Creating Your First LiveCycle Application

Contents

1. Introduction

This tutorial introduces you to the end-to-end process of creating an Adobe LiveCycle application to automate a business process. A *LiveCycle application* is a solution that consists of associated assets, such as form designs, images, data schemas, SWF files, and processes. When the application is deployed on a LiveCycle server, it can be invoked as a service. Services can be invoked from Adobe LiveCycle Workspace 9, programmatically using (Java,.NET, and Flex), using web services, by email, or using a watched folder.

In this tutorial, you will create, manage, and test a LiveCycle application using these tools:

Adobe LiveCycle Workbench 9.5: To create the application, create and debug processes, and import assets into the application.

Adobe LiveCycle Designer 9: To create the form design used in the application.

Applications and Services web pages within LiveCycle Administration Console: To import and configure permissions to applications.

Adobe LiveCycle Workspace 9: To test your application to see how it appears to a user.

Summary of steps

The steps to create an end-to-end application include these tasks:

*   Create an application.
*   Create a form design and process.
*   Test a human-centric process within Workspace.

Audience

This tutorial is intended for developers who are interested in creating a LiveCycle application. Before you step through this tutorial, it is recommended that you review the following information:

*   LiveCycle Overview
*   "LiveCycle Overview" in Tour de LiveCycle

Understanding the business problem

This tutorial provides the typical steps for understanding the business problem. In a real-life scenario, you would take time to understand your business problem before you start to plan and implement it.

Plan the application details: Plan the tasks that are required to create your application by determining the data-capture requirements and understanding the steps in the business process you are automating.

*   **Determine the data capture requirements:** You can use many ways to capture information from a user. In Workspace, you can use applications built with Flex, interactive forms created in Adobe Acrobat, form designs created in Designer, or Guides. In this tutorial, you will use Designer to create a form design with a provided data schema. Typically, you create a data schema to organize the data you capture. The schema is important because adds structure to the data.

- **Identifying the business process:** Before you develop your application, identify the business process and the major steps to automate the process.

Scenario for the tutorial

Users fill forms to start automated processes. Forms are used for data-capture activities. One process Fin@nce Corporation is automating is a preliminary loan-approval process. The preliminary loan-approval process helps to streamline future loans for clients by preapproving them for set amounts. Workspace is used to fill and approve forms.

In this tutorial, a simple preliminary loan approval application form is required as part of the business process. It includes the following information:

- **Loan Amount:** The amount of the loan
- **Applicant information:** The applicant's name and contact details
- **Other information:** A field indicating whether the loan is PENDING APPROVAL, APPROVED, or NOT APPROVED

In this tutorial, you will create a human-centric process. Human-centric processes are automated processes that involve people. The process you will implement is as follows:

1 An applicant fills an electronic form and sends it to Fin@nce Corporation.
2 The loan officer receives the form and approves or denies the loan by completing one of these actions:
   - Clicks the Approve button. This action sets the Approval Status field on the form to APPROVED. When a loan application is approved, a PDF file of the loan details is created and archived on the network.
   - Clicks the Deny button. This action sets the Approval Status field on the form to NOT APPROVED.

     Whether the form is approved or denied, it is routed back to the user so that they can see the status of the loan.

The following illustration summarizes the process:
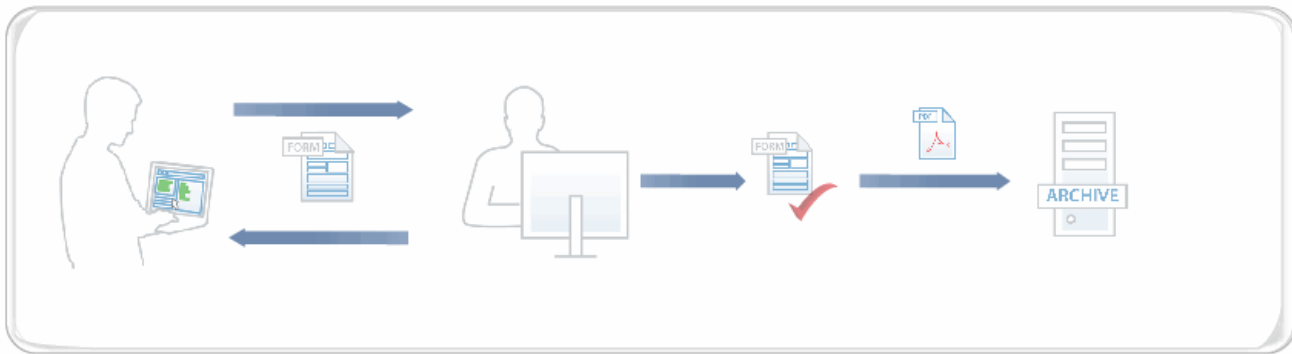
Stepping through the tutorial

Before you start this tutorial, you must install and configure your LiveCycle development environment. (See "Installing a development environment for the tutorial" on page 5.)

This tutorial is organized into several modules, each containing a number of tasks. Each module has an estimated time frame for completion. To maximize your understanding of the information in this tutorial, complete the modules in the following order:

Note: Optional sections are used only if you have separate environments for development and testing.

1 "Getting started with Workbench" on page 8 (10 minutes)
2 "Creating an application in Workbench" on page 12 (10 minutes)

3 "Creating a form design in Designer" on page 17 (20 minutes)

4 "Creating a process using Workbench" on page 24 (50 minutes)

5 "Deploying and configuring an application" on page 38 (10 minutes)

6 (Optional) Moving an application (10 minutes)

7 "Testing using in Workspace" on page 42 (10 minutes)

8 "Troubleshooting a process" on page 48 (15 minutes)

Installing a development environment for the tutorial

For the purposes of this tutorial, it is recommended that you use one server. For example, develop and test your application on the same server, which is typical of development environments.

Install a LiveCycle development environment by completing these tasks:

• Install a LiveCycle server. (See "Installing a LiveCycle server" on page 5.)

• Install Workbench. (See "Installing Workbench and the testing environment" on page 5.)

• Download and extract tutorial assets on the computer that you installed Workbench on. (See "Downloading and extracting the tutorial assets" on page 6.)

• Install sample users with roles for development and testing in the environment. (See "Understanding the sample environment" on page 6.)

Installing a LiveCycle server

You or an administrator can install a LiveCycle server to use for this tutorial.

The steps in this tutorial correspond to LiveCycle Service Pack 1. However, you can complete this tutorial using LiveCycle, without applying the service pack.

Follow these steps to install a LiveCycle server:

• Install the Turnkey installation. (See Installing and Deploying LiveCycle Using Turnkey at http://www.adobe.com/go/learn_lc_installTurnkey_9.) At a minimum, you must install the Adobe LiveCycle Output 9 and Adobe LiveCycle Process Management 9 modules to complete this tutorial. During the Turnkey installation, ensure that you import the samples, which configure a sample environment required to complete this tutorial.

• Install the LiveCycle Service Pack 1. For information on how to obtain Service Pack 1, refer to your Maintenance and Support welcome package or to the Customer Support portal at http://www.adobe.com/go/learn_lc_support for contact details.

Installing Workbench and the testing environment

The steps in this tutorial correspond to Workbench Service Pack 1. However, you can complete this tutorial using Workbench, without applying the service pack.

Install these tools, which are required to develop and test LiveCycle applications on your computer:

• Adobe Reader 9.2.0 or later on the computer where you test your application. Download Adobe Reader from the Adobe Reader download page at http://get.adobe.com/reader/.

Note: If you have Adobe Acrobat® 9.2 or later, you do not need to install Adobe Reader 9.2.0

• Adobe Flash® Player 9 or later on the computer where you test your application. Download Flash Player from the Flash download page at http://get.adobe.com/flashplayer/.

- Workbench on a computer that can access the LiveCycle server using a network connection. When you install Workbench, ensure that you also install Designer. For information about installing Workbench, see *Installing Your Development Environment* at http://www.adobe.com/go/learn_lc_installDevEnv_9.

  Note: You can also install Workbench on the same computer as the LiveCycle server if you have the necessary hardware and operating system requirements.

- Install Workbench Service Pack 1. For information on how to obtain the Service Pack 1, refer to your Maintenance and Support welcome package or to the Customer Support portal at http://www.adobe.com/go/learn_lc_support for contact details.

Downloading and extracting the tutorial assets

Download the createfirstapp.zip file at http://www.adobe.com/go/learn_lc_firstApplicationZip_9 to get the assets required to complete this tutorial.

Extract the createfirstapp.zip file to a folder on your computer, such as C:\firstappAssets. Ensure that you extract the ZIP file to the computer where you installed Workbench. Choose a folder name that is easy to remember because you will access the files at various times during this tutorial.

The createfirstapp.zip file also contains these files:

- financeCorpLogo.jpg
- firstAppLoanSchema.xsd
- FirstAppSolution.lca
- firstAppTestData.xml
- houseImage.jpg
- importFirstApp_readme.txt
- PreLoanForm.xdp
- PreLoanForm.xdp_dci
- PreLoanProcess.process
- PreLoanProcess.process_dependency

The FirstAppSolution.lca file is the completed application that you can compare your work to. To import the FirstAppSolution.lca file, follow the steps in the importFirstApp_readme.txt file.

Understanding the sample environment

When you use a LiveCycle Turnkey installation, a default user profile and default password are provided. When you installed your LiveCycle server, you must import the samples, which configure a sample environment for you. The sample environment includes a sample domain and users with roles that are required for developing and testing applications in LiveCycle. *Do not install the sample environment on a LiveCycle production server.*

If you did not install the samples for your turnkey installation, you can re-run the LiveCycle Configuration Manager and complete the LiveCycle ES2 Samples Import step. Alternatively, you can download, install, and run the Sample Setup Utility before you start this tutorial. (See LiveCycle DevNet Sample at http://www.adobe.com/go/learn_lc_samples_9.)

User roles in sample organization
The following users are part of the sampleorganization.com domain provided in the sample environment. You will use these users profiles to complete this tutorial.

User name	Password	Full name	LiveCycle role
atanaka	password	Akira Tanaka	Application Administrator Workspace user
jjacobs	password	John Jacobs	Workspace user
srose	password	Sarah Rose	Workspace user
kbowman	password	Kara Bowman	Workspace user

Note: To use other user profiles, ensure that you configure the roles appropriately. For example, you can use your own user profile instead of Akira Tanaka. Your own user profile must have the Application Administrator. (See "Setting up and configuring users" in LiveCycle Administration Help at http://www.adobe.com/go/learn_lc_admin_9?context=admin&topic=um_users_groups.)

2. Getting started with Workbench

Before you start this module, ensure that you completed setting up a development environment as described in "Installing a development environment for the tutorial" on page 5.

In this module, you will connect to a LiveCycle server using Workbench. To learn how to connect to a LiveCycle server using Workbench and understand its environment, complete these tasks in the following order:

* Log in to a LiveCycle server. (See "Connecting to a LiveCycle server" on page 8.)
* Become familiar with the Workbench environment. (See "Understanding the Workbench environment" on page 10.)

Connecting to a LiveCycle server

To create an application, it is necessary to connect to a LiveCycle server by using Workbench. The user profile that you use must have the Application Administrator role assigned to it. Contact your administrator if you are not sure which user profile to use.

Note: By default, Workbench times out after two hours, regardless of activity or inactivity. To edit the time-out setting, see "Configure advanced system attributes" in LiveCycle Administration Console Help.

Connect to a LiveCycle server

1 From the Start menu, select **Programs** > **All Programs** > **Adobe LiveCycle** > **Adobe LiveCycle Workbench ES2**.

2 Select **File** > **Login**. The Adobe LiveCycle Workbench ES2 (Version 9.5) login window appears.

**3** In the **Username** box, type `atanaka` and, in the **Password** box, type `password`.

Note: It is recommended that you use the Akira Tanaka (atanaka) user profile for this tutorial. (See "Understanding the sample environment" on page 6). However, you can use a different user profile instead of atanaka.

4 Configure the server to connect to:

* Beside the Log On To list, click **Configure**. The Manage Configured Servers dialog box appears.
* Click the **Add** ➕ button. The Adding New Server dialog box appears.
* In the **Server Title** box, type `LCES2` to describe your server. The name you enter is a mnemonic name that is meaningful only to you. The name is displayed as part of the Log On To list in the Workbench login window. As you type the name, the same text appears in the Hostname box.
* In the **Hostname** box, delete the displayed name if it is not the same as the LiveCycle server name. Type either the name or the IP address of the LiveCycle server. For example, type `LC-server`. If you do not have this information, contact your administrator.
* In the **Protocol** list, select either **Simple Object Access Protocol (SOAP/HTTP)** or **Simple Object Access Protocol SOAP/HTTPS**. The name you choose specifies the connection protocol. In typical development environments, use SOAP/HTTP. If you configured HTTPS when you installed your server, use Simple Object Access Protocol SOAP/HTTPS. If you do not have this information, contact your administrator.
* In the **Server Port Number** box, use the provided port number 8080. The value specifies the port number that is used to connect to the server. For a JBoss Turnkey installation, you typically use 8080. If you do not have this information, contact your administrator.

After you configure the server connection, the dialog box looks like this illustration, where LC-server is the name Hostname and the Server Port Number is 8080:
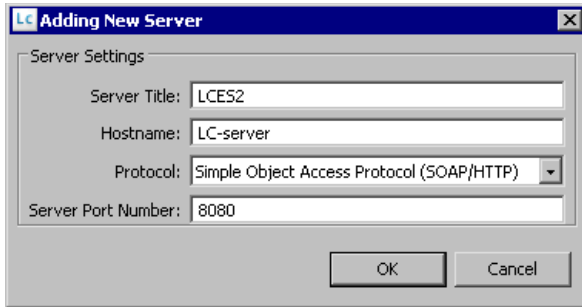
- Verify your settings and click **OK**.
- When you return to the Manage Configured Servers dialog box, click **OK**.

5 In the Adobe LiveCycle Workbench ES2 (version 9.5) login window, verify that you selected the correct server, such as LCES2, from the Log On To list.
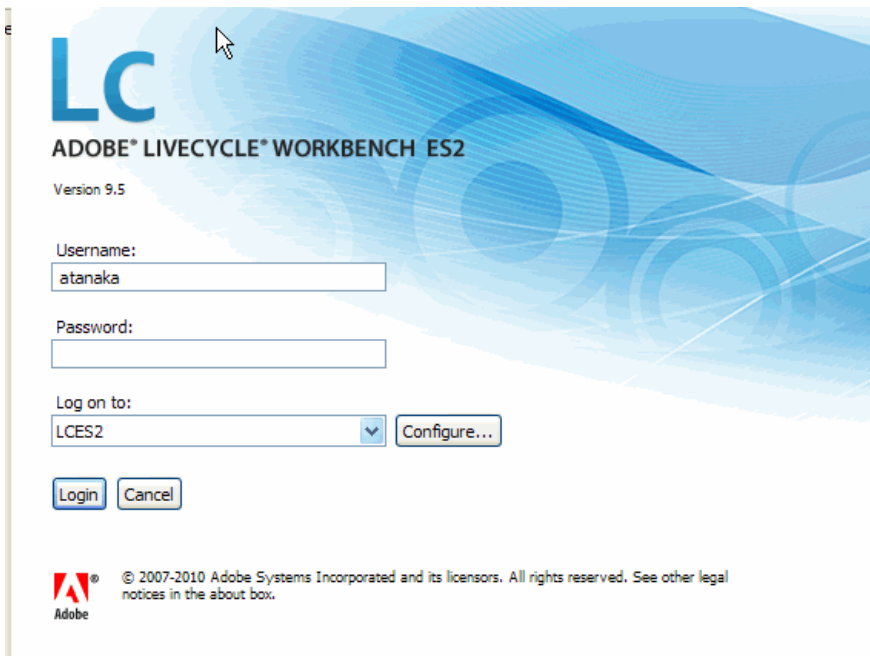
6 Click **Login**.

You have connected to a LiveCycle server.

Pausing the tutorial

You can pause the tutorial to continue at a later time by logging out of the server. These steps are provided as a reference if you want to pause your tutorial at the end of a module and continue at a later time. **For the duration of this tutorial, stay logged in to the server.**

Log of the LiveCycle server

1 In Workbench, select **File** > **Logout**. The logout window appears.

2 Select **File** > **Save** to save your work.

3 Click **Logout**.

4 Select **File > Exit** to close Workbench.

You have logged out of the server. If you want to continue with this tutorial, log back in to Workbench.

Understanding the Workbench environment

When you log in to Workbench, several views are displayed. Views are panes within the Workbench user interface and contain a set of elements a navigation tree, fields for setting values, and so on. Each view can be resized and provides its own menus and toolbars. A collection of views is called a *perspective.* The following illustration shows a completed process in the Process Design perspective. In your environment, the editor is not visible until you create a process.
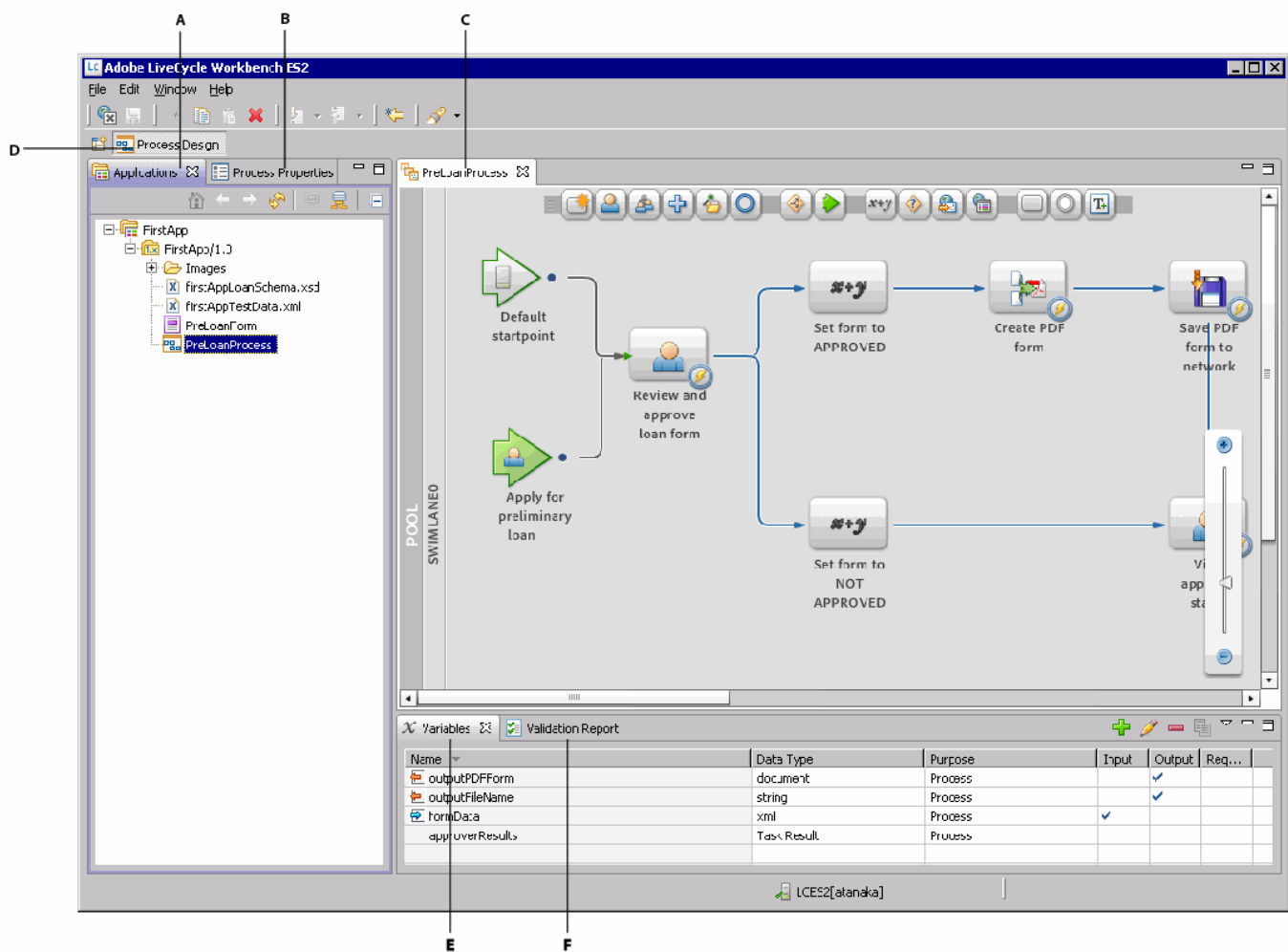
*A.*Applications view *B. Process Properties view* *C. Editor for process asset* *D. Process Design perspective* *E. Variables view* *F. Validation Report view*

Learning more

In this module, you learned to connect to a LiveCycle server using Workbench. You also learned where various views are located in the Workbench environment. The following resources provide additional information about the tasks you completed.

If you are using LiveCycle Workbench 9, see these topics in *Application Development Using LiveCycle Workbench Help* at http://www.adobe.com/go/learn_lc_workbench_9:

- Logging In to a LiveCycle ES2 server
- Working with Applications

If you are using LiveCycle Workbench 9.5, see these topics in *LiveCycle Workbench 9.5 Help* at http://www.adobe.com/go/learn_lc_workbench_95:

- Using Workbench > Logging In to a LiveCycle ES2.5 server
- Using Workbench > Working with Applications

3. Creating an application in Workbench

Before you start this module, ensure that you completed the tasks in "Getting started with Workbench" on page 8.

In this module, you will create a LiveCycle application. Then, you will add, manage, and organize assets in the application you create. To learn how to create an application, create a folder, and manage assets in Workbench, complete these tasks in the following order:

- Create an application. (See "Creating an application" on page 12.)
- Create a folder and import an asset in to your application. (See "Importing assets to an application" on page 13.)
- Use check-in and check-out to manage assets in your application. (See "Managing assets in LiveCycle" on page 15.)

Creating an application

An *application* is a container for organizing assets as a collection. Assets include processes, forms, images, schemas, or files that you want to store with your application. An application provides a convenient way for you to export and import all related assets as a unit. Storing assets as a unit ensures that the application has everything it requires to function when it is moved from one LiveCycle server to another.

Applications and assets are stored locally on your computer. When you connect to a LiveCycle server for the first time, no applications appear in the Applications view. After you create an application, it appears in the Applications view. The name of an application must be unique for each LiveCycle server. Each application can have multiple versions, which allow you to work on incremental changes to the application.

To get started, you will create an application in Workbench.

Create an application
1 Log into Workbench.
2 Select the **Applications** view.
3 Select **File** > **New** > **Application**. The New Application wizard appears.
4 In the **Application name** box, replace the default text by typing `FirstApp`.
5 (Optional) In the **Description** box, type a description for the new application, such as `Preliminary Loan Approval application`.
6 Click **Finish**. If a message is returned indicating that an existing application has the same name, append a string to your application's name to make it unique.
7 In the Applications view, a child folder appears to represent the version 1.0, or application version.

You have created an application. The Applications view looks like this illustration:

Note: Application versions appear with a naming convention of [application name]/[major version].[minor version].
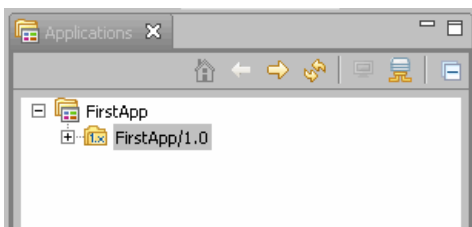
Importing assets to an application

After you create an application, you can add assets, such as forms, processes, images, and schemas. You can also add folders to better organize assets in your application. For the purposes of this tutorial, you will create one folder named *Images* and import an image into it.

Create a folder in an application

1   In the Applications view, ensure that your application is expanded.

2   Right-click **FirstApp/1.0**, and select **New > Folder**. The New Folder dialog box appears.

3   In the **Folder name** box, type `Images` to replace the default text, and then click **Finish**.

Add an asset to the folder in the application

1   Select **File > Import**. The Import wizard appears.

2   Click **General > File System** and click **Next**.

3   Beside the From Directory list, click **Browse**. The Import From Directory dialog box appears.

4   Navigate to and select the folder that you extracted the tutorial assets into and click **OK**. For example, select firstAppAssets.(See "Downloading and extracting the tutorial assets" on page 6.)

5   In the right pane, select check box beside the **financeCorpLogo.jpg** file.

6   Beside the Into Folder box, click **Browse**. The Import Into Folder dialog box appears.

7   Click **LCES2 > FirstApp > 1.0 > Images** and click **OK**.

   For Workbench 9: If you do not have Service Pack 1 or later applied to Workbench 9, the topmost folder for your application starts with the name you gave the server connection, separated by an underscore (_), with the name of application appended to it. In this situation, do the following: Click **LCES2_FirstApp > 1.0 > Images** and click **OK**.

8 Ensure that the **Create selected folders only** option is selected. The Import wizard looks like this illustration when you have Service Pack 1 or later installed.
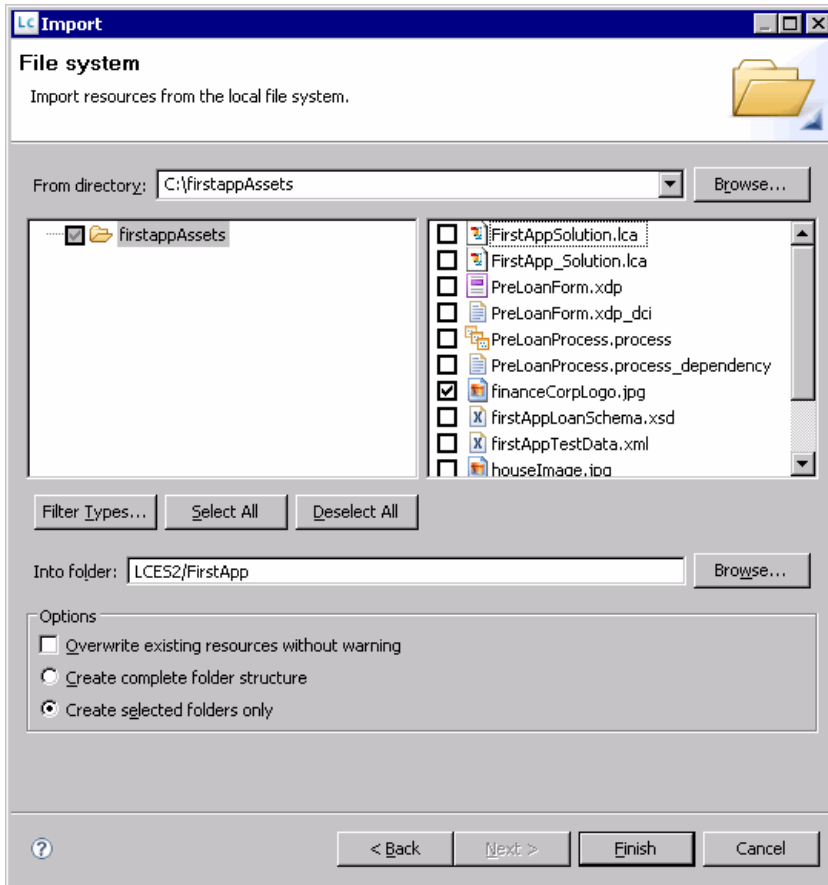
9 Click **Finish**.

The financeCorpLogo.jpg file is imported into your application. The imported asset appears with a plus sign on it, as shown in this illustration:
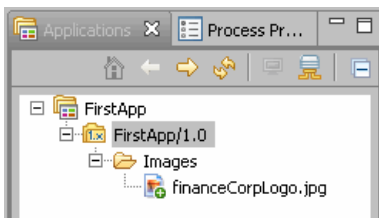
From Windows Explorer, you can import an asset by dragging it into an application version or a folder in the Applications view.

You have created a folder and imported an asset to it.

Managing assets in LiveCycle

When you create or edit assets in an application, you are modifying a copy on your computer. Check in the assets if you want to deploy it on the server or share it with other developers working on the same application. To edit an asset, check out the asset. Checkout is important for these reasons:

*   Notifies other developers working on the same application that an asset is being edited.
*   Prevents other developers from modifying the same asset and potentially overwriting each other's changes. The asset that is checked out becomes locked, which prevents others from modifying it.

These commands are used during this tutorial to manage your assets:

Check In: Uploads the last saved copy of your asset to the repository on the LiveCycle server. When an asset is first added to an application, it is available only to the user that added the asset. To make the asset available to other users, ensure you save the changes, and then check in the asset.

Check Out: Locks the asset for you to edit. Only one person can check out an asset. Check out an asset before you edit it. When an asset is checked out, a check mark appears on the lower-right corner of the asset icon. A green check mark indicates that you checked out the asset; a red check mark indicates that another user checked out the asset.

Revert: You can roll back to discard changes to an asset that is checked out. Discarding changes to an asset is useful if you want a copy of the last checked-in version of the asset. This command also checks in the asset; therefore, check out the asset to continue editing it.

After you check out an asset, select File > Open (or double-click the asset in the Applications view) and an editor opens. If you try to edit an asset that is not checked out, Workbench prompts you to check out the asset for editing.

Check in an asset

❖   In the Applications view, right-click **financeCorp.jpg** and select **Check In**.

Previously, the financeCorp.jpg file appeared with a plus sign on it. The plus sign disappears after you check in the file, as shown in this illustration.
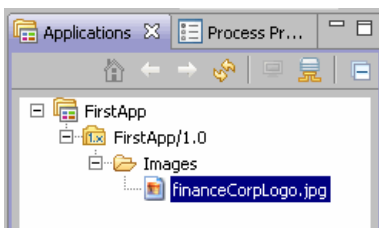
Check out an asset

❖   In the Applications view, right-click **financeCorpLogo.jpg** and select **Check Out**.

Previously, the financeCorp.jpg file appeared without a check mark on it. After you check out the file, it appears with a green check mark on the lower-right corner, as shown in this illustration. You can only edit assets after you check it out.
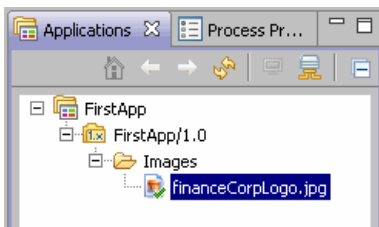
Revert an asset

1   In the Applications view, right-click **finaceCorpLogo.jpg** and select **Revert**. A Confirm Revert dialog box appears.

2   Click **Yes**.

You have checked in, checked out, and reverted an asset. The Revert command discards changes, copies the last checked-in version to your computer, and checks in the asset.

Learning more

In this module, you learned how to create an application, create a folder, and import an asset into your application. The following resources provide additional information about the tasks you completed.

If you are using LiveCycle Workbench 9, see the following topic in *Application Development Using LiveCycle Workbench Help* at http://www.adobe.com/go/learn_lc_workbench_9:

•   Working with Applications

If you are using LiveCycle Workbench 9.5, see the following topic in *LiveCycle Workbench 9.5 Help* at http://www.adobe.com/go/learn_lc_workbench_95:

•   Using Workbench > Working with Applications

4. Creating a form design in Designer

Before you start this module, ensure that you completed the tasks in .

In this module, you will create a form for applicants to use to start the approval process for a preliminary loan. You will create the form using Designer.

This module provides the minimal steps for creating a form design for use with a LiveCycle application. To fully understand the aspects involved in creating a proper form design, see the *Creating a purchase order form* tutorial.

To learn how to create a form design for automating a business process, complete these tasks in the following order:

- Create a form design using Designer. (See .)
- Modify the form design you created. (See .)
- Test the form design and save it. (See .)

If you are already familiar with the process of creating a form design, you can import the form design to your application instead of creating it. (See .)

Creating a new form design

Create the form design by using the New Form wizard in Workbench. After you create the form design, save it to your application. The wizard configures the form for use with Workspace by completing the following tasks:

- Sets the location to store the form in your application.
- Imports a data schema that you can use to bind data to.
- Adds form objects to the form design to enable it for use with Workspace.
- Specifies to submit data using XML Data Package (XDP) or a PDF document.
- Starts Designer to complete the creation of the form design.

Create the form design

1 Ensure that you are logged in to Workbench.

2 Select **File** > **New** > **Form**. The New Form wizard appears.

3 In the New Form pane, complete these steps:

- In the **Name** box, type `PreLoanForm` to replace the default text.
- In the **Description** box, type `Preapproval loan form`.
- In the pane below the Location box, select **FirstApp** > **FirstApp/1.0**, and click **Next**.

4 In the Specify Form Data Model pane, complete these steps:

- Select the **Import a data model** option.
- Beside the Import Data Model box, click **Browse**.
- In the Open dialog box, navigate to the location on your computer where you extracted the tutorial assets, select **firstAppLoan-Schema.xsd**, and then click **Open**.
- In the bottommost pane, below the Enter Or Select The Parent Folder box, select **FirstApp** > **FirstApp/1.0**, and then click **Next**.

5 In the Form Usage pane, accept the following default settings:

- The **From LiveCycle Workspace ES check box** option is selected.

- For the **Submit Adobe Acrobat/Reader Form As** option, ensure **xdp** is selected in the list.

6 Click **Next**. The Opening LiveCycle Designer dialog box appears, which starts Designer.

7 In the New Form Assistant wizard, complete these steps:

- In the Getting Started pane, select the **Use a blank form** option, and click **Next**.

- In the Document Setup pane, use the defaults (Page Size: Default, Orientation: Portrait, and Number Of Pages:1) and click **Next**.

- In the Summary pane, ensure that **xdp** is selected in the **Form Type** list, and then click **Finish**.

Designer opens the form design that includes the Submit button. The following illustration identifies areas in the Designer user interface that are used in this tutorial:
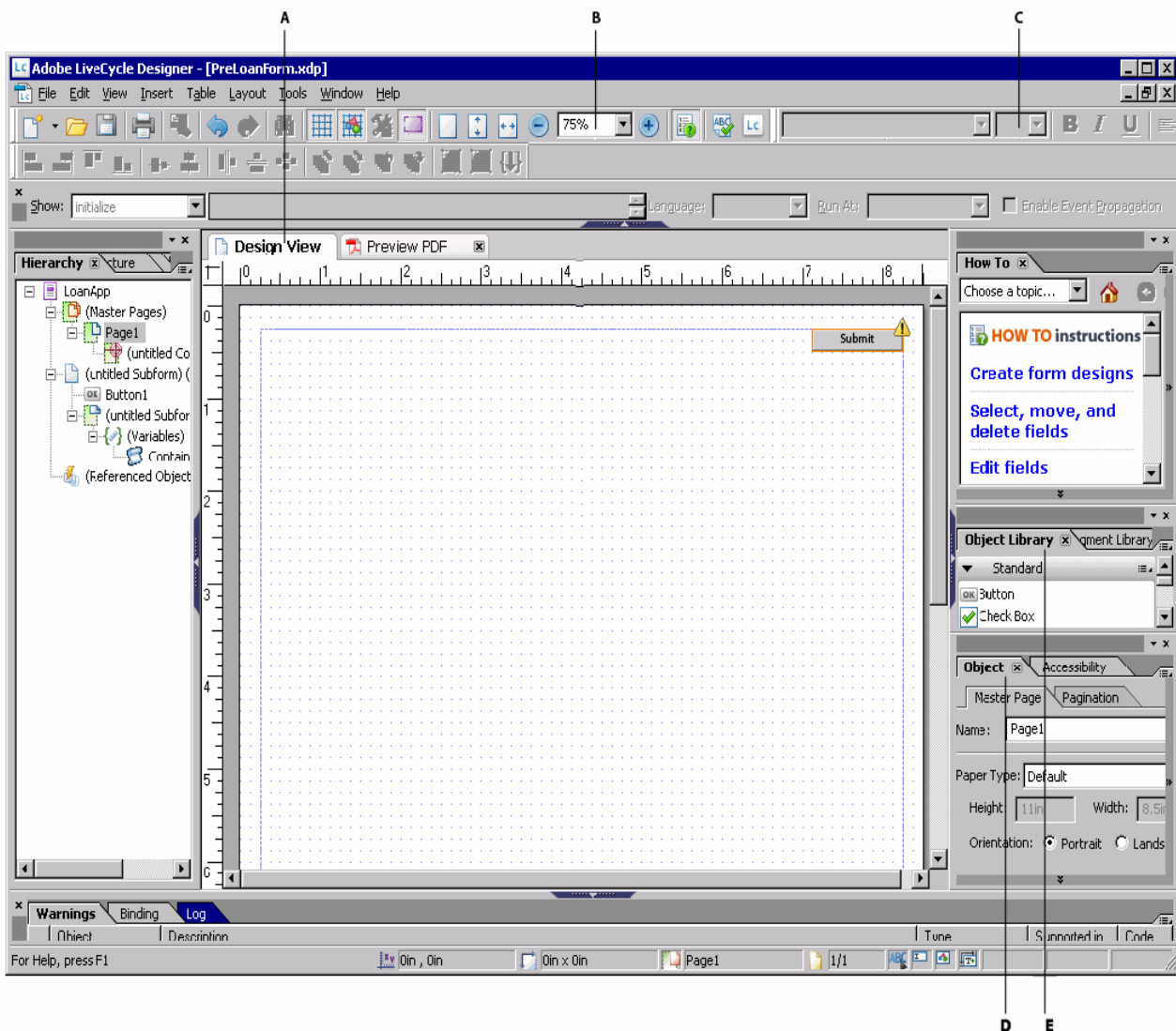
*A. Design View **B.** Change size of Design View **C.** Modify font size **D.** Object palette **E.** Object Library palette*

Save the form design

❖ In Designer, select **File** > **Save**.

Save your changes periodically and check in your assets. Ensure that you check out the form design when you want to make additional changes.

You have finished using the Form wizard to create a form design enabled for Workspace.

Modifying a form design

You now have a form design with a submit button in the upper-right corner. The submit button enables the submission of forms in Workspace. In addition, the form design uses a data schema to pass information from the form to the process. You will modify the form design to add fields where a user can provide input to, add titles and logos to identify the form, and enhance the presentation.

You will perform the following tasks to modify the form design:

Adding a title and logo to the form design

Providing an image and title helps the user identify the process they are starting or are involved in. You add a title and logo to the form design by adding form objects to the form design. *Form objects* represent entities on a form design, such as text, text fields, images, and check boxes.

Add a logo to the form design

1   In the topmost toolbar in Designer, between the plus and minus icons, select **100%** to change the size of the form design.

2   In the Object Library palette, in the Standard category, drag the **Image** object 🖼 to the upper-left corner of the form design.

3   In the Object palette, select the **Embed Image Data** check box.

4   Beside the URL box, click the **Browse** 📁 button. The Browse For Image File dialog box appears.

5   Navigate to the Images folder, select **financeLogo.jpg**, and then click **Open**.

Add a title to the form design

1   In the Object Library palette, in the Standard Category, drag the **Text** object on to the form design and position it beside the Image object you added earlier.

2   Delete the default text and type `Preliminary Loan Approval Form`. (By default, you can type text in the Text object.)

3   Triple-click the text you typed to highlight it.

4   In the topmost toolbar, change the font size to **24**, and then click **Bold** the 🅱 button.

5   Select **File** > **Save** to save the form design.

You have added a title and logo to your form design, which now looks like this illustration.
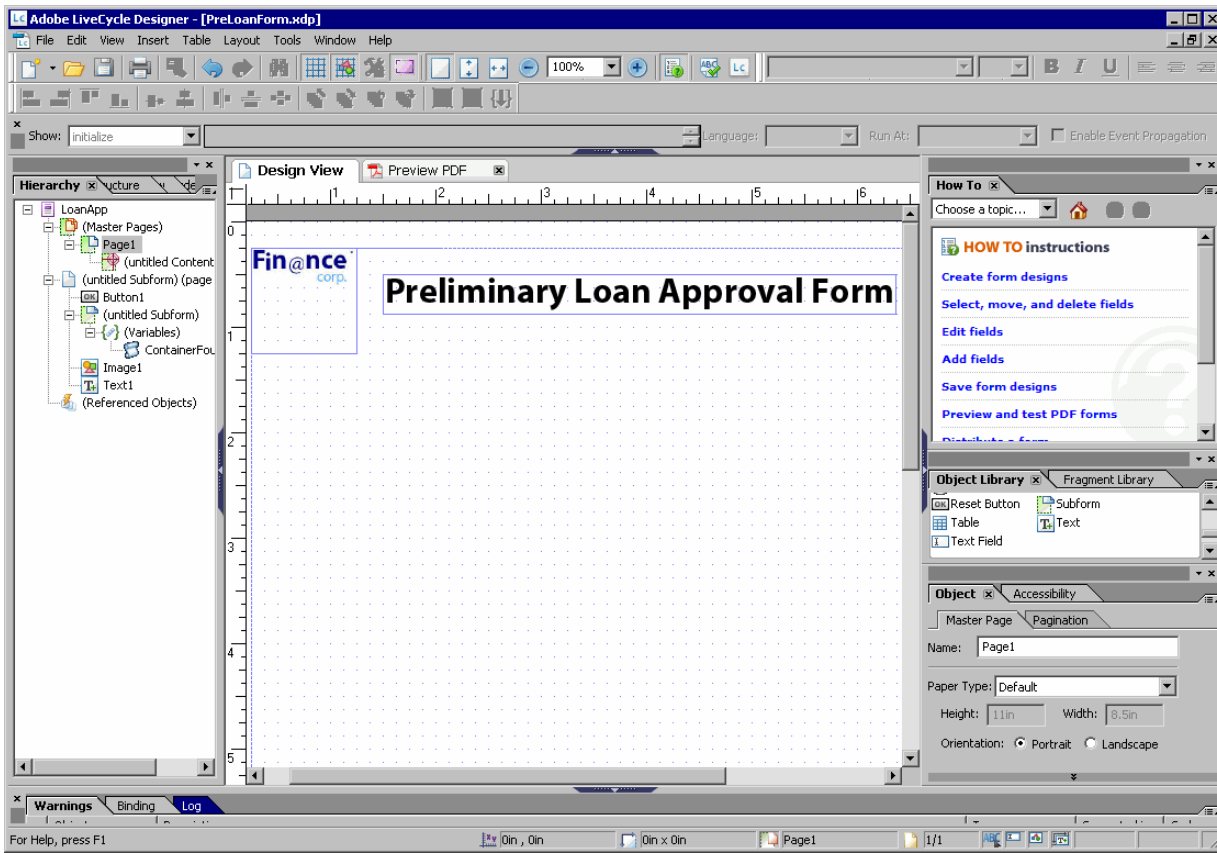
Adding fields to the form design

It is necessary to bind the form fields to a data schema for use as input to an automated process. Typically, this task involves several steps You can add multiple form objects to a form design by using a data schema file. Using a data schema file to create form objects in your form design saves you time for these reasons:

- Form objects are created based on the defined data type in the schema file.
- Each form object is automatically bound to the data source or entries in the data schema.
- Captions are created for each form object based on the element name in the data schema.

After you add the form objects, you can rearrange and modify the fields as required.

A best practice is to develop a stable schema before you start to create form designs.

Add form objects to a form design using a schema

1 In Designer, if the Data View palette is not available, select **Window** > **Data View**.

2 From the Data View palette, drag **DataConnection** to the leftmost part of the form design, below the image you added. Four fields appear in the form design.

Enhancing the appearance of a form design

Enhancing the appearance of the form design makes it easier to use and more visually appealing. You can improve the appearance of a form design by performing tasks such as changing fonts, colors, arranging fields, and setting default values.

Rearrange and modify the appearance of fields

1 In Designer, arrange the fields so that they appear one below another. To move a field, click and hover your mouse over the field until a cross appears, and then drag it to a new location on the form design. Arrange the fields from top to bottom in the following order:

- Name
- Loan Amount
- Phone Or Email
- Approval Status

2 Press the Control key and click the **Name**, **Loan Amount**, **Phone Or Email**, and **Approval Status** fields so that all four are selected.

3 Drag the rightmost border of the selected fields to the right to make all the fields larger.

4 Select **Layout** > **Align** > **Left** to left align the fields with the leftmost border.

5 Click in an empty area on the form design to deselect the fields.

Note: Errors in spelling appear with warning signs to the right of the field in your form design. You can ignore them for the purposes of this tutorial.

Set a field to read-only and set its default value

1 Select the **Approval Status** text field.

2 In the Object Palette, click the **Value** tab.

3 From the **Type** list, select **Read Only**.

4 In the **Default** box, type PENDING APPROVAL.

5 Select **File > Save**.

You have added form objects to the form design by using a schema file. You also rearranged the fields on the form design and set the Approval Status field to read-only with a default value of PENDING APPROVAL. The form design looks like this illustration.
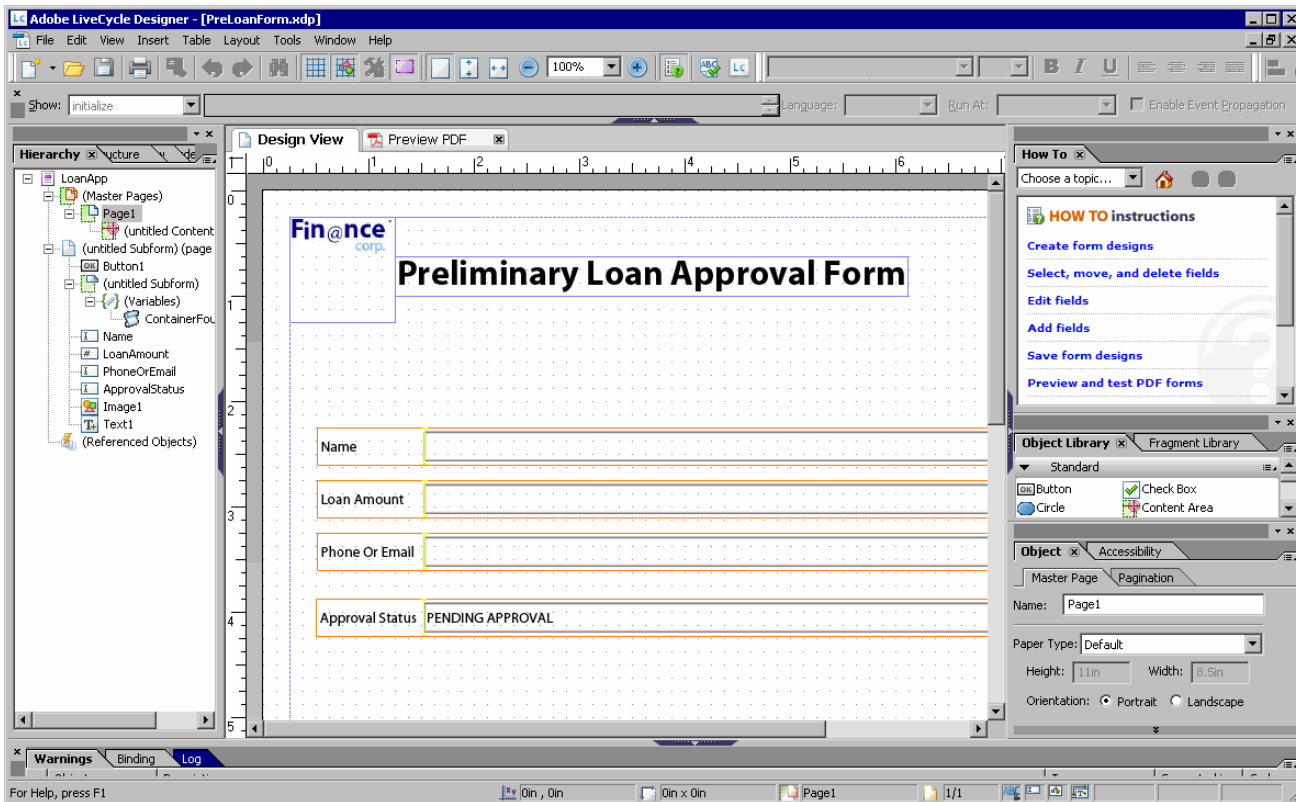
Testing the form design

Test your form to ensure that it functions as designed before you use it in a process. You can use the sample data file provided with the tutorial assets to preview your form design as a PDF form.

Test your form design using sample data

1   In Designer, select **File** > **Form Properties**. The Form Properties dialog box appears.

2   Click the **Preview** tab.

3   Beside the Data File box, click the **Browse** 📁 button. The Open XML File dialog box appears.

4   Navigate to the location where you extracted the tutorial assets, such as C:\firstappAsets, select firstAppTestData.xml file, and click **Open**. A dialog box appears with this text displayed: The Selected File Is Not Part Of Any LiveCycle Application. Would You Like To Copy It And All The Files It May Reference To The Current Application?

5   Click **OK**.

6   In the Form Properties dialog box, click **OK**.

7   Click the **Preview PDF** palette to view the form. If the Preview PDF palette is not visible, select **View** > **Preview PDF**.

    Note: If the Preview PDF palette goes to full screen mode, double-click it to return to Designer.

8   Click the **Design View** tab, and select **File** > **Save**.

9   Select **File** > **Exit** to exit Designer.

10   In Workbench, right-click **PreLoanForm** in the Applications view, and select **Check In**. Also, check in the firstAppTestData.xml and firstAppLoanSchema.xsd files.

You have completed testing the form design with XML data and checked-in the assets into your application.

Adding a prebuilt form design to an application

If you are familiar with the process of creating a form design, use the prebuilt form design provided with the tutorial assets.

Caution: Do not complete these steps if you created a form design by using the steps from "Creating a new form design" on page 17 *and* "Modifying a form design" on page 19.

Adding a prebuilt form design to the application.

1   In Workbench, select **File** > **Import**. The Import wizard appears.

2   Select **General** > **File System**, and then click **Next**.

3   Beside the From Directory list, click **Browse**. The Import From Directory dialog box appears.

4   Navigate to the folder that you extracted the tutorial assets into and click **OK**. (See "Downloading and extracting the tutorial assets" on page 6.)

5   In the right pane, select the **PreLoanForm.xdp**, **firstAppLoanSchema.xsd**, and **firstAppTestData.xml** files.

6   Beside the Into Folder, click **Browse**. The import Into Folder dialog appears.

7   Select **LCES2_FirstApp** > **1.0**, click **OK**, and then click **Finish**.

8   In the Applications view, right-click **PreLoanForm.xdp** and select **Check In**. Also check in the firstAppLoanSchema.xsd and firstAppTestData.xml assets.

Learning more

In this module, you learned how to create a form design in Designer. The following resources provide additional information about the tasks you completed.

See this section in *LiveCycle Designer Help* at http://www.adobe.com/go/learn_lc_designer_9:

•   Working with Form Designs

5. Creating a process using Workbench

Before you start this module, ensure that you completed the tasks in "Creating a form design in Designer" on page 17.

You create a process using Workbench. A process that involves users is called a *human-centric process*. Human-centric processes automate the routing of data from one user to another by assigning *tasks*. Part of creating a process is drawing a process diagram. A *process diagram* is the visual representation of the business process that you are automating. You will add elements to the process diagram to represent steps in the process you are automating. Elements can include operations, users, routes, and start points. After you draw the process diagram, you add variables and configure operations to implement the steps in the business process.

In this module, you will create a human-centric process to automate your business process. (See "Understanding the business problem" on page 3.)

The business process you automate uses the form design you created. (See "Creating a form design in Designer" on page 17.) To learn how to create a human-centric process, complete these steps in the following order:

- Create a process in Workbench. (See "Creating a process" on page 24.)
- Draw the process diagram and configure it to implement the business process. (See "Editing the process diagram" on page 25.)
- Validate the process you created. (See "Validating a process" on page 36.)

Note: If you are familiar with the process of creating processes, import the process and process-dependency files to your application instead. (See "Adding a prebuilt process to an application" on page 36.)

Creating a process

Use the New Process wizard to create a process. The wizard guides you through the steps for configuring the process and creating the necessary start points. Start points represent how a process is invoked or started. The process you create is started from Workspace.

Create a process using the New Process wizard

1. Ensure that you are logged in to Workbench.
2. Select **File** > **New** > **Process**. The New Process wizard appears.
3. In the New Process pane, perform these tasks:
   - In the **Name** box, type `PreLoanProcess` to replace the default text.
   - In the **Description** box, type `A process to handle preliminary loan approvals.`
   - Below the Enter Or Select Parent Folder box, select **FirstApp** > **FirstApp/1.0**, and then click **Next**.
4. In the Configure A Start Point pane, select the **When a user submits a task in Workspace** option and click **Next**.
5. In the Workspace Start Point Configuration pane, complete these steps:
   - Select the **Use an existing Form** option.
   - In the pane below the Enter Or Select An Asset box, click **FirstApp** > **FirstApp/1.0** > **PreLoanForm**, and click **Next**.
   - Ensure that the **Create a New Workspace Category** option is selected and, below it, delete the default text and type `Loans`.
   - In the **Workspace Process Name** box, delete the default text, and type `Apply for preliminary loan`.
   - In the **Description** box, type `Fill the form to start the approval process.` to replace the default text.
   - Beside Icon label, click the **Browse** 🗎 button.
   - Navigate to the location where you extracted the tutorial assets, select **houseImage.jpg**, click **Open**, and then click **Next**.

6   In the New Process Configuration Summary pane, click **Finish**.

After you complete the steps, an editor appears with the process diagram displayed in it. The process diagram contains these start points:

- A Default start point that specifies that the process is invoked programmatically from a web service (SOAP/REST), Java, or Adobe LiveCycle Data Services ES2.
- A Workspace start point that specifies that the process is invoked using Workspace.

The following illustration shows the editor and identifies areas of the process diagram referred to in this tutorial.
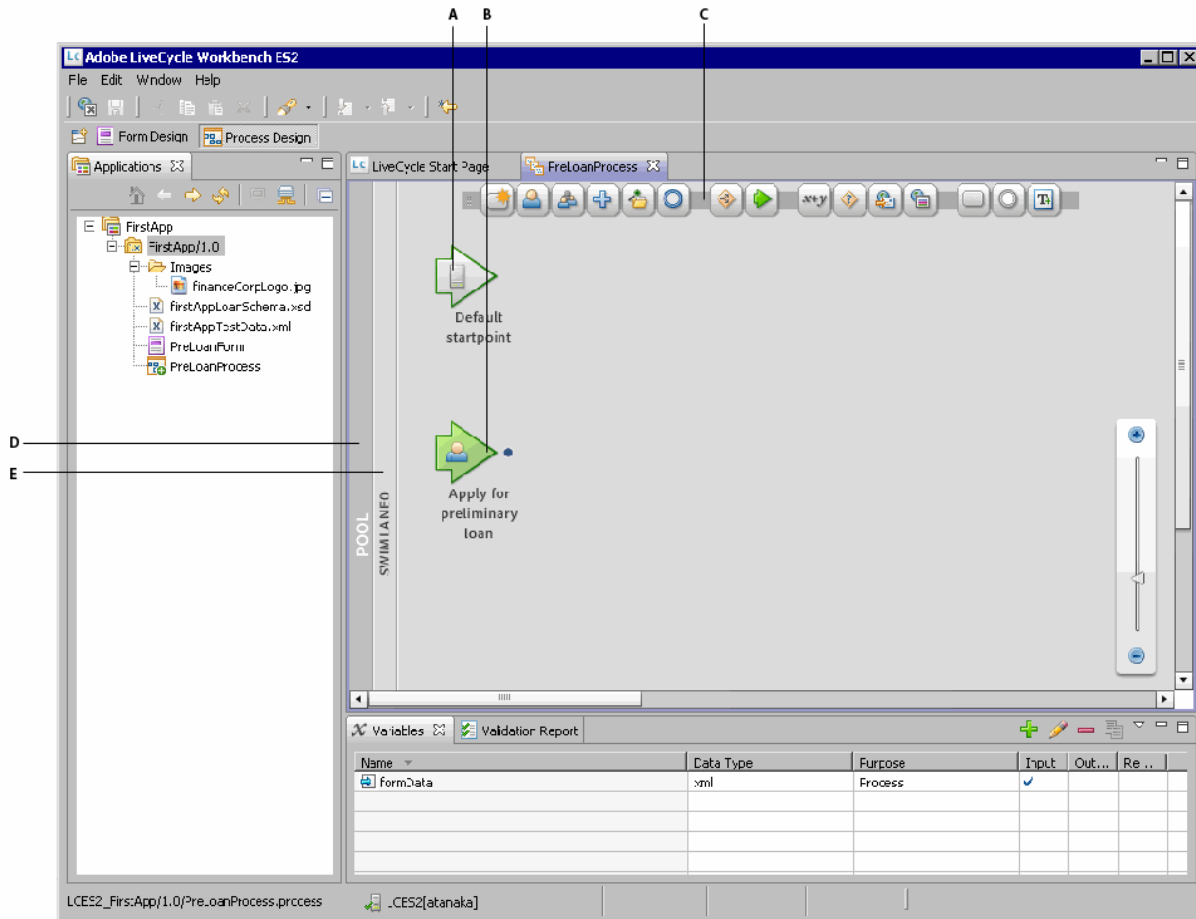
A. Default start point *B.* Workspace start point *C.*Activity toolbar *D.* Pool *E.*Swimlane

Editing the process diagram

The process diagram created for you includes these start points:

- A Default start point and Workspace start point (named *Apply For Preliminary Loan*)
- An input variable named *formData* that passes information into your process when it is invoked

After you create the process, you add and configure operations on the process diagram. You will complete these tasks to draw, edit, and configure process.

-

- "Drawing routes" on page 28
- "Defining variables" on page 29
- "Configuring operations" on page 30

Adding operations to the process diagram

A default process diagram consists of a pool that contains one swimlane. Swimlanes are horizontal areas that divide the pool. You can add more swimlanes to better organize operations and improve the readability of processes. Swimlanes have no effect on the execution of the process. For this tutorial, only one swimlane is used because the process is simple.

Add operations to define the steps or activities in your process. Operations can represent people involved in your automated process and steps in the process.

In the process diagram, you can drag common operations from the activity toolbar located at the top of the editor for the process. The activity toolbar is shown in this illustration.

Add Assign Task operations

1   In the editor for PreLoanProcess, drag the **Assign Task** operation from the activity toolbar and position it between and to the right of the Default Startpoint and Apply For Preliminary Loan start point (Workspace start point). Arrows from the start points appear and connect to the left side of the Assign Task operation. The green arrowhead on the left side of the operation indicates that this operation is the first step in the process, or known as the *Start Activity*.

2   Click the **AssignTask0** text, delete the text, type `Review and approve loan form`, and then press Enter.

3   Drag another **Assign Task** operation to the process diagram and position it near the lower-right edge of the process diagram.

4   Click the **AssignTask1** text, delete the text, type `View approval status`, and then press Enter.

After you complete the steps, the process diagram looks like this illustration.
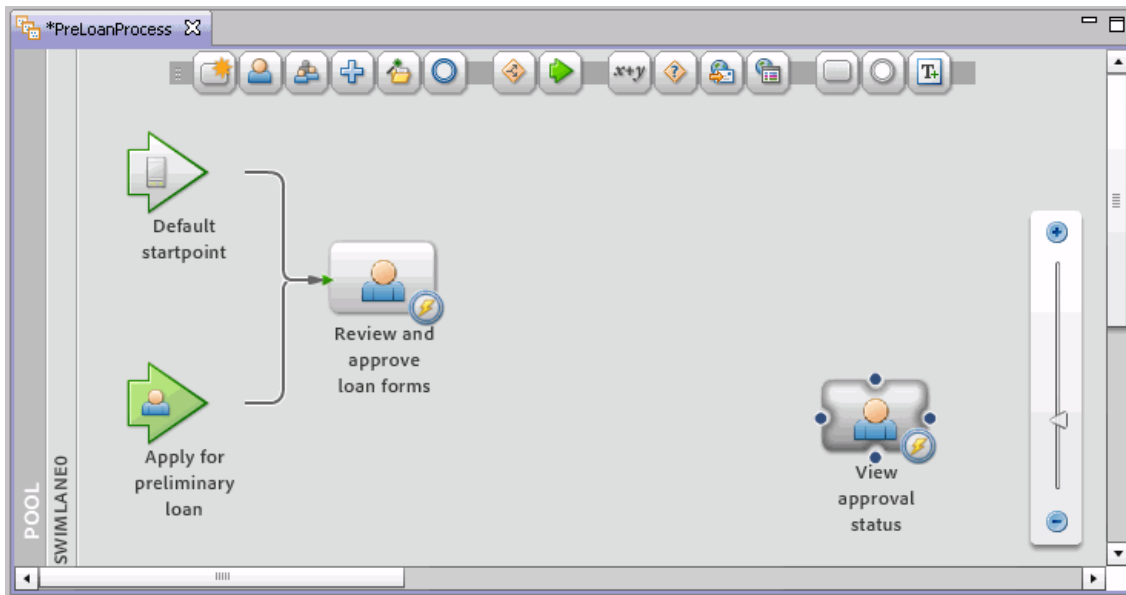
Add Execute operations

1   In the editor for PreLoanProcess, drag the **Set Value** operation from the activity toolbar and position it near the top of the swimlane and right of the Review And Approve Loan Form operation.

2   Click the **execute2** text, delete the text, type `Set form to APPROVED`, and then press Enter.

3   Drag another **Set Value** operation to the process diagram and position it near the bottom of the swimlane and to the right of the Review and Approve Loan Form operation.

4   Click the **Execute3** text, delete the text, and type `Set form to NOT APPROVED`, and then press Enter.

Add a GeneratePDFOutput operation

1   In the editor for PreLoanProcess, drag the **Activity Picker** from the activity toolbar, and position it to the right of the Set Form To APPROVED operation. The Define Activity dialog box appears and displays a list of services.

2   In the **Find** box, type `generate`. The services in the list become filtered.

3   In the list of results, select the **generatePDFOutput** operation (Output - 1.1), which is located under the Output service, and click **OK**.

4   Click the **generatePDFOutput** text, delete the text, type `Create PDF form`, and then press Enter.

Add a Write Document operation

1   In the editor for PreLoanProcess, drag the **Activity Picker** from the activity toolbar, and position it on the upper-right corner of the swimlane. (Beside the Create PDF form operation.) The Define Activity dialog box appears and contains a list of services.

2   In the list, click **Foundation**.

3   Click the **Write Document** operation (File Utilities - 1.0) and click **OK**.

4   Click the **Write Document** text, delete the text, type `Save PDF form to network`, and then press Enter.

5   Select **File > Save**.

You have added these operations to the process diagram:

•   Two Assign Tasks operations (User service) to represent the loan officer and the person to route the complete form to

•   Two Execute operations (Set Value service) to access and manipulate process variable data

•   A Generate PDF Output operation (Output service) to create a PDF form from the approved data

•   A Write Document operation (File Utilities service) to save the PDF file to a network location

The completed process diagram looks like this illustration.
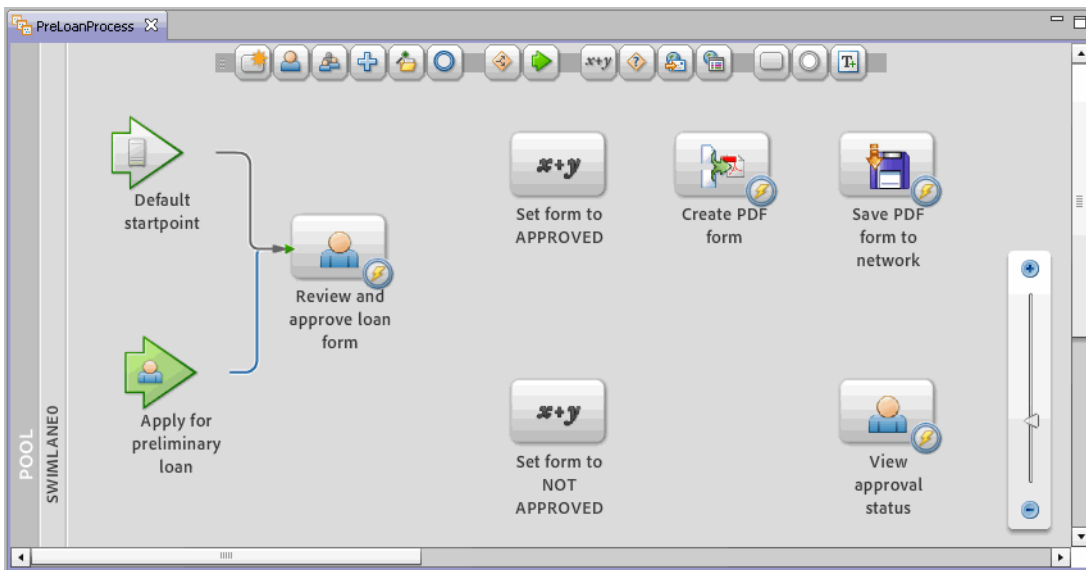
Drawing routes

Routes connect operations and define the order in which operations execute. Routes, which are represented by arrow lines, connect operations in the process diagram. You can draw multiple routes that originate from a step to indicate different paths in the process. You cannot draw routes from a start point to an operation. Routes are created automatically to the operation that is set as the Start Activity.

Draw routes

1 Select the **Review and approve loan form** operation. Four anchors appear around the operation. Each anchor is displayed as a round circle.

2 Drag the right anchor to the Set Form To APPROVED operation and release your mouse button. Notice that, as you drag to the Set Form To APPROVED operation, an arrow connects the two operations. (To delete a route, right-click the route and select Delete Route or press Delete.)

3 Repeat steps 1 and 2 to draw routes between these operations:

  • Set Form To APPROVED to Create PDF Form operation

  • Create PDF Form to Save PDF Form To Network operation

  • Save PDF Form To Network to View Approval Status operation

  • Review And Approve Loan Form to Set Form To NOT APPROVED operation

  • Set Form To NOT APPROVED to View Approval Status operation

4 Select **File > Save**.

After you draw the routes, the process diagram looks like this illustration:
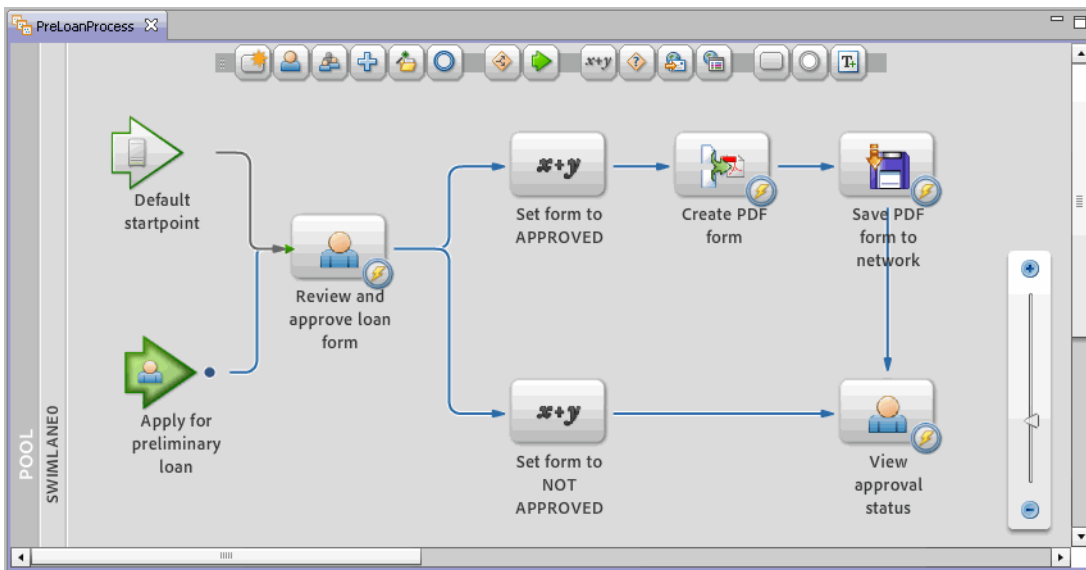
Defining variables

Variables pass data from operation to operation. When you create a variable, specify a name and data type. You can also specify whether the variable provides input, output, or both input and output values. *Input* variables specify the data that is passed to your process when it is invoked. *Output* variables are values provided by your process when the process completes.

An input variable named *formData* is created for you. The formData variable passes XML data from the form into your process. For this process, you will create output variables to store the contents of the PDF file and the filename of the PDF file by using the Variables view. Later in this module, you will create variables, as required, directly from the Process Properties view.

Create a variable for saving the PDF form

1   In the Variables view, click the **Create New Variable** ➕ button. The Variable dialog box appears.

2   In the **Name** box, type `outputPDFForm`.

3   In the **Title** box, type `Approved loan form`.

4   In the **Type** box, select **document**. The document data type is used to store binary file data.

5   In the Purpose area, select the **Output** check box to set the variable as an output value for the process, and click **OK**.

6   Create another variable to store the name of the file. Repeat the steps 1 to 5, but use these values for the new variable:

   Name: outputFileName

   Title: PDF filename

   Type: string

   Purpose: Select the Output check box.

7   Click **OK** and then select **File** > **Save**.

You have created two new variables to store data and pass as output parameters from your process. The Variables view looks like this illustration.

Name ▼	Data Type	Purpose	Input	Output	Req...
outputPDFForm	document	Process		✔	
outputFileName	string	Process		✔	
formData	xml	Process	✔		

Configuring operations

Configure properties for an operation to change the way it functions or executes. When you configure an operation, by default, you configure the *Basic* properties, which allow you to configure basic operation functionality. To configure advanced functionality, select *All* to see the all property groups and properties. The following illustration indicates common areas in the Properties Process view used in this tutorial.
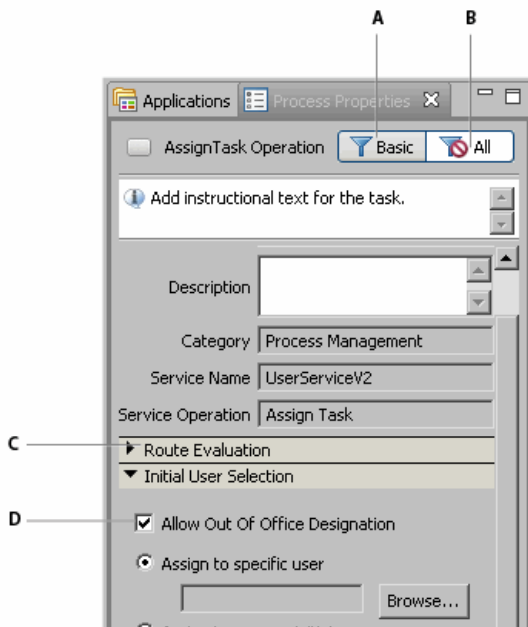
A. Show Basic properties *B.* Show All Properties *C.* Property group *D.* Property

You will configure the operations you added to the process diagram by completing these tasks:

- "Configuring Assign Task (User 2.0 service) operations" on page 30
- "Configuring Execute (Set Value service) operations" on page 32
- "Configuring the GeneratePDFOutput operation (Output service)" on page 34
- "Configuring the Document Write operation (File Utilities service)" on page 34

Configuring Assign Task (User 2.0 service) operations

Configure the Assign Task operations in the process diagram for these reasons:

- The user who is involved with the process. For example, configure the Review And Approve Loan Form operation to assign the task to John Jacobs, who is the loan officer.
- The instructions to display to the user in Workspace.
- The form and form data to display to the user.
- Buttons to complete the task in order to choose the next step in the process when multiple routes are available. The next step is based on the button that the user clicked. Typically, when multiple routes are configured from an operation, you add conditions to each route to determine the next step to take. The Assign Task operation can be configured to let the user choose the next step in the process. For example, for the Review And Approve Loan Form operation, you can click an Approve and Deny button.

Configure the Review and Approve Loan Form operation

1 In the editor for PreLoanProcess, click the **Review and approve loan form** operation.

2 In the Process Properties view, click the **Initial User Selection** property group and complete these steps to configure the user who is assigned the task:

- Beside the Assign To Specific User box, click **Browse**. The Select User dialog box appears.
- Select **John Jacobs** and click **OK**. If you did not configure the LiveCycle sample environment, select another user. (See "Under-standing the sample environment" on page 6.) Remember which user you selected because you need to log in as that user to perform testing later.

3 Click the **Task Instructions** property group. Complete these steps to specify the instructions displayed to the user in Workspace:

- In the rightmost list, ensure that **template** is selected. Use Template values to specify strings that are a combination of text that you type and values from process variables.
- Beside the Task Instructions box, click the **Insert or replace XPath expression** [Insert Expression...] button. The XPath Builder dialog box appears.
- Double-click **process_data** > **formData** > **xdp** > **datasets** > **data** > **LoanApp** > **Name** and click **OK**. In the Task Instructions box, {$/process_data/formData/xdp/datasets/data/LoanApp/Name$} appears. As shown in the following illustration, to double-click Name, you must expand the tree structure.
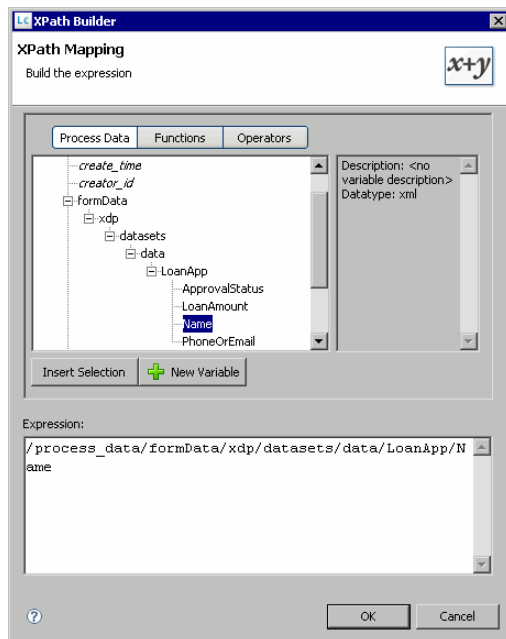
- After the {$/process_data/formData/xdp/datasets/data/LoanApp/Name$} text, add a space and type `has requested a loan. Please review the form and click Approve or Deny.` in the box.

4 Click the **Presentation & Data** property group and complete these steps to specify the data to merge with the form:

- Ensure that the **use an application asset** option is selected.

- In the Application Asset area, beside the Asset box, click the ellipsis ⊡ button. The Select Form Asset dialog box appears.

- Click **FirstApp** > **FirstApp/1.0** > **PreLoanForm** and click **OK**.

- In the **Variable** list, select **formData**.

5 Click the **User Actions** property group and complete these steps to add buttons for the user to complete their task:

- Click **Add A User Action** ⊞ button. The Action Properties dialog box appears.

- In the **Action Name** box, type Approve to replace the default text.

- In the **Destination** list, select **Set Form To APPROVED**, and click **OK**.

- Click the **Add A User Action** button to add a second button.

- In the **Action Name** box, type Deny to replace the default text.

- In the **Destination** list, select **Set Form To NOT APPROVED**, and click **OK**.

- Select the **This action needs confirmation, and will use the following text as the message** check box.

- In the box below the check box, type Do you really want to deny the loan request? Click **OK**.

6 Click the **Output** property group to configure the location to store the output results from the operation by completing these steps:

- Beside the Task Result, click the **Create A Variable** ⊞ button. The Variable dialog box appears. The type of Task Result is automatically provided.

- In the **Name** box, type approverResults and click **OK**.

- In the **Output data** list, select **formData** to save changes that were possibly made to the form.

Configure the View Approval Status

1 In the editor for PreLoanProcess, click the **View approval status** operation.

2 In the Process Properties view, click the **Initial User Selection** property group and select the **Assign to process initiator** option. Selecting this option routes the form back to the user who started the process.

3 Click the **Task Instructions** property group and type Your loan form has been reviewed. If the loan was approved, a Fin@nce representative will contact you within three business days. in the **Task Instructions** box. This step specifies the instructions displayed to the user who is reviewing the approval status of the form.

4 Click the **Presentation & Data** property group to configure the form design and data to merge by completing these tasks: (The merged form is displayed to the user in Workspace.)

- Ensure that the **use an application asset** option is selected.

- In the Application Asset area, beside the Asset box, click the ellipsis ⊡ button. The Select Form Asset dialog box appears.

- Click **FirstApp** > **FirstApp/1.0** > **PreLoanForm** and click **OK**.

- In the **Variable** list, select **formData**.

5 Click the **Output** property group and, in the **Output data** list, select **formData**.

6 Select **File** > **Save**.

You have configured the Review And Approve Loan Form and View Approval Status (Assign Task) operations.

Configuring Execute (Set Value service) operations

Use Execute operations to modify the form data, and create a filename for the PDF file. In the process, use the Execute operations to change the values of the Approval Status field in the formData variable. When the loan officer clicks the Approve button, the Approval Status field is set to APPROVED. When the loan officer clicks the Deny button, the Approval Status field is set to NOT APPROVED.

Configure the Set Form To APPROVED operation

1   In the editor for PreLoanProcess, click the **Set Form To Approved** operation.

2   In the Process Properties view, in the Mapping property group, click the **Edit the mapping data** [pencil icon] button. The XPath Builder dialog box appears.

3   Set the Approval Status field to APPROVED by completing these steps:

   •   On the Process Data tab, double-click **process_data > formData > xdp > datasets > data > LoanApp > ApprovalStatus**. The XPath expression displayed as an entry in the table on the left pane.

   •   For the first entry, click **[expression]** under the Expression column in the table.

   •   In the **Expression** box located at the lower-right of the dialog box, type `"APPROVED"`. (Ensure that you include the quotation marks.)

4   Create the filename for the PDF file by completing these steps:

   •   Click the **Add a list entry** [plus icon] button. A new entry appears below the previous entry in the table.

   •   On the **Process Data** tab, double-click **process_data > outputFileName**. The location /process_data/@outputFileName appears in the Location box.

   •   Click **[expression]**, click the **Functions** tab, and double-click **String Functions > concat(string, string, string*)**.

   •   In **Expression** box, double-click the first **string** so that it is highlighted, and type `"approvedby_"`. (Ensure that you include the quotation marks.)

   •   Double-click the second **string**, click the **Process Data** tab, and double-click **process_data > approverResults > object > completedBy > commonName**.

   •   Double-click the third **string**, type `".pdf"`, and delete the extra asterisk (*). (Ensure that you include the quotation marks.)

   After you complete the steps, the Expression box looks like this illustration.
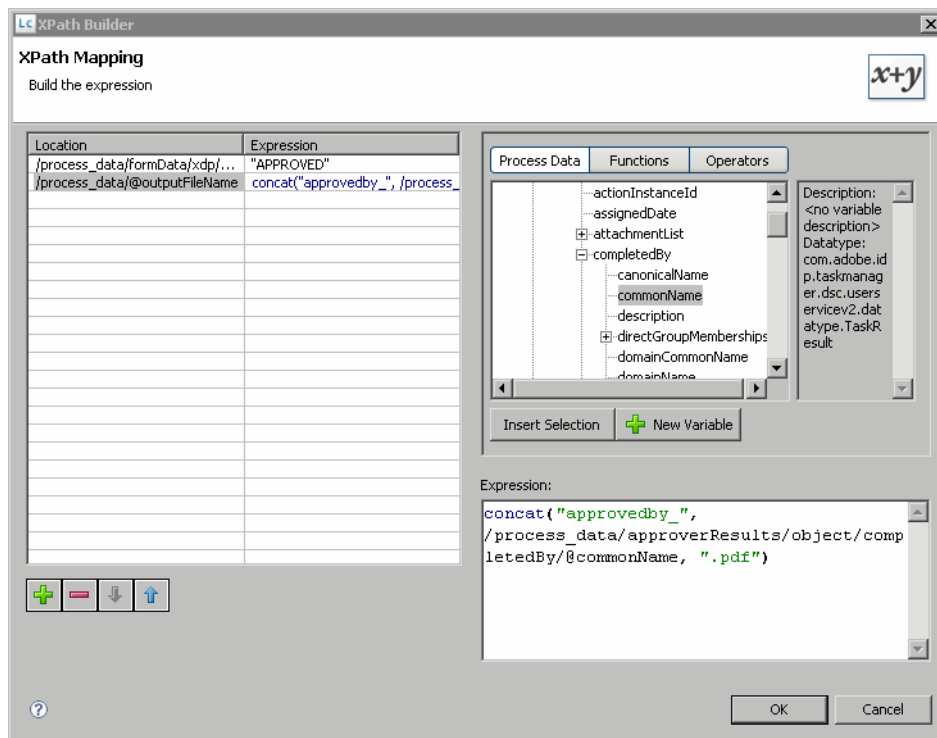
5   Click **OK**. In the Process Properties view, the entries you added appear in the table below the Mappings property.

Configure the Set Form To NOT APPROVED operation

1 In the editor for PreLoanProcess, click the **Set form to NOT APPROVED** operation.

2 In the Process Properties view, in the Mapping property group, click the **Edit the mapping data** ✎ button. The XPath Builder dialog box appears.

3 Set the Approval Status field in the form to NOT APPROVED by completing these steps:.

- On the Process Data tab, double-click **process_data** > **formData** > **xdp** > **datasets** > **data > LoanApp** > **ApprovalStatus**. The XPath expression is displayed as an entry in the table on the left pane.

- Click [**expression**] and, in the **Expression** box, type "NOT APPROVED". (Ensure that you include the quotation marks.)

- Click **OK**.

4 Select **File** > **Save**.

You have configured the Set Form To APPROVED and Set Form To NOT APPROVED (Execute) operations.

Configuring the GeneratePDFOutput operation (Output service)

Use the GeneratePDFOutput operations to merge a form design with XML data. The result is a non-interactive PDF form that can be printed or archived to the network as a PDF file.

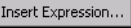**Configure the Create PDF Form operation**

1 In the editor for PreLoanProcess, click the **Create PDF form** operation.

2 In the Process Properties view, click the **Input** property group and complete these steps to configure the properties:

- Beside the Form box, click the ellipsis ⋯ button. The Select Asset dialog box appears.

- Click **FirstApp > FirstApp/1.0 > PreLoanForm** and click **OK**.

- In the **Input Data** list, select **formData**.

3 Click the **Output** property group and, in the **PDF Output** list, select **outputPDFForm**.

4 Select **File** > **Save**.

You have completed the Create PDF Form (generatePDFOutput) operation to create a PDF file from the form design and form data.

Configuring the Document Write operation (File Utilities service)

Use the Document Write operations to write files to the file system.

Configure the Save PDF form to network operation

1 In the editor for PreLoanProcess, click the **Save PDF form to network** operation.

2 Specify the name that the PDF form is saved to by completing these steps:

- In the Input property group, beside the Pathname Pattern property name, select **template** from the list. The default is Literal Value.

- In the box below the Pathname Pattern property name, type C:\followuploans\. Ensure that you position the cursor after the backslash.

- Click the **Insert or replace XPath expression** Insert Expression… button below the box. The XPath Builder dialog box appears.

- On the Process Data tab, double-click **process_data** > **outputFileName**. The XPath expression /process_data/@outputFileName appears in the Expression box.

- Click **OK**.

3 Specify the document variable to save the PDF file to by completing these tasks:

- Beside the **Document** property name, select **Variable**. The default is Literal Value.

- In the list below the **Document** property, select **outputPDFForm**.

4 Below the Make Unique property, select the **Append a suffix to the file name to make it unique if it has been used** check box.

5   In the **Output** property group and, in the **Result** list, select **outputFileName**.

6   Select **File** > **Save**.

7   Click the Applications view, right-click **PreLoanProcess,** and select **Check In.**

After you configure the Save PDF Form To Network (Document Write) operation, the property sheet looks like this illustration.
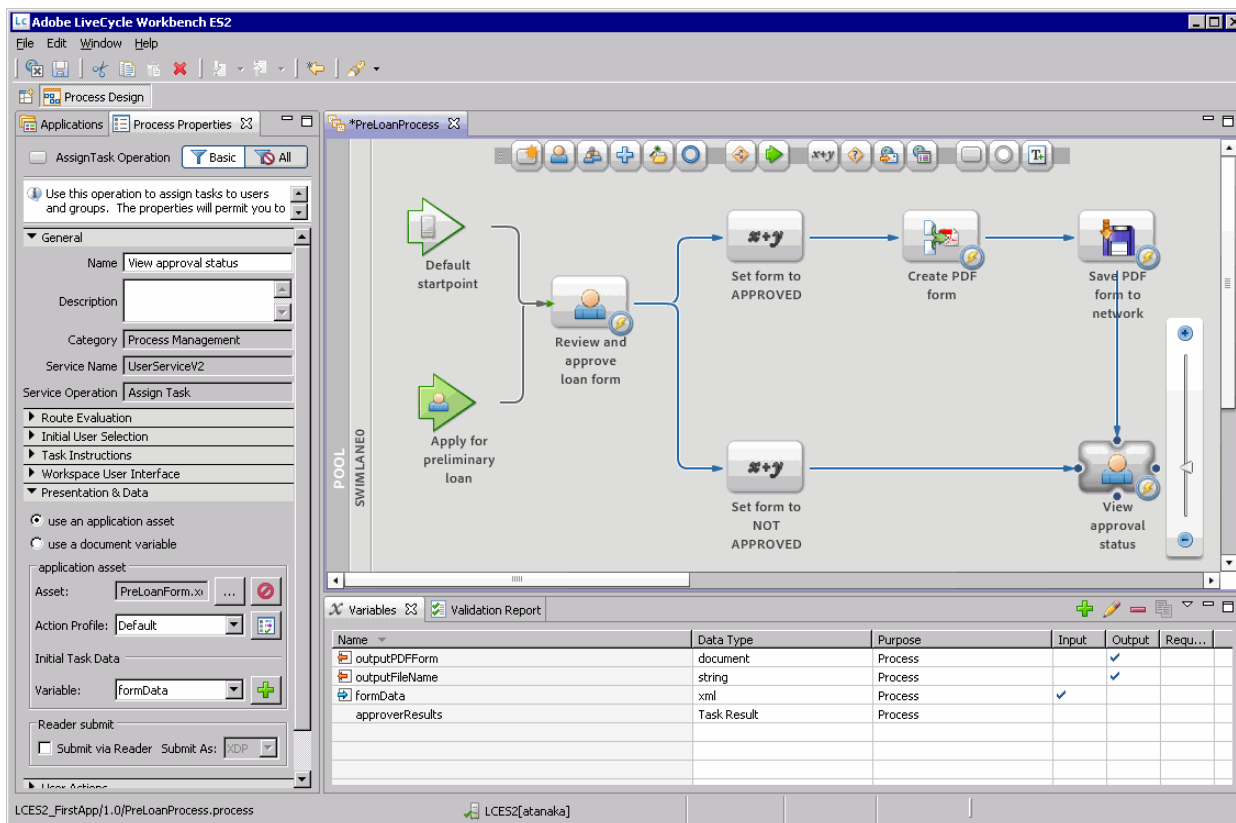

You have configured all the operations on the process diagram. Your completed process diagram looks like this illustration

Validating a process

Before you deploy and test your application, you can check whether your application has any design-time issues. As you create your process, validation occurs. You can see the results in the Validation Report view.

Check that process is valid

1   Select the **Validation Report** view and verify that no errors or warnings exist.

2   If warnings or errors exist, click the message, identify the problem, and resolve it. When necessary, go back to the task and review the steps to help identify the problem.

3   Right-click **FirstApp** > **FirstApp/1.0** > **PreLoanProcess** and select **Validate** to refresh the Validation Report view.

You have validated the process.

Adding a prebuilt process to an application

To skip the tasks for creating the process, import the process directly into your application.

Caution: Do not complete these steps if you completed "Creating a process" on page 24 and "Editing the process diagram" on page 25. Before you import the process, you must have a form design in your application; otherwise, you see errors.

Adding a prebuilt process to the application.

1   In Workbench, select **File** > **Import**. The Import wizard appears.

2   Select **General** > **File System**, and then click **Next**.

3   Beside the From Directory list, click **Browse**. The Import From Directory dialog box appears.

4   Navigate to the folder that you extracted the tutorial assets and click **OK**. (See "Downloading and extracting the tutorial assets" on page 6.)

5   In the right-pane, select **PreLoanProcess.process** and **PreLoanProcess.process_dependency**.

6   Beside the Into Folder box, click **Browse**. The Import Into Folder dialog box appears.

7   Select **LCES2_FirstApp** > **1.0**, click **OK**, and then click **Finish**.

8   In the Applications view, right-click **PreLoanProcess** and select **Check In**.

You have imported a prebuilt process. It would be a good idea to validate the imported process before you start the next module. (See "Validating a process" on page 36.).

Learning more

In this module, you learned how to draw a process diagram, design a human-centric process, add variables, configure operations, and configure start points in Workbench. The following resources provide additional information about the tasks you completed:

If you are using LiveCycle Workbench 9, see these topics in *Creating Processes Using LiveCycle Workbench Help* for more information about drawing process diagrams and configuring processes:

*   About processes

*   Creating processes

*   Designing human-centric processes

*   Service reference

If you are using LiveCycle Workbench 9.5, see these topics in *LiveCycle Workbench 9.5 Help* for more information about drawing process diagrams and configuring processes:

• Getting started with process design

• Creating and managing processes

• Designing human-centric processes

• Service reference

6. Deploying and configuring an application

Before you start this module, ensure that you completed the tasks in "Creating a process using Workbench" on page 24.

When you deploy an application, the processes in the application become available as *services*. A service has endpoints that permit you to invoke it. *Endpoints* are created using the start points specified in the process diagram. Default start points create Java EJB, Web Service (SOAP), LiveCycle Remoting, and REST endpoints that permit you to invoke a process programmatically. When you add a *Workspace start point* to your process diagram, a *Task Manager endpoint* is created after you deploy the application.

When an application is first deployed, only user profiles configured with the Application Administrator role can invoke the service. To configure the security for a service, use Applications and Services in LiveCycle Administration Console. In LiveCycle, administrators manage the security for applications.

In this module, you will deploy your application. After the application it deployed, it becomes a service that you will configure. To learn how to deploy your application and configure the security for a service, complete these tasks in the following order:

- Deploy the application as a service. (See "Deploying an application using Workbench" on page 38.)
- Configure the security for the service. (See "Configuring security on a service using LiveCycle Administration Console" on page 39.)

Note: To deploy your application to different LiveCycle servers for testing, move your application before you deploy it. (See "Moving an application" on page 50.)

Deploying an application using Workbench

Before you deploy an application, check in all the assets to upload them to the LiveCycle server. Checking in the assets ensures that when you deploy your application, it is using the most recent version of the assets.

Note: If you have assets checked out, you are prompted to check them in. Select Check In All Files and click OK.

Deploy an application

❖ In Workbench, in the Applications view, right-click **FirstApp** > **FirstApp/1.0** and select **Deploy**.

After you deploy your application, FirstApp/1.0 appears with a small, horizontal, green arrow on it as shown in this illustration.
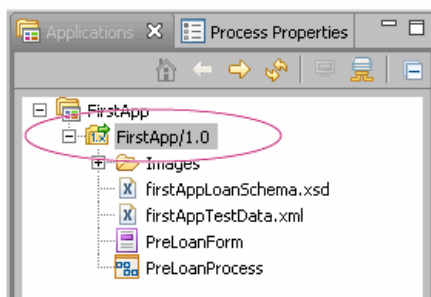
The process that you created becomes activated as a service. Endpoints are created after you deploy your application. A Task Manager endpoint is created because a Workspace start point exists on the process diagram.

Configuring security on a service using LiveCycle Administration Console

Although you can deploy an application in Workbench, you cannot test the endpoints when the users who access Workspace do not have Super Administrator, Application Administrator, or Workspace User privileges. End users typically do not have administrator privileges. To make your test environment more realistic, configure users profiles with the same privileges that you would provide in a typical production environment.

To permit users to invoke a service, especially if the users are in separate domains, configure access by using Applications and Services in LiveCycle Administration Console. In addition to configuring security on a service, Applications and Services permits you to complete these tasks:

- Manage applications or LiveCycle archive files
- Manage the categories displayed in Workspace
- Configure endpoints

Note: When the application is redeployed, endpoint settings that you configure in Applications and Services are reset to the values specified by the Workspace start point. However, security settings remain unchanged. Service security settings are removed if you undeploy the application.

Configure security on the FirstApp/PreLoanProcess service

1 Log in to LiveCycle Administration Console using a web browser. Type the URL `http://[server]:[port]/adminui`, where `[server]` represents the network name of the LiveCycle server and `[port]` represents the port number of the LiveCycle server. For example, type http://LC-server:8080/adminui.

2 Press Enter. The LiveCycle Administration Console login page appears.

3 On the Administration log screen, complete these steps:
   - In the **User ID** box, type `atanaka`.
   - In the **Password** box, type `password`.
   - Click **Login.**

   It is recommended that you use the Akira Tanaka (atanaka) user profile for this tutorial. (See "Understanding the sample environment" on page 6). If you use a different user profile, you can use it instead of the Akira Tanaka user profile.

4 On the LiveCycle Administration Console home page, click **Services** > **Applications and Services** > **Service Management**.

5 On the Service Management page, in the **Category** list, select **FirstApp**, and then click **Filter**.

6 In the filtered results, click **FirstApp/PreLoanProcess:1/0**.

7 On the Configure FirstApp/PreLoanProcess page, click the **Security** tab.

8 Click **Add Principal**.

9 On the Select Principal page, click the **All Principals In Domain SampleOrganization, PublicationsGroup** link. (You may need to click **Next** to see this link.)

10 Select **INVOKE_PERM** and click **Add**.

The Configure FirstApp/PreLoanProcess page looks like this illustration after All Principals In Domain SampleOrganization, PublicationsGroup is added.
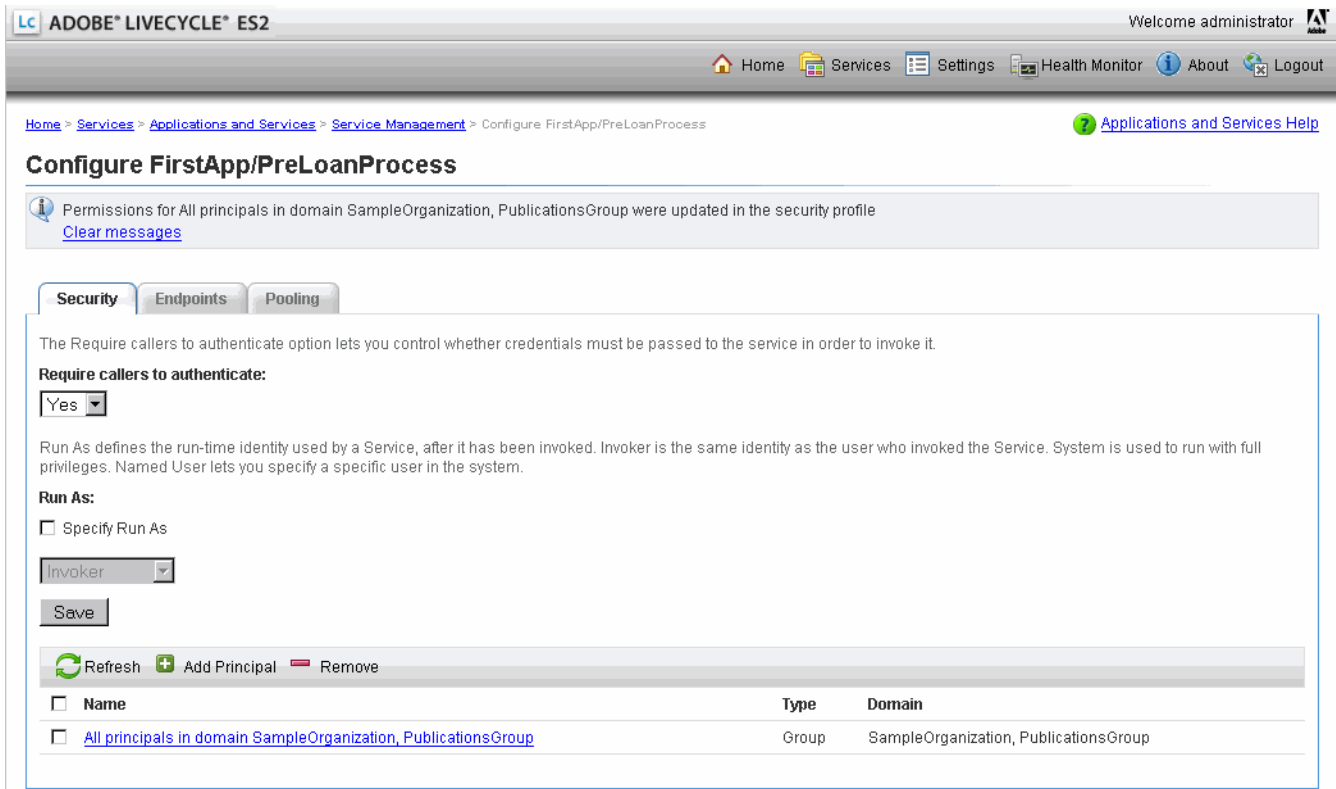
11 Click **Save** and click **Logout**.

You have configured your service so that all users who are part of the SampleOrganization, PublicationsGroup domain can invoke the PreLoanProcess service.

Learning more

In this module, you learned how to deploy an application and configure the service. The following resources provide additional information about the tasks you completed.

If you are using LiveCycle Workbench 9, see the following topic in *Application Development Using LiveCycle Workbench Help* at http://www.adobe.com/go/learn_lc_workbench_9 for information about deploying processes:

•   Deploying applications

If you are using LiveCycle Workbench 9.5, see the following topic in *LiveCycle Workbench 9.5 Help* at http://www.adobe.com/go/learn_lc_workbench_95:

•   Using Workbench > Working with applications > Deploying applications

See these topics in *LiveCycle ES2 Administration Help* at http://www.adobe.com/go/learn_lc_admin_9 for information about modifying security settings and configuring endpoints:

- Modifying security settings for a service
- Configuring Task Manager endpoints

7. Testing using in Workspace

Before you start this module, ensure that you have completed the tasks in .

You will test the human-centric process using Workspace. To learn how to use Workspace to test your service, complete these tasks in the following order:

Logging in to Workspace

To test your service in a more realistic manner, log in to Workspace using different user profiles:

User Name	User ID/Password	Description
Sarah Rose	srose/password	Represents the person applying for the loan.
John Jacobs	jjacobs/password	Represents the loan officer who reviews and approve loans.

Log in to Workspace:

1   In the URL line of a web browser, type `http://[server]:[port]/workspace` where `[server]` represents the network name of the LiveCycle server and `[port]` represents the port number of the LiveCycle server.

For example, on the computer where you installed LiveCycle ES or ES2.5 by using the turnkey method, in the URL line of the web browser, type `http://LC-server:8080/workspace`. The Workspace login screen appears.

2   In the **User ID** box, type the user ID to use and, in the **Password** box, type the password, and then click **Login**.

You have logged in to Workspace.

## Starting a preliminary loan application

In Workspace, you will fill a form to start the preliminary loan approval service by clicking a process card. The process card is accessible from the Loan category you were assigned while creating the process in Workbench.

Fill a form as Sarah Rose

1   Log in to Workspace as Sarah Rose. (See .) The Welcome screen appears.

2   On the Welcome screen, click **Start Process**.

3 On the **Start Process** tab, click **Loans**. The Apply For Preliminary Loan is displayed as a process card as shown in the following illustration. The process card contains the image you specified in Workbench when you created the process.
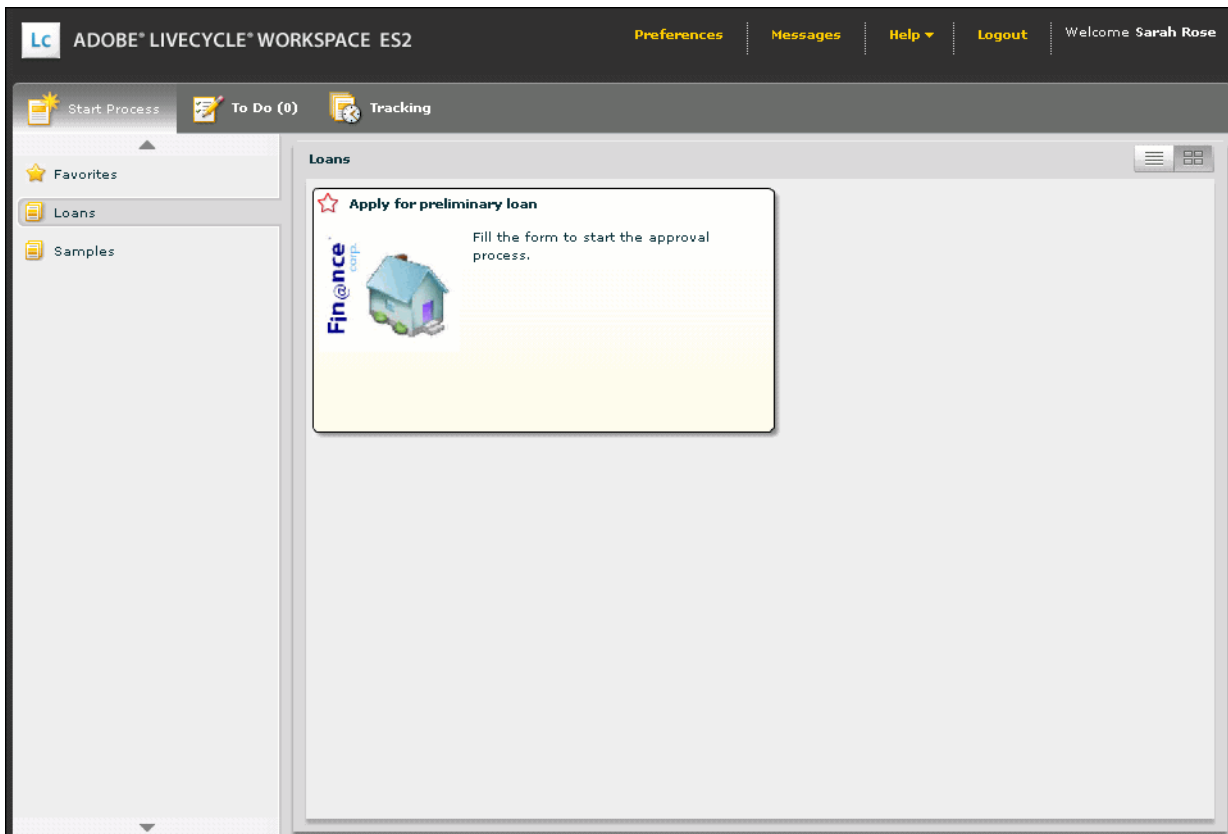
4 Click the **Apply for preliminary loan** process card. The Task Details and Form tabs appear and the Form tab is selected. On the Form tab, the form is opened using Adobe Reader.

5 Fill the form by completing these tasks:

- In the **Name** box, type `Sarah Rose`.
- In the **Loan Amount** box, type `100000`.
- In the **Phone Or Email** box, type `(123)123-1234`.

After you fill the PDF form, your completed form looks like this illustration.
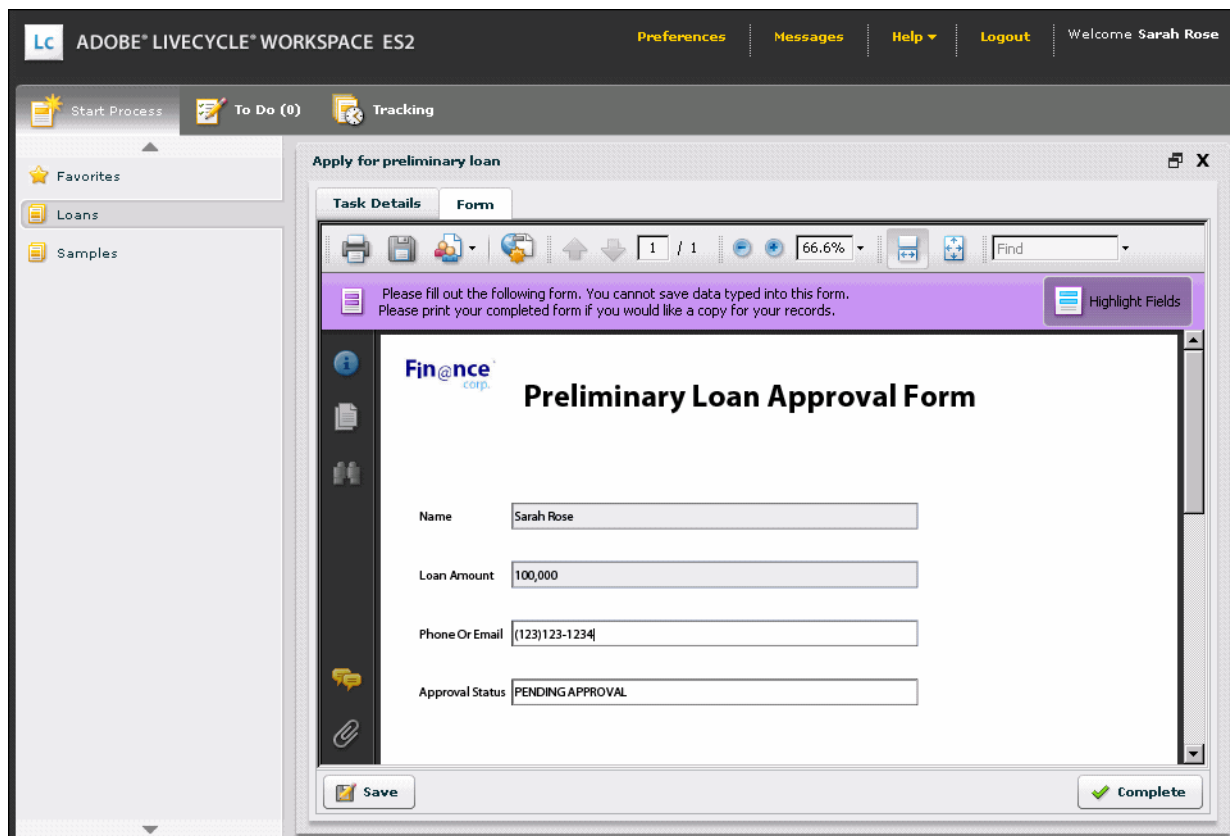
6   Click **Complete**.

Note: The Submit button is no longer in the form. The Submit button on the form is hidden so that the user submits forms by using buttons provided by Workspace.

7   Click **Logout**. A Please Confirm dialog box appears with the prompt: Are You Sure You Want To Logout?

8   Click **OK**.

You have filled a preliminary loan approval form.

Approving the preliminary loan form

When the form is sent to the loan officer, a task is displayed as an item in the To Do list. The loan officer approves or denies the loan request.

Log in as the loan officer and approve the loan

1   Log in to Workspace as John Jacobs. (See .) The Welcome screen appears.

2   On the Welcome screen, verify that a number appears in To Do, and then click **To Do**.

3 Click the **Review and approve loan form** process card. The Form tab displays the form to approve. Notice that Approve and Deny
buttons are displayed in this illustration:
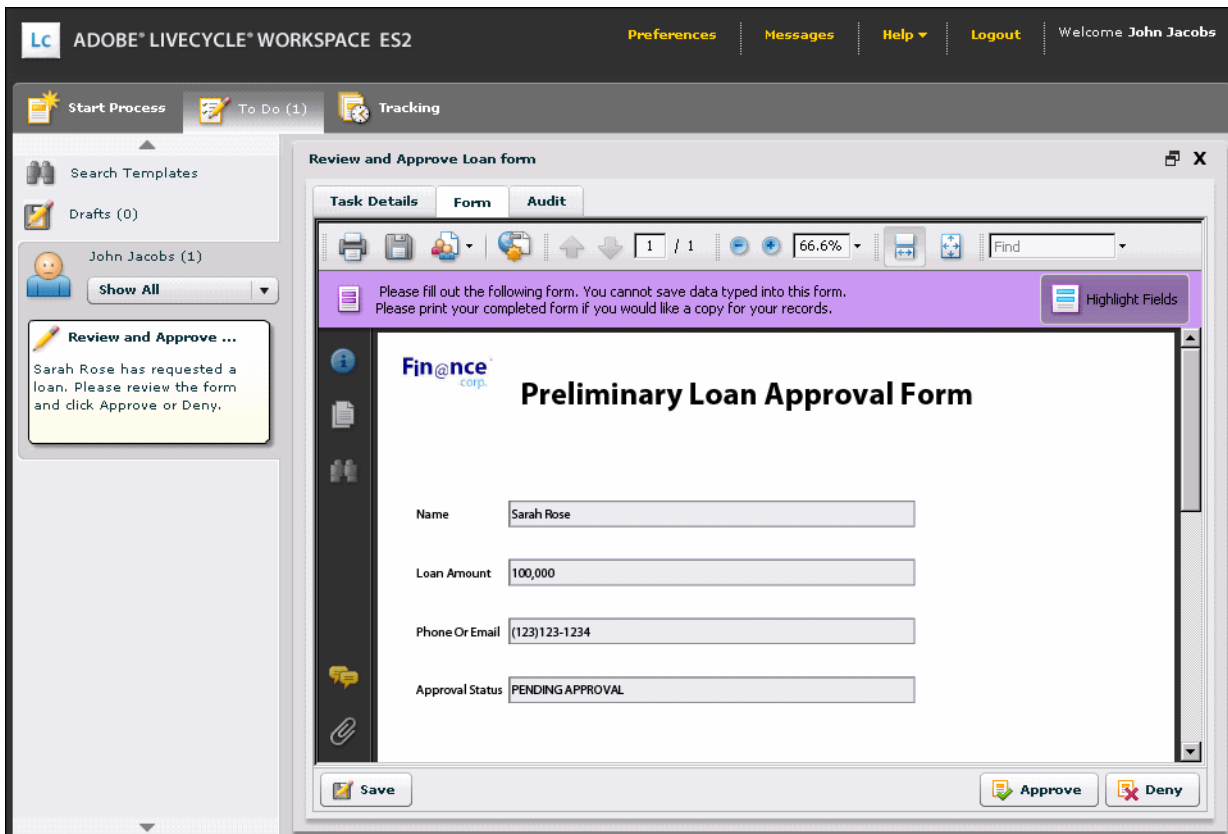
4 Click the **Approve** [Approve] button.

5 Click **Logout**. A Please Confirm dialog box appears with the prompt: Are You Sure You want To Logout?

6 Click **OK**.

You have approved the loan application as the loan officer, John Jacobs.

Verifying the loan status and confirming PDF file creation

Verify that the loan was approved and a PDF file created and archived to a network location.

Note: To view the archived PDF file, log in to the LiveCycle server.

Verify that the form was saved to a network location

1 On the LiveCycle server, navigate to C:\followuploans\ using Windows Explorer. If you provided another location, navigate to that
location on the computer or network.

2 Verify that a filename, approvedby_John Jacobs.pdf, exists as shown in this illustration.

Note: If you tested multiple times, multiple files named Approved_byJohn Jacobs with a unique number appended to the end of each file appear.
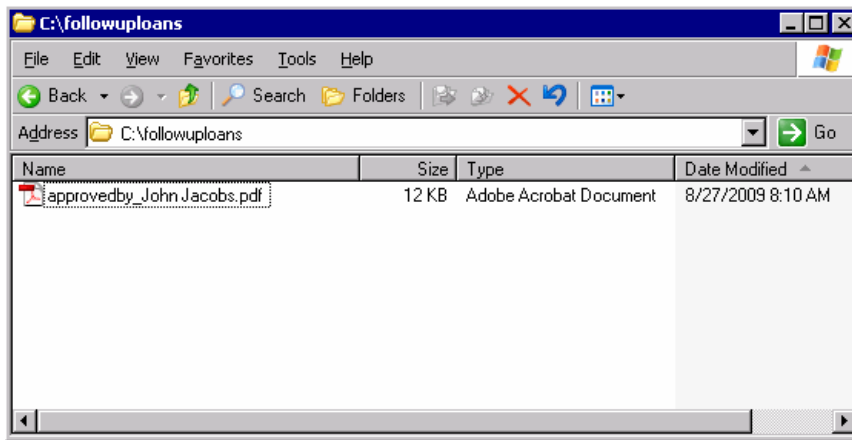
View the approval status

1 Log in to Workspace as Sarah Rose. (See .). The Welcome screen appears.

2 On the Welcome screen, confirm that a number (other than zero) appears in To Do, and click **To Do**.

3 Click the **View approval status** process card.

4 Verify that the Approval Status field is set to APPROVED. Now, the approved form looks like this illustration.

5 Click **Complete**.

You verified that the loan approval form was routed back to the user who applied for the loan. In addition, you verified that a PDF file was created and archived to a network location.

Congratulations! You have successfully completed this tutorial! You have created, deployed, and tested your first LiveCycle application. Optionally, you can record the process as it runs and view the recording. (See "Troubleshooting a process" on page 48.).

Learning more

In this module, you learned how test the application you created. The following resources provide additional information about the tasks you completed.

See these topics in *Application Development Using LiveCycle Workbench Help* at http://www.adobe.com/go/learn_lc_workbench_9*:*

- Starting processes
- Working with To Do lists

To further your understanding of application development and various topics described in this tutorial, see "Learning more about LiveCycle ES2.5" on page 52.

8. Troubleshooting a process

Before you start this module, ensure that you completed the tasks in "Deploying and configuring an application" on page 38.

Workbench provides an environment to debug and view the execution of a process. You will use Workbench to record the process as it executes and view the recording. To start and view a recording of a process, complete "Starting and viewing a recording" on page 48.

Starting and viewing a recording

You can record a process as it executes and view a recording of the process. If you have multiple processes in an application, you can record all of them. Viewing a process permits you to see the runtime values of variables as you step through the recording of the process execution. Viewing the runtime variable values allows you to troubleshoot and optimize the process.

Start and view a recording

1   In Workbench, right-click **FirstApp** > **FirstApp/1.0** > **PreLoanProcess** in the Application view, and select **Record and Playback** > **Start Recording**.

2   Test the process in Workspace 9. (See "Testing using in Workspace" on page 42.)

3   Return to Workbench and select **Record and Playback** > **Stop Recording**.

4   In the Applications view, right-click **FirstApp** > **FirstApp/1.0** > **PreLoanProcess** and select **Record and Playback** > **Play Process Recording**. In the Play Process Recording dialog box, an entry appears for the test you completed.

5 Select the process recording and click **OK**. The editor for your process appears as shown in this illustration.
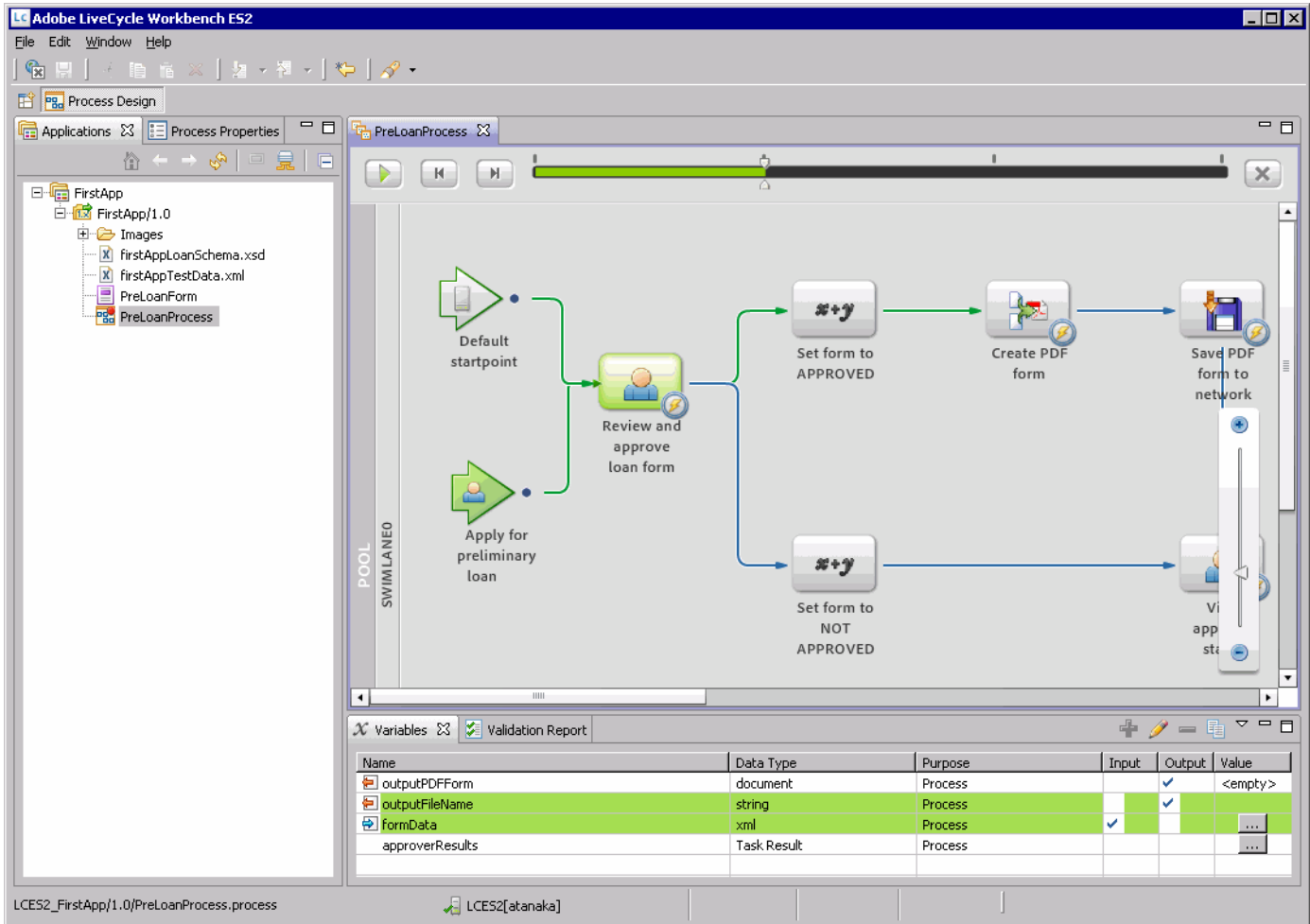
6 Click the **Play** ▶ button to play the entire recording or the N**e**xt ⊞ button and **Back** ⊞ button to step forward and backward through the process, respectively. When you finish viewing the recording, click the **Close** ⊠ button to exit the recording playback mode. As the process executes, values change in the Variables view. If an ellipsis button appears under the Value column for a variable, you can click it to see the value.

Because your process completed successfully, the colored routes between each operation are green. If a step fails to complete, the route turns red.

Learning more

In this module, you learned how to record a process as it runs and view the recording. The following resources provide additional information about the tasks you completed:

See the following topic in *Application Development Using LiveCycle Workbench Help* at http://www.adobe.com/go/learn_lc_workbench_9:

• Recording and playing back processes

You have completed all the sections in this tutorial. To further your understanding of application development and various topics described in this tutorial, see "Learning more about LiveCycle ES2.5" on page 52.

9. Moving an application

Before you start this module, ensure that you completed the tasks in "Deploying and configuring an application" on page 38.

*Caution: Do not complete the tasks in this module if you are using the **same** LiveCycle server for development and testing.*

When your development environment is not suited for testing with realistic scenarios, move the application to a different LiveCycle server. To learn to move your application, complete these tasks in the following order:

- Export the application to an archive file. (See "Exporting an application" on page 50.)
- Import the archive file to a LiveCycle server. (See "Importing an application" on page 50.)

Exporting an application

Use Workbench to export a deployed application. The application is exported as an archive file, which contains the latest versions of all assets in the application.

Export an application:

1   In Workbench, in the Applications view, right-click **FirstApp** > **FirstApp/1.0**, and select **Create LiveCycle Archive**. The Create New LiveCycle Archive wizard appears.

2   In the Create An Application Archive pane, select the **Create a new archive file option**, and click **Next**.

Note: If you have files checked out, you are prompted to check in all files before you can export your application.

3   In the Archive Properties pane, verify that the **FirstApp** and **FirstApp/1.0** check boxes are selected.

4   Beside the Local File System Destination box, click the ellipsis ⋯ button. The Save As dialog box appears.

5   Navigate to a location on your computer where you want to save your file, such as C:\.

6   In the **File name** box, type a name, such as `FirstAppEx`, to replace the default name.

7   Click **Save** and then click **Finish**.

8   When the Export Success dialog box appears, click **OK**.

You have exported a deployed application as an archive file. You can check the file system to see the .lca file.

Importing an application

You will use the Applications and Services page in LiveCycle Administration Console to import an application to the LiveCycle server. When you log in to LiveCycle Administration Console, the user profile that you use must be assigned either the Application Administrator or Super Administrator role.

Log in to LiveCycle Administration Console to import archive file

1   In the URL line of a web browser, type `http://[server]:[port]/adminui` where `[server]` represents the network name of the LiveCycle server and `[port]` represents the port number of the LiveCycle server.

For example, in the URL line of the web browser, type `http://localhost:8080/adminui`.

2   Press Enter.

3   On the Administration log screen, complete these steps:

- In the **User ID** box, type `atanaka`
- In the **Password** box, type `password`
- Click **Login**.

It is recommended that you use the Akira Tanaka (atanaka) user profile for this tutorial. (See "Understanding the sample environment" on page 6). If you use a different user profile, you can use it instead of the Akira Tanaka user profile.

4 On the LiveCycle Administration Console home page, click **Services** > **Applications and Services** > **Application Management**.

5 On the Application Management page, click **Import**.

6 On the Application/Archive Import page, click **Browse**. The File Upload dialog box appears.

7 On your computer, navigate to the location of the Application archive file such as C:\FirstAppEx.lca, click **Open**, and then click **Preview**.

8 On the Preview Application page, select the **Deploy assets to runtime when import is complete** check box, and then click **Import**.

9 On the Application Management Page, verify that the FirstApp application is deployed and click **Logout**. If the FirstApp application successfully deployed, a green check mark in the Deployed column appears beside it, as shown in this illustration.
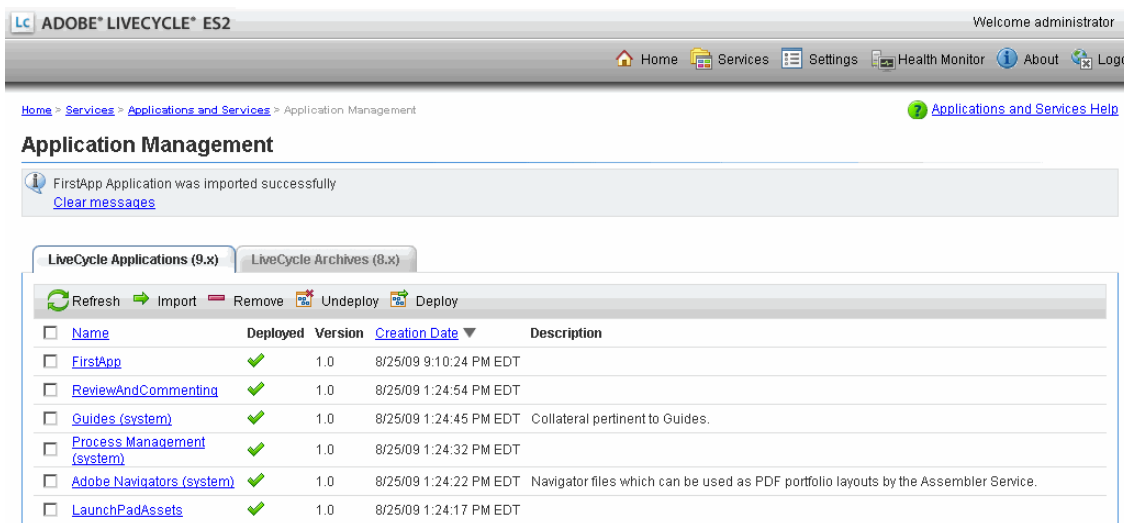
You have imported the application into another LiveCycle server.

Learning more

In this module, you learned how to move an application from one server to another. The following resources provide additional information about the tasks you completed.

For information about validating and deploying processes, see the following topic in *Application Development Using LiveCycle Workbench Help* at http://www.adobe.com/go/learn_lc_workbench_9:

- Preserving process results using Archive Wizard
- Moving applications into another environment

For more information about importing and configuring applications, see the following topic in *LiveCycle ES2 Administration Help* at http://www.adobe.com/go/learn_lc_admin_9:

- Importing and managing applications and archives

10. Learning more about LiveCycle ES2.5

This tutorial provided you an introduction to the process of developing an end-to-end application in LiveCycle. To learn more, refer to the list of suggested resources. These resources provide the next steps to improve your understanding of LiveCycle application development.

Related Resources

Workbench
If you are using LiveCycle Workbench 9, see *Creating Processes Using LiveCycle Workbench Help* as well as *LiveCycle Workbench 9.5 Help.*

If you are using LiveCycle Workbench 9.5, see *LiveCycle Workbench 9.5 Help.*

You will also find these videos useful:

- Introduction to LiveCycle ES2 Applications
- Introduction to LiveCycle Workbench ES2
- Introduction to LiveCycle ES2 Processes
- Troubleshooting Processes in the Development Environment
- Using Models in LiveCycle ES2 Applications

Designer
Additional information about Designer is available in *LiveCycle Designer 9 Help*.

These topics are suggested for further exploration:

- Working with form designs: Learn more about designing forms.
- Scripting: Learn about using scripts in form designs.

Learn more about designing forms by completing this tutorial:

- *Creating a purchase order form*

You will also find these videos useful:

- Introduction to Action Builder
- Script to Add Sub-Totals
- Introduction to Patterns and Formatting Field Values

LiveCycle Administration Console
You can learn about how to administer LiveCycle in *LiveCycle ES2 Administration Help*.

These topics are suggested for further exploration:

- Configure and manage applications and services: Learn more about managing and configuring applications.
- Configuring Workspace: Learn more about how to administer Workspace.
- Configuring LiveCycle Process Management ES2: Learn more about administering processes and tasks that are assigned to users.

Workspace
You can learn more about using Workspace in *LiveCycle Workspace Help*

Other resources

- LiveCycle documentation site at http://www.adobe.com/go/learn_lc_documentation_9: The location to access more LiveCycle documentation.

- LiveCycle Services at http://www.adobe.com/go/learn_lc_services_9: A description of all the services that can be installed on LiveCycle.

- LiveCycle DevNet at http://www.adobe.com/devnet/livecycle: A resource for developing with LiveCycle

- LiveCycle samples at http://www.adobe.com/go/learn_lc_samples_9: A resource for samples that you can import and for various code examples

- LiveCycle Doc team blog at http://blogs.adobe.com/livecycledocs/: Updates on LiveCycle straight from the documentation team