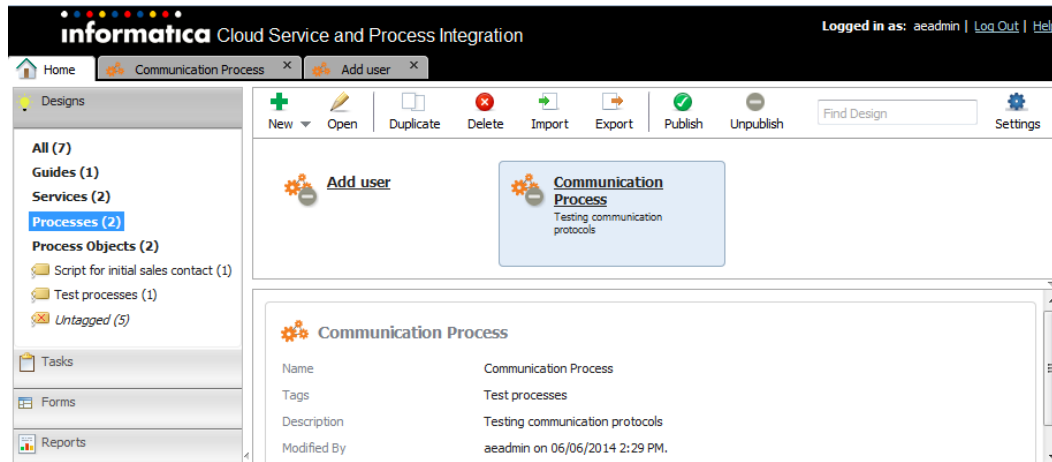
To create a new item, select **New** from the Designs toolbar. Information on creating and editing design items is elsewhere in this help.

CHAPTER 3

Process Designer Tutorial

This tutorial walks you through the steps you need to use to create the basic foundation for just about any guide you want to create. The following figure shows a part of the guide you will create:
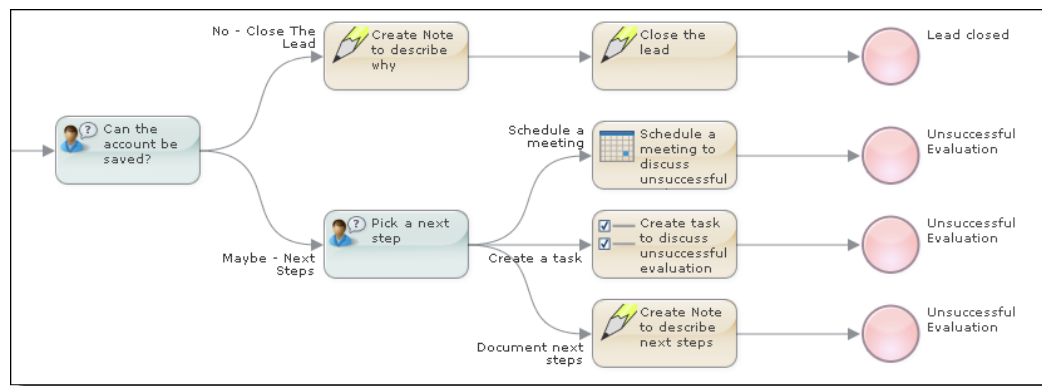
The rounded rectangles and circles represent different steps that could occur. Here, the rectangles represent actions that could occur and the circles represent the end of the interaction. The lines starting at the left side of a step and ending on the right side of another step are the paths (called branches) that users would take as they execute the guide.

Just by using the tutorial, you will understand how the different kinds of steps are used. You will be using the following:

- **Screen steps**: The most fundamental of the Guide Designer steps. You will use them to display instructions, ask questions, and have the user fill in forms. It can also display information contained within a database and perhaps even update it.

- **Service Call steps**: A Screen step is limited to updating information that already exists. The Service Call step can also read information from your database and update it. However, its power is that it can also create new records and perform other activities such as using your email system. Developers create Service Call steps that interact with their applications and environments. For example, a Service Call step could be created that sends mail to people.

- **Jump Steps**: As you are designing a guide, there are times that you want to jump from one branch of the tree to another, which is what a jump step lets you do. One example is that the steps in one branch schedule a meeting. Other branches will also need to schedule a meeting. Rather than each have scheduling steps, you could jump to the steps in another branch. You can also use jump steps to repeat a set of steps using different data. For example, a branch could create a contact. If there are additional contacts, users could jump back to the step that lets them create it.

- **Embedded steps**: While a guide can be self-contained, there are many reasons to have one guide be embedded in another. One example is the scheduling set of steps just mentioned. Instead of using a jump step to go to these steps, you could embed a guide that contains all of the scheduling steps into a branch.

(If you understand programming concepts, there is an analogy to a subroutine. There are some important differences and they are described in the Guide Designer documentation.) A less obvious use for embedded steps is to repeatedly perform a set of activities. For example, you want to update all of your contacts. Here you would create a guide that updates one contact, and then send a list of contacts to it one at a time.

- **End Steps**: An end step marks the place where no more processing will occur within a guide. It can also display information to the user. For example, if a guide created a contact, the end step could display the contact's information.

The Sales Call with Contact Prompt/Sales Call Follow-up Guides

This tutorial will have you recreate the Sales Call Follow-up guide that is one of the sample guides shipped with Process Developer.

The Sales Call Follow-up guide has 23 steps. Only 6 of them are shown in this figure.
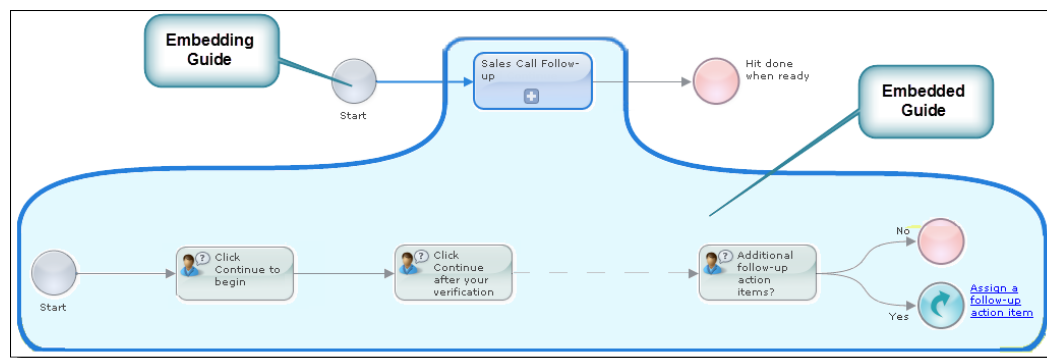
Embedding one guide within another guide isn't a metaphor: it is literally true. All of the steps within the embedded guide are copied into the embedding guide. However, the embedding guide remembers where these steps came from. If the embedded guide changes, the embedding guide will let you know that it has changed so that you can republish the embedding guide.

The next figure shows all of the embedded guide's steps.
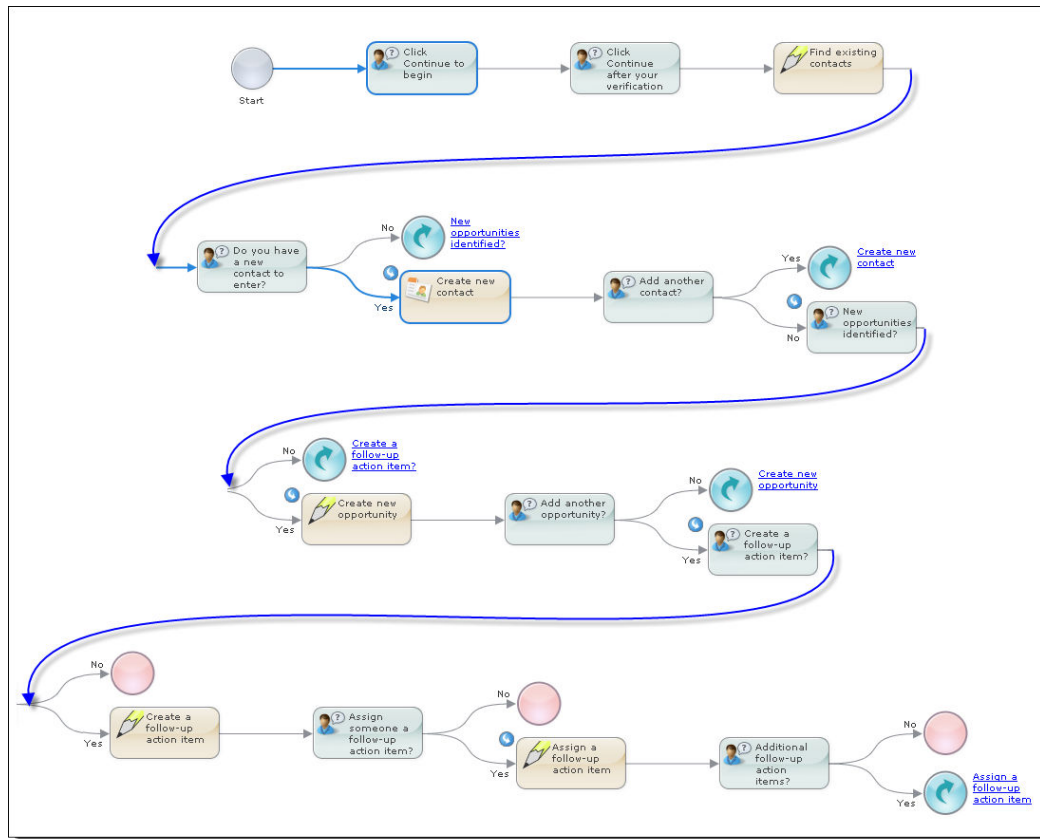
The execution of this guide can be divided into four parts, and each of these parts is shown in its own area in the next figure:

- **Part 1**: Preliminary steps, some of which is introduction.
- **Part 2**: Entering new contacts.
- **Part 3**: Entering new opportunities.
- **Part 4**: Creating and assigning action items.
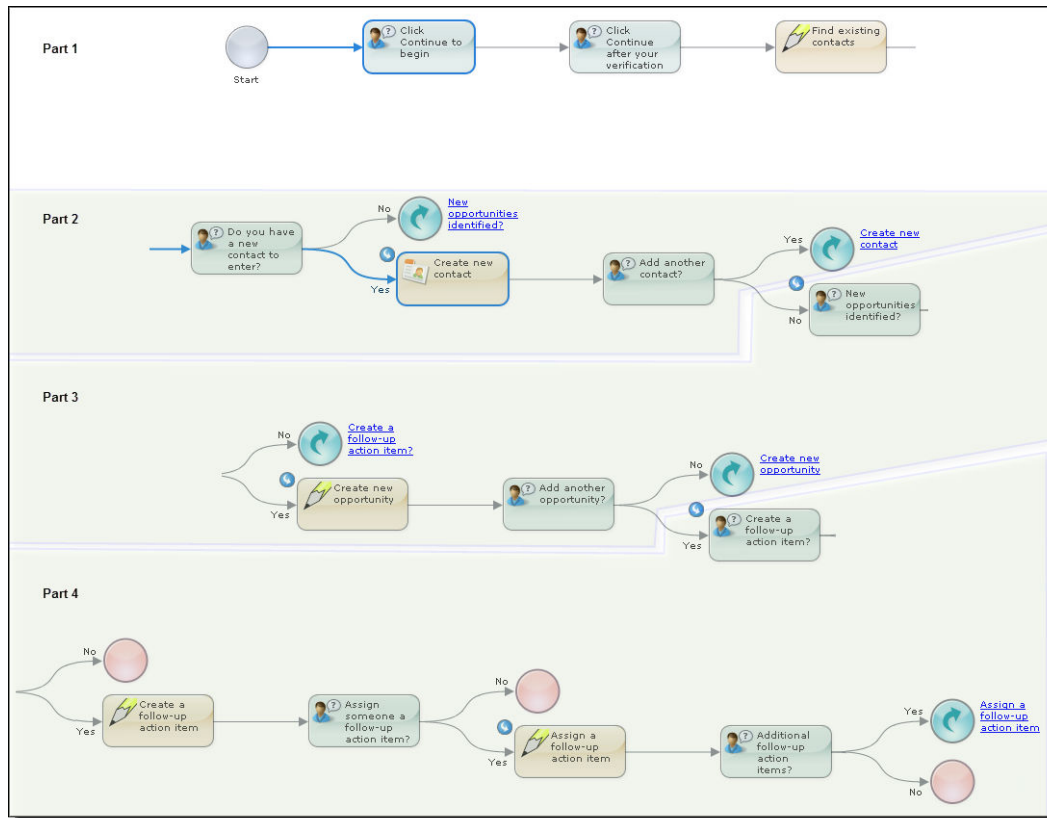
The curved arrows within circles in this figure are jump steps. They tell Guide Designer that it should "jump" to a step and execute that step and the steps that follow. Guide Designer represents them this way instead of drawing lines between the steps as the step canvas could quickly become a tangled maze of lines. However, arrows representing the jumps are shown in the next figure as the jumps are very straightforward in this guide.
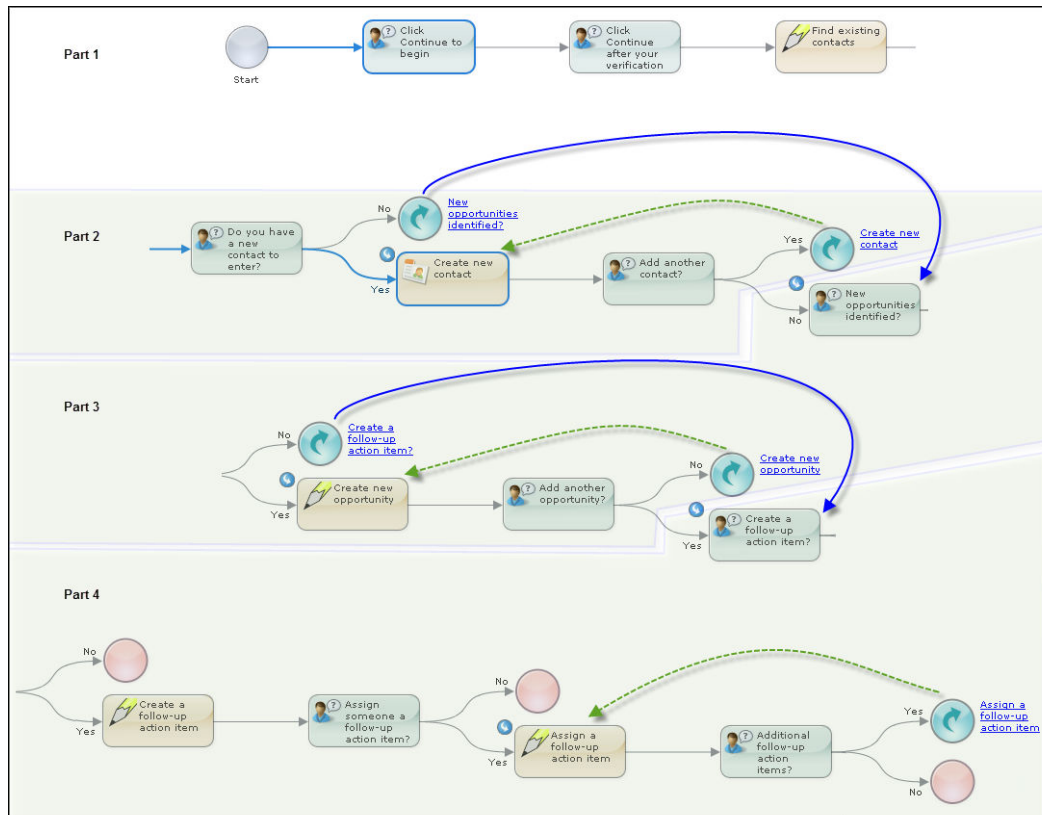
The blue arrows show forward jumps and the green arrows show backward jumps.

What You Will Be Building Here

As the guide shown in the previous topic is large, this tutorial will only describe some of the steps, leaving the remainder as exercises. The next figure shows most of the steps that are described in detail in this tutorial.
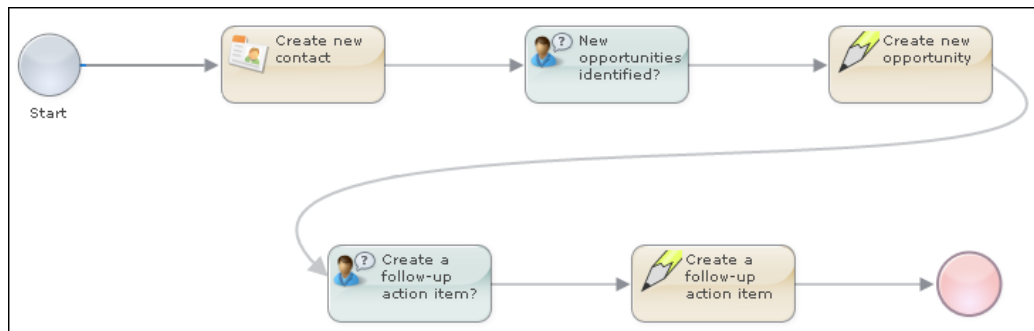
Lesson 1: Creating a Guide

This lesson explains how to create a guide. Other lessons will have you deploy and run what you create.

Part 1: First Steps

Note: Before following along and creating the tutorial guide, you must have previously created the Guide Designer orchestration project. You will do this from within the Process Developer. You will find complete information in an Process Developer Cheat Sheet and an overview in Installing the Guide Designer Orchestration Project.

Begin creating a guide by pressing the **New** button.
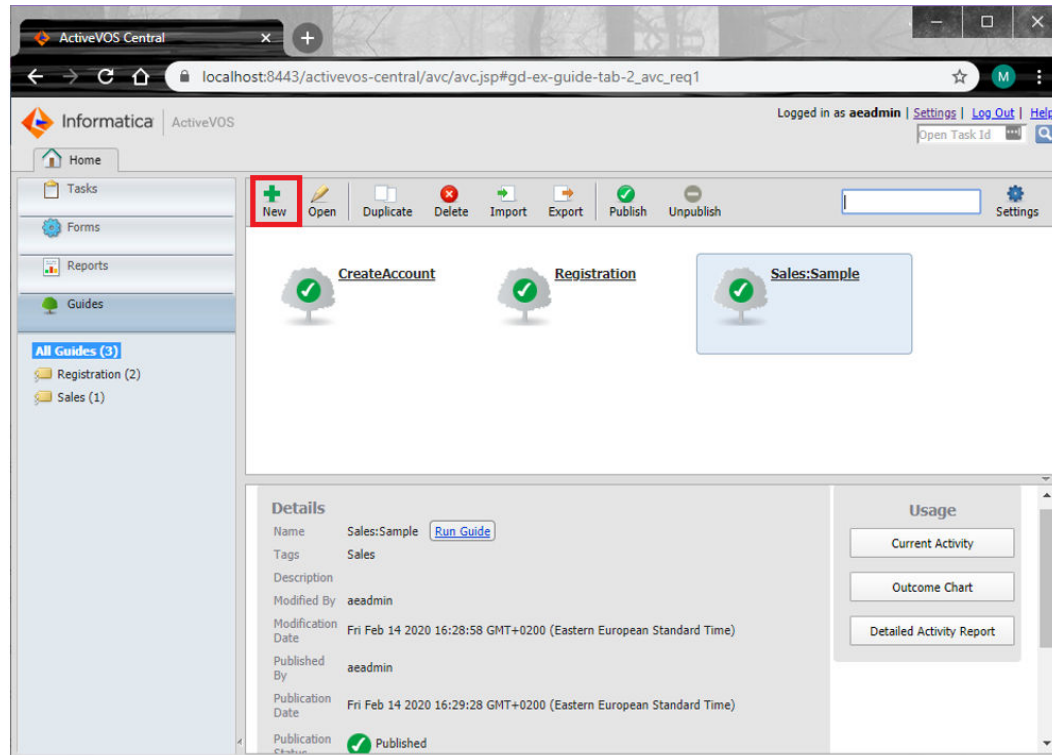
A new tab displays and this tab contains the **Guide Properties** dialog box. Underneath it (or perhaps to the side) are two steps that give you a place to begin.
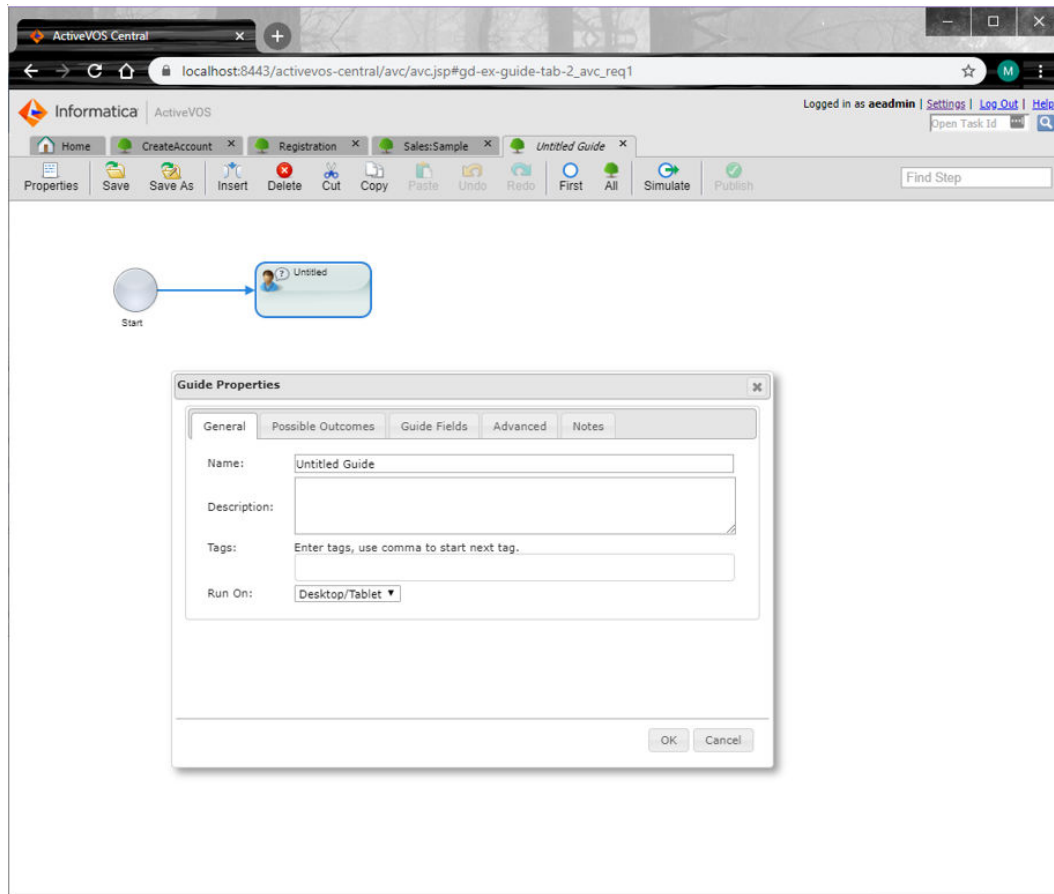
Enter the following information into this dialog, and when you're done, press **OK**.

Field	Action
Name	Type "Sales Call Follow-up Tutorial".
Applies To	Select **Contact** from the picklist. What you select here makes data within a database table available to the guide.
Description	Type "A simplified Sales Call Follow-up guide".
Tag	Type "Sales Call Follow-up". A "Tag" is a category under which the guide appears when Guide Designer presents guides to the user.
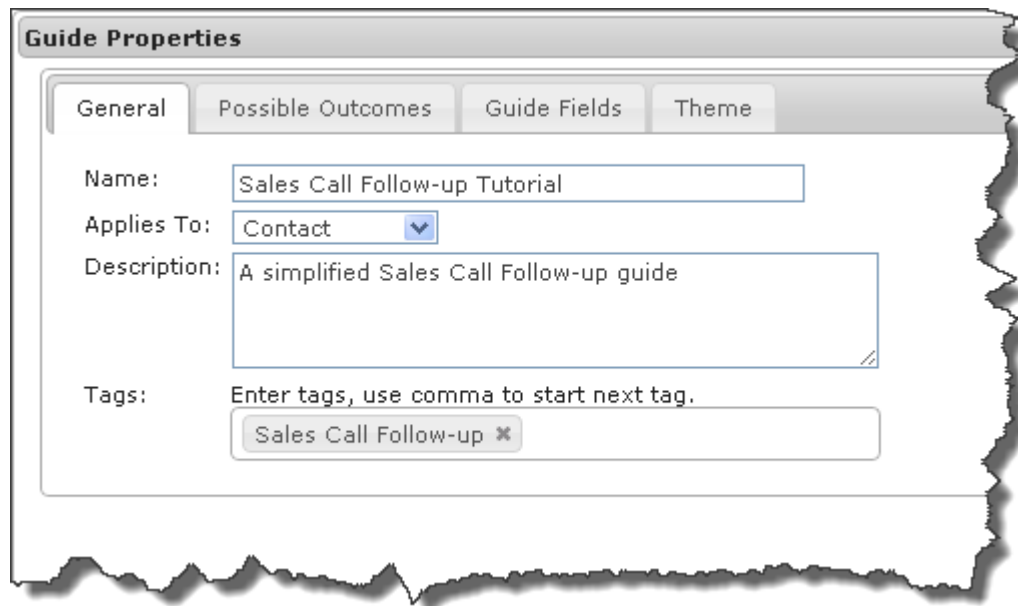
Part 2: Beginning the Guide The First Step

All actions that Guide Designer performs are defined by the steps that you place within it. This topic describes the Create new contact an Service Call step. Here it is in relation to the steps that you will create:
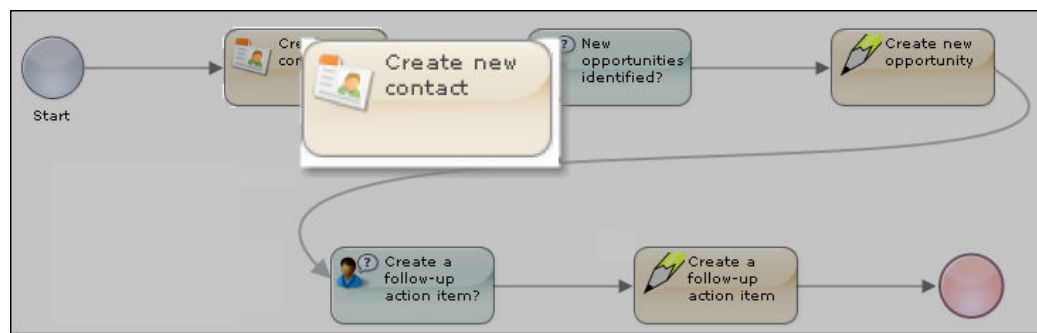
To help you get started, the Guide Designer added two steps to the canvas. The first (the one on the left within a circle) is a "start step". There are no actions that you will ever perform on a start step. As it name indicates, it is the place where Guide Designer will begin executing your guide. The second is an untitled step. This is actually where you begin. You will add additional steps in other parts.

The *Create new contact* step adds information about a contact to your application's database. Guide Designer will do this automatically when users complete this step.

The procedure is as follows:

1. Double-click on the untitled step.

2. **Step Type**: Select *Service Call* from the picklist.

3. **Title**: Type "Create new contact".

4. **Action**: Select **Create**, then in the displayed menu click **Create Contact**.
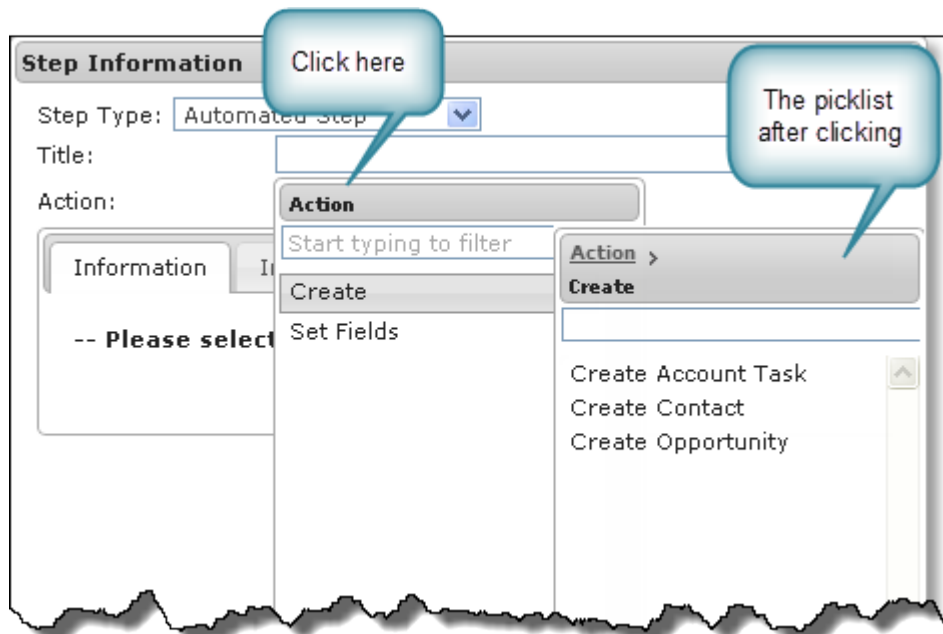
The *Information tab* now contains input and output fields. The field's data type is displayed and if it is in an input field, a description of the field.

5. Click the *Input tab*.
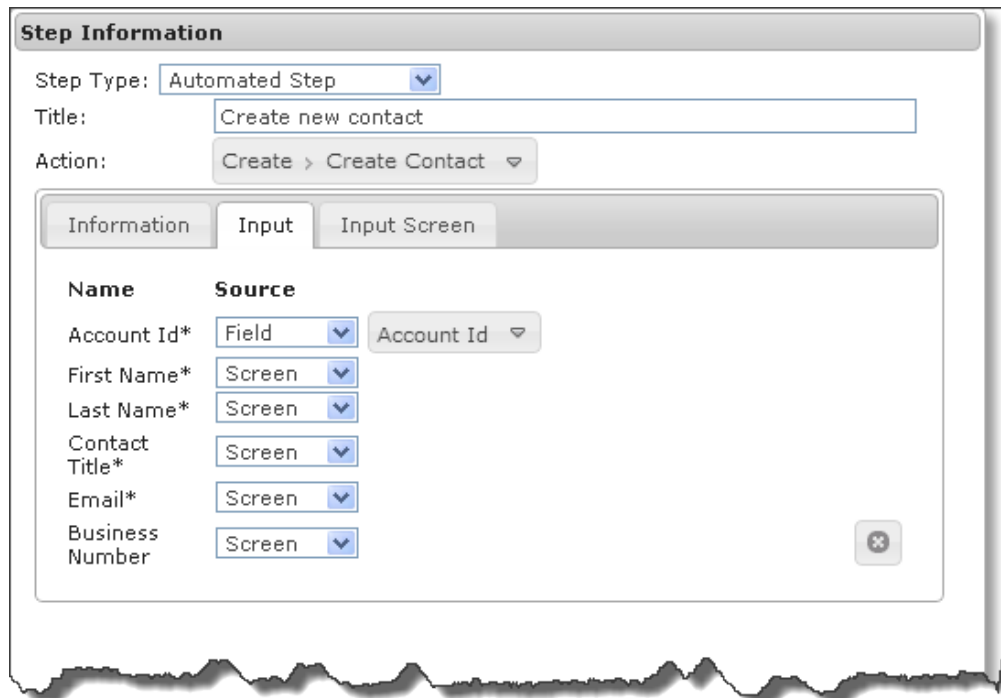
The *Input tab* has all of the field's within a contact record. All but Business Number are required--notice the asterisk. If a Business Number isn't needed, you can delete it from by clicking the "x" to the right. Not all Service Call input tabs contain all of the table's fields. (The developer who connected Guide Designer to an application defines which are displayed and which are required.)

6. Set the source for Account Id to be **Field**, then select Account Id from the Field picklist. The rest should have their sources set to **Screen**. When the source is set to **Field**, Guide Designer uses the value in the current context. When it is set to **Screen**, you will need to insert a place into which the user can type information within the Input Screen tab.
    Note: For information on setting sources, see *Setting Source Value*s elsewhere in this help.

7. Click on the *Input Screen tab*.
    Guide Designer places a button for each of the sources you set to Screen within the Input tab. You will need to insert these buttons into the canvas.
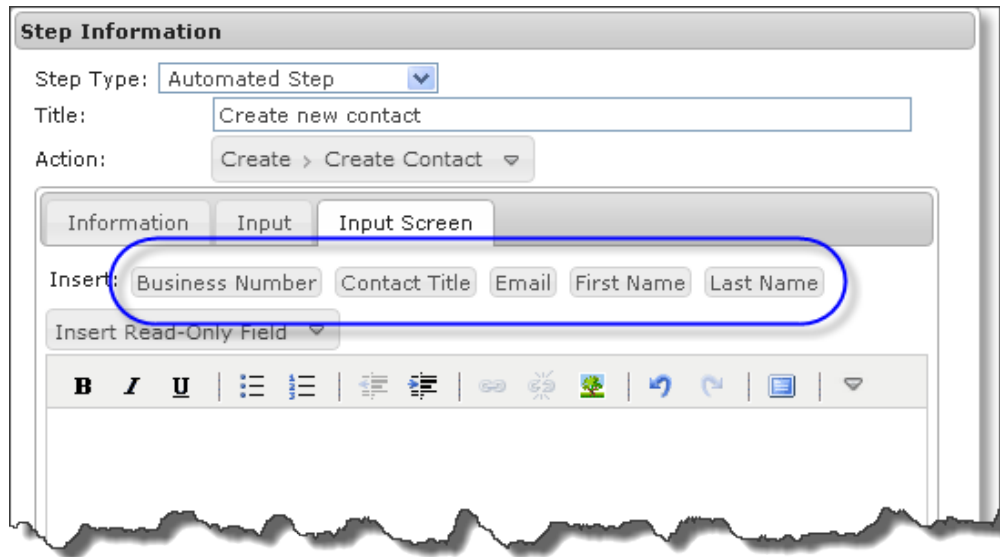
The area beneath the editing buttons is called a "canvas". You will use it in much the same way as you would a page within a text editor such as Microsoft Word.

Begin by typing text that tells the user what to do: "Enter the contact's information". Next, type press your keyboard's Enter key. Click on the buttons in the order shown in the following figure, pressing the Enter key between each. Notice that after the field is inserted, the button is shown in gray.

Step Information

Step Type: Automated Step

Title: Create new contact

Action: Create › Create Contact

| Information | Input | Input Screen |

Insert: Business Number Contact Title Email First Name Last Name

Insert Read-Only Field

B *I* U | list | list | indent indent | link unlink image | undo redo | window |

Enter the contact's information

First Name First Name*

Last Name Last Name*

Email Email*

Business Number Business Number

Contact Title Contact Title*

Continue Button: Continue

Preview   OK Cancel Apply

8. At any time, you can see what a step looks like by pressing the **Preview** button in the bottom-left corner.
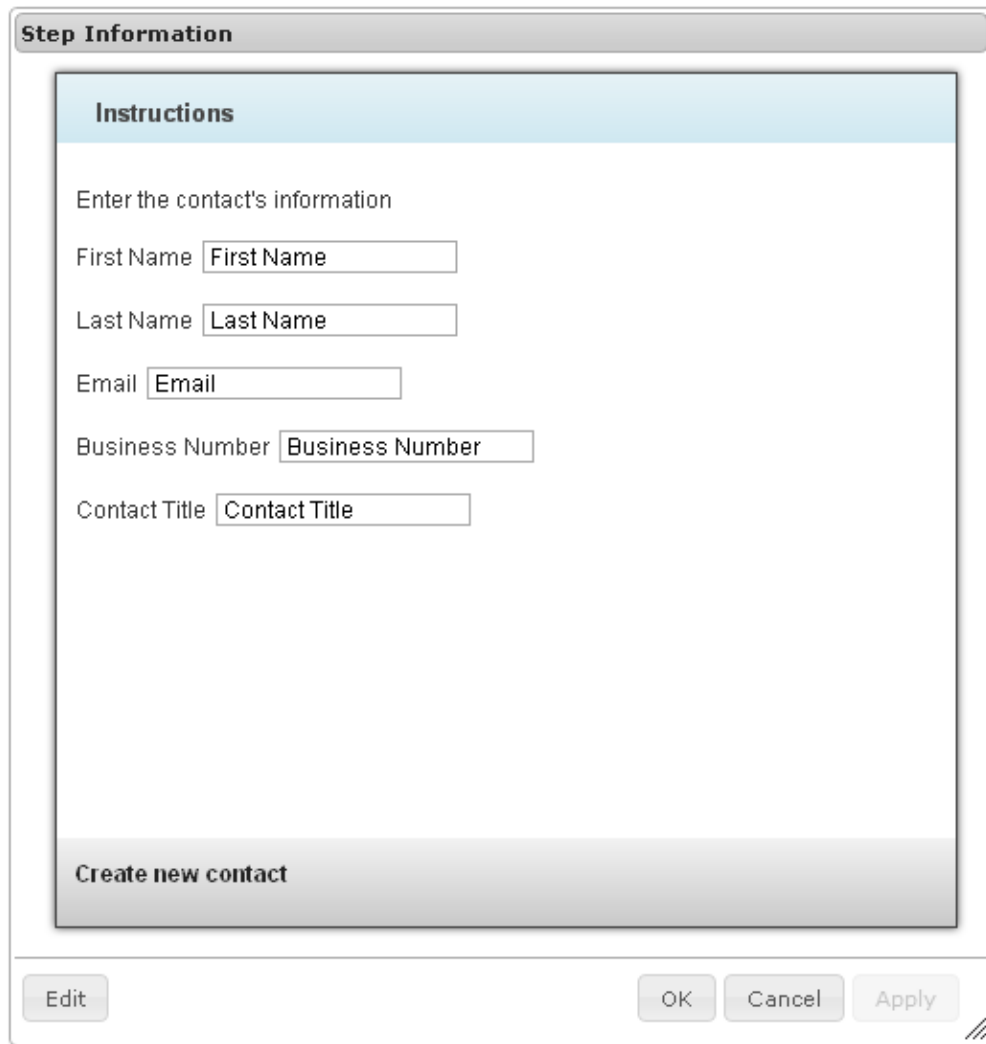
The **Preview** button is now an **Edit** button. Press it to return to the Service Call step's canvas.

9. Press the **OK** button in the Service Call step.

10. Go to the menu bar and press **Save**.
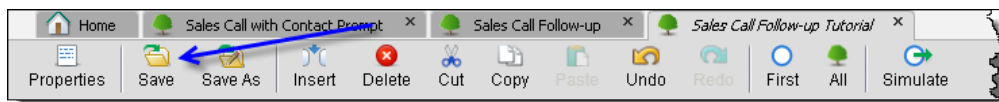
After you press **OK**, the Guide Designer uses the title you had typed as the name of the Service Call step. The Guide Designer also adds another untitled step. In the next topic, you will set this to be a screen step.

Part 3: Adding the New opportunities identified Screen Step

This topic adds a a second step to the guide. This step will be a screen step and will follow the *Create new contact* Service Call step.
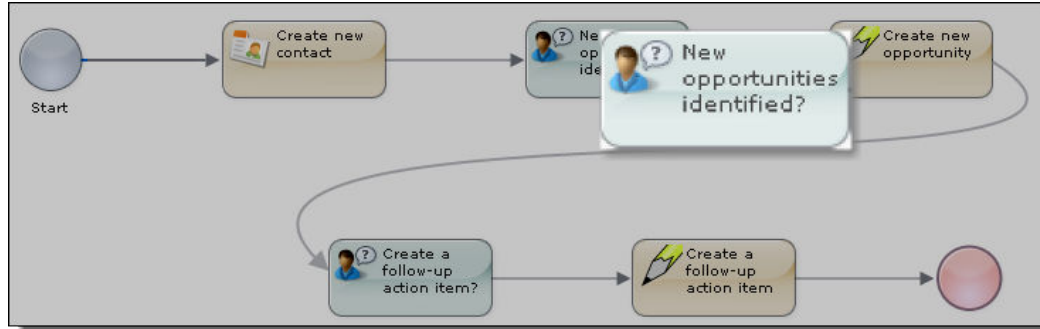
Here are some of the more common ways in which screen steps are used. Here are the most common ways it is used:

- It can just contain information text.
- It can ask a question.
- It can contain read-only fields. These fields display information contain in your application's database.
- It can contain updateable fields. An updateable field can also display database information or it can be empty. When the user changes existing field information or add new text, Guide Designer adds the changes to your application's database.

Here is a procedure for creating this screen step:

Field	Action
Step Type	Select *Screen Step*
Canvas	Type "Did you identify new opportunities during this call"
Prompt	Type "New opportunities identified?"
Answers	Type *No* and *Yes*

Here's what your step should look like:

These buttons work the same way as they do in a word processor..

If you type answers in the wrong order, press this button to change their order.
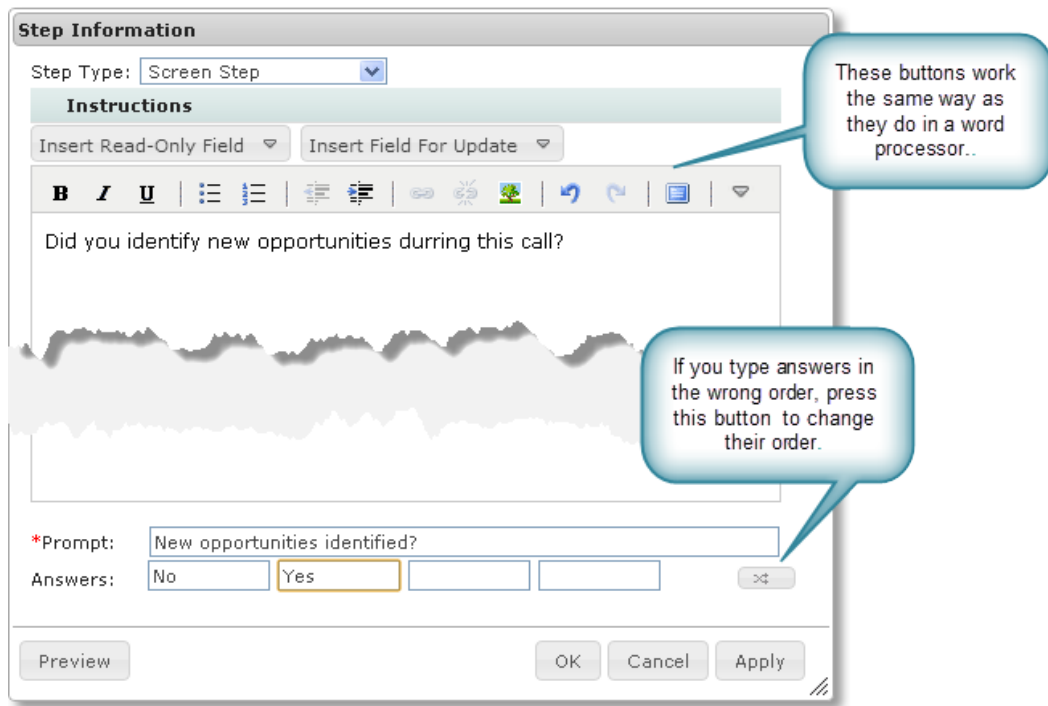
After clicking **OK** (and you should also save your changes), Guide Designer uses the prompt as the step's title. It also adds two untitled steps to the right, one for each answer.
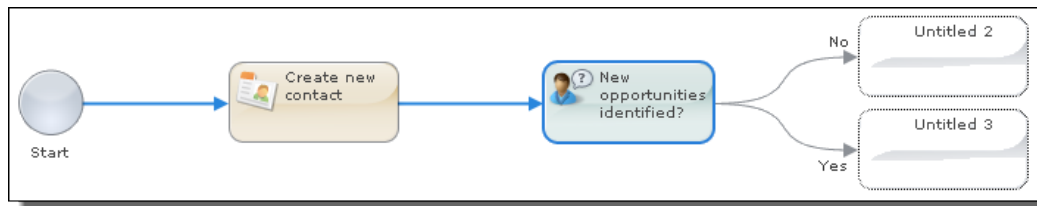
For now, you'll ignore the *No* branch.

Exercise: "Exercise 1: Adding the Click continue to begin Screen Step" on page 146: Here you will add a screen step that is very similar to the one you added in Part 2.

Part 4: Adding the Create new opportunity Service Call Step

In this part, you'll add another Service Call step.
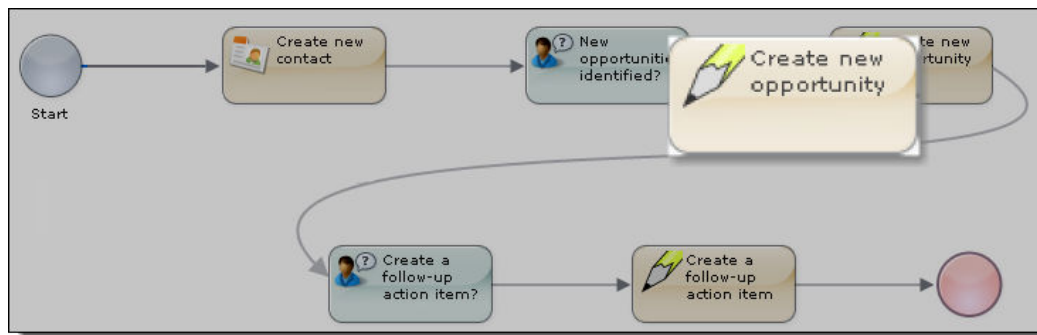
Here's how you should fill in the Input tab:

Field	Action
-	Click on the untitled step that is the target of the Yes branch that was created in the previous topic
Step Type	Service Call Step
-	Click on the Input tab
Title	Type "Create a new opportunity"
Action	Select **Create > Create Opportunity**
Account Id	Set **Source** to **Field**, and then select *Account Id*
Name	Set **Source** to **Screen**
Close Date	Set **Source** to **Screen**
Stage	Set **Source** to **Screen**
Add Input	Select *Amount*--it was added to the fields above it--then set its **Source** to **Screen**

Note: For information on setting sources, see *Setting Source Values* elsewhere in this help.

Here's what the *Input tab* should look like:
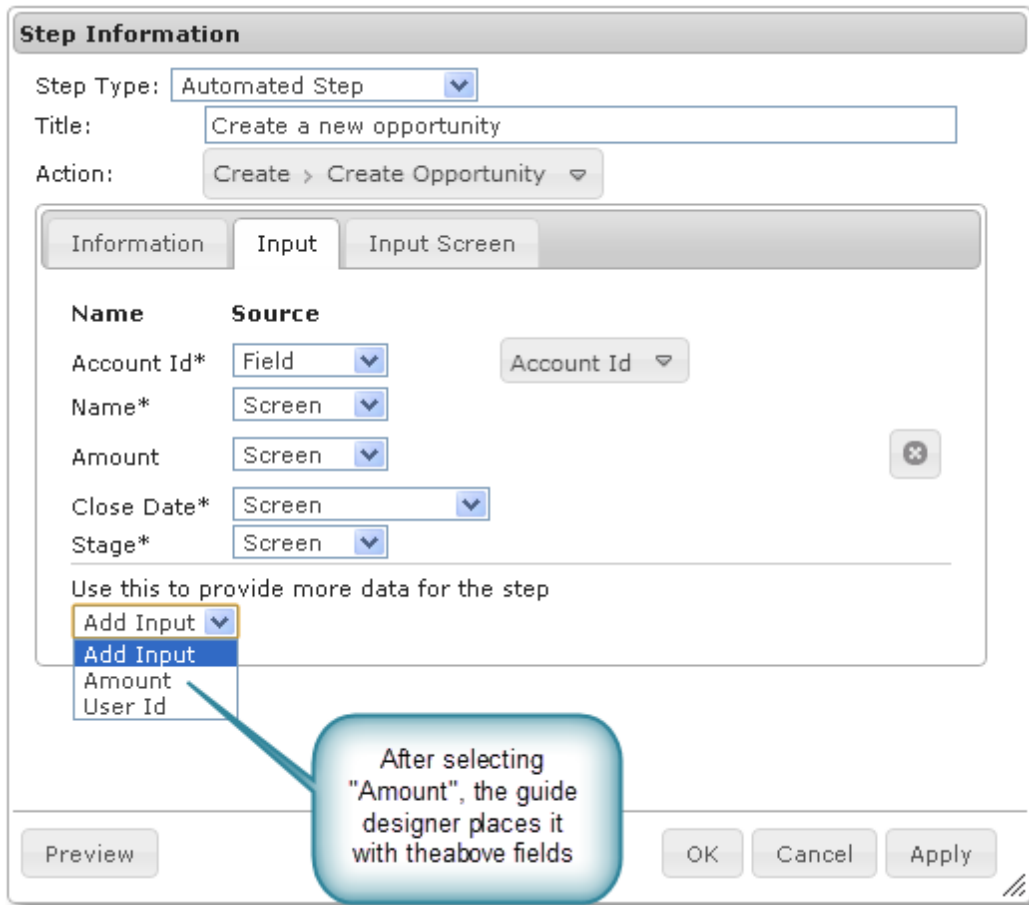
The order of the fields here may be different than what appears in your Input tab.

Now create the Input Screen:

Field	Action
--	Click the Input Screen tab
Canvas	Type "Enter the opportunity"
Insert	Click on each button--don't forget to press the Enter key between items

Here's what the step should look like:

Step Information

Step Type: Automated Step

Title: Create a new opportunity

Action: Create › Create Opportunity ▽

| Information | Input | Input Screen |

Insert: Amount Close Date Name Stage

Insert Read-Only Field ▽

B *I* U | ≔ ≔ | ⇥ ⇥ | ∞ ⚡ 🖼 | ↺ ↻ | ▣ | ▽

Enter the opportunity

Name [Name*] (...)

Amount [Amount] (...)

Close Date [Close Date*] (...)

Stage [Stage*] (...)

Continue Button: [Continue]

Preview   OK   Cancel   Apply

Change the text of the three of the labels and use the "B" toolbar button on all four labels to set the text to bold. Here's what you canvas could look like.

Enter the opportunity

Opportunity Name: [Name*] (...)

Booking amount: [Amount] (...)

Expected close date: [Close Date*] (...)

Stage: [Stage*] (...)

Note: For information on using this toolbar, see **Using the Screen Editor**.

Let's make the page a bit more attractive by adding an image.

1. By default, images float right. Placing the cursor to the right of text often has the best visual results.

2. Select an image (it doesn't matter which one), and then click the **Insert Image** button.
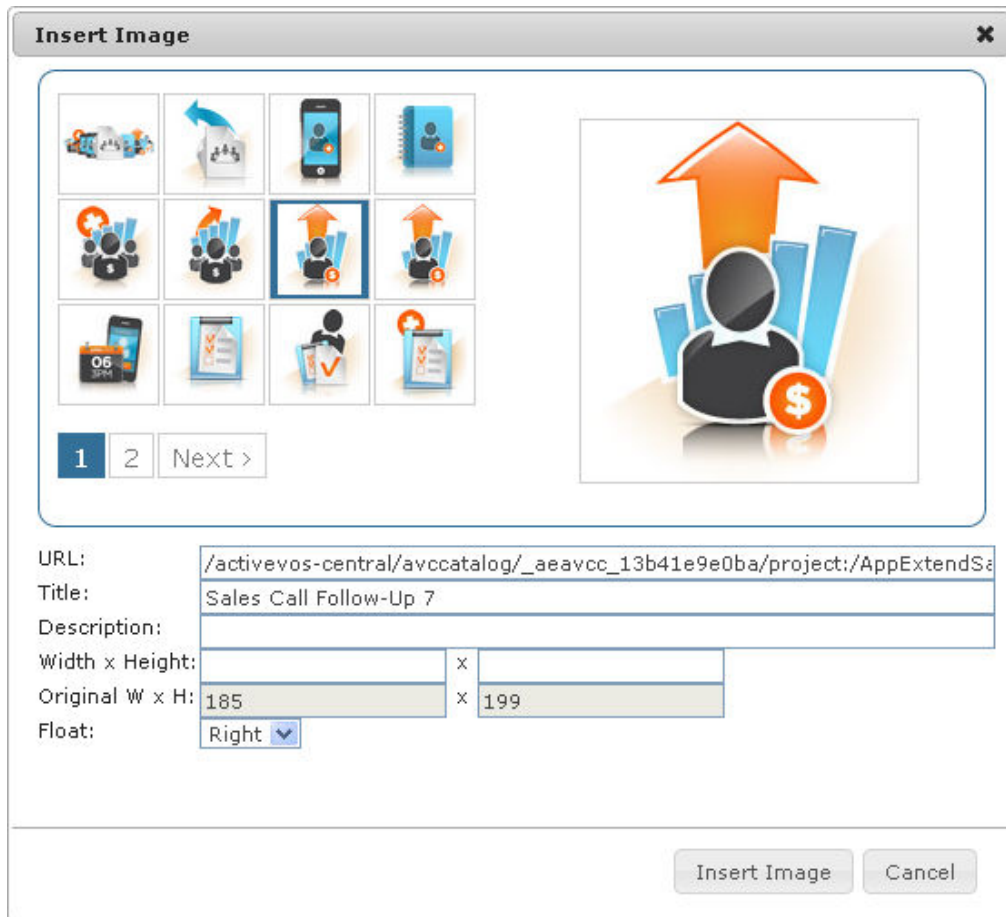
The canvas should not look something like this:

After clicking OK at the bottom of the **Step Information** dialog, the right part of your guide should look like this:
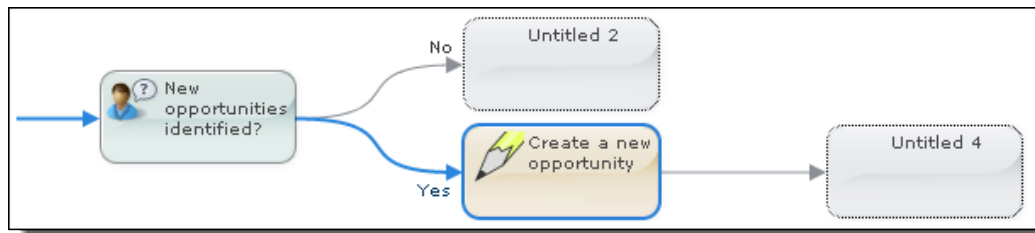
Press the **Save** button.

Exercises

- "Exercise 2: Adding the Click continue after your verification Step" on page 147
- "Exercise 3: Adding the Do you have a new contact to Enter? Step" on page 148

Part 5: Adding the Create a follow-up action item Screen Step

You'll add a second screen step in the same way you added one previously.
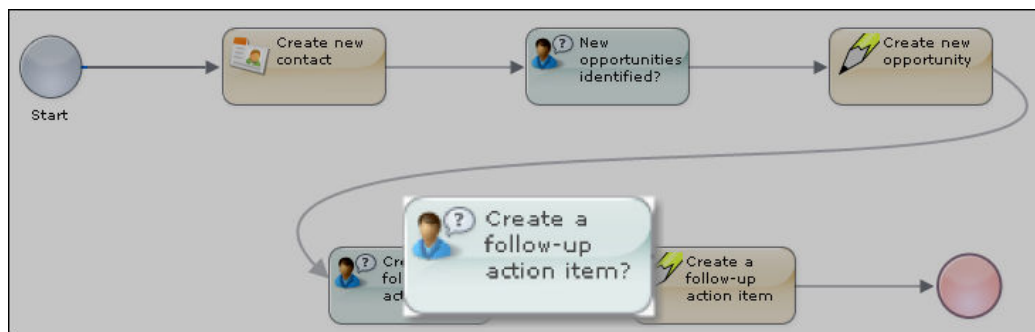
Do the following:

Field	Action
-	Click on the untitled step
Step Type	Select Screen Step
Canvas	Type "Do you want to create a follow-up action item?"
Insert Picture	It doesn't matter which one you choose
Prompt	Type "Create a follow-up action item?"
Answers	Type *No* and *Yes*

Your step should look something like this:

After clicking **OK**, the right side of your guide should look as follows:
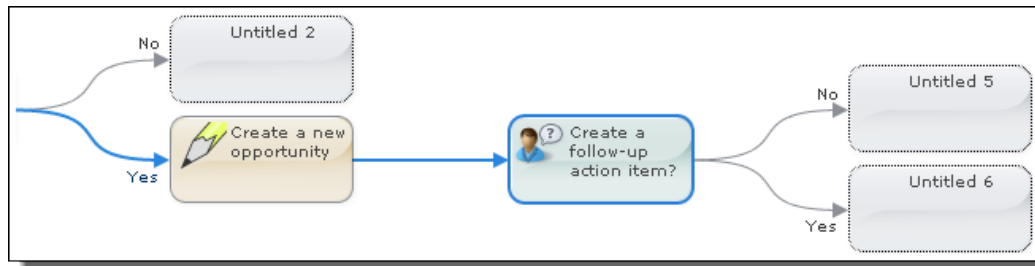
Part 6: Adding the "Create a follow-up action item" Service Call Step

This topic has you creating a third Service Call step. You will learn to use the **Insert Read-Only Field** picklist that occurs within many steps.
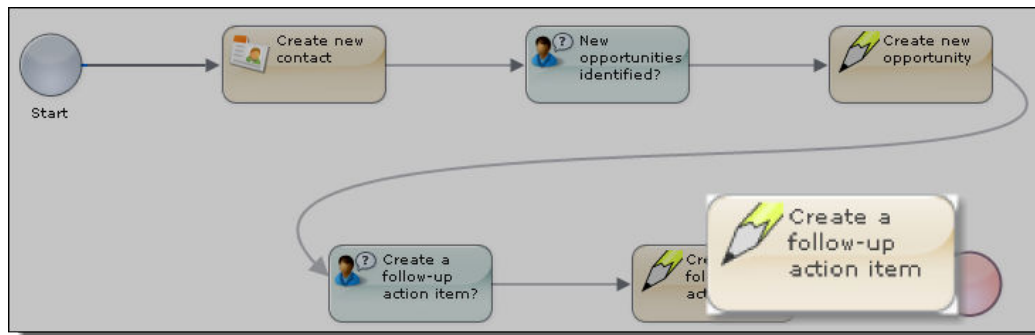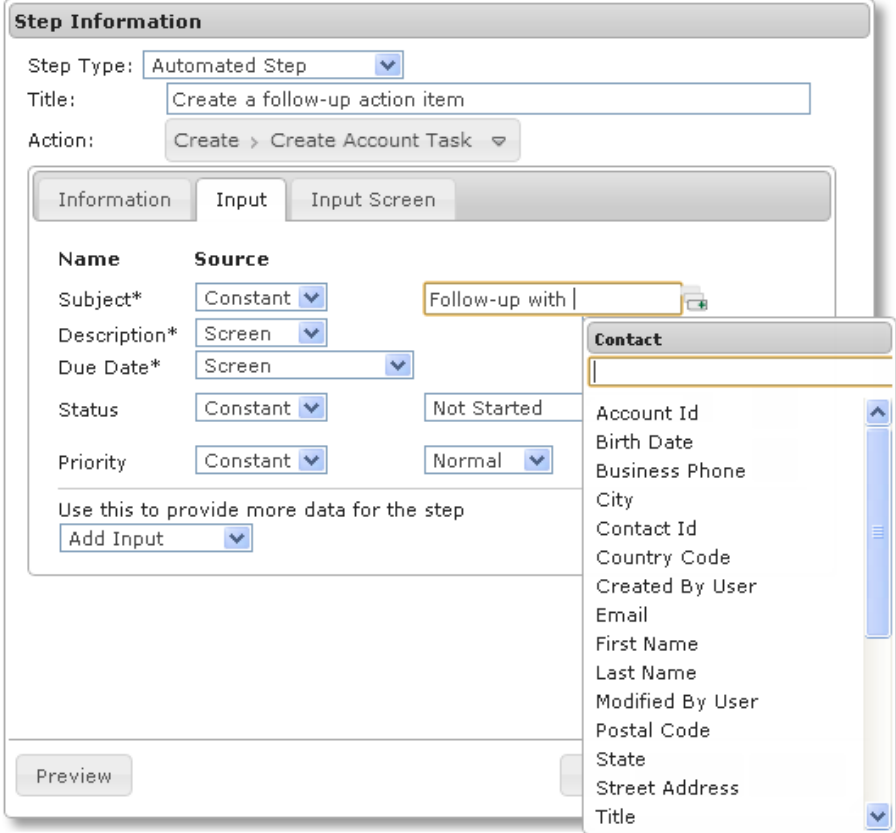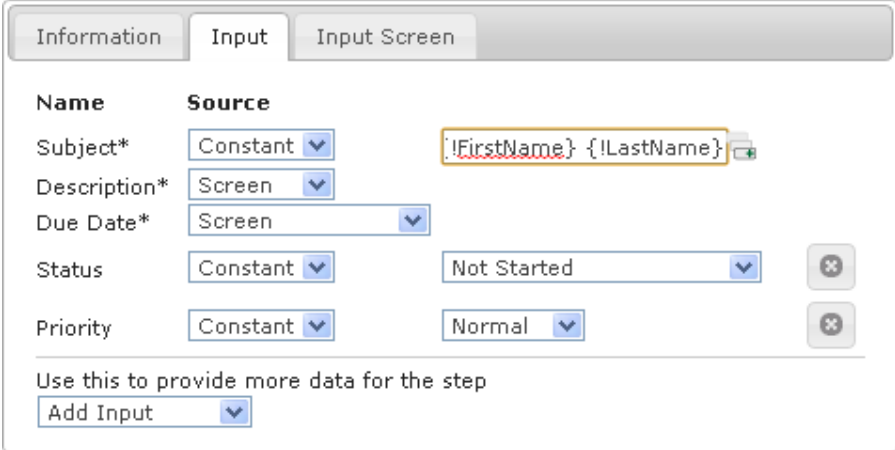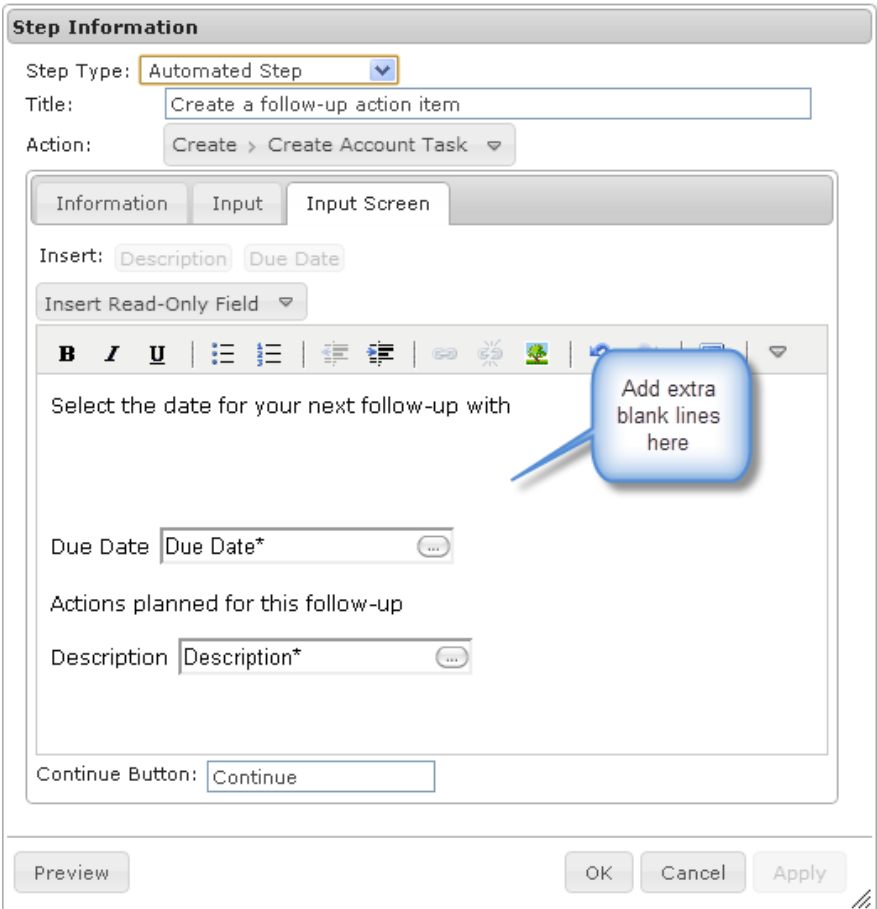
In this step, you'll set the **Source** for a field to **Constant**. (You've already set **Source** to either **Field** or **Screen**.) When you set a field's **Source** as **Constant**, Guide Designer will set the field to what is entered here and when the record is added, this constant will be the value written into the field.

Note: For information on setting sources, see *Setting Source Values* elsewhere in this help.

Here's what you should do:

Field	Action
-	Click on the *Yes* branch following the *Create a follow-up action* item step
Step Type	Select *Service Call* from the picklist
Title	Type "Create a follow-up action item"
Action	Create > Create Account Task
Input tab	Select it if it isn't selected
Description	Set **Source** to **Screen**
Due Date	Notice the two fields to the right 1. Click in the left field to see what displays 2. Set **Source** to **Screen**
Add Input	Click *Status* within the picklist

Field	Action
Status	1. Set **Source** to **Constant**. 2. Select "Not Started" in the picklist to the right of **Constant**.
Add Input	Click *Priority* within the pickllst
Priority	1. Set **Source** to **Constant**. 2. In the picklist to the right of **Constant** , select *Normal*
Subject	1. Set **Source** to **Constant**. 2. Type "Follow-up with " in the text box to the right of **Constant**. 3. A small icon appears after the text field; click it Here's what the *Input tab* should look like: While a **Constant** field is written directly into a record, it can also include the text of other fields.

Field	Action
	Add two **Contact** fields: *First Name* and *Last Name*. You will need to use your End key to see what you're doing when you insert the second field. Here's what the top of the *Input tab* should look like:

Select the *Input Screen* tab, and make the canvas look as follows:

When the user sees a step, it should have context information. For example, in this Service Call step, the user should see the name of the person who the person receiving this task will be contacting. You can do this by inserting the contact's name into the canvas. Do this by Selecting *First Name* and *Last Name* within the **Insert Read-Only Field** picklist.
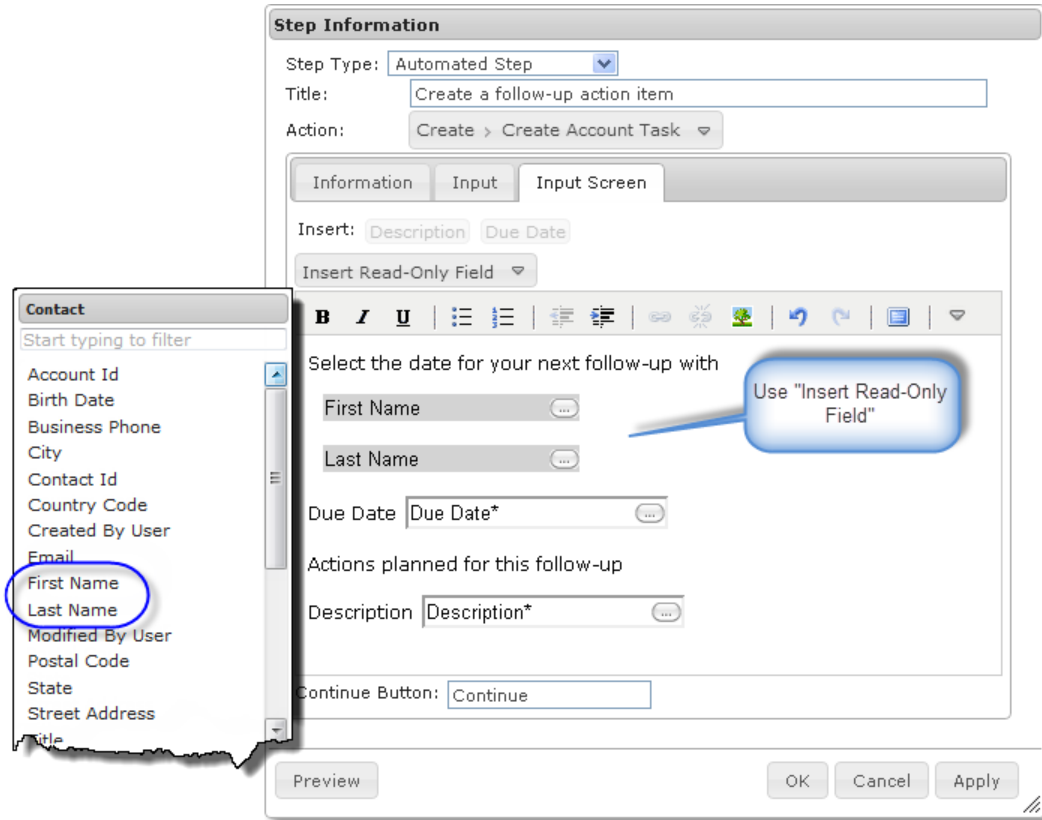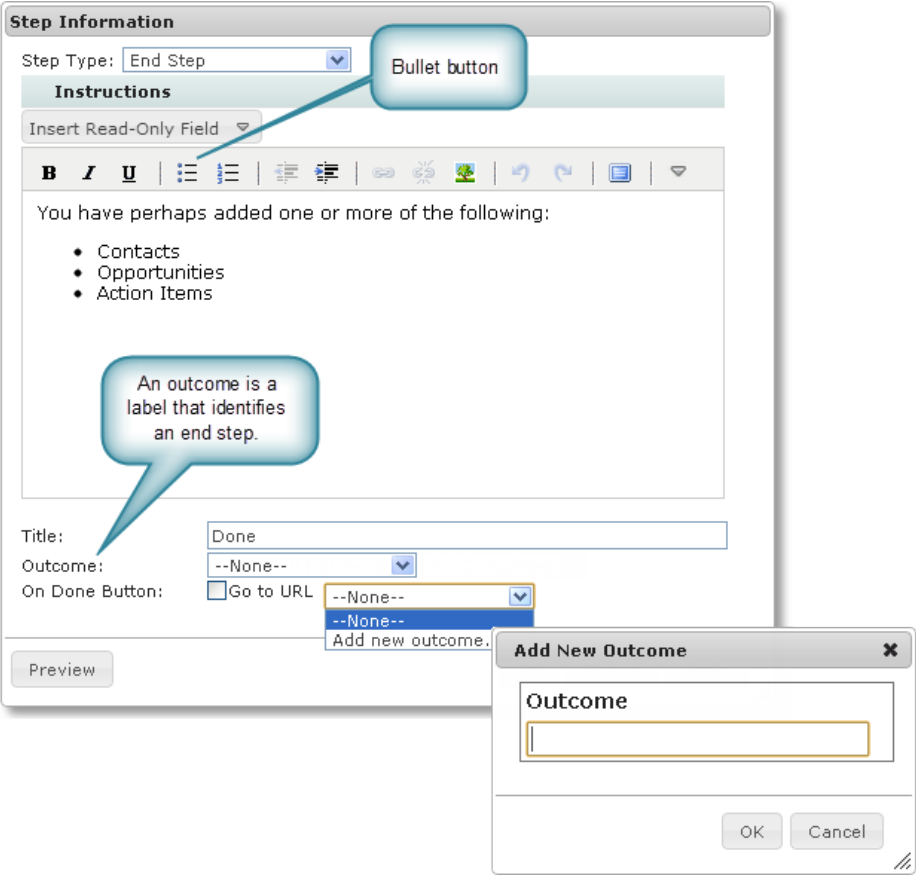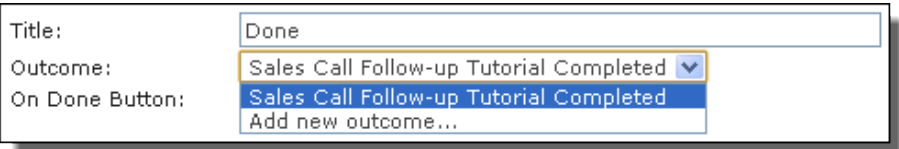
Part 7: Creating an End Step

This topic shows how to create an End Step.
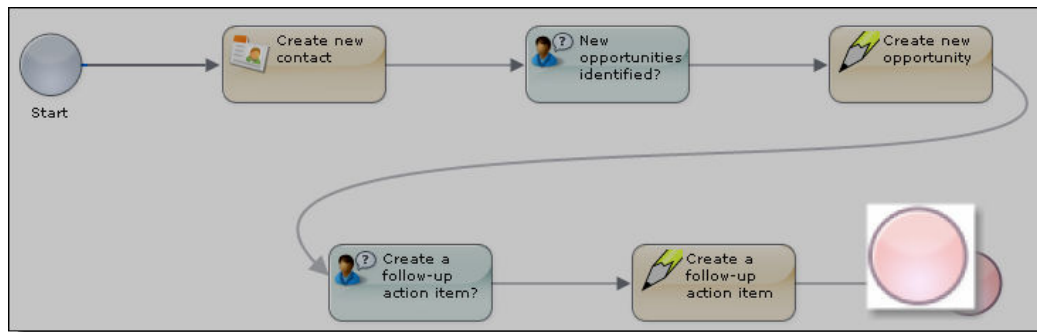
Field	Action
-	After you created the *Create a follow-up action item* step, the Guide Designer added an untitled step (as usual). Click on this untitled step

Action	Select *End Step*
Canvas	Type the text shown in the next figure. You will need to use the bullet button on the toolbar
Title	Type "Done"
Outcome	Click on the picklist. Within the picklist, select *Add new outcome*. The Guide Designer displays its **Add New Outcome** dialog.
Outcome	Type "Sales Call Follow-up Tutorial Completed" in the Add New Outcome dialog, then click **OK**. After you add an outcome, it will appear in the Outcomes picklist.

Guides can have multiple outcomes and an outcome could be associated with more than one end step. As a guide executes, Guide Designer monitors the guide's progress, recording the steps that the user executes. When the guide finishes, it records the label of the end step that was executed (The guide being built here only has one outcome: Sales Call Follow-up Tutorial Completed. Most guides have many.) When an administrator asks for a report, the outcome indicates how the interaction ended.

While you can interactively add outcomes, you can also enter them into **Guide Properties** dialog. If you add an outcome in an end step, the Guide Designer also enters it within the Guide Properties dialog.
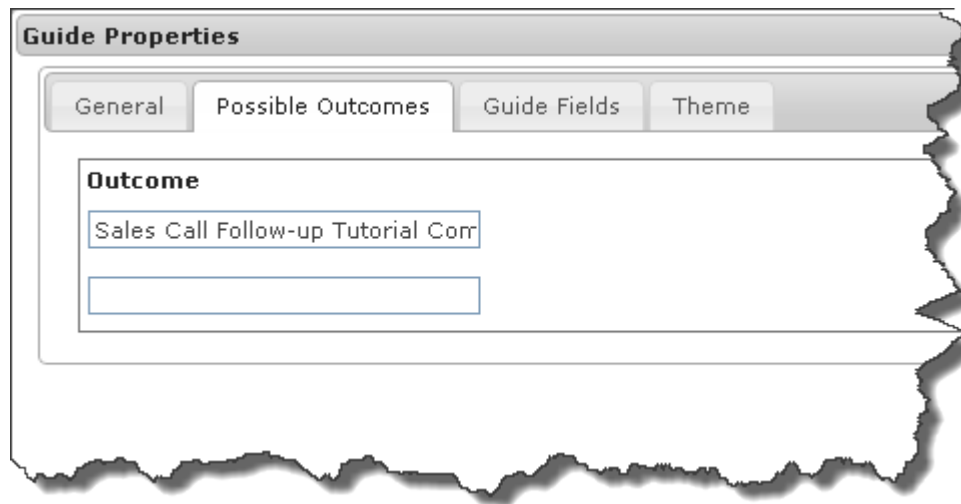
Exercises

-
-

Part 8: Adding a Jump State

If you recall the diagram of the whole guide, there were places that jumped forward and others that jumped backward. As you will see, creating a jump step is very easy.

In the following figure, the step in the upper-right corner is a *Jump Step*. Notice the curved arrow icon.
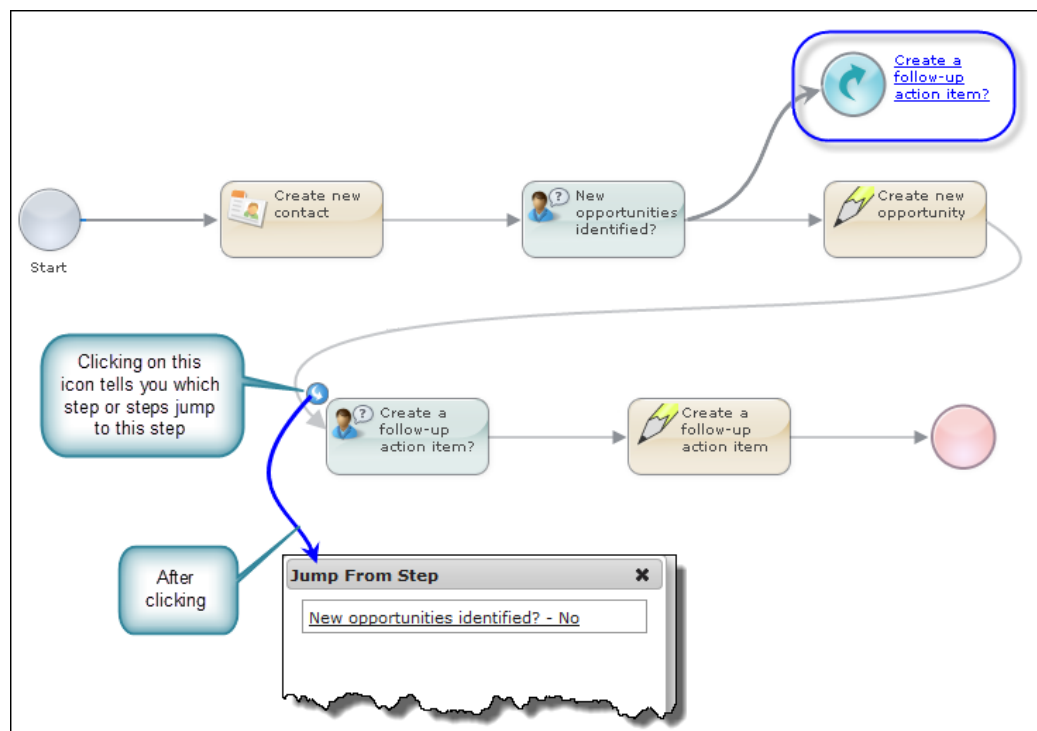
This step will jump from New opportunities identified step to the Create a follow-up action item? screen step. Notice the small curved arrow icon pointing into this *Screen Step*. This lets you know that the step is the target of a jump set.

Here's how to add it:

1. Click the untitled step that is ends the *No* branch emerging from the *New opportunity identified* screen step.

2. Click on the untitled step.

3. **Action**: Jump Step.

4. **To**: Select *Create a follow-up action item?* from the picklist.

5. Click **OK**

When users execute the *New Opportunities identified* screen step, they have two choices: *No* and *Yes*. If the user selects *No*, the next step that is displayed is *Create a follow-up action item?*

Exercises

- "Exercise 6: Adding the Create a follow-up action item step" on page 151
- "Exercise 7: Adding the Assign someone a follow-up action item" on page 152
- "Exercise 8: Adding Additional Follow-up Actions Items" on page 153

Part 9: Adding the Find existing contacts Service Call Step

After a meeting, users should validate that the information for their contacts is correct. To do this, they will need to select the contact from a list. This topic introduces the following concepts:

- Creating an output field using the **Guide Properties** dialog
- Defining the output field
- Using a Set Fields action within a Service Call step
- Creating the list

Creating an Output Field

Guides read, create, and update your application's data. This information is stored in fields, and these are the items that you see within the **Insert Read-Only Field** and **Insert Field for Update** picklists. These fields are defined within the application. However, there are times when a guide needs to temporarily store information. To do this, you will need to add fields, fields that are only used within the guide. The place where you add them is the *Guide Fields tab* within the **Guide Properties** dialog.
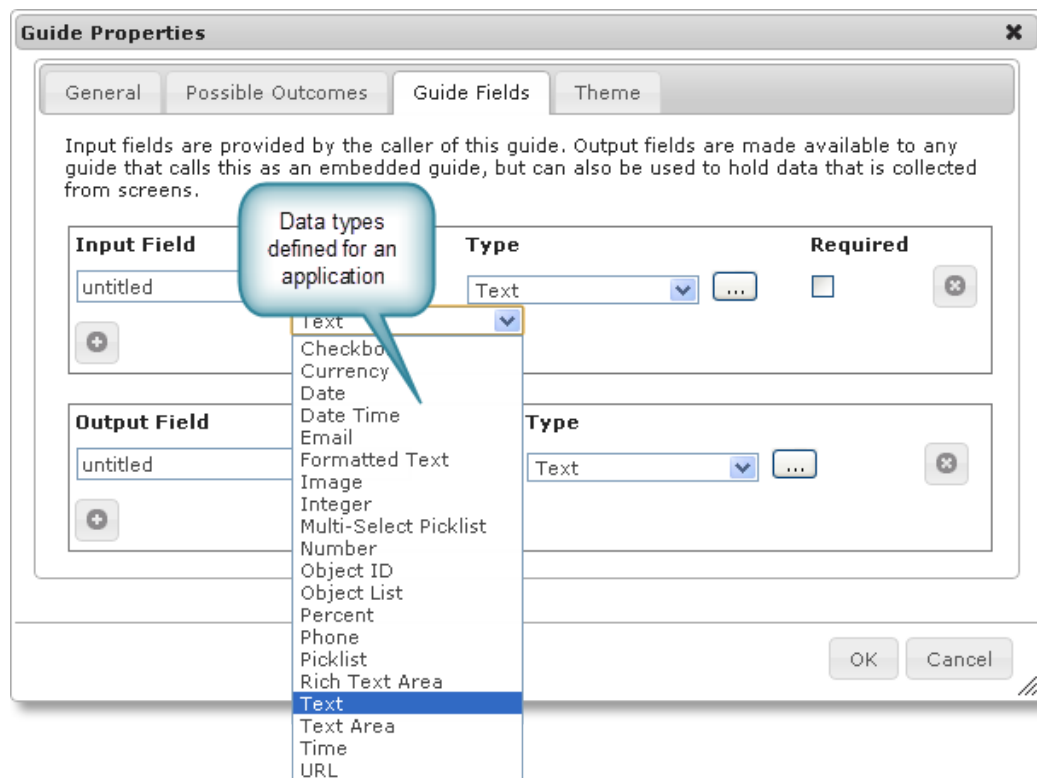
You can create input and output fields.

- **Input Field**: Used to pass data to an embedded guide. This is not discussed in this tutorial.
- **Output Field**: Used in two ways. The field can create a field into which you can assign data. The second is to receive data from an embedded guide. (This also is not discussed in this tutorial.)

All fields have a data type. Guide Designer has the built-in data types shown in the picklist in the previous figure. Developers can add additional data types.

Some of these data types are ones used in most programming languages; for example, text and numbers. Others such as checkbox and picklist are elements that are used in interactive forms. Object ID and Object List are unique to Guide Designer. An object ID is a built-in type into which Guide Designer can write data that

identifies an object. Object IDs can be passed to or returned from Service Call steps and guides as fields. The guide can also be started with an object ID association, which is set using **Applies To** in the **Guide Properties** dialog.

An Object List is also a built-in type. It is a multi-valued data type where each item identifies one object ID. This list can be operated on as a set and the guide can individually execute on each element of the list or displayed as a table to the user.

This guide needs to display a list of contacts, and this information will be stored in an Object List.

1. Select the **Guide Properties** dialog and click the *Guide Fields tab*

2. Press the "+" symbol within the Output Field area

3. In the left-most area, type "Contact List"

4. Select *Object List* from the Type picklist

Defining an Output Field

All fields, both those created and those contained within an object, have settable properties. Normally, you just ignore them. However. you need to set properties when defining a list. Here is what you do to define this Contact List field:

1. Click the "..." button to the right of the Object List item selected previously. The Guide Designer displays the following dialog:
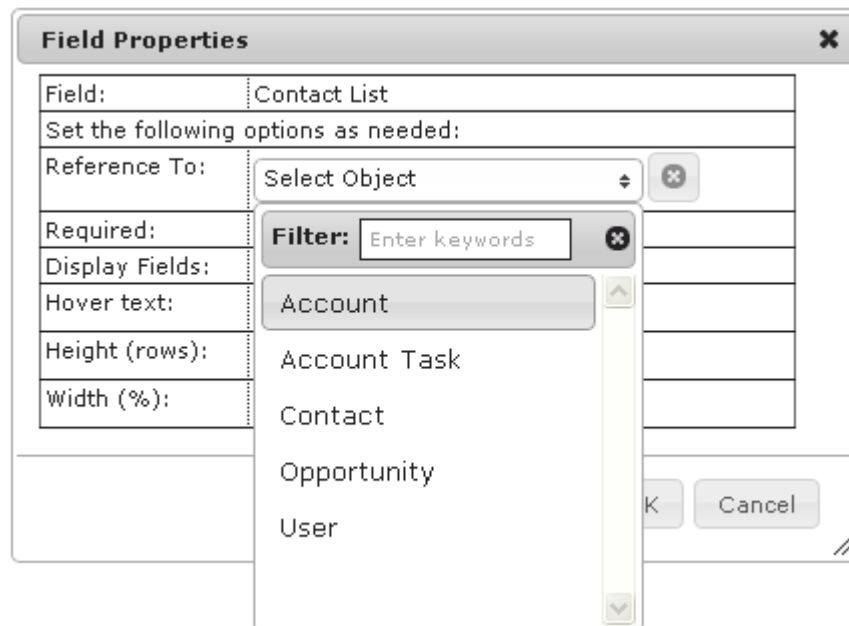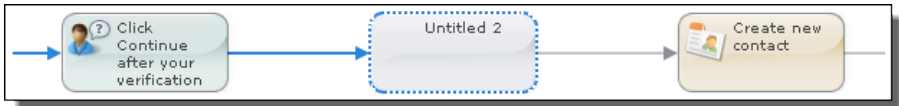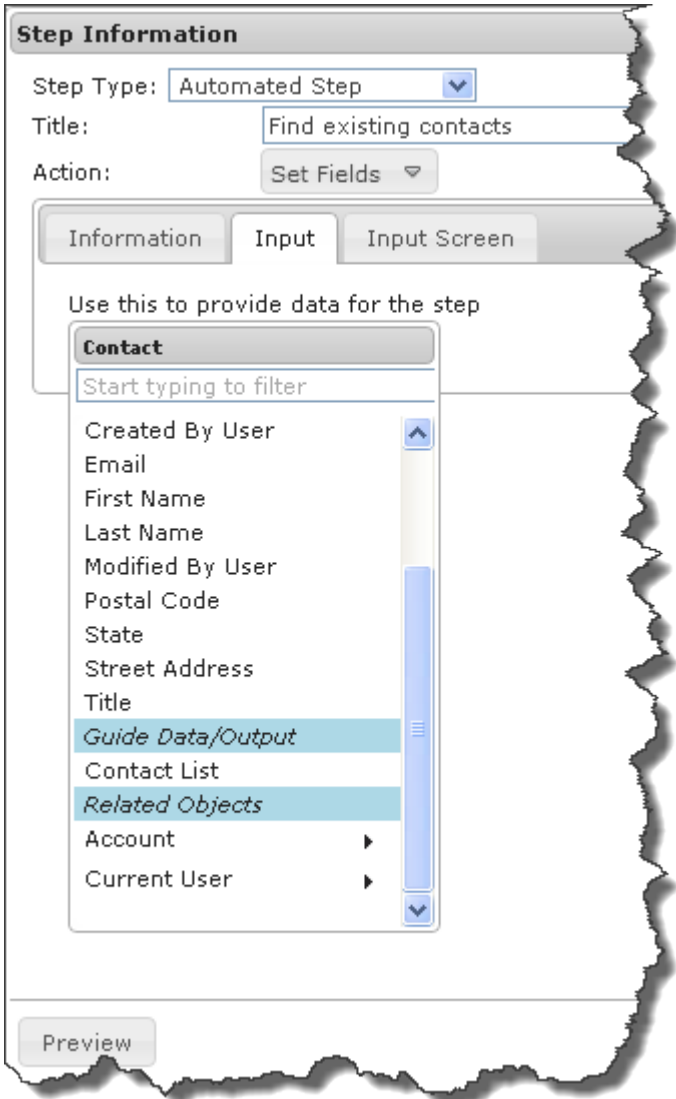
2. Take a moment to look at the properties you can set. Only two need be set for this guide. Set the **Reference To** property to *Contact* and set the **Height** property to 6. The Reference To property makes fields within that object available. Height indicates how many items are displayed in a step. If there are more than 6, Guide Designer adds a scroll bar.

3. Click **OK** in this dialog.

4. Click **OK** in the Guide**Properties** dialog.

Using a Set Fields Action Within an Service Call Step

In the Guide Fields tab of the **Guide Properties** dialog, you defined the Contact List field. You now will define the items that will appear in it.

Here's how to set up a Service Call step.

Field	Action
-	Select the *Create new contact* step, and then click the **Insert** button.
Step Type	Select *Service Call Step* from the picklist
Title	Type "Find existing contacts"

Field	Action
Action	Select **Set Fields** from the picklist
Add Input	Select *Contact List* from the picklist

You will need to scroll the list. Notice that the Guide Designer has added a *Guide Data/Output* section and placed *Contact List* within it.

Creating a List

In other topics, source values could be **Field**, **Constant**, and **Screen**. In this context, the picklist contains Field, List, and Screen.
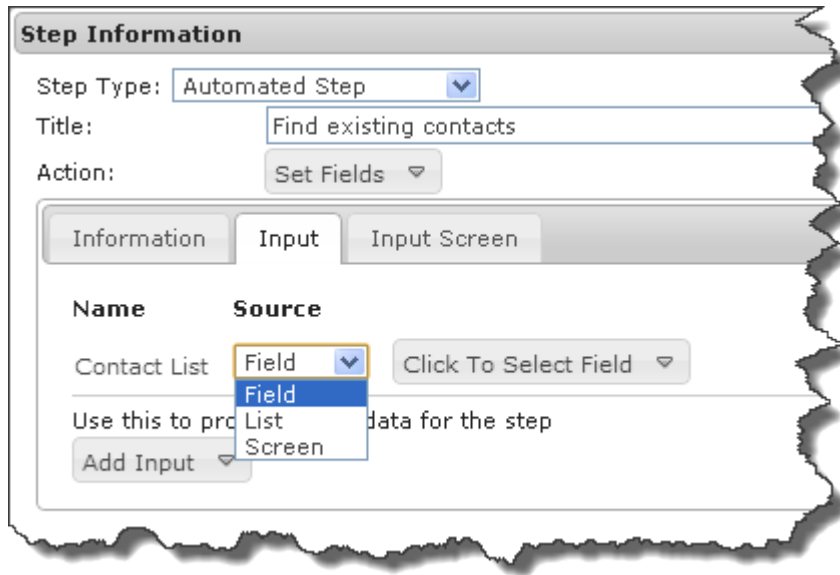
Here's how the step appears after you select **List**.
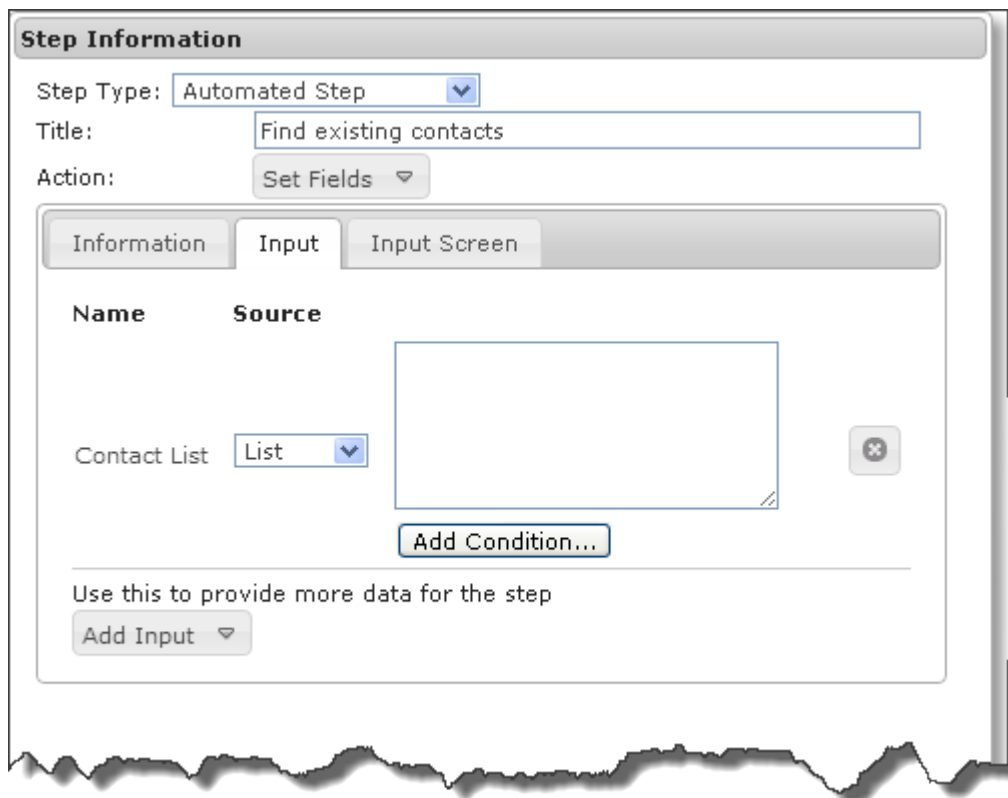
What you will need to do now is add one or more conditions that select contacts contained within your application. You could directly type an SQL WHERE clause into this text area. However, the Guide Designer can give you considerable help if you use the **Add Condition** dialog. Display it by clicking the dialog's button. The following composite figure shows the contents of each of the picklists within this dialog:
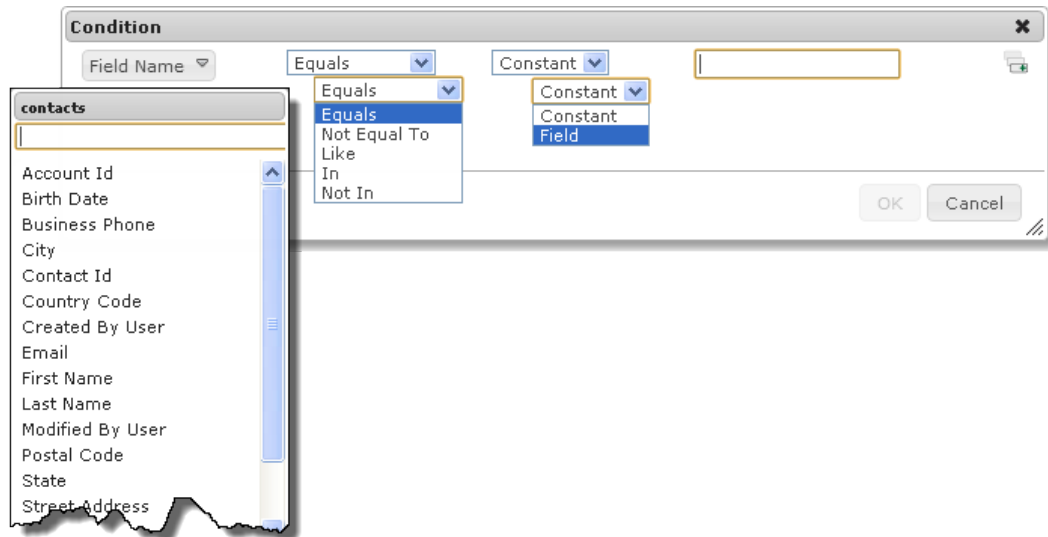
As every contact has an Account Id field, you can obtain a list of contacts associated with a contact by using the Account Id field. Here is how you set this dialog.
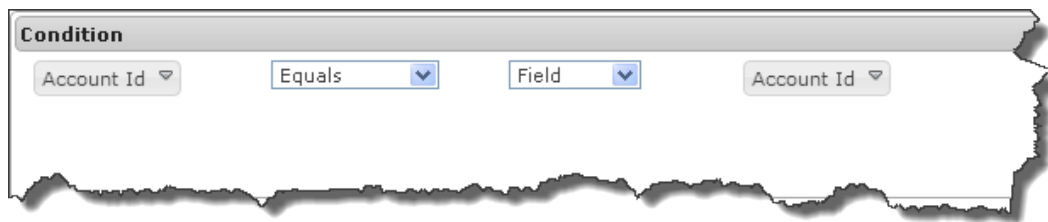
After clicking OK, the Step Information dialog looks as follows:
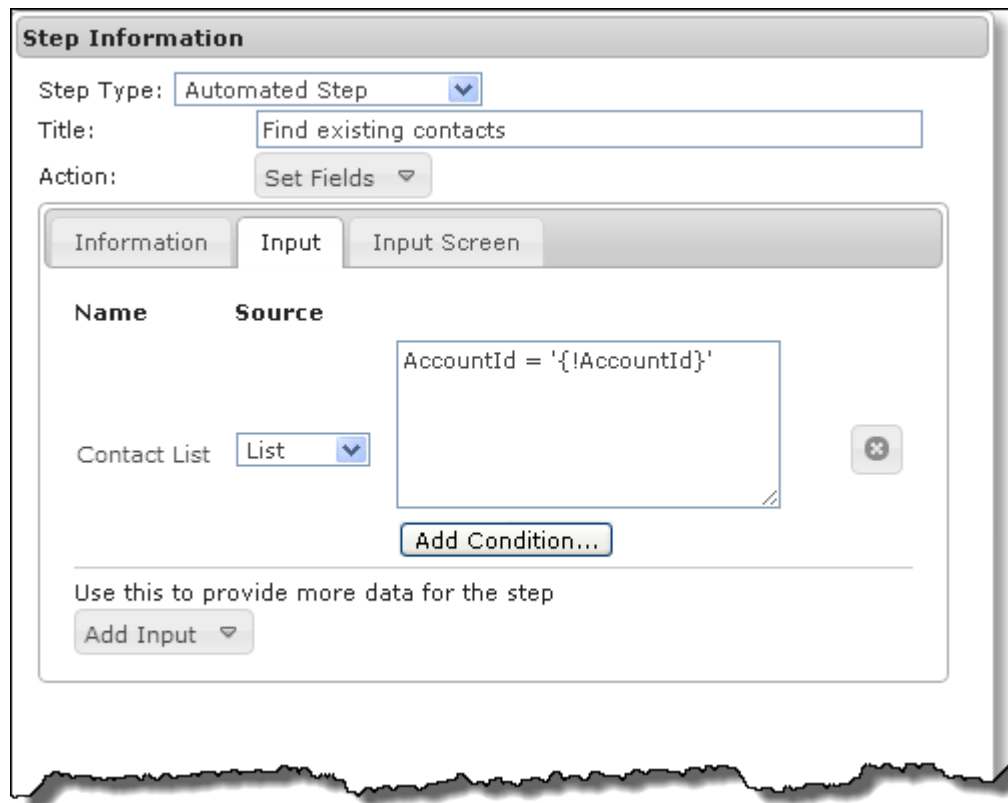
Lesson 2: Simulating Your Guide

You've already seen how Guide Designer can preview what a guide looks like. It also allows you to run a simulation of your guide to see both how it looks to your users and test what happens when you choose different answers.

1. Click **First** to go to the beginning of the guide, then click **Simulate**.

2. The first simulated screen that you designed appears. The actual data for First and Last name, Email, and Business Phone do not appear in Simulation mode, but they will appear when the user runs the guide.
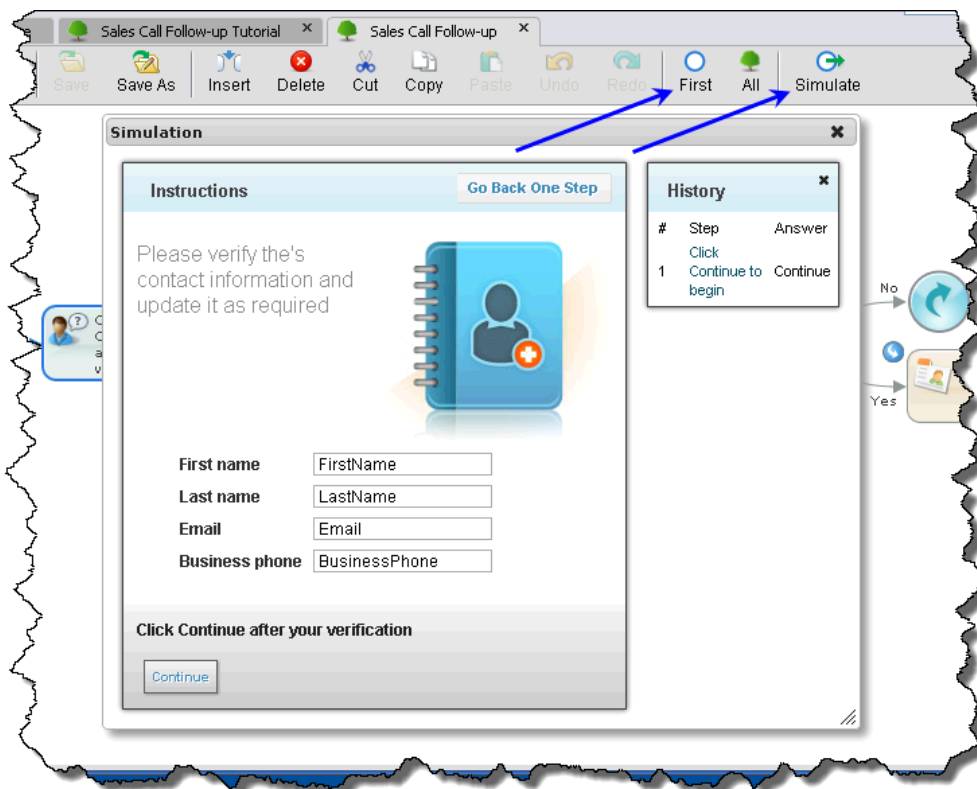
3. Click the **Continue** button. When you come to a step that has a Yes and No buttons, clicking on one of the buttons sends you down that path.

If you type the button at the bottom of the Service Call step, anything that you may enter is not saved since this is a simulation. Similarly, no records are created or any kind of data updated.

You can continue the simulation on your own by clicking through the screens and Service Call steps that you built. Notice that during simulation, you can always go back to a previous step by clicking the **Go Back One Step** button. Your user can also see the full History of the screen flow, and go back to any point in it by clicking on the step in the History. Try using **Go Back One Step** and going back using the History.
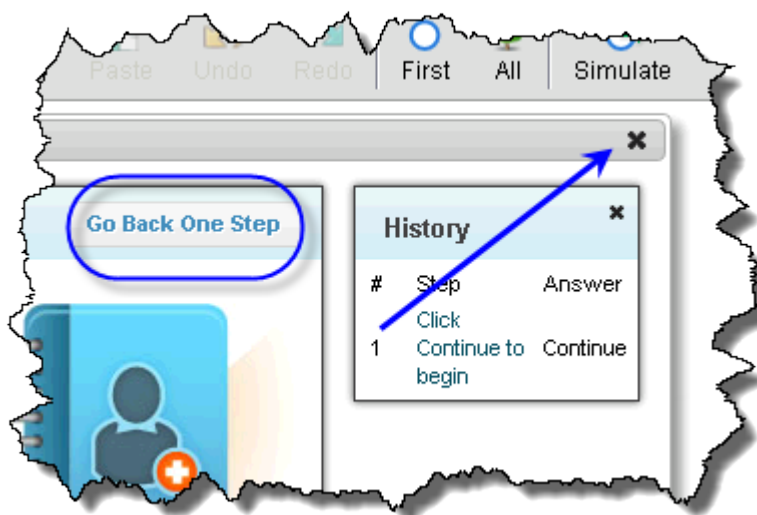
Note: When a user presses the Go Back One Step, Guide Designer can undo the action performed by a step. (This is application specific.) For example, if a Contact was created, the Contact is removed.

Exit simulation mode by clicking the **X** in the top right corner. (Clicking on the *X* within the History panel simply removes the panel from the display.)

Lesson 3: Saving and Publishing Your Guide

You can save your guide at any point by clicking the Save button, and you should have been doing this as you created exercises. Now it's time to publish the guide to your organization.

1. With your guide saved, select the Home tab in the Guide Designer.
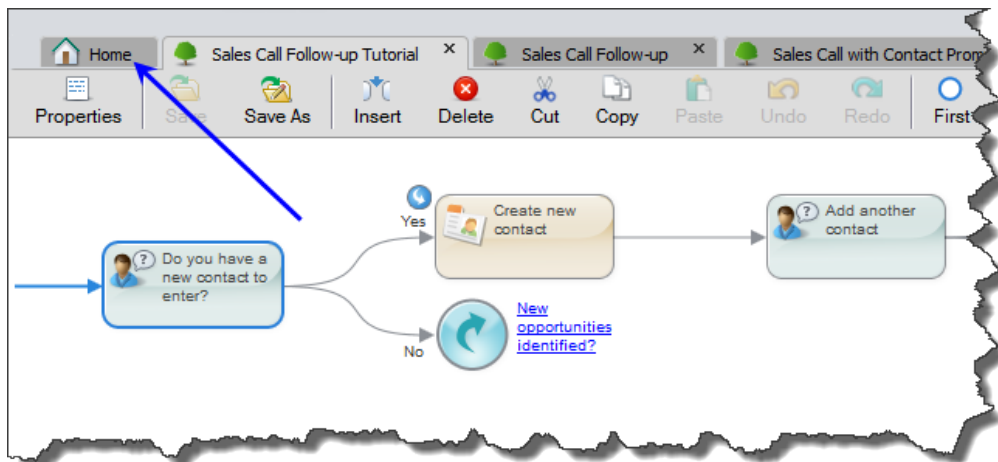
2. The Home tab shows all guides you've imported and created. You will notice that your Sales Call Follow-Up Tutorial guide is shown as unpublished, whereas you may have published other guides, as is indicated by a green check mark.
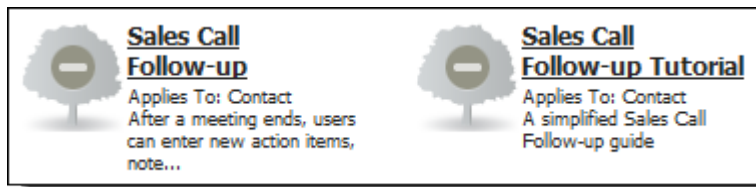
3. To make Sales Call Follow-Up Tutorial available to your users, select it, and then press the **Publish** button.
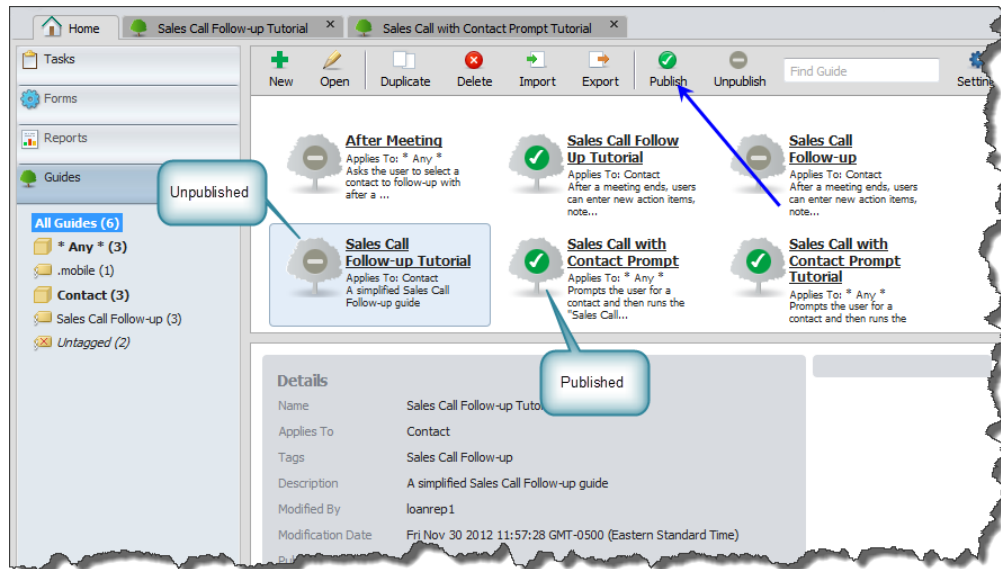
That's all there is to publishing.

Tip: You can make changes to your guide any time and re-publish it. The Details panel always shows you when a guide is out-of-date or current. (An out-of-date guide is one with modifications that have not been published.)

Notice the **Details** panel underneath the Guides area. It shows information about your guide's status.
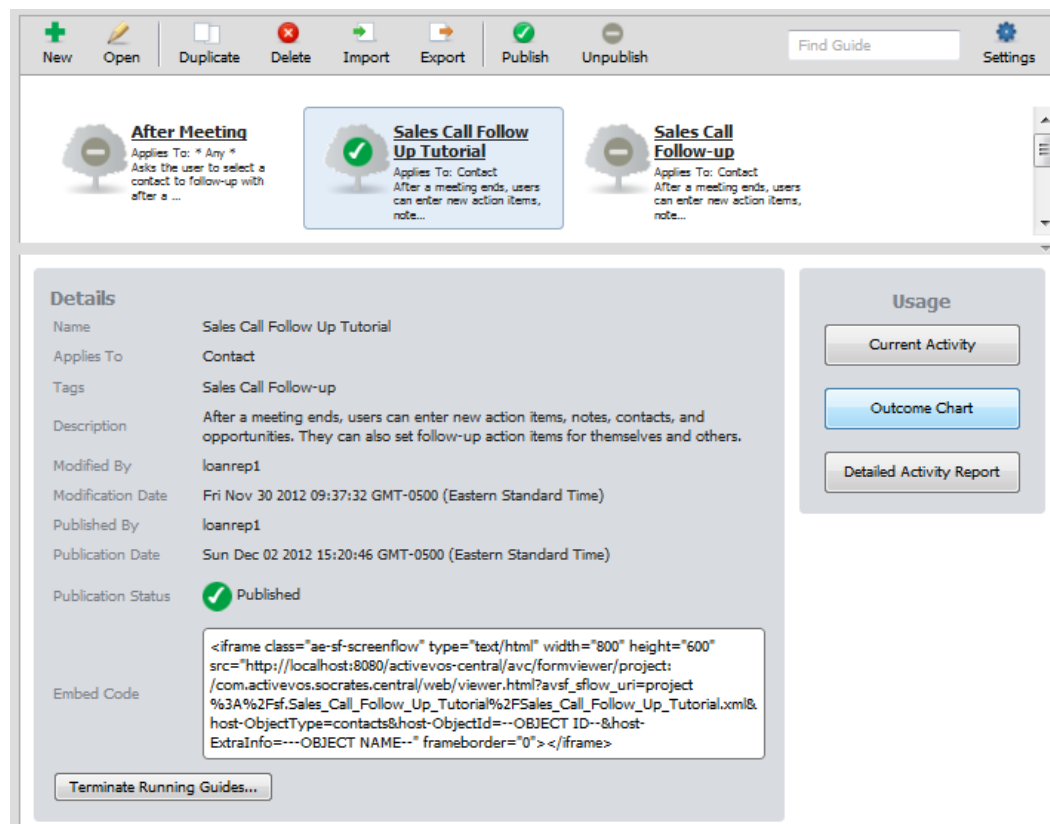
Lesson 4: Running Your Guide

Now that you have published your guide, it's time to run it. There are two possibilities:

1. Select the URL within the Run Guide area. If running directly from a link, you will need to give it to all of your users.

2. Place the URL within an IFrame, and then embed the code for this within your application.
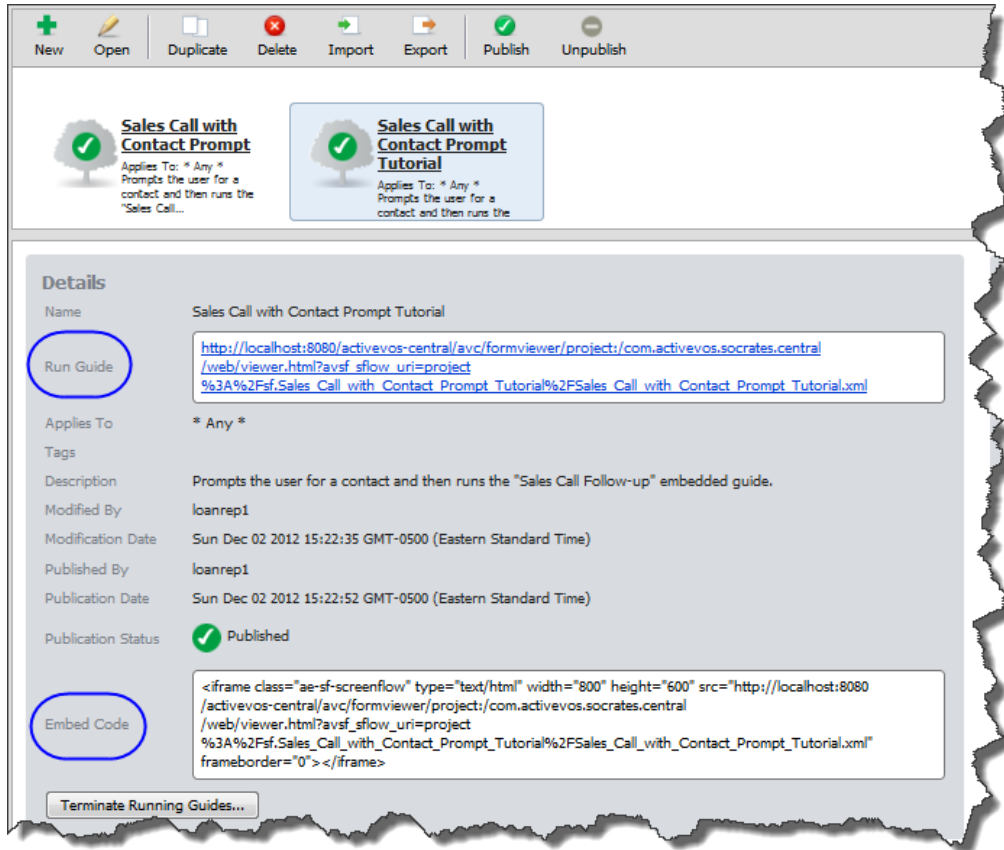
Note: One of the reasons for embedding one guide within another and having the embedding guide's Applies To set to *** Any *** is that it can be run directly. This is useful for testing. Notice also that the guides you create do not automatically appear in the Forms tab on the left.

After clicking on the link, you'll see a a guide that lets you contact. You can display a list of contacts by selecting the push pin. Guide Designer then displays a lookup dialog. After you type text and press Go, Guide Designer display contacts that match what you type. You can type characters within a name, and Screnflow can still find names. For example, "ll" matches "Jill".
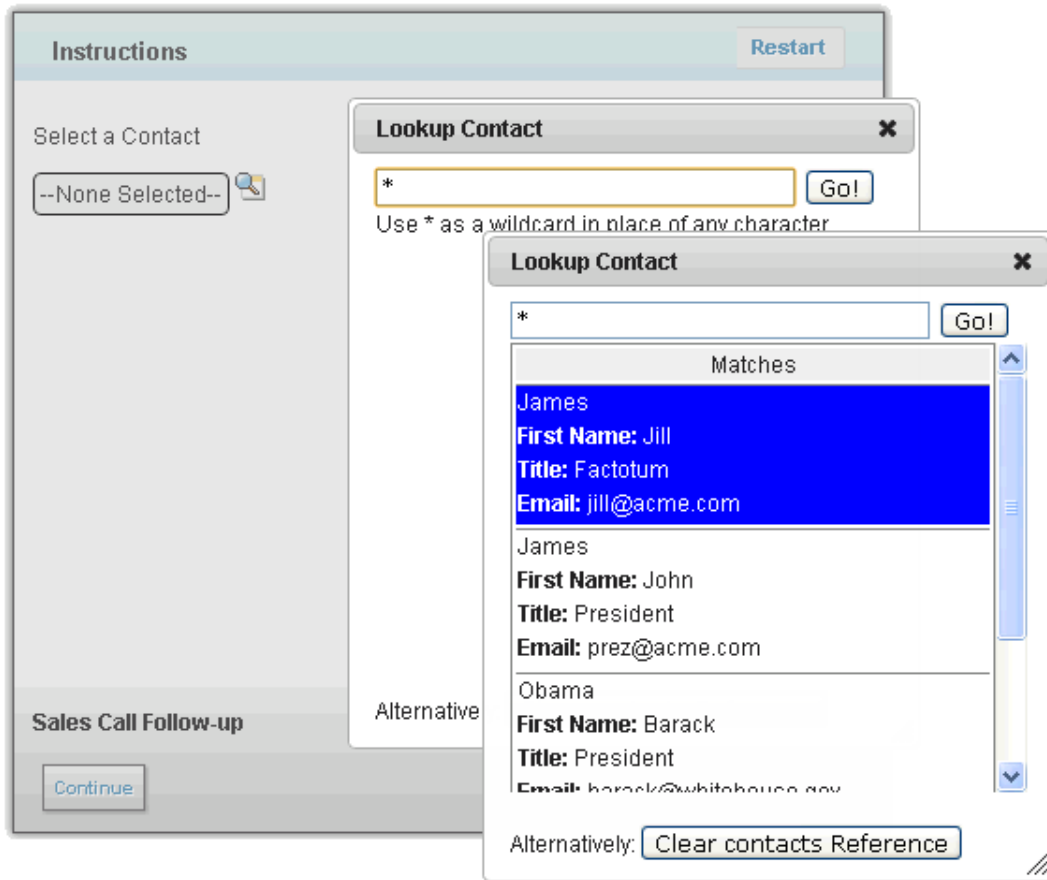
After selecting a contact, here's what you will see:

Here are a few more screens that will be displayed:
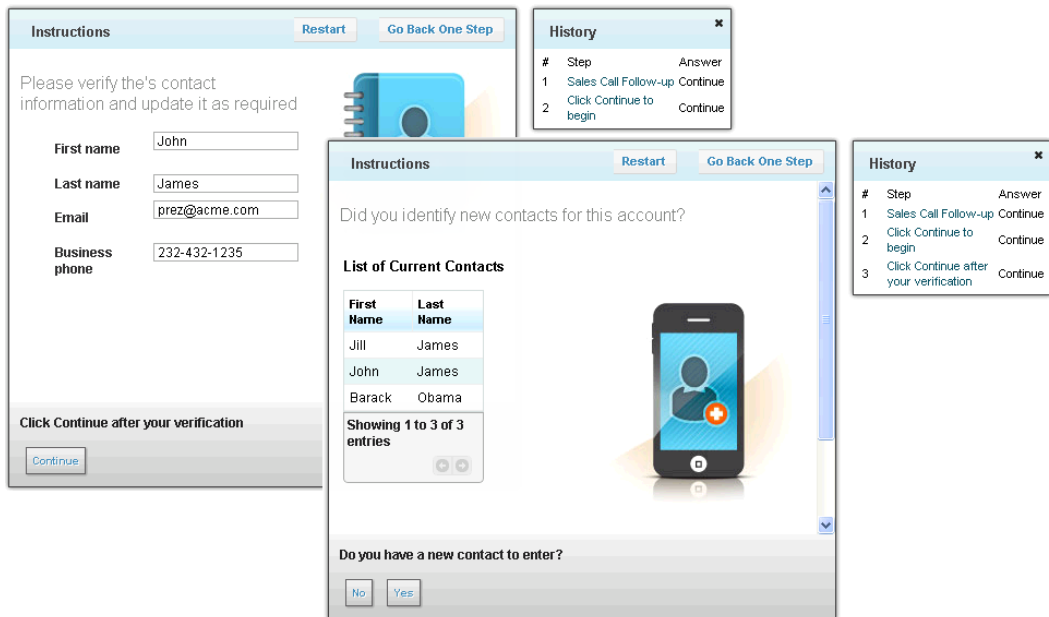
Now that you have your guide up and running, go through it, checking out the different paths through it, seeing what happens, sometimes not adding information, and at other times adding one or more contacts, opportunities, and follow-up tasks. You should also change a contact's information and check that it was changed.

Tip: To prepare for the final section of this tutorial, View Reports, run this guide several times on some of your leads. If possible, if other guides are available, run this guide as well. By executing your new guide several times, you will generate data that Guide Designer can use when it creates reports.

Exercises

The exercises in this part are used with Lesson 1 of this tutorial.

Exercise 1: Adding the Click continue to begin Screen Step

While many guides begin with a Service Call step, many more begin with a *Screen Step* that explains what the guide does. In the exercise, you will add a screen step before the *Create new contact* Service Call step.

The way in which you insert a step before another step is to select the step (here it is *Create new contact*) and then press the Insert button on the menu bar.
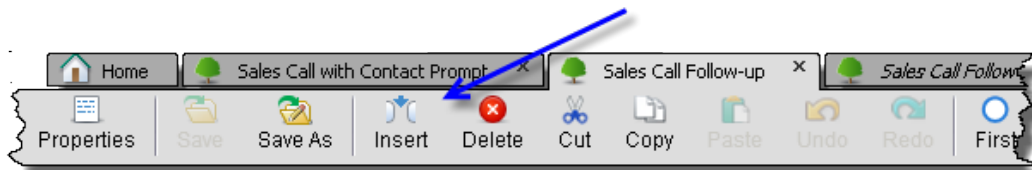

Now create a screen step that looks like this:

Note: For information on using this toolbar, see *Using the Guide Designer Editor* elsewhere in this help.

The beginning of your guide should look like this after you insert this screen step:
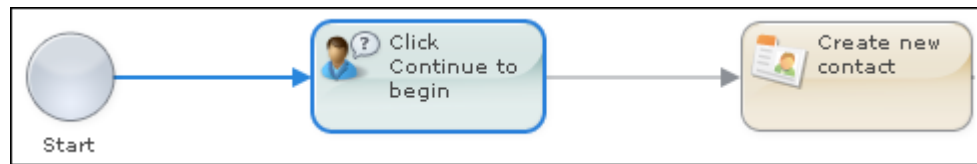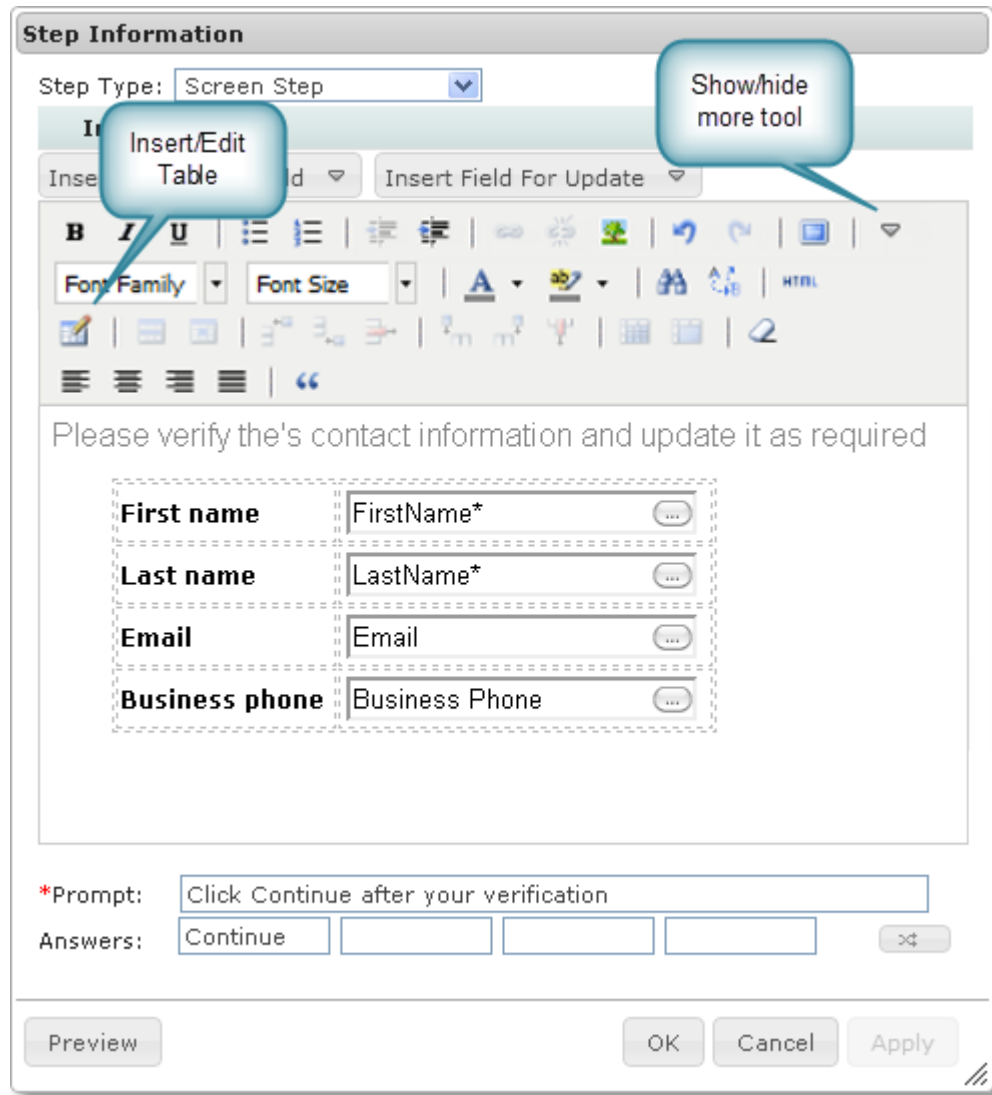

Exercise 2: Adding the Click continue after your verification Step

Place this screen step immediately before the *Create new contact* step. Do this by selecting this step and then clicking the **Insert** button.
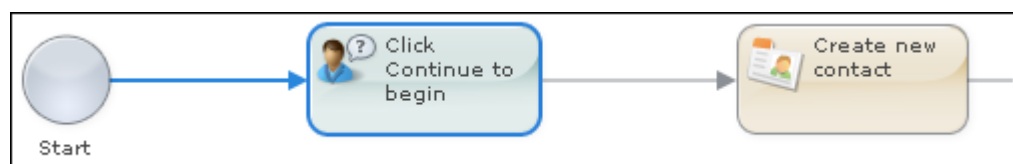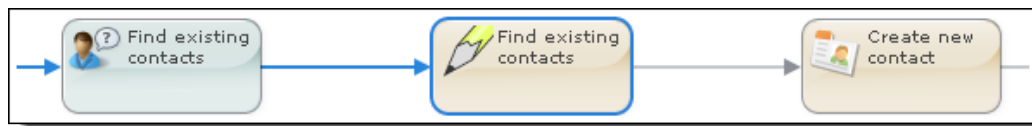

A new untitled step is created. Click on it and create a screen step that resembles the following figure. Here are some hints:

- Click the **Show/hide more tools** button. Three additional rows of buttons appear.

- Click the **Insert/Edit Table** button and inset a table that has two columns and four rows.

- Use the **Insert Field For Update** picklist to insert the four fields.
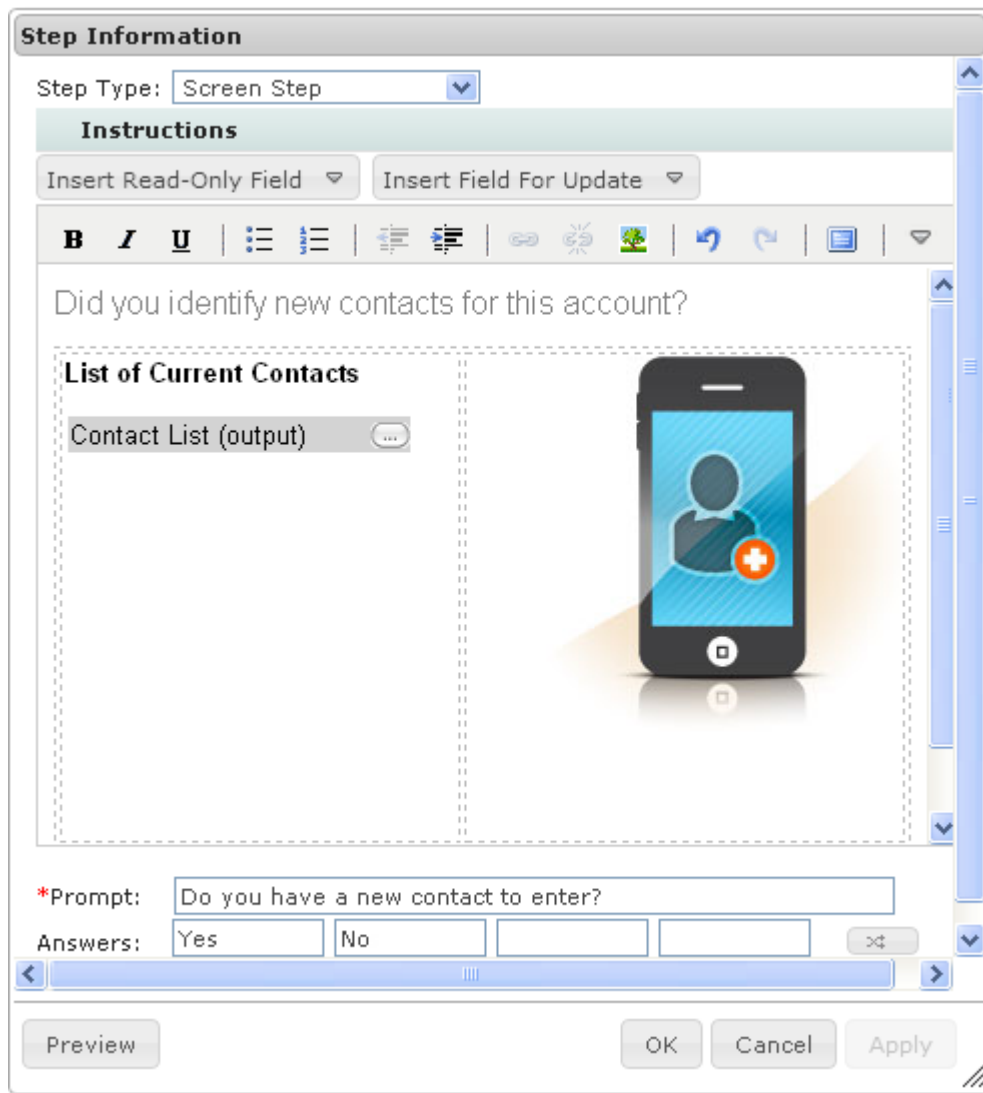
Exercise 3: Adding the Do you have a new contact to Enter? Step

Place this screen step immediately before the *Create new contact* step. Do this by selecting this step and then clicking the **Insert** button.
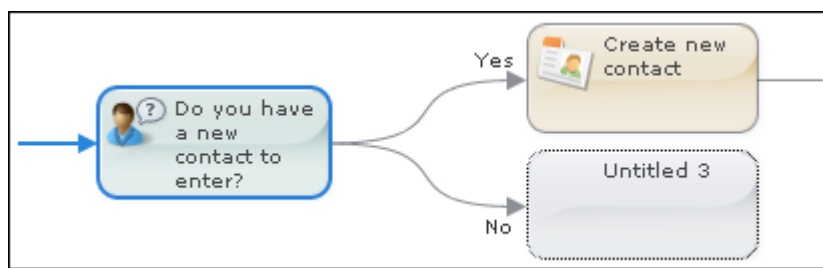

A new untitled step is created. Click on it and create a screen step that resembles the following figure. Here are some hints:

- Use a table to position elements.
- Use the **Insert Read-only Field** picklist to insert the *Contact List* field. The Guide Designer displays this field in gray and tells you that it is an output field.
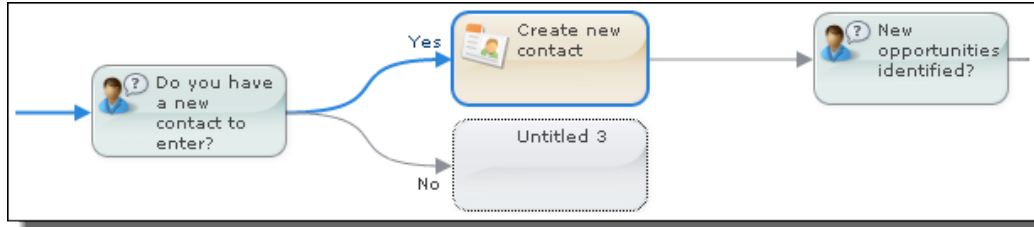
Step Information

Step Type: Screen Step

Instructions

Insert Read-Only Field ▽   Insert Field For Update ▽

B *I* U | ☰ ☰ | ☰ ☰ | ⚭ ⚯ | 🖼 | ↺ ↻ | ▣ | ▽

Did you identify new contacts for this account?

List of Current Contacts

Contact List (output)   (...)

*Prompt: Do you have a new contact to enter?

Answers: Yes   No   [ ]   [ ]   ⤭

Preview   OK   Cancel   Apply

It is extremely important that you type the answers in the order shown in this dialog. When you insert a step that will have branches before another step, the Guide Designer sets the step immediately after this one as the top branch.

Yes → Create new contact

? Do you have a new contact to enter?

No → Untitled 3

If you had entered them in a different order, the No branch would point to the *Create new contact* step.

Exercise 4 Do you want to add another contact Screen Step

Place this *Screen Step* immediately before the *New opportunities identified* step. Do this by selecting this step and then click the **Insert** button.

A new untitled step is created. Click on it and create a screen step that resembles the following figure. Again, it's important to type the answers in the order shown here.

Exercise 5: Adding the Add another opportunity Screen Step

A new untitled step is created. Click on it and create a screen step that resembles the following figure. Again, it's important to type the answers in the order shown here.


Exercise 6: Adding the Create a follow-up action item step

Place this screen step immediately before the *Done* End Step. Do this by selecting this step and then clicking the **Insert** button. As before, pay attention to the order in which the answers are entered.
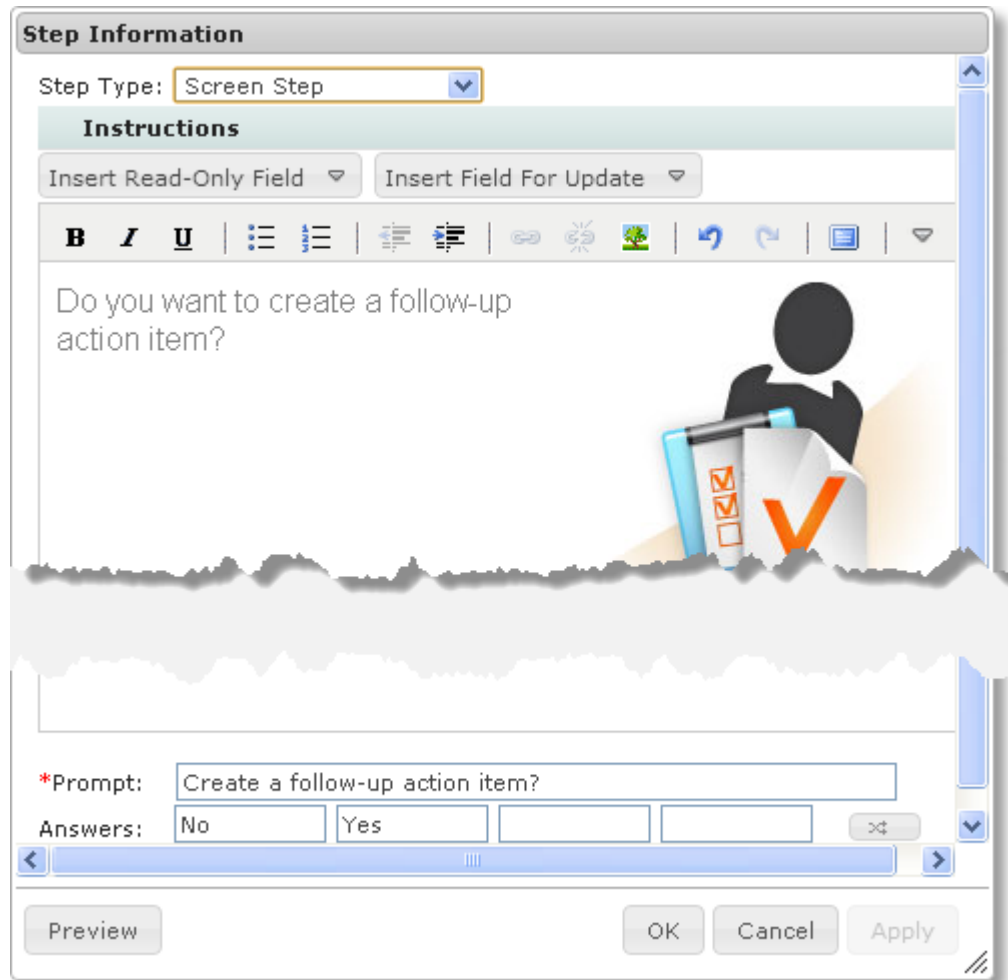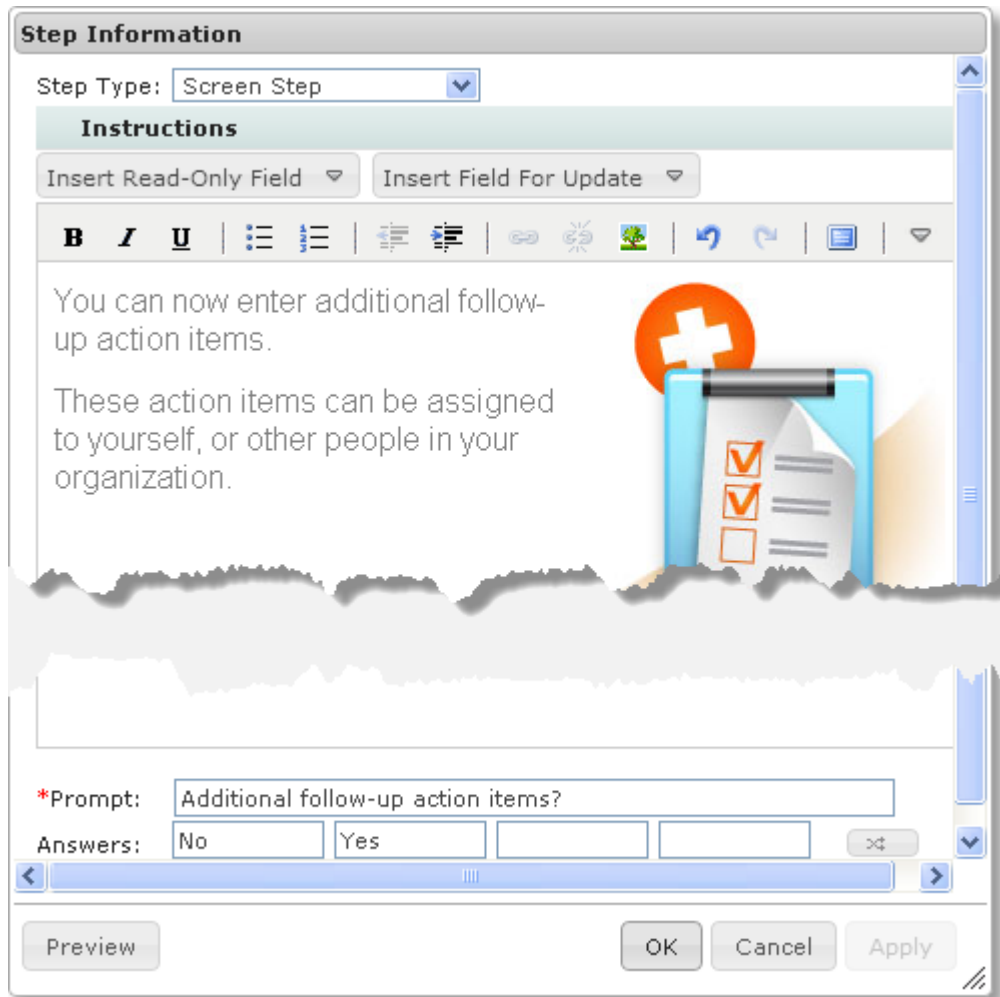
Exercise 7: Adding the Assign someone a follow-up action item

Select the newly created untitled step and click on it. Create a screen step that looks as follows:
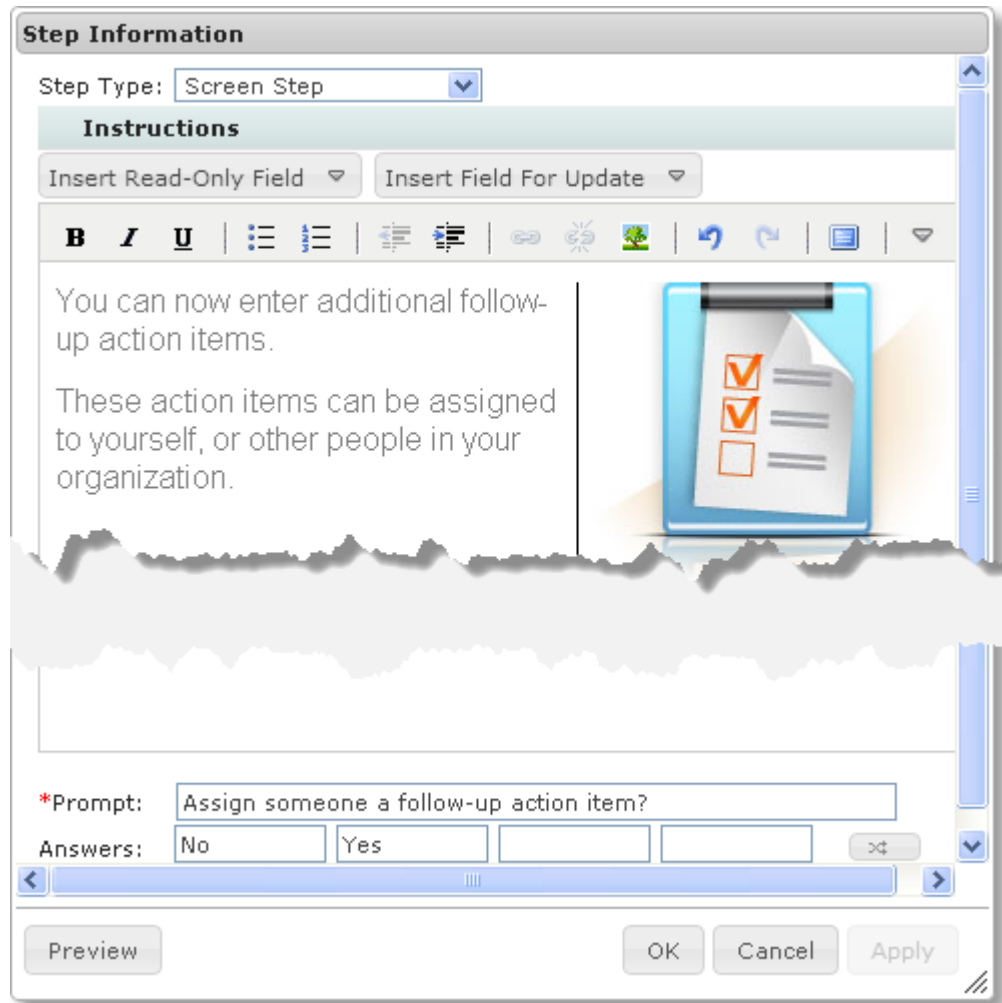
Exercise 8: Adding Additional Follow-up Actions Items

Now for the last screen step. Select the untitled step created by the previous exercise, and edit it to look like the screen step in the following figure.

The *No* branch will be an End Step. Rather than directly entering it, select the End Step crated a few exercises ago, select the **Copy** button in the toolbar, select the untitled step at the end of the *No* branch, then select the **Paste** button.

The Designer lets you copy (or cut) steps and entire trees.

C H A P T E R 4

For Administrators

Administrative topics in this section are:

Adding a Role Name to Identity Service

Your administrator must add the role name `abTaskClient` to your identity service and add that role for you.

Showing and Hiding Process Central Components

You can show or hide the following components of Process Central:

- Process Central header (includes logo, Help, Settings, Search, Log Out)
- Tasks
- Requests
- Reports
- Guides

Show or hide these components using URL parameters. Hare are three examples:

- **Show Only Tasks**

  ```
  http://localhost:8080/activevos-central/avc/avc.jsp?aemgrs=tasks
  ```
- **Show only Tasks and Requests**

  ```
  http://localhost:8080/activevos-central/avc/avc.jsp?aemgrs=tasks,requests
  ```
- **Hide Process Central header**

  ```
  http://localhost:8080/activevos-central/avc/avc.jsp?aeembed=true
  ```

URL parameters include:

- tasks
- requests
- reports

- socrates (for Guide Designer)

Setting Features in the app-config.xml File

You can set some general Guide Designer features by editing values in the `app-config.xml` file. This file is located within the `config` subdirectory in the project's workspace files. These elements are within the features element and have default values of either true or false. These are discussed in "Features" topic within the Guide Designer API text. However, here is a summary:

- **guidePermissions**: If this value is set to true, the Settings dialog will have a Permissions tab that lets guide designers set who can access the guides within a project.
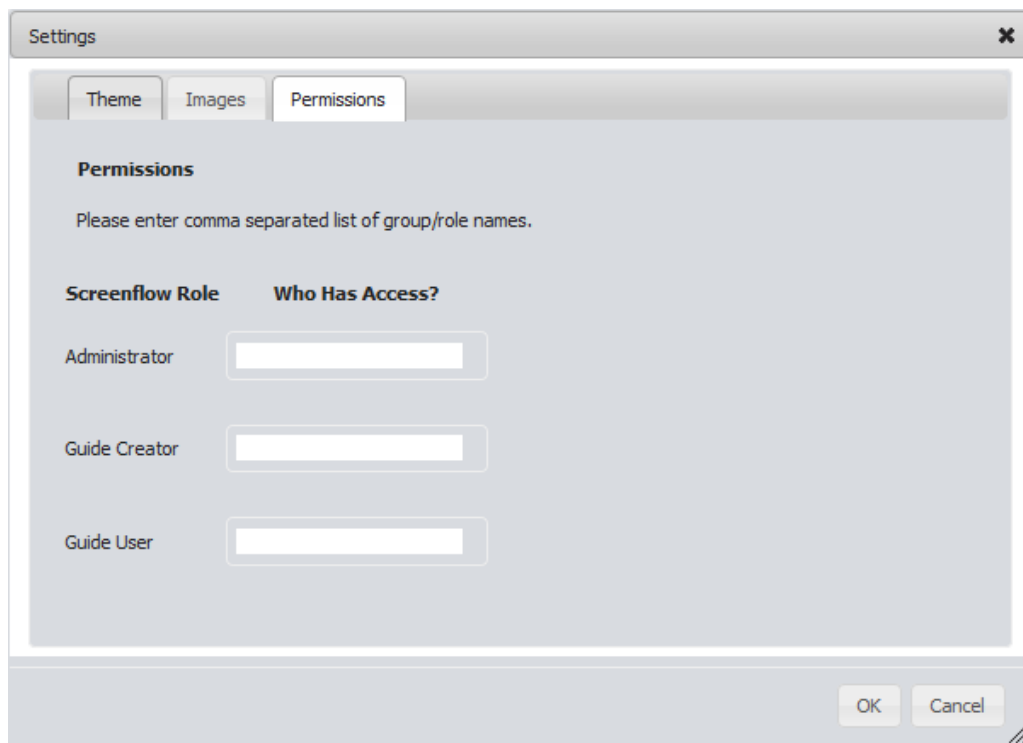
Use this dialog to set the groups and role names that will have access to this project's guides. The default is false.

- **handleRedirectOnDone**: If enabled, a guide can redirect to another URL when the guide concludes. The default is false.
- **polymorphism**: If enabled, the guide designer lets users choose more than one object as a reference to. The default is false.
- **redirectToObject**: When a guide reaches an End step, the behaviors that can coccur can be set. The default is false.
- **runAsUser**: If enabled and your host supports data access and modification requests, an elevated user can perform these requests. The default is false.
- **showEmbedLink**: If enabled, a guide can show HTTP link for embedding the guide. The default is true.