Notices

For details, see the following topics:

- Documentation copyright notice

Documentation copyright notice

©2013 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

These materials and all Progress® software products are copyrighted and all rights are reserved by Progress Software Corporation. The information in these materials is subject to change without notice, and Progress Software Corporation assumes no responsibility for any errors that may appear therein. The references in these materials to specific platforms supported are subject to change.
Notices

Apama, Business Empowerment, Business Making Progress, Corticon, Corticon (and design), DataDirect (and design), DataDirect Connect, DataDirect Connect64, DataDirect XML Converters, DataDirect XQuery, Empowerment Center, Fathom, Making Software Work Together, OpenEdge, Powered by Progress, PowerTier, Progress, Progress Control Tower, Progress Dynamics, Progress Business Empowerment, Progress Empowerment Center, Progress Empowerment Program, Progress OpenEdge, Progress Profiles, Progress Results, Progress RPM, Progress Software Business Making Progress, Progress Software Developers Network, ProVision, PS Select, RulesCloud, RulesWorld, SequeLink, SpeedScript, Stylus Studio, Technical Empowerment, WebSpeed, Xcalia (and design), and Your Software, Our Technology–Experience the Connection are registered trademarks of Progress Software Corporation or one of its affiliates or subsidiaries in the U.S. and/or other countries. AccelEvent, Apama Dashboard Studio, Apama Event Manager, Apama Event Modeler, Apama Event Store, Apama Risk Firewall, AppsAlive, AppServer, BusinessEdge, Cache-Forward, DataDirect Cloud, DataDirect Spy, DataDirect SupportLink, Future Proof, High Performance Integration, OpenAccess, ProDataSet, Progress Arcade, Progress ESP Event Manager, Progress ESP Event Modeler, Progress Event Engine, Progress RFID, Progress Responsive Process Management, Progress Software, PSE Pro, SectorAlliance, SeeThinkAct, SmartBrowser, SmartComponent, SmartDataBrowser, SmartDataObjects, SmartDataView, SmartDialog, SmartFolder, SmartFrame, SmartObjects, SmartPanel, SmartQuery, SmartViewer, SmartWindow, WebClient, and Who Makes Progress are trademarks or service marks of Progress Software Corporation and/or its subsidiaries or affiliates in the U.S. and other countries. Java is a registered trademark of Oracle and/or its affiliates. Any other marks contained herein may be trademarks of their respective owners.

See Table of Contents for location of Third party acknowledgements within this documentation.
# Table of Contents

**Preface**...........................................................................................................13  
  About this documentation.................................................................13  
  Documentation conventions...........................................................13  
  Product support contact information................................................14  

**Chapter 1: Introduction to Progress OpenEdge Business Process Modeler.**17  
  Business Process Modeler supported platforms...........................................18  
  System requirements..................................................................................18  

**Chapter 2: Getting started with Business Process Modeler.................**19  
  Starting Business Process Modeler..........................................................19  
  Exploring Business Process Modeler......................................................21  
    Using the Business Process Modeler menubar.....................................22  
    Business Process Modeler Tasks pane................................................25  
    Business Process Modeler palette.......................................................27  
    Using keyboard shortcuts.................................................................28  
  Localizing Business Process Modeler....................................................29  

**Chapter 3: Developing Business Process Modeler projects.................**31  
  Project facets.........................................................................................31  
    Project facet properties.......................................................................32  
  Creating a BPM project...........................................................................32  
    Creating a BPM process......................................................................33  
    Creating a Web application.................................................................37  
  Managing multiple processes...............................................................38  
    Importing a process file......................................................................39  
    Adding a new BPM process...............................................................41  
    Adding a new Web application...........................................................41  
    Importing a process file......................................................................42  
    Using a process file as a template......................................................43  
      Using a process file from recent releases........................................43  
      Importing XPDL files........................................................................43  
    Using a project as a template.............................................................44  
    Creating a simulation project..............................................................45  

**Chapter 4: Designing a process template diagram.................................**49
Creating a process template diagram....................................................................................................49
Changing shapes...................................................................................................................................52
Connecting worksteps.............................................................................................................................53
Using multiple links ...............................................................................................................................55
Adding notes.........................................................................................................................................57

Chapter 5: Managing gestures..................................................................................................................59
Using mouse gestures...............................................................................................................................60
Creating custom gestures.........................................................................................................................61
Using the Gesture Editor...........................................................................................................................61
Saving custom gestures..............................................................................................................................64
Using custom gestures...............................................................................................................................64

Chapter 6: Performing additional Business Process Modeler operations.................................................65
Publishing the process...............................................................................................................................65
Managing pages.......................................................................................................................................66
Generating 360 degree view.......................................................................................................................66
Opening a subprocess.................................................................................................................................70
Printing the process template...................................................................................................................70
Exporting the process template...............................................................................................................71
Generating a process summary..................................................................................................................72
Managing phases.....................................................................................................................................76
Creating a phase......................................................................................................................................76
Resizing a phase......................................................................................................................................78
Performing additional phase operations...................................................................................................78

Chapter 7: Working with the Overview page............................................................................................79
Introducing the Overview page..................................................................................................................79
Reviewing the Overview page....................................................................................................................80
Managing tasks.......................................................................................................................................82
Adding a task.........................................................................................................................................82
Performing other operations.....................................................................................................................83

Chapter 8: Introducing Path Analysis.......................................................................................................85
Using the Path Analysis view.....................................................................................................................85
Exploring the Path Analysis view..............................................................................................................86

Chapter 9: Defining process properties.....................................................................................................89
Setting process properties.........................................................................................................................89
Setting Business Process Modeler preferences.........................................................................................91
Using the Activity page..............................................................................................................................92
Chapter 13: Setting workstep properties

About worksteps..............................................................................................................218
Workstep properties...........................................................................................................220
Defining properties of Start workstep....................................................................................220
Specifying general properties of Start workstep.................................................................221
Using the Fields tab of Start workstep properties...............................................................222
Using the Messaging tab of Start workstep properties.......................................................223
Defining properties of Activity worksteps..............................................................................224
Specifying general properties of Activity workstep............................................................224
Using the Fields tab of Activity workstep properties.........................................................228
Defining collaboration tasks.................................................................................................228
Defining properties of Adapter worksteps............................................................................230
Specifying general properties of Adapter workstep............................................................230
Using the Dataslots tab of Adapter workstep properties....................................................231
Defining properties of Subprocess worksteps....................................................................233
Specifying general properties of Subprocess workstep......................................................233
Using Dataslot tab of Subprocess workstep properties......................................................234
Defining properties of Message worksteps...........................................................................235
Defining properties of End worksteps..................................................................................235
Using Messaging tab of End workstep properties..............................................................236
Using the Dataslots tab for Web applications....................................................................236
Defining properties of Decision gateways..........................................................................236
Defining Exclusive Decisions..............................................................................................237
Defining properties of Or-Join gateways.............................................................................238
### Chapter 14: Introducing Monitoring process

- Designing a monitoring process
- Configuring monitoring workstep
- Using dataslots

### Chapter 15: Defining workstep presentation format

- Defining workstep presentation
- Defining Auto-generated HTML presentations
- Defining custom presentations
- Defining Flow presentations
- Using the Form presentation
- Defining Mobile presentation
- Defining GUI presentation
- Defining Tablet presentation
- Defining Other presentation

### Chapter 16: Using alerts

- Reviewing the Alerts tab
- Defining an alert
- Associating an alert with a workstep

### Chapter 17: Using Form Editor

- Opening the Form Editor
- Using a form template
- Setting form properties
- Using the Layout tab
- Defining the form's layout
- Adding a table
- Adding a tabbed pane
- Adding a Field Set
Chapter 18: Configuring and running simulation

Adding a process model.................................................................340
Exploring the Progress OpenEdge BPM Simulation perspective........342
Configuring a simulation scenario.................................................343
  Adding a scenario......................................................................344
  Modifying simulation settings for a scenario..............................344
  Importing and exporting a scenario.........................................345
  Configuring simulation parameters for a process......................349
  Managing performers...............................................................355
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managing resources</td>
<td>357</td>
</tr>
<tr>
<td>Defining objectives</td>
<td>358</td>
</tr>
<tr>
<td>Setting calendar</td>
<td>360</td>
</tr>
<tr>
<td>Performing other operations</td>
<td>361</td>
</tr>
<tr>
<td>Choosing a randomization method</td>
<td>361</td>
</tr>
<tr>
<td>Specifying a date</td>
<td>361</td>
</tr>
<tr>
<td>Running a process simulation</td>
<td>362</td>
</tr>
<tr>
<td>Viewing process information</td>
<td>365</td>
</tr>
<tr>
<td>Viewing project information</td>
<td>367</td>
</tr>
<tr>
<td>After completion of simulation</td>
<td>370</td>
</tr>
<tr>
<td>Generating a simulation report</td>
<td>371</td>
</tr>
<tr>
<td>Viewing simulation report details</td>
<td>373</td>
</tr>
<tr>
<td>Comparing simulation results</td>
<td>377</td>
</tr>
<tr>
<td>Chapter 19: Working with User Management tool</td>
<td>381</td>
</tr>
<tr>
<td>Types of members</td>
<td>382</td>
</tr>
<tr>
<td>Starting User Management tool</td>
<td>383</td>
</tr>
<tr>
<td>Printing organization chart</td>
<td>384</td>
</tr>
<tr>
<td>Managing nodes</td>
<td>384</td>
</tr>
<tr>
<td>Managing users</td>
<td>385</td>
</tr>
<tr>
<td>Managing groups</td>
<td>387</td>
</tr>
<tr>
<td>Managing queues</td>
<td>389</td>
</tr>
<tr>
<td>Importing users</td>
<td>390</td>
</tr>
<tr>
<td>Chapter 20: Working with Business Calendar</td>
<td>391</td>
</tr>
<tr>
<td>Using the Business Calendar tool</td>
<td>392</td>
</tr>
<tr>
<td>Managing calendars</td>
<td>392</td>
</tr>
<tr>
<td>Managing schedules</td>
<td>393</td>
</tr>
<tr>
<td>Chapter 21: Managing business objects</td>
<td>397</td>
</tr>
<tr>
<td>Using the Business Object Manager</td>
<td>397</td>
</tr>
<tr>
<td>Importing business objects</td>
<td>401</td>
</tr>
<tr>
<td>Exporting a business object</td>
<td>404</td>
</tr>
<tr>
<td>Using business objects in a link</td>
<td>405</td>
</tr>
<tr>
<td>Using business objects in applications</td>
<td>405</td>
</tr>
<tr>
<td>Using business objects in a form</td>
<td>405</td>
</tr>
<tr>
<td>Using business objects in scripts</td>
<td>410</td>
</tr>
<tr>
<td>Chapter 22: Product support contact information</td>
<td>413</td>
</tr>
<tr>
<td>Chapter 23: Documentation copyright notice</td>
<td>415</td>
</tr>
</tbody>
</table>
Chapter 24: Business Process Modeler tutorial

Designing a process template diagram
Reviewing the process requirements
Defining the process flow
Creating the process model
Defining the process information flow
Defining process performers
Defining workstep properties
Assigning dataslots to worksteps
Saving and installing the process
Installing the published application
Creating an instance and performing a task

Chapter 25: Tutorial: Using the Graphical Event Logic (GEL) Editor

Step 1: Designing the form
Step 2: Adding the events
Step 3: Configuring actions

Appendix A: Third-party acknowledgments
Preface

For details, see the following topics:

- About this documentation
- Documentation conventions
- Product support contact information

About this documentation

This guide provides documentation for Progress OpenEdge Business Process Modeler, version 11.3. Progress OpenEdge Business Process Modeler is a component of the Progress OpenEdge Business Process Server line of software products.

Progress OpenEdge Business Process Modeler is a graphical, easy-to-use tool that enables line-of-business analysts and other business users to design processes, run simulations of them, and to store and retrieve processes from a central shared repository.

Documentation conventions

This document uses the following conventions and terminology notations.
**Table 1: Conventions in this manual**

<table>
<thead>
<tr>
<th>Convention (styles and terms)</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>bold</strong></td>
<td>Indicates titles of command buttons, check boxes, options, lists, dialog boxes, and portal page names.</td>
</tr>
<tr>
<td><strong>bold italic</strong></td>
<td>Represents notes that alert you on specific Business Process Server elements and similar advisory information.</td>
</tr>
<tr>
<td><strong>italic</strong></td>
<td>Indicates folder paths, file names, and book titles.</td>
</tr>
<tr>
<td><strong>monospace</strong></td>
<td>Represents code segments or examples.</td>
</tr>
<tr>
<td>backward slash “\”</td>
<td>Indicates the path in Windows environment. For UNIX environment, replace with forward slash “/”.</td>
</tr>
<tr>
<td>Modeler_Home</td>
<td>Represents the installation folder of OpenEdge Business Process Modeler.</td>
</tr>
<tr>
<td>Workspace_Home</td>
<td>Represent the Workspace folder of Business Process Modeler located in the Modeler_Home folder.</td>
</tr>
</tbody>
</table>

**Product support contact information**

If the product documentation does not provide a solution to your specific issue, or if you need clarification on the issue, then contact our Product Support team. You can contact the team through the Internet, telephone, or postal mail, as per the details provided in Table 2 on page 14.

**Table 2: Product Support Contact Information**

<table>
<thead>
<tr>
<th>To contact by</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web site</td>
<td><a href="http://progresslink.progress.com/supportlink">http://progresslink.progress.com/supportlink</a></td>
</tr>
<tr>
<td></td>
<td>If you are an existing customer, then you can log in to the above site for product support. If you are a first time user, then you need to create an account first.</td>
</tr>
<tr>
<td>Telephone¹</td>
<td>1-781-280-4999 for US, Latin America and Canada</td>
</tr>
<tr>
<td></td>
<td>1-781-280-4543 for the Product Support Fax Line</td>
</tr>
<tr>
<td>Postal Address¹</td>
<td>Progress Software Corporation</td>
</tr>
<tr>
<td></td>
<td>14 Oak Park Drive</td>
</tr>
<tr>
<td></td>
<td>Bedford, MA 01730, USA.</td>
</tr>
</tbody>
</table>

To enable us to quickly answer your questions, please provide the following information:

- Your name, installation site address and the license key for Business Process Server software.
- Your Business Process Server version and build number.

¹ For support telephone numbers and offices in your region, visit the support web site above. This contact information is for customer support only.
• Your operating system, application server and browser, with version and service pack details, if any.

• Your database management system and version, and information on JVM and JDBC used.
Introduction to Progress OpenEdge Business Process Modeler

This chapter introduces Progress OpenEdge Business Process Modeler (herewith referred to as Business Process Modeler), a tool which enables you to:

- Design a business process using the worksteps, connectors, and other features provided in the Business Process Modeler user interface. A process must contain a Start workstep, intermediate worksteps that can include Activity, Adapter and Subprocess worksteps, and one or more End worksteps.

- Define performers for the new process. These process-level performers are available for each instance of the process.

- Assign performers to worksteps. Types of performers include individual users, groups of users, or queues that perform Activity worksteps; adapters that perform Adapter worksteps; and other processes that act as performers of Subprocess worksteps.

- Define the information flow for the new process by creating dataslots, which can provide information into (inputs) or out of (outputs) worksteps.

- Define the presentation (look and feel) for Activity worksteps through the Form Editor.

- Configure and run simulations of the process to analyze performer and resource usage, to identify repetitive loops and inefficient bottlenecks, and to predict process performance. Business Process Modeler also allows you to generate an HTML report on the simulation results.

- Save the process to your local server after completing process design, or save it to the Process Repository, where it can be accessed by other Business Process Modeler users. You can also...
retrieve the latest version of other processes that have been stored in the Process Repository, and use them as templates when preparing a new process.

- Open the saved process in Progress Developer Studio for OpenEdge, a Business Process Server component that enables users to expand the process design by adding such features as conditions that execute Decision gateways, rules that further define the process, and advanced workstep properties that include excluding performers and setting overdue actions. From Progress Developer Studio for OpenEdge, the process can be published to Business Process Portal, where it is available to users as an installed application.

For details, see the following topics:

- Business Process Modeler supported platforms
- System requirements

**Business Process Modeler supported platforms**

Business Process Modeler has been extensively tested on the platforms, tools, and applications supported by Progress Business Process Server 11.3. Table 3 on page 18 provides Business Process Modeler-specific supported platforms.

**Table 3: Process Modeler Supported Platforms**

<table>
<thead>
<tr>
<th>Category</th>
<th>Software</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Process Server Version</td>
<td>11.3</td>
</tr>
<tr>
<td>Operating Systems</td>
<td>MS Windows 2000/Vista/XP Professional SP2 and 7</td>
</tr>
</tbody>
</table>

**System requirements**

Table 4 on page 18 describes the requirements for installing and running Business Process Modeler on Windows.

**Table 4: System requirements**

<table>
<thead>
<tr>
<th>Item</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Free disk space</td>
<td>500 MB free disk space</td>
</tr>
<tr>
<td>Business Process Modeler installation/operation</td>
<td>1 GB of RAM</td>
</tr>
<tr>
<td>Processor power</td>
<td>1GHz</td>
</tr>
</tbody>
</table>
Getting started with Business Process Modeler

You can use Business Process Modeler to design templates for basic business processes and store or retrieve them from the process repository. Business Process Modeler also enables you to manage performers at the process or workstep level and to design and run simulations of processes and individual worksteps.

For details, see the following topics:

- Starting Business Process Modeler
- Exploring Business Process Modeler
- Localizing Business Process Modeler

Starting Business Process Modeler

Business Process Modeler uses the Eclipse-based Integrated Development Environment (IDE), a comprehensive cross-platform environment to provide a stable, universal platform for application development. This open, scalable, and standards-based development environment reduces the time spent in application development and supports a user’s choice of version control system and application server. The current version of Business Process Modeler is based on the Eclipse Foundation SDK (version 3.7) IDE.
To start Business Process Modeler:

1. From the **Start** menu, select **Programs > Progress > OpenEdge > Process Modeler**, to open a blank Business Process Modeler GUI in the Progress OpenEdge Business Process Modeler perspective.

   **Figure 1: Blank Business Process Modeler interface**

2. Use the **File** menu to perform the following operations:

   **Table 5: Process Modeler Operations**

<table>
<thead>
<tr>
<th>Submenu</th>
<th>Function</th>
<th>For more information, see</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>New &gt; BPM Project</strong></td>
<td>Creates a new BPM project.</td>
<td><strong>Creating a BPM project</strong> on page 32</td>
</tr>
<tr>
<td><strong>New &gt; Process Model</strong></td>
<td>Creates a new BPM Process or Web application.</td>
<td><strong>Creating a BPM process</strong> on page 33 and <strong>Creating a Web application</strong> on page 37.</td>
</tr>
<tr>
<td>or <strong>New &gt; Web Application</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>New &gt; Simulation Project</strong></td>
<td>Creates a simulation project.</td>
<td><strong>Creating a simulation project</strong> on page 45.</td>
</tr>
<tr>
<td><strong>Open</strong></td>
<td>To browse to an existing process in your <strong>Workspace_Home folder</strong>; or in another location of your existing processes.</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** To access or use Business Process Modeler, you must have a valid license. For the stand-alone Business Process Modeler, your administrator can grant you specific permissions. For more information on Licenses, contact our Product Support Team.
Exploring Business Process Modeler

This section describes the Business Process Modeler interface including its views, menus, and toolbars. Figure 2 on page 21 displays a sample process template created for a process model.

Figure 2: The Business Process Modeler User Interface

Figure 2 on page 21 identifies the commonly used views and components, and these are further described in Table 6 on page 21.

Table 6: Process Modeler components

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Menu bar</td>
<td>Provides access to Business Process Modeler functions. For details, see Using the Business Process Modeler menubar on page 22.</td>
</tr>
<tr>
<td>Toolbar</td>
<td>Consists of icons providing shortcuts to commonly used functions.</td>
</tr>
<tr>
<td>Project Explorer View</td>
<td>Displays the contents of active projects. It consists of a hierarchical view of all the files that make up the active project.</td>
</tr>
<tr>
<td>Palette Bar</td>
<td>The bar above the Content pane includes icons for selecting, connecting, grouping and moving the elements in the process template diagram. For details, see Business Process Modeler palette on page 27.</td>
</tr>
<tr>
<td>Business Process Modeler Tasks pane</td>
<td>The Tasks pane contains six options—Select and Change Layout, Create Phases and Swim Lanes, Draw Shapes, Connect Shapes, Assign Participants, Create Annotations. These options help you in the design of a process model. For details, see Business Process Modeler Tasks pane on page 25.</td>
</tr>
</tbody>
</table>
The Content pane contains the main work space in which the selected project or file is displayed. The Content pane also includes:

- **File tabs**, located at the top of the pane, which display the opened files. Click on a File tab to display the file in the Content pane.

- **Content pane tabs**, located at the bottom of the pane, which include:
  - **Diagram** tab, which displays the process template.
  - **Overview** tab, which displays the process elements in a tabular format.
  - **Process Analysis** tab, using which you can perform process path analysis.
  - **Dataslots** tab, where you can define process dataslots.
  - **Performers** tab, where you can define performers at the process level.
  - **Alerts** tab, where you can define and manage Alerts.
  - An arrow, at the bottom right corner of the Content pane, which provides a bird’s eye view of the entire process template in a 3x3 grid. You can drag the purple box (showing the current process view) to navigate to a specific process view location.

<table>
<thead>
<tr>
<th>Properties View</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Click your cursor in a blank spot in the diagram to display process properties. You can also click a workstep to view the properties of any selected workstep.</td>
</tr>
</tbody>
</table>

### Using the Business Process Modeler menubar

Table 7 on page 22 lists the components and commands available in the Business Process Modeler toolbar.

**Table 7: Toolbar Components and Commands in Business Process Modeler**

<table>
<thead>
<tr>
<th>Component Name</th>
<th>Icon / Shortcut</th>
<th>Menu</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td><img src="icon.png" alt="Icon" /></td>
<td>File &gt; New</td>
<td>Select Process Model, Web application, or Simulation Project option to open the appropriate wizard to create the project (as discussed in Developing Business Process Modeler projects on page 31). You can also launch the wizard for Process Model by clicking <strong>Process Model</strong> (&gt;Create new simulation project&lt; icon or the Simulation Project wizard by clicking <strong>Create new simulation project</strong> (Create new simulation project&lt; icon.</td>
</tr>
<tr>
<td>Open</td>
<td><img src="icon.png" alt="Icon" /></td>
<td>File &gt; Open</td>
<td>Opens the Open File dialog box, where you can locate and open an existing process template.</td>
</tr>
<tr>
<td>Component Name</td>
<td>Icon / Shortcut</td>
<td>Menu</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>----------------</td>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Close</td>
<td>Ctrl + W</td>
<td>File &gt; Close</td>
<td>Closes the current process template.</td>
</tr>
<tr>
<td>Close All</td>
<td>Ctrl+Shift+W</td>
<td>File &gt; Close All</td>
<td>Closes all process templates.</td>
</tr>
<tr>
<td>Import Projects into Workspace</td>
<td></td>
<td>File &gt; Import Projects into Workspace</td>
<td>Imports projects from previous Business Process Modeler releases into the current release.</td>
</tr>
<tr>
<td>Save</td>
<td>Ctrl + S</td>
<td>File &gt; Save</td>
<td>Saves the process template currently in the Content pane, in the ProcessModeler\workspace&lt;Application&gt;\processtemplates folder.</td>
</tr>
<tr>
<td>Save As</td>
<td>Ctrl+Shift+S</td>
<td>File &gt; Save As</td>
<td>Saves process templates in a format you specify.</td>
</tr>
<tr>
<td>Save to Repository</td>
<td></td>
<td>File &gt; Save to Repository</td>
<td>Saves the current process to a repository.</td>
</tr>
<tr>
<td>Save Project to Repository</td>
<td></td>
<td>File &gt; Save Project to Repository</td>
<td>Saves the project to a repository.</td>
</tr>
<tr>
<td>Save All</td>
<td></td>
<td>File &gt; Save All</td>
<td>Saves all process templates in Business Process Modeler.</td>
</tr>
<tr>
<td>Print</td>
<td>Ctrl + P</td>
<td>File &gt; Print</td>
<td>Displays the Print Preview mode (see Printing the process template on page 70) and the Print dialog box, enabling you to print a process template, and define its printing properties. Printing multiple pages is supported only when using JDK 1.2 or later.</td>
</tr>
<tr>
<td>Properties</td>
<td>Alt + Enter</td>
<td>File &gt; Properties</td>
<td>Opens the Process Properties dialog box for the process template.</td>
</tr>
<tr>
<td>Undo</td>
<td>Ctrl + Z</td>
<td>Edit &gt; Undo</td>
<td>Undoes or recreates the last action performed in the Content pane, up to 100 operations. Supports lane, shape, connector, and cut/copy/paste operations, as well as Align, Space, and Collapse operations. Note: All other actions (for example, entering process property information in Process Properties dialog boxes) cannot be undone. If an operation that cannot be undone is performed, then the undo queue is cleared and the previous operations cannot be undone. You can perform Undo and Redo operation in Form Editor also.</td>
</tr>
<tr>
<td>Redo</td>
<td>Ctrl + Y</td>
<td>Edit &gt; Redo</td>
<td>Cuts a selected item from Business Process Modeler clipboard. Applies to diagram objects only.</td>
</tr>
<tr>
<td>Cut</td>
<td>Ctrl + X</td>
<td>Edit &gt; Cut</td>
<td>Copies a selected item to Business Process Modeler clipboard. Applies to diagram objects only.</td>
</tr>
<tr>
<td>Component Name</td>
<td>Icon / Shortcut</td>
<td>Menu</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------</td>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Paste</td>
<td>![Paste Icon]</td>
<td><strong>Edit &gt; Paste</strong></td>
<td>Pastes the clipboard items into Business Process Modeler Content pane. Applies to diagram objects only.</td>
</tr>
<tr>
<td>Ctrl + V</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Select All</td>
<td>Ctrl + A</td>
<td><strong>Edit &gt; Select All</strong></td>
<td>Selects all workstep items in Business Process Modeler Content pane.</td>
</tr>
<tr>
<td>Show views</td>
<td>Ctrl + T</td>
<td><strong>View &gt; Project Explorer</strong></td>
<td>Shows/hides the Project explorer view, the Properties view, the Problems view, the Repository Browser; and the Outline view; or restores the Default view.</td>
</tr>
<tr>
<td>Ctrl + I</td>
<td></td>
<td><strong>View &gt; Properties</strong></td>
<td></td>
</tr>
<tr>
<td>Ctrl + L</td>
<td></td>
<td><strong>View &gt; Problems</strong></td>
<td></td>
</tr>
<tr>
<td>Ctrl + R</td>
<td></td>
<td><strong>View &gt; Repository Browser</strong></td>
<td></td>
</tr>
<tr>
<td>Ctrl + O</td>
<td></td>
<td><strong>View &gt; Restore Default View</strong></td>
<td></td>
</tr>
<tr>
<td>Users</td>
<td></td>
<td><strong>Tools &gt; Users</strong></td>
<td>Enables you to add and manage users, groups, and queues.</td>
</tr>
<tr>
<td>Calendars</td>
<td></td>
<td><strong>Tools &gt; Calendars</strong></td>
<td>Enables you to define an organization-level calendar.</td>
</tr>
<tr>
<td>Business Objects</td>
<td></td>
<td><strong>Tools &gt; Business Objects</strong></td>
<td>Enables you to create a new business object, or import, export, modify, or remove an existing business object.</td>
</tr>
<tr>
<td>Help</td>
<td>![Help Icon]</td>
<td><strong>Help &gt; Help Contents</strong></td>
<td>Launches the online help system, which assists you with application development in Business Process Modeler.</td>
</tr>
</tbody>
</table>
Business Process Modeler Tasks pane

The Business Process Modeler Tasks pane includes primarily the Design Tasks panel, which lists the links used when designing your process template diagram. Clicking any of these links displays the associated panel; for instance, the Layout Tasks panel and the Description panel for the Select and Change Layout link.

**Figure 3: Process Modeler Tasks pane**

![Process Modeler Tasks pane](image)

**Note:** To hide the Tasks pane, click the Hide (▶) icon beside it. To show the Tasks pane again, click the Show (▼) icon. You can also expand (and then collapse) any of the panels. To expand a panel, click the Expand (►) icon. To collapse a panel, click the Collapse (▼) icon. Alternatively, you can expand (or collapse) a panel by clicking the panel header.

- **Select and Change Layout.** Enables you to align and space process template elements and to apply auto-layout. For more information, see Aligning and spacing worksteps on page 246.

- **Create Phases and Swim Lanes.** Enables you to add a phase or swim lane to the process template diagram. For more information, see Using swim lanes on page 128. This option is not available for Web applications.

- **Draw Shapes.** This link lists the worksteps and other process template elements (Start, Activity, Decision, Or-Join, End, etc.) that you can drag into the Content pane to create the workflow for your process. For more information, see Using shapes on page 26.

- **Connect Shapes.** Use this link to draw connections between process template elements. The Connect Tasks panel shows three types of connectors for Normal flow, Compensation flow, and Timeout flow, which you can use to connect worksteps. If the connection is not valid, then the connection cursor is displayed in red. Once you select a connector type, you can continue to make connections. To discontinue connecting, click any other Tasks pane link. For more information, see Connecting worksteps on page 53.

- **Assign Participants.** This link displays the performers grouped under four categories namely, Users, Adapters, Sub-Processes, and External. By default, Process Modeler displays organization-level users and a predefined user, “Creator” for Process Models, as well as a group of pre-defined managed adapters and a custom adapter. You can use this link to add,
delete, or modify the properties of a performer. For more information, see Defining performers on page 118.

- **Create Annotations.** This link enables you to add an Annotation, Sticky Note, or a Group notation to the process template diagram. For more information, see Adding notes on page 57.

**Note:** For a brief description on how to use each of these links, expand the Description panel.

### Using shapes

Table 8 on page 26 provides information about the Shapes listed when you click the Draw Shapes link in the Tasks pane. These shapes follow standard BPMN notation and include Activities (represented by a rounded rectangle); event objects such as Start or End (represented by a circle); and Gateways (represented by a diamond) that indicate a change in the workflow such as the joining, merging, splitting, or deciding of the flow's direction.

**Table 8: Shapes in Business Process Modeler Tasks Pane**

<table>
<thead>
<tr>
<th>Shape</th>
<th>Represents</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Start" /></td>
<td>Start</td>
<td>An event object that specifies the beginning of a business process. The default name is Start. Each Business Process Modeler process must have exactly one Start workstep.</td>
</tr>
<tr>
<td><img src="image" alt="Activity" /></td>
<td>Activity</td>
<td>Represents the basic unit of work that you can assign to one of the following performer types: User (a valid human performer or group of performers), Adapter, or Subprocess.</td>
</tr>
<tr>
<td><img src="image" alt="Decision" /></td>
<td>Decision</td>
<td>This gateway represents a decision point in the process flow. If the specified condition is not met, then the default branch is taken.</td>
</tr>
<tr>
<td><img src="image" alt="Exclusive Or-join" /></td>
<td>Exclusive Or-join</td>
<td>This gateway allows the process flow to proceed only once from multiple predecessor worksteps to a successor workstep, and terminates any other human-performed predecessor worksteps. This shape is not available for Web applications.</td>
</tr>
<tr>
<td><img src="image" alt="Or-join" /></td>
<td>Or-join</td>
<td>This gateway connects multiple predecessor worksteps and one successor workstep. The successor workstep is performed if any of the predecessor worksteps have been completed.</td>
</tr>
<tr>
<td><img src="image" alt="And Gateway" /></td>
<td>And Gateway</td>
<td>This gateway can act as an And Join or as a Split (or And-Fork). When it connects multiple predecessor worksteps to a single successor workstep, it acts as an And-Join. When it connects one predecessor workstep to multiple successor worksteps, it acts as a Split (or an And-Fork). The successor worksteps are performed in parallel and only if the predecessor workstep was completed. This enables you to synchronize the completion of multiple predecessor worksteps. This gateway is not available for Web applications.</td>
</tr>
</tbody>
</table>
### Business Process Modeler palette

The Business Process Modeler palette contains the following icons.

- **Select shapes (鼹)** (or **Select**) icon: Use this icon to select an Activity or other shape in the process template diagram to perform operations including copying, moving, and resizing shapes.

- **Pan the diagram (礻)** (or **Panning**) icon: Use this icon to move a portion of the diagram into the viewing area. Once you select the Panning icon, the cursor in the Content pane is displayed as a “hand”. The Panning mode is persistent until you select another mode.

- **Show Gestures (礻)** icon: Use this icon to display the list of mouse gestures, that you can use to quickly add shapes, layouts, and perform common process-related actions. For more information, see [Using mouse gestures](#) on page 60.

- **Color (礻)** icon: Use this icon to apply color to a selected shape, swim lane, or note. Click the icon to display a color palette, from where you can select one of the listed colors, or click **More** to display more color options.

- **Zoom In (礻)** / **Zoom Out (礻)** icons: Use these icons to zoom in or zoom out the view of the current process template in the Content pane. You can select a pre-defined magnification level (200%, 100%, 75%, 50%, or 25%) from the combo box.

- In the case of Process Model templates only, the drop-down list displays the current process. This is useful in the case of inline subprocesses (see [Defining an inline subprocess](#) on page 124). Click the **One Level Up (礻)** icon or select the parent process or higher subprocess from the adjacent drop-down list.

- **Check Diagram (礻)** icon allows you to verify if the process template diagram is correct.

- **Show 360° View (礻)** icon allows you to generate the 360° view of the current process template. For more information, see [Generating 360 degree view](#) on page 66.

- **Preview All (礻)** icon allows you to view all the presentation forms that have been defined for the process and developed in the Form Editor.

- **Print Preview (礻)** icon allows you to preview the current process template before printing it. For more information, see [Printing the process template](#) on page 70.

- **Export (礻)** icon allows you export the current process template as a Web page (HTML), an SVG, or a JPEG file. For more information, see [Exporting the process template](#) on page 71.

---

<table>
<thead>
<tr>
<th>Shape</th>
<th>Represents</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Message" /></td>
<td>Message</td>
<td>An intermediate workstep in a process, indicating that the workstep is waiting for a message or is used to send a message when it is activated. Message worksteps are used for notational purposes only in Business Process Modeler.</td>
</tr>
<tr>
<td><img src="image" alt="End" /></td>
<td>End</td>
<td>An event object that specifies that the application has come to a normal end. Business Process Modeler allows multiple End worksteps.</td>
</tr>
</tbody>
</table>
Using keyboard shortcuts

Table 9 on page 28 lists the keyboard shortcuts you can use during application development.

Table 9: Keyboard Shortcuts in Business Process Modeler

<table>
<thead>
<tr>
<th>Keys</th>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alt + Enter</td>
<td>File &gt; Properties</td>
<td>Enables you to define the current process properties.</td>
</tr>
<tr>
<td>Ctrl + A</td>
<td>Edit &gt; Select All</td>
<td>Selects all items in the Content pane.</td>
</tr>
<tr>
<td>Ctrl + C</td>
<td>Edit &gt; Copy</td>
<td>Copies a selected item onto the clipboard, which you can paste into the Content pane. Applies to diagram objects only.</td>
</tr>
<tr>
<td>Ctrl + F</td>
<td></td>
<td>Enables you to find a specified item within displayed text in any text frame.</td>
</tr>
<tr>
<td>Ctrl + I</td>
<td>View &gt; Properties</td>
<td>Enables you to show/hide the Properties view.</td>
</tr>
<tr>
<td>Ctrl + N</td>
<td></td>
<td>Enables you to create a new file, folder or project, by selecting one of the wizards.</td>
</tr>
<tr>
<td>Ctrl + O</td>
<td>View &gt; Outline</td>
<td>Enables you to show/hide the Outline view.</td>
</tr>
<tr>
<td>Ctrl + P</td>
<td>File &gt; Print</td>
<td>Enables you to print a selected item.</td>
</tr>
<tr>
<td>Ctrl + R</td>
<td>View &gt; Repository Browser</td>
<td>Enables you to show/hide the Repository Browser pane.</td>
</tr>
<tr>
<td>Ctrl + S</td>
<td>File &gt; Save</td>
<td>Enables you to save the changes you made to your active business process model or Web application.</td>
</tr>
<tr>
<td>Ctrl + T</td>
<td>View &gt; Project Explorer</td>
<td>Enables you to show/hide the Project Explorer view.</td>
</tr>
<tr>
<td>Ctrl + V</td>
<td>Edit &gt; Paste</td>
<td>Pastes the clipboard item (cut or copied) into the Business Process Modeler Content pane. Applies to diagram objects only.</td>
</tr>
<tr>
<td>Ctrl + W</td>
<td>File &gt; Close</td>
<td>Closes the current process template diagram.</td>
</tr>
<tr>
<td>Ctrl + X</td>
<td>Edit &gt; Cut</td>
<td>Cuts a selected item onto the Business Process Modeler clipboard, which you can paste into the Content pane. Applies to diagram objects only.</td>
</tr>
<tr>
<td>Ctrl + Y</td>
<td>Edit &gt; Redo</td>
<td>Redoes the last operation performed in the workspace.</td>
</tr>
<tr>
<td>Ctrl + Z</td>
<td>Edit &gt; Undo</td>
<td>Undoes the last operation performed in the workspace.</td>
</tr>
<tr>
<td>Alt + F7</td>
<td>Check Diagram</td>
<td>Checks the active diagram and verifies its validity.</td>
</tr>
</tbody>
</table>

Note: All other actions (for example, entering process property information in Properties dialog box) cannot be undone. If an operation that cannot be undone is performed, then the undo queue is cleared and the previous operations cannot be undone.
Localizing Business Process Modeler

You can adapt Business Process Modeler to a language of your choice. For instructions on how to localize Business Process Modeler menus, titles of windows, labels of controls, messages, and other interface elements to a specified language, see Chapter 11: “Localizing Business Process Server” in the *Customization Guide*. 
Developing Business Process Modeler projects

This chapter describes how to create projects using Business Process Modeler. For details, see the following topics:

- Project facets
- Creating a BPM project
- Managing multiple processes
- Importing a process file
- Using a process file as a template
- Using a project as a template
- Creating a simulation project

Project facets

Business Process Modeler implements the Faceted Project Framework, which enables you to create BPM projects comprising of units of functionality (or facets) that can be easily added or removed.
A facet represents a unit of functionality in a BPM project. By default, you can create a BPM project with any of the following facets:

- **Process**: to create a BPM project with a BPM process facet.
- **Web Application**: to create a BPM project with a Web application facet.
- **Common Resources**: to create a BPM project with a common resources facet.

Business Process Modeler allows you to view project facet properties for any BPM project, as well as apply a facet. For more information, see Project facet properties on page 32.

### Project facet properties

Business Process Modeler enables you to view the project facet properties for an added BPM project in the Properties page. Additionally, you can apply a facet to the project.

You can view the Properties page by selecting the created BPM project in the Project Explorer view and clicking File > Properties. You can view the list of the following project facets from the Project Facets page:

- **Java**: This facet is locked by default. You cannot remove this facet.
- **Process**: This facet is selected if the BPM project contains a BPM process.
- **Web Application**: This facet is selected if the BPM project contains a Web application.

You can apply a facet to the project by selecting the respective checkbox and clicking Apply. You can restore the default facet properties by clicking Revert.

### Creating a BPM project

To create a BPM project:

1. From the File menu, select New > BPM Project to launch the BPM Project wizard.
2. From the first page of the BPM Project wizard, enter the name of the new project in the Project name box. The name can contain only alphanumeric characters, and must not contain blank spaces or special characters, except underscores (“_”). Project name must start with an alphabetic character.
3. By default, the project is stored in the default workspace folder. To choose another location, clear the **Use default location** checkbox, and then click **Browse** to choose another location.

4. From the **Project type configuration** drop-down list, select the type of configuration (or facet) to be associated with this project. The available options are:
   - **Process**: to associate a BPM process facet with this project.
   - **Web Application**: to associate a web application facet with this project.

5. Click **Next** to continue. The remaining pages of the BPM Project wizard varies according to the project type configuration selected. To complete the BPM Project wizard, refer to the appropriate section provided in Table 10 on page 33.

### Table 10: Types of projects

<table>
<thead>
<tr>
<th>Project Type</th>
<th>See</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process</td>
<td>Creating a BPM process on page 33.</td>
</tr>
<tr>
<td>Web application</td>
<td>Creating a Web application on page 37</td>
</tr>
</tbody>
</table>

### Creating a BPM process

You can use the BPM Project wizard (Figure 4 on page 33) to create a BPM process.
To create a BPM process:

1. From the first page of the BPM Project wizard, select Process from the Process type configuration drop-down list.
2. Click Next to open the BPM Process page.

Figure 5: BPM Process page

If you want to use an existing process as a template to design a new process, from the Template section, select one of the following options:

- **Use an existing file from file system as template**, then click Browse to navigate to an existing file (of type SPT, XML, or XPDL) that you can use as a template for your new application. For more information on using a template to create an application, see Using a process file as a template on page 43.

  **Tip:** We recommend using an existing SPT file as a template for creating the new process. The SPT file contains the process template and all associated forms. The XML file is a basic version that does not contain associated forms.

- **Use an existing process from workspace as template** option, then click Browse to select an existing project (example, Approval) that is available in the Project Explorer that you can use as a template for your new application. For information on using an existing project as a template, see Using a project as a template on page 44.

- **Use a Microsoft Project file as template** option, then click Browse to navigate to a Microsoft Project file (*.mp*), which you can use as a process for your application.

  **Note:** You can also import existing projects or extract from an archive file using the Import functionality. For more information, see Importing a process file on page 39.
3. Click Next to open the BPM Process Properties page, in which you can modify the application name and enter the relevant process information in the respective boxes.

Table 11 on page 35 describes the application information fields.

**Table 11: Application properties**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>(Optional) The label is the title of the application that users see in Business Process Portal. For example, it is listed in the Administration module under Applications &gt; BP Server, and the Home module in BPM Portal displays this name together with its Description in the installed applications listed under the Applications tab. If no label is entered, the entry in the Name box is shown in Business Process Portal.</td>
</tr>
<tr>
<td>Version</td>
<td>(Optional) Use this box to add (or edit) the version number of the application.</td>
</tr>
<tr>
<td>Manager</td>
<td>(Required) Displays @CREATOR as the default manager. You can specify another manager, if required. The manager, either a single user or a user group, serves as a back-up performer if the originally assigned workstep performer is not available. If an individual is entered in the Manager box, then any tasks assigned to an unavailable performer are assigned to that user. If a group is entered in the Manager box, then any tasks assigned to an invalid performer are made available to the members.</td>
</tr>
<tr>
<td>Group</td>
<td>(Optional) Indicate which group of users is allowed to view and execute this project. You can only specify one group.</td>
</tr>
<tr>
<td>Duration</td>
<td>(Required) To specify the length of time you estimate is required to complete the process, use the appropriate Duration boxes to specify the duration in terms of days, hours, minutes, and seconds. The specified duration is used to determine the due date. After the due date you specified is passed, the process instance becomes overdue.</td>
</tr>
<tr>
<td>Category</td>
<td>(Optional) Use this box to differentiate between different types of projects.</td>
</tr>
<tr>
<td>Subcategory</td>
<td>(Optional) Use this box to further differentiate between projects in the same category.</td>
</tr>
<tr>
<td>Author</td>
<td>(Optional) Identifies the application developer who created the process template.</td>
</tr>
</tbody>
</table>
4. Click **Next** to open the **Add BPM Process module to Business Process Server(s)** page of the BPM Project wizard. You can use this page (optional) to add the Business Process Servers and configure the publishing settings of the BPM process module.

**Figure 6: Add BPM Process module to Available Business Process Server(s) page**

![Add BPM Process module to Business Process Server(s) page]

- **a)** To publish the BPM process as a module on any connected Business Process Server from the **Servers** view, select the server (if defined) from the **Available Business Process Server(s)** section.

- **b)** In the **Default Publishing Style** section, select the default option for publishing the process module to the selected server.

- **c)** To publish the BPM process as a Web service, select the **Publish as web service** checkbox.

- **d)** If you want to publish the BPM process with the user permissions you defined in the previous version of the same process, then select the **Inherit ACL** option.
5. Click **Next** to open the **BPM Process Description** page of the BPM Project wizard, in which you can enter the description of the BPM process. This is optional.

**Figure 7: BPM Process Description page**

6. Click **Finish** to complete the creation of the BPM process project.

### Creating a Web application

You can use a Web application to:

- Develop a process that can be integrated as an individual workstep into a complex BPM process. For instance, an approval process that requires several levels of management review and approval, can be defined as a Web application. It can be then used as a single workstep in a process template: thus you can save development time and computer overhead.

- Develop a stand-alone process that may include adapters to access BP Server server and other external sources.

- Replace the Start workstep in a process template.

You can use the **BPM Project** wizard (**Figure 4 on page 33**) to create a Web application.

**To create a Web application:**

1. From the first page of the **BPM Project** wizard, select **Web Application** from the **Process type configuration** drop-down list.
2. Click **Next** to open the **Web Application** page.

**Figure 8: Web Application page**
If you want to use an existing process as a template to design a new process, from the Template section, select one of the following options:

- **Use an existing file from file system as template**, then click Browse to navigate to an existing file (e.g., BS_SVBSurvey.swt) located in your workspace folder that you can use as a template for your new application. For more information on using a template to create an application, see see Using a process file as a template on page 43.

**Tip:** We recommend using an existing SWT file as a template for creating the new process. The SWT file contains the process template and all associated forms. The XML file is a basic version that does not contain associated forms.

- **Use an existing web application from workspace as template**, then click Browse to select an existing Web application project in the Project Explorer that you can use as a template for your new process. For information on using an existing project as a template, see Using a project as a template on page 44.

- **Use a Microsoft Project file as template**, then click Browse to navigate to a Microsoft Project file (*.mp), which you can as a process for your application.

3. Click Next to open the Web application properties page, in which you can modify the name of web application, and enter optional information like label, category, sub-category, and author in the respective boxes. For details regarding this information, refer to Creating a BPM process on page 33.

4. Click Next to open the Add Web Application module to Business Process Server(s) page of the BPM Project wizard.

If you want to publish the Web application as a module on any connected Business Process Server from the Servers view, select the server (if defined) from the Available Business Process Server(s) section. This step is optional.

5. Click Next to open the Web Application Description page of the BPM Project wizard, in which you can enter the description of the Web application. This is optional and is similar to Figure 7 on page 37 .

6. Click Finish to complete the creation of a Web application.

### Managing multiple processes

In Business Process Modeler, you can create multiple BPM processes or web application modules within an existing or (new) BPM project. In the Project Explorer view, all the resources or files pertaining to a process or a web application module are stored in a folder with the same name as the process within the project directory.
For instance, if you create a BPM project, "project1" (as discussed in Creating a BPM project on page 32), a default process, "project1" is created. You can add additional processes (for example, "process1" and "process2") to this project. After adding, you can view the files of the added processes in separate folders within the "project1" project folder in the Project Explorer view, as shown in Figure 9 on page 39.

**Figure 9: Project Explorer - added processes**

![Image of Project Explorer](image)

With multiple process support in a BPM project, each new process acts as a separate project module, and all the properties related to a particular process are stored within the module definition.

You can create any number of processes (within Eclipse capability) in a BPM project. You can also delete added processes from a project. Deleting a process deletes all the resources associated with that particular process from the project directory.

### Importing a process file

Business Process Modeler provides the **Import** functionality, which enables you to import external projects into your current workspace. You can also extract an archive file as a project into your workspace.

**To import a project:**

1. From **File** menu, click **Import Projects into Workspace** to launch the **Import** wizard.
2. To navigate to existing projects, click **Browse** beside the **Select root directory** option to open the **Browse for Folder** dialog box.

   Alternatively, select the **Select archive file** option and click **Browse** to navigate to any of the supported archive files to import. Click **OK** to return to the **Import** dialog box.
3. From the Projects section, select the project(s) to be imported into your workspace. You can select one, many, or all projects.

**Figure 10: Import Projects**

4. Select the Copy projects into workspace checkbox (available only for Select root directory option) to create a copy of the external projects in your workspace.

5. Click Finish to start the import.

The application appears in the Project Explorer View.

6. Expand the application folder displaying the SPT or SWT file in the process templates folder. Double-click the file to display it in the Content pane.
Adding a new BPM process

You can create a BPM process and add it to an existing BPM project.

   Figure 11: BPM Process Wizard, Page 1

2. In the Project name box, you must specify the BPM project to which you want to add this BPM process.
   a) Click Browse beside the Project name box to open the Select Project dialog box, which lists the existing BPM projects.
   b) Select the BPM project to which you want to add the process, then click OK. The selected BPM project is displayed in the Project name box.

3. Complete the rest of instructions in the BPM Process wizard, as described in Creating a BPM process on page 33.

On completion of this wizard, the new process is added and displayed in the specified BPM project folder structure in the Project Explorer view, as shown in Figure 9 on page 39.

Adding a new Web application

You can create a Web application and add it to an existing BPM project.

1. From the File menu, select New > Web Application. The Web Application wizard appears. 
   Figure 12: Web Application Wizard, Page 1

2. In the Project name box, you must specify the BPM project to which you want to add this Web application.
   a) Click Browse beside the Project name box to open the Select Project dialog box, which lists the existing BPM projects.
   b) Select the BPM project to which you want to add the process, then click OK. The selected BPM project is displayed in the Project name box.

3. Complete the rest of instructions in the Web Application wizard, as described in Figure 8 on page 37.
On completion of this wizard, the new Web application is added and displayed in the specified BPM project folder structure in the Project Explorer view, as shown in Figure 9 on page 39.

Importing a process file

Business Process Modeler provides the Import functionality, which enables you to import external projects into your current workspace. You can also extract an archive file as a project into your workspace.

To import a project:

1. From File menu, click Import Projects into Workspace to launch the Import wizard.
2. To navigate to existing projects, click Browse beside the Select root directory option to open the Browse for Folder dialog box.
   Alternatively, select the Select archive file option and click Browse to navigate to any of the supported archive files to import. Click OK to return to the Import dialog box.
3. From the Projects section, select the project(s) to be imported into your workspace. You can select one, many, or all projects.

   Figure 13: Import Projects

4. Select the Copy projects into workspace checkbox (available only for Select root directory option) to create a copy of the external projects in your workspace.
5. Click Finish to start the import.
   The application appears in the Project Explorer View.
6. Expand the application folder displaying the SPT or SWT file in the process templates folder. Double-click the file to display it in the Content pane.
Using a process file as a template

You can reduce process development time by using an existing process file as the template for your new process in Business Process Modeler. You can browse to a specific location (for example, your workspace folder), which contains Business Process Modeler projects. The following topics describe how to use an existing process file as a template to create a new process in Business Process Modeler:

• Using a process file from recent releases on page 43.
• Importing XPDL files on page 43.

You can also retrieve a process file from the Process Repository (for more information, see Using the Repository Browser tab).

Caution: We recommend that you do not use the File > Open command to create a new process template, because associated folders within the project are not created.

Using a process file from recent releases

To use a process file from a recent release (7.5 or later) as template to create a new process in Business Process Modeler:

1. In the first page of the BPM Project wizard (see Figure 4 on page 33), enter the name of the new project in the Project name box and click Next.
2. To use an existing process file as a template, select the Use an existing file from file system as template option, then click Browse to open the Select File dialog box, which displays the existing process files.
3. Select the SPT file in any of the process folders, which you want to use as a template for your new process.
4. Select a process file from your Workspace_Home folder and click Open.

The process template name (for instance, Assignment) and its path are shown in the File box; a number is automatically inserted in the Version box; the version number is one increment higher than the selected process file. The process template name and version (example, Assignment_V2) is entered in the remaining read-only fields.
5. Follow Step 4 to Step 7 in Creating a BPM project on page 32.
6. Select File > Save All to save changes to all open projects and files.

Importing XPDL files

Business Process Modeler supports the XML Process Definition Language (XPDL), which is a standardized format used for interchanging business process definitions across different workflow products.
XPDL enables an efficient exchange of BPMN diagrams by:

- Storing all aspects of a BPMN diagram, including graphical information such as the X and Y position of the nodes, as well as the executable aspects required to run a process.
- Including extensions to handle the new BPMN 1.1 constructs, as well as clarification of conformance criteria for implementation.

Business Process Modeler allows you to import process templates, which conform to XPDL version 2.1 specifications.

**Note:** Business Process Modeler does not support importing process templates containing vendor-specific information. For information regarding vendors and product versions supporting XPDL version 2.1, contact our Product Support team.

To use an exported XPDL file as a template to create a new process in Business Process Modeler:

1. In the second page of the BPM Project wizard (see Figure 5 on page 34), select the **Use an existing file from file system as template** option, then click **Browse** to open the **Select File** dialog box.
2. From the **Files of type** drop-down list, select the **XPDL** option, then select the XPDL file, which you want to use as a template for your new process.
3. Follow the instructions listed in **Using a process file as a template** on page 43, to use the XPDL file as a template for your new process template.

You can export process templates to XPDL version 2.1, as described in **Exporting the process template** on page 71.

**Using a project as a template**

You can significantly reduce process development time by using an existing Business Process Modeler project as the template for your new project. If you use this option, you can choose to import not only the process file, but also all associated interfaces, rules, and adapters. You can only use this option for existing project files that are displayed in the Project Explorer view.

**To use an existing project as template to create a new process:**

1. In the second page of the BPM Project wizard (Figure 5 on page 34 for BPM process and Figure 8 on page 37 for Web application), enter the name of the new project in the appropriate box.
2. To use an existing project as a template, select the **Use an existing process from workspace as template** option (for BPM processes) or the **Use an existing web application from workspace as template** option (for Web application), then click **Browse** to open the **Select project** dialog box, using which you can select the project to be used as a template for your new application.

The process template name (for instance, Assignment) is shown in the **Project** box, and a number in the **Version** box is automatically inserted that is one increment higher than the selected process file. The process template name and version (example, Assignment_V2) is entered in the remaining read-only fields. You can change the project name, or change or delete the version number.
3. Click Next to continue.
4. After entering the description (optional), click Finish in the BPM Project wizard.

A new project folder is listed in the Project Explorer view, along with the process template file with an SPT extension; example, NewApp_v2 and NewApp_v2.spt.

Creating a simulation project

Business Process Modeler provides the Simulation project, which supports all simulation functions, and allows for easier integration of additional functionalities in future releases.

To create a Simulation project in Business Process Modeler:

1. From the File menu, select New > Simulation Project to launch the New Simulation Project wizard.

2. In the first page of the New Simulation Project wizard, enter the name of the simulation project in the Project Name box. The name can contain only alphanumeric characters, and must not contain any blank space or special character except underscores (“_”). Ensure that the project name starts with an alphabetic character. Optionally, enter a label and description in the respective boxes.

Figure 14: New Simulation Project wizard: page 1
3. Click **Next** to open the second page of the New Simulation Project wizard, in which you can define the simulation properties for this project.

   a) In the **Start Simulation** section, select either of the following options to define when to start the simulation:

      • **At current system time** option to activate the simulation at the current time of the system.

      • **At time** option to activate the simulation at a specified time other than system time. After clicking this option, you need to specify a date and time. To do so, click the **ellipsis** button beside the **At time** option. For details, see **Specifying a date** on page 361.

   b) From the **Stop Simulation** section, select any of the following options to define when to stop the simulation:

      • **On completion of process instances** option to stop the simulation when all the instances of the processes in the simulation project are completed.

      • **At time** option to stop the simulation at a specified time. You can use this option only if you have selected the **At time** option in the **Start Simulation** section. You need to specify a stop time later than the simulation start time. To specify date and time, click the **ellipsis** button beside the **At time** option. For details, see **Specifying a date** on page 361.

      • **After duration** option to specify a relative end time for simulation, in terms of duration. You can specify the duration in terms of hours, minutes, and seconds by entering values in the respective boxes. In this case, the simulation continues even after all the instances are completed, for the specified duration.

4. Click **Next** to open the third page of the New Simulation Project wizard, in which you can define the project calendar settings for this simulation project.

   You can use the **Project Calendar** page to define the default calendar settings for your simulation project.

   a) From the **Work Days** section, select (or clear) the appropriate checkboxes for the work days.

   b) The **Sessions** section table displays the default time segments. The default workhours are 9:00 - 13:00 and 14:00 - 18:00.

   • To add a session, click **Add** to open the **Session** dialog box. Specify the start time (hours:minutes) and the end time (hours:minutes) of the session in the respective boxes. Click **OK** to add the session to the **Sessions** section table.

   • To edit any of the listed sessions, select the session, then click **Modify** to open the **Session** dialog box, in which you can modify the start and end time.

   • To remove any of the listed sessions, select the session, then click **Delete**. Business Process Modeler prompts you for confirmation. Click **OK** to remove the session.

   c) Click **Next** to continue.

   **Note:** The settings defined here are applied to the default scenario in your simulation project. You can modify these settings for the default scenario, as well as for any added scenarios in the Simulation perspective, after creating the simulation project. For details, see **Modifying simulation settings for a scenario** on page 344.
5. In the fourth page of the New Simulation Project wizard, you can add the processes, which you want to include in this simulation. You can also configure the simulation settings for each added process.

   a) To add a process to the simulation project, click the **Add** icon in the **Select** section to open the **Required Project Selection** dialog box. From the list of current processes, select the process to be added, then click **OK**.

   The added process is displayed in the **Select** section. You can add as many processes as required.

   **Note:** To remove a process from the simulation project, select the process, then click the **Remove** icon.

   For each added process, you can configure the simulation start settings (available only for multiple processes), the instance count, and the type of randomization.

   b) From the **Start** section, select the **At time** option to set an absolute time and date, as described in Step 3a. The start time for a process must be later than the absolute start time (if specified) for the simulation project. Alternatively, select the **After duration** option to specify a relative start time for the process simulation, as described in Step 3b.

   **Note:** You cannot set an absolute time and date for the process if you have specified the simulation start time as current system time.

   c) Use the **Count** section to specify the number of process instances. You can either keep the instance count unlimited or specify a maximum value in the **Limited To** box. You cannot specify an unlimited instance count if you have selected the **On completion of process instances** option in the previous wizard page.

   d) From the **Type** drop-down list in the **Distribution** section, choose a method of randomizing the duration of each interval between process instances in the simulation. For information regarding randomizing methods, see **Configuring and running simulation** on page 339.

6. Click **Finish** to complete the creation of the Simulation project.

7. If you are currently not in the Progress OpenEdge BPM Simulation perspective, then Process Modeler prompts you to switch to this perspective. Click **Yes** to open the simulation project in the Progress OpenEdge BPM Simulation perspective. Alternatively, click **No** to open the simulation project in your current perspective. If you select the **Remember my decision** checkbox, then Business Process Modeler does not display any prompt and executes your selected command the next time you create or open a simulation project file (**SST**). For further details regarding configuring and running a simulation, see **Configuring and running simulation** on page 339.

   **Note:** You can modify the above selected configuration for opening the simulation configuration perspective using the **Simulation > UI** page in the **Preferences** dialog box. For information, see **Using the Simulation page** on page 95.
Designing a process template diagram

This chapter describes how to design a process template diagram in Business Process Modeler.

**Note:** Alternatively, you can design a process template without the knowledge of BPMN shapes. For details, see *Working with the Overview page* on page 79.

For details, see the following topics:

- Creating a process template diagram
- Connecting worksteps
- Adding notes

Creating a process template diagram

*Developing Business Process Modeler projects* on page 31 describes how to use appropriate wizards to create the project files for process models and Web applications. This section describes how to design a process template for both types of projects and the differences between them.
After completing the New Process Model (Creating a process model) or the New Web application (Creating a Web application) wizards, the Business Process Modeler perspective is launched, with the new process template displayed in the Content pane.

Figure 15: Creating a Process Template

Figure 15 on page 50 displays the Business Process Modeler interface for a Process Model project.

If you did not use the Template section in the wizards, then the project opens a blank process template file (with an SPT or an SWT extension) in the Content pane, as shown in Figure 15 on page 50. If you used the Template section, then the process you selected as a template appears in the Content pane.

The Tasks pane provides Shapes, Performers, and other elements that are available for the selected process. The name of the current process template is displayed above the Content pane, and the Content pane tabs at the bottom enable you to view the template diagram, dataslots, performers, or simulation data for the process.

A process template typically consists of a single Start workstep, one or more Activity worksteps, and at least one End workstep. The following procedure describes how to create a process template by adding shapes, connectors, and other component templates to the Content pane.

1. From the Tasks pane, click the Draw Shapes link and then drag any of the listed shapes into the blank Content pane to add them to the process template diagram, beginning with the Start shape (◇).
   a) Select an Activity shape (◇) to represent worksteps performed by a human performer (only for Process Models), an adapter, or a subprocess and drag them into the Content pane. Add more Activity Shapes as required.
   b) Similarly, add an End workstep (◇). The process template must include at least one End workstep.
Similarly, you can add the remaining Shapes depending on your process template requirements.

2. From the Tasks pane, use the **Connect Shapes** link to connect the worksteps. The worksteps and other shapes contain up to 12 points of connection, enabling you to make multiple connections to worksteps and shapes. For more information on workstep connectors, see **Connecting worksteps** on page 53.

Figure 16 on page 51 displays a sample process template for a Process Model.

**Figure 16: Viewing the New Process Template**

Table 12 on page 51 lists the operation you can perform to complete the process flow diagram.

**Table 12: Completing a process flow diagram**

<table>
<thead>
<tr>
<th>Operation</th>
<th>See</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define dataslots</td>
<td>Using dataslots on page 97.</td>
</tr>
<tr>
<td>Add workstep performers</td>
<td>Defining and assigning performers on page 115.</td>
</tr>
<tr>
<td>Define an alert</td>
<td>Using alerts on page 269.</td>
</tr>
<tr>
<td>Add process simulation parameters</td>
<td>Configuring and running simulation on page 339.</td>
</tr>
<tr>
<td>Define workstep properties</td>
<td>Setting workstep properties on page 217.</td>
</tr>
<tr>
<td>Define workstep presentation</td>
<td>Defining workstep presentation format on page 257.</td>
</tr>
</tbody>
</table>

3. From the Modeler palette, click the **Check Diagram** (✓) icon to determine the validity of the process.

If the new process template is not valid, then invalid items are listed in the Problems view; and a message is displayed listing the invalid items. If you have correctly defined the process template, then “The diagram is complete” message appears. Click **OK**.

4. To save changes to all open projects and files, from the **File** menu, click **Save All**.
The major differences between editing functions for a Process Model and a Web application are described below:

- There are fewer options in the Tasks pane. Unlike a Process Model, creating phases and Swim Lanes are not supported in Web applications.
- There are fewer Shapes available for Web applications. The AND Gateway and Exclusive Or-Join shapes are not supported.
- The Alerts tab is not present as this feature is not supported in Web applications.

### Changing shapes

Business Process Modeler allows you to change the shapes after adding them to your Content pane. This eliminates the need to delete an inappropriate shape and drag the correct shape to the Content pane. Table 13 on page 52 lists the valid shapes that you can change to for each process template element.

#### Table 13: Changing shapes

<table>
<thead>
<tr>
<th>Shape</th>
<th>Change to (in BPM projects)</th>
<th>Change to (in Web applications)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>All other shapes, except End.</td>
<td>All other shapes, except End.</td>
</tr>
<tr>
<td>Activity</td>
<td>All other shapes, except Start and End.</td>
<td>All other shapes, except Start and End.</td>
</tr>
<tr>
<td>Decision</td>
<td>Activity, AND Gateway, Message.</td>
<td>Only Activity.</td>
</tr>
<tr>
<td>XOR Join</td>
<td>All other shapes, except Start and End.</td>
<td>Not Applicable.</td>
</tr>
<tr>
<td>OR Join</td>
<td>All other shapes, except Start and End.</td>
<td>All other shapes, except Start and End.</td>
</tr>
<tr>
<td>AND Gateway</td>
<td>All other shapes, except Start and End.</td>
<td>Not Applicable.</td>
</tr>
<tr>
<td>Message</td>
<td>All other shapes, except Start and End.</td>
<td>Not Applicable.</td>
</tr>
<tr>
<td>End</td>
<td>All other shapes, except Start.</td>
<td>All other shapes, except Start.</td>
</tr>
</tbody>
</table>

To change a shape:

1. Right-click the shape that you want to change, and then point to the Change To option.

   Business Process Modeler displays the list of valid shapes that you can change to.

2. Click the shape that you want to change to from the options provided. The shape is changed and the altered shape element is displayed in the Content pane.
Connecting worksteps

You can use the Connect Shapes link in the Tasks pane to connect the several shapes or process template elements in your process template. The connections (or links) between the process elements determine the workflow of the process. Each shape contains up to 12 points of connection, enabling you to make multiple links to worksteps and shapes. Note that when you click the Connect Shapes link, the Tasks pane displays three types of connectors for—Normal flow, Compensation flow, Timeout flow—, which you can use.

Figure 17: Tasks Pane with the connector mode selected

Table 14 on page 53 describes these link types and their appearance when used on the template workstep:

Table 14: Link types

<table>
<thead>
<tr>
<th>Link</th>
<th>Appearance</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal (Default)</td>
<td><img src="image" alt="Normal Connector" /></td>
<td>The Normal connector type provides standard connectors to worksteps. In the case of a workstep with multiple outgoing links, the default link (indicated by the slanting line in adjoining figure) indicates a link with no condition.</td>
</tr>
<tr>
<td>Normal (Conditional)</td>
<td><img src="image" alt="Conditional Connector" /></td>
<td>This is another Normal connector type. In the case of a workstep with multiple outgoing links, this conditional link indicates a link with a condition. The workflow selects this link path if the condition is satisfied.</td>
</tr>
</tbody>
</table>
Compensation

Indicates the compensatory steps to be taken in the event of an Error occurring during the execution of the process or a Recovery. The link from the source workstep of a Compensation Flow is marked with a \( \rightarrow \). When you add the Compensation Flow to a diagram, the target workstep of the Compensation Flow automatically becomes a Rollback Point (see Defining a Rollback on page 244) and is marked with a Compensation symbol (\( \rightarrow \)).

Timeout

Indicates the direction of a workflow after Overdue Actions have been executed and the Last Overdue action is completed. Add the Timeout Flow link to the diagram and the source workstep of a Timeout Flow is marked with a \( \rightarrow \). For more information, see Defining a Timeout in worksteps on page 245.

Note: Keep the Optimize connectors when moving shapes checkbox clear to maintain the point of connection to the shape when you move the connected shape. Alternatively, if you select this checkbox, the shortest path between shapes is used for the connection. For example, if the position of "Activity2" in the process diagram is closer to "Activity1" than the position of "Activity3," then any connector from Activity1 is joined to Activity2.

After choosing one of the above flow types, the representation of the cursor in the Content pane changes to reflect the type of flow you selected. Table 15 on page 54 indicates the type of links that can be used with typical process template components and the number of links (single, multiple, or none), which are allowed to originate from each element.

Table 15: Shape and Output Link Matrix

<table>
<thead>
<tr>
<th>Shape/Workstep</th>
<th>Default</th>
<th>Conditional</th>
<th>Compensation</th>
<th>Timeout</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>Multiple allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
</tr>
<tr>
<td>Decision</td>
<td>One allowed</td>
<td>Multiple allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
</tr>
<tr>
<td>Split</td>
<td>Multiple allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
</tr>
<tr>
<td>And-Join</td>
<td>One allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
</tr>
<tr>
<td>Or-join</td>
<td>One allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
</tr>
<tr>
<td>Xor-Join</td>
<td>One allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
</tr>
<tr>
<td>Activity workstep</td>
<td>Multiple allowed</td>
<td>Multiple allowed</td>
<td>One allowed</td>
<td>One allowed</td>
</tr>
<tr>
<td>Subprocess worksteps</td>
<td>Multiple allowed</td>
<td>Multiple allowed</td>
<td>One allowed</td>
<td>One allowed</td>
</tr>
<tr>
<td>Message</td>
<td>Multiple allowed</td>
<td>Multiple allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
</tr>
<tr>
<td>End</td>
<td>Not allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
<td>Not allowed</td>
</tr>
</tbody>
</table>
Using multiple links

You can use multiple incoming and outgoing links for a workstep without using an intermediary Or-Join, AND or Decision. In addition, Business Process Modeler supports multiple outgoing links for Start worksteps; and multiple incoming links for Decisions and End worksteps.

The following restrictions apply to workstep connections:

- You cannot have more than one link of the same type with the same source and target worksteps.
- A link cannot originate from and terminate at the same workstep.
- The Or-Join and XOR-Join gateways can have only one outgoing link. When an AND Gateway has multiple incoming links, it acts as an And-Join and can have only one outgoing link.

**Note:** Even though you can now design a business process without using the Or-Join or Decision gateways, these worksteps are still available for users who want to add them to processes to clarify or highlight areas of the process’s workflow.

Using multiple incoming links to worksteps

You can add multiple incoming links to worksteps, as shown in Activity 3 in Figure 18 on page 55. You can also add multiple incoming links to Decision, Or-Join and End worksteps. These multiple incoming links are the equivalent of an Or-Join shape and, in this case, indicates that Activity 3 will be performed when either Activity 1 or Activity 2 is completed.

**Figure 18: A process diagram with multiple connections**

![Process diagram with multiple connections](image)

Using multiple outgoing links from worksteps

You can add multiple outgoing links to worksteps, as shown by the links out of Activity 3 in Figure 18 on page 55. You can also add multiple outgoing links from Decision, Or-Join, and Start worksteps. With this type of link, you can add conditions that direct the process flow.

**Note:** All links are default links unless you clear the Default checkbox in the Link Properties dialog box (for more information, see Defining link properties on page 240).
If all of the outgoing links from a workstep have no specified condition, then these links are the equivalent of an And shape, indicating that once the predecessor workstep is completed, the successor worksteps are activated simultaneously. For example, the links to Activity 1 and 2 have no condition and these two activities are both started after the Start workstep is completed.

If one or more of the outgoing links from a predecessor workstep has (or have) a specified condition, then these links are the equivalent of a Decision shape. Once the predecessor workstep is completed, one or more successor worksteps can be activated if the links to them fulfill specified conditions. But if these conditions are not met in any of the links, then all links with no condition are executed. For example, the links from Activity 3 to Activities 4, 5, and 7 each have conditions and one or more of these activities are executed when the specified conditions are fulfilled; the links from Activity 3 to Activity 6 has no specified condition and this activity will be executed when the specified conditions for Activities 4, 5, and 7 are not fulfilled.

Note: If there are multiple outgoing links in a Web application, then you must assign a unique name to each outgoing link.

Using multiple incoming/outgoing links with gateways

Gateways are diamond-shaped objects in a BPMN process diagram that represent a change in the workflow—it may indicate the joining, merging, splitting, or deciding of the flow’s direction. In Business Process Modeler, gateways include Decision (including Exclusive Decisions), XOR Join, OR Join, and AND Gateway. Decisions (◇) have a single incoming link and multiple outgoing links. Or Join (⊙) and XOR Join (⊗) gateways have multiple incoming links and a single outgoing link. When an AND Gateway (◇) has multiple incoming links, it must have a single outgoing link—in this case, the AND Gateway acts as an And Join. When an AND Gateway has a single incoming link, it can have multiple outgoing links—in this case, the AND Gateway acts as an And Fork (or Split). These variations in incoming/outgoing links for gateways are illustrated in the following figure.

Figure 19: Examples of Gateway Links

If you want to start multiple, parallel worksteps after an Or Join or XOR Join gateway, then you must insert an AND Gateway immediately after the Or Join or XOR Join gateway.

Defining Loops in the workflow

You can now define loops in the workflow without using multiple links to and out of worksteps, and without using Decision shapes. For more information, see Defining a Loop condition on page 226.
Adding notes

You can add a note (a Sticky Note, an Annotation, or a Group notation) to any element in a process template diagram. Sticky Notes and Annotations generally have the same functions, and they only differ in visual representation. You can use a Sticky Note or Annotation to indicate specific features or conditions, or as prompts or reminders for other users (for example, *Must be reviewed by all managers*). Use the Group notation to select and insert a caption that:

- Describe a group of worksteps that share a similar task (for example, all activities performed by the HR Department in Hiring), or
- Require the same performer (for example, highlighting all worksteps *Performed by Managers Group in HR Department*).

To insert a note into a process:

1. Click the **Create Annotations** link in the Tasks Pane and drag the Sticky Note (**sticky note**), Annotation (**annotation**), or Group notation (**group notation**) shape into the Content pane. You can also drag and drop any of these notes onto a workstep or other shape, which creates a connection between it and the note.
2. Click the note to view its properties in the **Properties** view.

**Figure 20: Properties of Notes – General tab**

- In the **General** tab, you can change the appearance of the note (for example, Sticky Note to Group). Use the **Type** drop-down list to specify the new type for the existing note.
- Click **View/Edit** link to open the **View/Edit** dialog box, in which you can enter the text of the note, or the **grouping's** caption in the text area. For a Group notation, enter a caption for the group association (for example, *Review all grouped worksteps*). Use the toolbar icons to format the text as needed.
- You can also check the spelling and insert a link, by right-clicking the text area and selecting the appropriate options. For instance, to add a link to a note, select **Insert > Link** to open the **Link Properties** dialog box. Specify the text to be used as the hyperlink and the URL.
- Click **OK** to apply the changes to the selected note and return to the **Properties** view.
- From the **Background** section, you can select a background color only for Sticky Notes. You cannot change colors for Annotation and Group notations. You can also click **More** to display more color options.
**Note:** You can also change the background color of Sticky Notes by selecting the sticky notes in your Content pane and then clicking the **Color** icon from the Modeler palette. From the color palette, you can select one of the listed colors, or click **More** to open the **Color Chooser** dialog box (Creating swim lanes on page 129) to define your custom color.

f) From the **Associate With:** drop-down list, you can associate the Sticky Note or Annotation notation with a specific workstep by selecting from the worksteps listed. To disassociate a note from a workstep, select **(none)** from the drop-down list.

After defining the property of the note, the notes now appear in the process template diagram.

**Figure 21: Notes Associated with Worksteps**

3. Use your cursor to move or resize the Sticky Note, Annotation, or Group notation as required.
Managing gestures

Business Process Modeler provides mouse gestures to quickly add shapes, connectors, and layouts to your Content pane, as well as perform commonly-used actions in your process template.

This chapter describes how to use mouse gestures in your process template diagram in Process Modeler, and to create custom gestures using the Gesture Editor.
For details, see the following topics:

- Using mouse gestures
- Creating custom gestures
- Saving custom gestures
- Using custom gestures

Using mouse gestures

You can view the set of predefined mouse gestures in the Default Gestures dialog box.

1. Create or open the process template diagram (SPT or SWT) in which you want to use mouse gestures.

2. Click the Show Gestures (?) icon from the Modeler palette, to open the Default Gestures dialog box. Figure 22 on page 60 displays the mouse gestures predefined for a Process Model.

Figure 22: Default Gestures in Process Model project

Note: Web applications support fewer shapes and layout gestures.

3. Point to each of the shapes, layouts, and actions to view the gesture details in the Gesture Details section. The red dot indicates the starting point for the gesture.

4. Click OK to close the Default Gestures dialog box.

To use any gesture in your Content pane, drag your right mouse button in the required direction. To connect shapes using gestures, press CTRL key and then drag your right mouse button between shapes to connect them.
Creating custom gestures

Business Process Modeler provides the Gesture Editor, which allows you to create custom gestures and replace the default gestures, illustrated in Figure 23 on page 61.

Using the Gesture Editor

To start the Gesture Editor:

1. From the Tools menu, click Gestures to open the Gesture Editor.
2. To populate the Gesture Editor with predefined actions and parameters, click File > New.

Figure 23: Gesture Editor

The Gesture Editor consists of the following three panes:

- Actions pane, containing the list of predefined gesture actions, as listed in Table 16 on page 61. For information regarding the shapes used, see Using shapes on page 26.
- The middle pane, in which you can view (and edit) the action selected in the Actions pane, as well as draw one or many gestures for each action.
- Gestures pane, which displays the gestures associated with each action.

Table 16: List of predefined actions

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>shape_start</td>
<td>Creates a Start workstep.</td>
</tr>
<tr>
<td>shape_activity</td>
<td>Creates an Activity workstep.</td>
</tr>
</tbody>
</table>
### Designing gestures

**To design a gesture in the Gesture Editor:**

1. From the **Actions** pane (Figure 22 on page 60), select the action for which you want to design the gesture.
2. Modify, if required, the default text in the **Label** and **Description** boxes to provide additional information for the gesture. You can enter a static label or description; for instance, "End" for label and “Adds an End workstep” for description.

Business Process Modeler converts any value starting with the "$" sign with the corresponding string in the localization file (example, designer.properties).
3. In the drawing area provided, drag your mouse to draw the gesture shape to be used for the corresponding action. To accept the drawn shape and clear the drawing area, click Accept. To discard the drawn shape, right-click in the drawing area.

4. After you click Accept, the drawn shape is displayed as a gesture in the Gestures pane. To remove the added gesture from the Gestures pane, select the gesture and click Discard.

Depending on the complexity of the action, we recommend that you create multiple gesture samples. For simpler actions, 2 or 3 gesture samples are adequate. For complex actions, you must create more samples.

Repeat Step 3 to Step 4 to add multiple gesture samples. Step Figure 24 on page 63 displays gesture samples created for the Start workstep.

Figure 24: Sample Gestures for Start workstep

You can use the checkbox for each gesture in the Gestures pane to select the icon to be displayed for this action in the help for the customised gestures. If you select multiple checkboxes, then the gesture that you select first is displayed as the gesture icon (Step Figure 24 on page 63) and the remaining gestures are displayed in the Gesture Details section.

Guidelines for gestures

The guidelines to be followed when designing a gesture are listed below:

- The starting point of a gesture is important. You can use the starting point to distinguish several similar shaped gestures.
- You can use the direction to create two different gestures for the same shape. For instance, you can use a clockwise-drawn circle for a Start workstep, and anti-clockwise circle for the End workstep.
- Always create multiple gesture samples for each action. This ensures that the gesture recognition algorithm compensates most of the variations, thus resulting in reliable recognition.
- Do not try to be too precise when drawing the gesture sample. Do not make the sample size too big or too small. Draw them as you would when using them in practice.
- Draw each gesture with a single move of the mouse. Releasing your mouse indicates the end of the gesture.
Saving custom gestures

Business Process Modeler stores all gestures (default and custom) in the `Workspace_Home\.com.savvion.studio\gestures` folder. The default gesture set is stored as `default.ptp`. All custom gestures are stored as GLF files (Gesture Library File). For helping you get started, Business Process Modeler provides a sample gesture set (`sample.glf`) file.

**To save your custom gesture:**

1. Use the **Gesture Editor** to create your custom gestures for each required action.
2. Click **File > Save** to open the **Save** dialog box.
3. Type the gesture file name, then click **Save**.

You can open your custom gesture file (using **File > Open** option in **Gesture Editor**) and modify the same.

Using custom gestures

You can now use your saved custom gesture in your process template diagram by replacing the default gesture set (`default.ptp`) with your custom gesture file.

1. Click **Tools > Preferences** to open the **Preferences** dialog box (described in Setting Business Process Modeler preferences on page 91).
2. Expand **Business Process Modeler > Diagram**.
3. From the **Gestures** section, change the default gestures file (`default.ptp`) by clicking **Browse** beside the **Gestures file** box, then select your custom gestures file in GLF format.
4. To close the **Preferences** dialog box, click **OK**.
5. Open the process template diagram (`SPT` or `SWT`) in which you want to use your custom gesture.
6. Click the **Show Gestures** (❓) icon from the Modeler palette, which now displays your custom gesture set.

You can now use your custom gesture set to add shapes to your process template.
Performing additional Business Process Modeler operations

This chapter describes how to use Business Process Modeler for performing additional operations. For details, see the following topics:

- Publishing the process
- Managing pages
- Generating 360 degree view
- Printing the process template
- Exporting the process template
- Managing phases

Publishing the process

After you have defined and saved the process template in Business Process Modeler, it is placed in the `ProcessModeler\workspace\models` folder. You can open and edit the process in Progress Developer Studio for OpenEdge. After saving the changes, you can publish the process from Progress Developer Studio for OpenEdge to Business Process Portal. For information regarding publishing applications, see the `Progress Developer Studio for OpenEdge User's Guide`. 
After the process is published, it is displayed as an installed application in the BP Server Applications page or the BPM Workflow Applications page in the Administration module in Business Process Portal. After a user with administration privileges installs the application in Business Process Portal, a valid application user with the correct permissions can then open it as a Business Process Server application.

Managing pages

Business Process Modeler provides a default number of nine pages (in a 3x3 grid) in the Diagram tab to design your process template diagram. Business Process Modeler automatically adds pages if your process template diagram does not fit within the 3x3 grid.

You can now add and manage pages in the Diagram tab, thus enabling you to control the page size more efficiently.

- From the Diagram tab, right-click a blank portion of your process template diagram, point to Pages option, then select any of the following options:

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add to right</td>
<td>Adds a page to the right. For the default 3x3 grid, this operation extends the grid size to 3x4.</td>
</tr>
<tr>
<td>Add to bottom</td>
<td>Adds a page to the bottom. For the default 3x3 grid, this operation extends the grid size to 4x3.</td>
</tr>
<tr>
<td>Fit to process</td>
<td>Use this option to save paper when you want to print a process template. Select this option to reduce the page count to fit the process template diagram.</td>
</tr>
</tbody>
</table>

Generating 360 degree view

Business Process Modeler provides a 360° view, which is a multi-layered Chart view of the current process template diagram. You can generate the 360° view especially for complex process templates, which consist of multiple activity (and adapter) worksteps and complex workstep traversal paths.
To generate the 360° view:

- From the Modeler palette (see Business Process Modeler palette on page 27), click the **Show 360° View** icon to launch the 360° View of the current project.

**Figure 25: Sample 360° View**

![Sample 360° View](image)

The 360° view diagram consists of color-coded sections, each representing a workstep. Each section displays the label of the workstep along with the estimated time (in days, hours, and minutes) when the workstep is activated. The estimated time is calculated from the beginning of the process. For processes containing multiple paths, each path is depicted as a sub-section in the chart.

**Note:** The estimated start time of each workstep can be calculated only when the duration of the preceding worksteps are set in absolute values (in days, hours, minutes, or seconds). The estimated start time cannot be calculated when workstep duration is bound to a INTEGER dataslot. For setting workstep duration, you can use the **Overdue in** box in the **General** tab of the Activity workstep's **Properties** view (as discussed in Specifying general properties of Activity workstep on page 224).

The 360° view displays each workstep type with the following color code.

**Table 17: 360° view color code**

<table>
<thead>
<tr>
<th>Workstep Type</th>
<th>Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity</td>
<td>Blue</td>
</tr>
<tr>
<td>Adapter</td>
<td>Violet</td>
</tr>
<tr>
<td>Start / End / Gateways</td>
<td>Brown</td>
</tr>
<tr>
<td>Subprocess</td>
<td>Orange</td>
</tr>
</tbody>
</table>

You can perform the following operations in the 360° view.
Table 18: 360° view operations

<table>
<thead>
<tr>
<th>To...</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pan the diagram</td>
<td>Drag your mouse.</td>
</tr>
<tr>
<td>Zoom in (or out) on the diagram OR view all workstep labels</td>
<td>Use your mouse scroll wheel.</td>
</tr>
<tr>
<td>Rotate the diagram around its center point</td>
<td>Drag your mouse with the right button.</td>
</tr>
<tr>
<td>Open the subprocess in the 360° view</td>
<td>Click the subprocess workstep section. For additional details, see Opening a subprocess on page 70.</td>
</tr>
<tr>
<td>Calculate a traversal path duration</td>
<td>Move your mouse over the diagram to highlight the selected path. The estimated duration of the selected path is displayed in the Path Length box.</td>
</tr>
</tbody>
</table>

The 360° view also contains the following icons:

- **Show/Hide Loops ( )** icon: For any process containing loops, the default 360° view does not include the worksteps that follow when a loop is executed. You can click this icon to display these worksteps in semi-transparent state. In this case, the worksteps gradually progress with...
fading color. Figure 26 on page 69 illustrates an example of using this icon with Assign_A_Task_V1 sample process.

**Figure 26: Assign_A_Task_V1 process with loops**

![Timeline View](image)

- **Timeline View ( )** icon: Click this icon to switch to the Timeline view, which displays the thickness of each workstep corresponding to the time it is likely to take to complete.

**Figure 27: Timeline View**

![Timeline View](image)

The worksteps in the Timeline view (Figure 27 on page 69) with greater thickness require more time to execute. As Activity worksteps are performed by humans and are likely to take the maximum time to complete as compared to the other workstep types, the Timeline view shows Activity worksteps with greater thickness. You can zoom in on the diagram to view the other workstep types in detail.
Opening a subprocess

You can also view an inline subprocess in the 360° view, by clicking the subprocess workstep section of the parent process diagram. For information regarding an inline subprocess, see Defining an inline subprocess on page 124. The 360° view of the parent process is replaced with that of subprocess.

Note: This feature is not available for Web applications because Web applications do not support an inline subprocess.

Business Process Modeler supports an inline subprocess within another inline subprocess. You can use the breadcrumb trail in the 360° view to track the process hierarchy.

Figure 28: Breadcrumb trail

You can click the process name button in the breadcrumb trail to reload the 360° view of the parent process in the hierarchy.

Printing the process template

You can generate a print preview of a process template diagram before printing it.

To generate the print preview:

1. Open the process template to be printed.
2. From the Modeler palette (see Business Process Modeler palette on page 27), click the Print Preview icon to view the process diagram in the Print Preview mode.

   In the Print Preview mode, the Tasks pane displays the print-related functions and other options.

   Figure 29: Print Preview mode

3. To reduce or enlarge the image scale for printing, increase (or decrease) the scaling percentage in the Scaling factor box in the Options panel.

   Alternatively, select the Fit to width/height option in the Scale Mode drop-down list, then enter the width and height in pages in the respective boxes to set the image scale to fit to the specified pages.
4. Define the page setup for printing by clicking Page setup link from the Print Preview Tasks panel to open the Page Setup dialog box.

The dialog box that appears depends on the type of printer you are using. Select the size of the printed document—Letter (8.5 x 11 inches), Legal (8.5 x 14 inches), 11 x 17 inches,—its source, and specify Portrait or Landscape orientation. Use the fields in the Margins panel to adjust the page’s margins.

5. Click the Print this diagram link to print the displayed process template.

The Print dialog box opens, in which you can select a number of printing options, including number of copies and page range. The dialog box, which appears depends on the type of printer you are using. If you are using Windows, then click Properties to define additional print properties.

6. To exit the Print Preview mode, click the Exit print preview link.

Exporting the process template

You can export the process template in the following formats:

- Web Page (HTML)
- SVG image
- JPEG image
- XPDL file
- Microsoft Project file
- Process Summary in PDF or MS Excel format

You can even export large process template diagrams as a single file.

To save the process template’s file and export it in a supported format:

1. Open the process template, which you want to export.
2. From the Modeler palette (see Business Process Modeler palette on page 27), click the Export icon.

The Tasks pane displays the Export Tasks panel, which includes the following links.

Table 19: Links

<table>
<thead>
<tr>
<th>Links</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save as a single JPEG image</td>
<td>To export the process as a JPEG image.</td>
</tr>
<tr>
<td>Generate a Web page</td>
<td>To export the process as an HTML page.</td>
</tr>
<tr>
<td>Save as an SVG image</td>
<td>To export the process as a Scalable Vector Graphics image.</td>
</tr>
<tr>
<td>Generate XPDL</td>
<td>To export the process as an XPDL file. You can use the exported XPDL file as template for creating a new process. For more information, see Importing XPDL files on page 43.</td>
</tr>
</tbody>
</table>
### Links and Description

<table>
<thead>
<tr>
<th>Links</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generate Microsoft Project file</td>
<td>To export the process as a MS Project file (*.mpx). You can use the exported MPX file as template for creating a new BPM process project or Web application. For more information, see Creating a BPM process on page 33 and Creating a Web application on page 37. <strong>Note:</strong> The process, which uses an exported MPX file, does not support these process template elements: And, Or-Join, Exclusive Or-Join, Timeout and Compensation flow connectors, Swim lanes, phases, managed adapters, and External Activity (and External Adapter).</td>
</tr>
<tr>
<td>Save as Process Summary</td>
<td>To generate a process summary of the current process in PDF and MS Excel formats. For more information, see Generating a process summary on page 72.</td>
</tr>
</tbody>
</table>
3. From the Export Tasks panel, click Save as Process Summary to open the Document Generation dialog box.

Figure 30: Document Generation dialog box

4. Select the Optional Sections, which you want to retain in your process summary.

5. From the Output drop-down list, select the format in which you want to generate the process summary output. The available formats are PDF and Excel.

6. Click Generate to generate the Process Summary.

Note: Process Summary files are saved directly under the application folder; for instance, if you generate a PDF summary for the Assignment application, the PDF file and the process diagram file are stored under Workspace_Home\Assignment\summary.pdf and Workspace_Home\Assignment\report.jpg respectively.
Viewing process summary

Figure 31 on page 74 displays the Process Summary in PDF format. The generated process summary in Excel format contains the same information as the PDF summary.

Figure 31: Process Summary Document in PDF Format

The first page displays a Process Properties section, including data on General properties, Attributes, and a History of the process (that is, when it was created, saved, checked in, etc.). From the Process Diagram section, you can click the Click here to view process diagram in a different window link to open the process diagram file.

Note: Process Summary reports for Web applications do not contain all the sections that are displayed for process models. This is because features that are unsupported in Web applications, such as the Simulation section, are not shown in Process Summary reports for Web applications.

To explore the process summary:

1. Use the bookmarks in the left pane to open a section for a specific workstep, or to view performers, dataslots, alerts, or notes (if notes are present in the process).
2. To see the process summary for a specific workstep, expand the Worksteps bookmark and then click the workstep name to display the following information:
   - Workstep details (Step Figure 32 on page 75) including its general properties (including Loop and skippable conditions if present) in the General section.
• Information about dataslots in the **Fields** section. Click the **View All User Dataslot Properties** link to open the **Dataslots** section.

• Information regarding workstep collaboration (if any) in the **Collaboration** section. This information is displayed only if you have selected the **Collaboration** check box in the **Document Generation** dialog box (Figure 30 on page 73).

For Subprocess worksteps, click the **Click here to view <AppName> subprocess document** link under the Subprocess Document heading to open the Process Summary document for the nested process.

**Figure 32: Process Summary in PDF Format: Workstep Page**

3. Click **Links** bookmark to view all the links in the process, along with the source and target worksteps for each link, and associated conditions (if any).
4. Click **Performers** bookmark to open the Performers section, which lists information about all the performers available to the process.

**Figure 33: Process Summary in PDF Format, Performers Page**

5. You can also click **Dataslots, Alerts** (visible when the process contains alerts), or **Notes** (visible when the process contains notes).

6. Click **File > Exit** or the **Close** button (miş) to exit the Process Summary PDF.

---

**Managing phases**

A phase is a set of tasks, which signifies the completion of a project segment. You can use the phase function to divide all the tasks in a process into sets of tasks (or subtasks). For example, in a typical Sales process, all subtasks which help achieve Customer Prospecting can be considered a phase.

Phases are represented by vertical lines and are similar to implementing Swim lanes (as discussed in **Using swim lanes** on page 128). You can define a phase and its worksteps in the **Diagram** tab. In the **Overview** tab, you can view the phase to which a particular task belongs.

**Note:** Phases are not supported in Web applications.

---

**Creating a phase**

You can create a new phase for a Process Model project in the **Diagram** tab.
To create a phase:

1. Open the Process Model project (SPT) in the Diagram tab.
2. From the Tasks pane (Business Process Modeler Tasks pane on page 25), click the Create Phases and Swim Lanes link in the Design Tasks panel, and then click the Phase icon ( ) to open the Phase Properties dialog box.

![Phase Properties dialog box](image)

**Figure 34: Phase Properties dialog box**

**Note:** You can also use the ‘P’ gesture to add a phase. For details, see Using mouse gestures on page 60.

3. Enter the ID and Name of the new phase in the respective boxes.
4. Click OK to add a new phase to the process template diagram. Move all the worksteps that belong to this phase into the added phase, as seen in the Figure 35 on page 77.

**Note:** You cannot move a phase.

![Business Process Modeler interface with added phases](image)

**Figure 35: Business Process Modeler interface with added phases**
Resizing a phase

You may need to resize a phase either to add a shape to the phase or to quickly move a shape into (or out of) this phase.

- To increase the width of a phase, select the Move shapes when resizing phases check box, before resizing the phase. Shapes adjacent to the phase move as you resize the phase.
- To make the adjacent shapes a part of this phase or to remove a shape from this phase, clear the Move shapes when resizing phases checkbox, before resizing the phase. Shapes adjacent to the phase do not move as you resize the phase.

To resize a phase, position your cursor over the space between two phase headers, then use the slider to adjust the width to the left or to the right.

Performing additional phase operations

You can perform additional operations on a phase by right-clicking the phase header (see the popup menu in Figure 35 on page 77). Additional operations include changing the phase header display, modifying existing phase properties, and removing a phase.

- To change the phase display header, from the View menu (Figure 34 on page 77), select either Name or ID to display your selection in the header of each phase.
- To change the properties of the selected phase, click Properties to open the Phase Properties dialog box (Figure 34 on page 77), in which you can modify the current phase properties.
- To remove the selected phase, click Remove Phase. All the worksteps in this phase are removed along with the phase.
Working with the Overview page

Business Process Modeler allows process modeling and analyzing through the Diagram view. In certain situations, this procedure can be cumbersome, as there is a learning curve involved for the business user, who needs to understand the usage and semantics of BPMN shapes before defining a meaningful process.

Consider an instance where a business user wants to quickly outline a large process with a large number of tasks. Instead of adding shapes and defining their properties in the Diagram view, a business user would probably rather work with a project-oriented view of the process by defining a set of tasks, the task/subtask relationships, and additional details without ever needing to switch to the Diagram view. The Diagram view is still a useful view especially for complex flows. However, it is not the only view, nor is it the only editor for defining new worksteps. Business Process Modeler provides the Overview page, which is an effective tool to support project-oriented processes.

For details, see the following topics:

- Introducing the Overview page
- Managing tasks

Introducing the Overview page

The Overview page (or tab) provides a project-oriented approach - including activity timelines - to designing a process, where you can add a set of tasks (or activities), define their relationship, and configure the properties, without switching to the Diagram tab.
The Overview page provides the following benefits:

- You can reduce process design time by over 30% by using the Overview page.
- You can filter process elements in the Overview page for analysis. For example, you can choose to view only Tasks or Adapters.

**Figure 36: Diagram and Overview pages**

Figure 36 on page 80 displays the diagram and tabular view of the sample process, “Assign_A_Task_V2.” As illustrated, both the views provide different views of the same process. The following are the important common features:

- Any changes made in either of the views are reflected in the other view, when activated.
- All the process template elements in the Diagram view (except for Notes, Swim lanes, and Phases) are also available in the Overview page.
- The same Properties view for the process and for each workstep is available in both views.
- As in Diagram view, you can group tasks in the Overview page and then convert it to an inline subprocess.

**Reviewing the Overview page**

You can design a process template from scratch in the **Overview page** of the Business Process Modeler interface.

**Figure 37: Overview page**

For a blank process template, the **Overview page** does not contain any elements or tasks. For a process template that contains elements, the **Overview page** contains the corresponding elements or tasks (see right image, Figure 37 on page 80).
The Overview page displays the following default information about each task in a tabular format:

### Table 20: Task information

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the task name, with the adjoining icon indicating the shape used. To change the task name, double-click the cell and enter the new task name. An inline subprocess workstep displays an expandable icon, which you can expand to view its worksteps.</td>
</tr>
<tr>
<td>Type</td>
<td>Indicates the type of task or workstep. To change the type, double-click the cell and then select one of the options available. The available options depend on the workstep and process type (as listed in Changing shapes on page 52).</td>
</tr>
<tr>
<td>Successors</td>
<td>Displays the successor tasks (in case of a single outgoing connector) or multiple tasks (in case of multiple outgoing connectors). To define the successor tasks, click in the cell and then click the ellipsis button. From the left pane of the Add/Modify Successors dialog box that appears, select the tasks to be added as successors and click ➔ to move them to the right pane. To remove tasks added as successors, select the tasks in the right pane and click ⬅️. You can define successor for any task type, except End.</td>
</tr>
</tbody>
</table>
| Duration     | Displays the default duration (2 hours) of Activity, Adapter, and Embedded Sub-Process task types. To modify the duration, click in the cell and then click the ellipsis button. Select either of the following options:  
  - Enter a duration, to open the Duration dialog box, in which you can enter the number of days, hours, or minutes for the workstep duration.  
  - Use a dataslot, to open the Select Dataslots dialog box, which lists the INTEGER dataslots available in the process. Select a INTEGER dataslot to define a dynamic duration for this workstep. |
| Priority     | To change the workstep priority, click in the cell, then select any of the predefined options: Low, Medium, High, or Critical. Click the Use a dataslot option to choose a dynamic dataslot value at runtime. |
| Performer    | Displays the default performer for Activity, Adapter, and Subprocess worksteps. Double-click the cell to modify the performer. For more information regarding performers, see Defining and assigning performers on page 115. |
| Milestone Alerts | Applicable only to Start, Activity, Adapter, and Subprocess type tasks. Displays the alerts (if any) associated with the completion of a task (or a milestone). To associate an alert, click in the cell and then click the ellipsis button. From the left pane of the Milestone Alerts dialog box that appears, select the alerts to be added and click ➔ to move them to the right pane. To remove any added alert, select the alert in the right pane and click ⬅️. You can associate multiple milestone alerts with the same task. To associate all of the defined alerts with the task, click ➔. To remove all of the associated alerts with the task, click ⬅️. For information regarding alerts, see Using alerts on page 269. |
| Phase        | Indicates the phase (if any) to which the task belongs. For information regarding phases, see Managing phases on page 76. |

You can sort the Overview page content by clicking on the column header. You can also search the task list based on a keyword, by entering the keyword in the Search box ( 🔍️ ).
You can also filter the listed tasks by selecting the appropriate filtering criteria. You can apply filtering on basis of the type of tasks (for example, Adapter) and on basis of advanced properties (for example, “Worksteps with skip condition” and “Worksteps that have overdue actions”).

To filter the tasks:

1. Click the Click to filter ( الفلسطينية) icon.

   The Filter dialog box appears.

   **Figure 38: Filter dialog box**

   ![Filter dialog box](image)

2. To enable filtering, select the Enable filtering check box.

   To filter the list of tasks on basis of the workstep type, select one (or more) check boxes in the Type list (left image, Figure 38 on page 82). To filter the list of tasks on basis of advanced workstep properties, select one (or more) check boxes in the Advanced list (right image, Figure 38 on page 82).

3. Click OK to apply the selected filtering to the list of tasks in the Overview page.

### Managing tasks

You can create and edit an entire process template in the Overview tab. You can also set the properties of individual tasks (or worksteps) in the Properties view, as described in Setting workstep properties on page 217. You can use the toolbar in the Overview tab (Figure 37 on page 80) to add and manage the tasks of the process template.

### Adding a task

To add a task (or a workstep):

1. Click the Click to add icon ( الفلسطيني)، inserting a default Activity task. To add a task after an existing task, select the existing task and then click the Click to add icon.

2. Type the task name in the Name column. Task names can only include alphanumeric characters and underscores (_).

3. In the Type column, click in the cell and then select from the available workstep options, as required.
Step Table 21 on page 83 lists the available options.

**Table 21: Workstep Type**

<table>
<thead>
<tr>
<th>Option</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>AND Gateway</td>
<td>To add an AND gateway. This gateway is not available for Web applications.</td>
</tr>
<tr>
<td>Activity</td>
<td>(default) To add an Activity workstep.</td>
</tr>
<tr>
<td>Adapter</td>
<td>To add an adapter workstep. Displays the Adapter dialog box, in which you can define (or select) any adapter.</td>
</tr>
<tr>
<td>Decision</td>
<td>To add a decision gateway.</td>
</tr>
<tr>
<td>Embedded Sub-Process</td>
<td>To add an inline subprocess. This option is not available for Web applications.</td>
</tr>
<tr>
<td>End</td>
<td>To add an End workstep. This type is only available when there is no successor task specified under the Successors column.</td>
</tr>
<tr>
<td>External Activity</td>
<td>To add a monitoring Activity workstep.</td>
</tr>
<tr>
<td>External Sub-Process</td>
<td>To add an external subprocess.</td>
</tr>
<tr>
<td>External Adapter</td>
<td>To add a monitoring Adapter workstep.</td>
</tr>
<tr>
<td>Managed</td>
<td>To add a managed adapter workstep.</td>
</tr>
<tr>
<td>Message</td>
<td>To add a Message workstep. This workstep type is not available for Web applications.</td>
</tr>
<tr>
<td>OR Join</td>
<td>To add an Or-Join gateway.</td>
</tr>
<tr>
<td>Start</td>
<td>To add a Start workstep. This option is available only for the first added task, and not for subsequent added tasks.</td>
</tr>
<tr>
<td>XOR Join</td>
<td>To add an Exclusive Or-Join gateway. This gateway is not available for Web applications.</td>
</tr>
</tbody>
</table>

4. To add more tasks, click the Click to add icon.

The Successor column for the previous task now displays the default task name (“Activity 2”) of the added task.

**Performing other operations**

After adding tasks, you can perform any of the following operations using the toolbar in the Overview tab.

- You can expand (and collapse) the entire task list in order to view (or hide) all elements. To expand the list, click the Click to expand all nodes icon. To collapse the list, click the Click to collapse all nodes icon.

- To remove a task, select the task and click the Click to remove icon.

- To group tasks into an inline subprocess, select the tasks and click the Group tasks icon. This inserts a subprocess task of type, “Embedded subprocess” and contains the grouped tasks.
between a Start and End workstep. To ungroup the tasks, select the subprocess task and click the **Ungroup Tasks** icon.

**Note:** You cannot add nor delete group tasks for sorted and filtered (by workstep type and by advanced properties) data.

- To change the position of a selected task, click the **Move up** icon or the **Move down** icon. Changing the position of a task does not alter the task’s predecessor and successor.

- To view any task in the Diagram view, select the task and then switch to the Diagram view.

- You may need to restore the added tasks to its original order. To do so, click the **Restore Default Order of Tasks** icon.

**Note:** You can design an entire process template in the **Overview** tab, without switching to the **Diagram** tab. To ensure that all changes in the Overview page are integrated seamlessly in the Diagram view of the process, see that the Auto Layout functionality is enabled in the **Preferences** dialog box (see Setting Business Process Modeler preferences on page 91).
Introducing Path Analysis

This chapter describes how you can perform process path analysis using the Path Analysis view. For details, see the following topics:

- Using the Path Analysis view
- Exploring the Path Analysis view

Using the Path Analysis view

Business Process Modeler provides the Path Analysis functionality in the Path Analysis view, which enables you to analyze the timing of process execution.

Path Analysis provides the following functionalities:

- Ability to drill down through a specific path and estimate the time required for the process execution to reach a particular point in the process.
- Displays workstep attributes including duration, priority, and performer.
- Provides process timeline information with:
  - Graphical display of workstep duration.
  - Grid and markers for duration units including days, weeks, and months.
  - Alternate coloring between weeks, to identify week boundaries.
• Displays all process paths from Start to End workstep using nested paths.
• Displays the estimated start time for each workstep, which is calculated by adding the sum of duration of all previous worksteps in that path.

Figure 39 on page 86 shows a sample approval process that requires the reviewer to decide if the request meets specific critical criteria. If the performer of the ReviewRequest workstep decides it is non-critical, then it is sent by e-mail to a group at a remote location for submission.

Figure 39: A Sample Approval Process Template

The Review Request workstep assigned to any member from "MgrGroup" has an assigned work time of two hours. The Submit Request workstep assigned to any member from the "ebmsgroup" has a work time of 50 minutes.

If any member from the "MgrGroup" decides the request is critical, then it is sent to the Exec Approve workstep (with assigned work time of 50 minutes): a senior executive (performer, “exec”) then approves or rejects it. The approved requests are sent to the Submit Request2 workstep (work time of 50 minutes). Any member of the “MgrGroup” can view the request status in the Request Status workstep (with default duration of 2 hours). Each of the Adapter worksteps has an estimated duration of 10 minutes. The entire process has an estimated duration of 6 hours.

Exploring the Path Analysis view

To open the Path Analysis view for any process template (for instance, the process shown in Figure 39 on page 86), open the process template and then click the Path Analysis content pane tab.
The Path Analysis view displays the name of each task and the following default columns.

### Table 22: Column Information In Path Browser

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the task name, with the adjoining icon indicating the shape used.</td>
</tr>
<tr>
<td>Est. Start</td>
<td>Indicates the time interval in which the task is estimated to start. If there is no time interval, this column displays “(none)” (for instance, in the case of the Start workstep). This value is derived by adding the duration for previous tasks. For instance, the estimated start of 4 hours for the Notify Reject task is a sum of the duration for Review Request and Exec Approve tasks. An asterisk beside an estimate (in this case, for “Request Status” and “End1” tasks), indicates that the estimate may not be accurate because there are parallel paths to reach that point.</td>
</tr>
<tr>
<td>Duration</td>
<td>Displays the time required to complete the task. The colors representing the duration are indicated as follows and are relative to the duration of the other tasks.</td>
</tr>
<tr>
<td></td>
<td>• Red for the tasks with the maximum duration (in this case, Review Request and Exec Approve).</td>
</tr>
<tr>
<td></td>
<td>• Green for the tasks with the minimum duration (in this case, Notify Reject and Notify Submit).</td>
</tr>
<tr>
<td></td>
<td>• Yellow for the tasks with the average duration (that is, an average of maximum and minimum duration values). For some processes, if all the tasks have the same duration, their duration is also represented in yellow.</td>
</tr>
<tr>
<td></td>
<td>• All durations that measure between the average and maximum values are represented in gradient colors from yellow to red.</td>
</tr>
<tr>
<td></td>
<td>• All durations that measure between the minimum and average values are represented in gradient colors from green to yellow.</td>
</tr>
<tr>
<td>Priority</td>
<td>Displays the workstep priority. Supported values are: Low, Medium, High, Critical, or dataslot name.</td>
</tr>
<tr>
<td>Performer</td>
<td>Displays the performer for Activity, Adapter, and Subprocess worksteps.</td>
</tr>
<tr>
<td>Timeline</td>
<td>Provides a visual representation of the Estimated Start and Duration data for each task, without displaying the actual numbers. Each timeline starts from the Estimated Start value and continues for the specified Duration. The timeline is tabulated based on the total process time (in this case, 6 hours) with each task timeline representing the work breakdown structure. Timelines also depict the dependencies between tasks (that is, a particular task cannot start before another task is completed). If the timeline for a particular task is displayed in orange (in this case, the Request Status task), it indicates that the estimated start time may not be accurate because there are parallel paths to reach that point.</td>
</tr>
</tbody>
</table>

**Note:** You can expand (and collapse) the process path in order to view (or hide) all elements. To expand the list, click the Click to expand all nodes icon. To collapse the list, click the Click to collapse all nodes icon.

The Path Analysis view displays all the possible paths that a process execution can take. When you search a particular workstep type (or name) by entering the keyword in the Search box, the Path Analysis view displays only those paths that contain the keyword you have searched.
You can also filter the paths by selecting the appropriate filtering criteria. You can apply filtering on basis of the type of tasks (for example, Adapter) and on basis of advanced properties (for example, "Worksteps that have loop" and "Worksteps that can be skipped").

**To apply filtering:**

1. Click the **Click to filter** icon.
   
   The **Filter** dialog box appears, which is similar to Figure 38 on page 82.

2. To enable filtering, select the **Enable filtering** check box.
   
   To filter the paths on basis of the workstep type, select one (or more) check boxes in the **Type** list. To filter the list of tasks on basis of advanced workstep properties, select one (or more) check boxes in the **Advanced** list.

3. Click **OK** to apply the selected filtering to the paths in the **Path Analysis** view.
Defining process properties

This chapter describes how you can define process properties of process models or Web applications. You can define process properties for the first time or modify some of the properties that you have defined when creating the application (refer Creating a process model and Creating a Web application in Developing Business Process Modeler projects on page 31). For details, see the following topics:

- Setting process properties
- Setting Business Process Modeler preferences

Setting process properties

Business Process Modeler provides the Properties view where you can set (and modify) properties of the process and process template elements (as described in Setting workstep properties on page 217).

Note: To display the Properties view in Business Process Modeler perspective, click View > Properties.
The procedures for defining the properties of a process are similar for Process Models and Web applications.

1. Open the Process Model (SPT) or Web application (SWT) file, whose properties you want to define.
2. Click a blank area of the Diagram tab to view the process properties in the Properties view. By default, the General tab appears.

**Figure 41: Properties View – General tab**

![Properties View – General tab](image)

**Note:** You cannot modify the process name in the Name box. You can define (or modify) the rest of the process properties in the General tab. For information on each of these properties, refer to Table 23 on page 93.

3. Click the Description tab and enter a description of the application (optional) in the text area.
4. Click the History tab to review past actions in the process. This read-only tab provides information on when the process was created or checked into the process repository.
5. Click the Attributes tab to review existing attributes and modify their values. You can create attributes only in the Administration module of Business Process Portal.

**Figure 42: Properties View – Attributes Tab and Attributes dialog box**

![Properties View – Attributes Tab and Attributes dialog box](image)
To review and modify attributes and their values:

a) From the Attributes tab, you can view process attributes and values. If many attributes listed are listed, then you can enter a string of letters in the Search box and an attribute name containing that string is displayed in the table provided.

b) By default, the Attributes tab lists the ProcessType attribute with value, BusinessProcess or SystemProcess. To add attributes, click More to open the Attributes dialog box, which lists the default attributes—Executable, Goal, Influencers, Objective, Organizational Unit, Process Type, Risks, Rewards—as well as any attributes that have already been defined in BPM Portal.

c) If you can access Business Process Portal, then click the Add ( ) icon to go to Business Process Portal, in which you can add more attributes, if required.

Click the Refresh ( ) icon to refresh the view of the listed attributes.

d) Select one or more attributes (except Process Type) listed in the Attributes dialog box and click OK to add it to the Attributes tab.

e) To modify the value of an attribute, select the attribute and click Modify. You can only modify an attribute if it has been defined as editable in Business Process Portal.

f) Click Clear to clear the value of a selected attribute. To remove an attribute, click More to open the Attributes dialog box, and clear the checkbox of the attribute that you want to remove from the process.

6. Use the Monitoring tab (available only for process models with a monitoring workstep) to define the configurations for a monitoring process. For more information, see Introducing Monitoring process on page 249.

Setting Business Process Modeler preferences

Business Process Modeler provides the Preferences dialog box, a Configuration interface that enables you to easily modify the more commonly used configuration parameters. Once you have defined these configuration parameters in the Preferences dialog box, they are automatically set for each new project.

To set your preferences in Business Process Modeler:

1. From Tools menu, click Preferences, to open the Preferences dialog box.

2. Expand OpenEdge > Business Process Modeler in the left pane: the following options are displayed:

   • Activity: to set the default properties of an Activity workstep. For more information, see Using the Activity page on page 92.

   • Application: to set the default properties and attributes for a process. For more information, see Using the Application page on page 92.

   • Diagram: to set the default size of the Diagram view, appearance, and other settings. For more information, see Using the Diagram page on page 93.

   • Export: to set the width and height (in pixels) of the process template diagram, when exported in JPG format. For more information, see Using the Export page on page 93.

   • Form: to configure data source and label settings in Form Editor. For more information, see Using the Form page on page 94.

   • Simulation: to set the default simulation-related properties in Simulation perspective. For more information, see Using the Simulation page on page 95.

3. After defining the above preferences, click OK to close the Preferences dialog box.
Note: To apply the changes made in each page, click Apply. Click Restore Defaults if you want to restore the values of all options in the page to the default values.

Using the Activity page

You can use the Activity page (Figure 43 on page 92) to configure the default properties of an Activity workstep.

**Figure 43: Activity Page in Preferences dialog box**

1. From the Presentation drop-down list, select the default presentation format (Form or Auto-generated) for Activity workstep.
2. Select the default priority (Low, Medium, High, or Critical) for the Activity workstep in the Priority drop-down list.
3. Use the Duration boxes to change the default duration (2 hours) of the Activity workstep.

Using the Application page

You can use the Application page to configure the default properties of a process, the default attributes, and database settings.

1. From the Default Properties section, set the default properties for a process including the default manager, group, duration, category, subcategory, and author.
2. From the Default Attributes section, view (or modify) the default server location from where you can import attributes.
3. From the Database section, view (or modify) the database system to be used for dataslots in the process.
Using the Diagram page

You can use the Diagram page to configure the following default properties of the Diagram tab in the Business Process Modeler interface.

Table 23: Diagram Page Settings

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Size</td>
<td>To set the configuration parameters for width and height of the diagram page. Values range from 1 to 10.</td>
</tr>
<tr>
<td>Show pages</td>
<td>Select to display the page boundary, as it appears on your default printer. Use this feature to create appropriate page breaks in a lengthy process template diagram.</td>
</tr>
<tr>
<td>Show grid</td>
<td>Select this to view the process template diagram with a grid. The Business Process Modeler grid is always active, whether visible or not. In the Grid size box, set the space between grid points. Although grid size ranges from 1 - 50, use values that are multiples of 4. By default, grid size is set to 12. Note: Due to Java limitations, a process template diagram may not display correctly if you select a grid size of 10 or less.</td>
</tr>
<tr>
<td>Fill shapes</td>
<td>Select this to view a solid fill for diagram components. Clear this option to view transparent components. You can also view a gradient fill for diagram components by selecting the Gradient fill option.</td>
</tr>
<tr>
<td>Snap to Grid</td>
<td>Select this to align a new component to the grid.</td>
</tr>
<tr>
<td>Snap to shapes</td>
<td>Select this to align each new workstep with the center of neighboring components. If you want to move a shape when the Snap to shapes feature is activated, then select the workstep and use the arrow keys to move the workstep.</td>
</tr>
<tr>
<td>Auto Layout</td>
<td>Select the quality of the auto layout functionality in the Overview page (as described in Working with the Overview page on page 79). Options are low (fast), normal, and high (slow). Clear the Run auto layout after using Tabular View checkbox to disable the auto layout functionality.</td>
</tr>
<tr>
<td>Gestures</td>
<td>Allows you to enable or disable the Gestures functionality (see Using mouse gestures on page 60). You can view and modify the default location of the Gestures file.</td>
</tr>
</tbody>
</table>

Using the Export page

You can use the Export page to enter the width and height (in pixels) of the process template diagram when exported in JPG format. For information on exporting, see Exporting the process template on page 71.
Using the Form page

You can use the Form page to configure the default properties of Form Editor. For operations in Form Editor, see Using Form Editor on page 275.

Figure 44: Form Page

- Select the Create advanced control on Data Source drag and drop checkbox. After enabling this functionality, any dataslot that you drag to your form creates an advanced control. For more information, see Using data binding on page 293.

- Use the Show asterisk for “Required” data fields checkbox to display (or hide) an asterisk beside the label of each required dataslot, when added as a control to the form. You can view the result of this setting in Business Process Portal or using form preview (see Figure 185 on page 277) in Form Editor.

- Use the Show workstep header checkbox to display (or hide) the default header for a form in a Web application workstep. After enabling this functionality, you can view the default header (with the Instruction field) on opening the form from a Form-enabled workstep in a Web application.

- Use the Preview section to configure settings for form preview. The Port Number box displays the default port number used for form preview. To modify the port number, click the ellipsis button beside the Port Number box. For information regarding form preview, see Previewing forms on page 335.
Using the Simulation page

You can use the Simulation page to configure the simulation settings for the simulation project. For details regarding simulation, see Configuring and running simulation on page 339.

Figure 45: Simulation page in Preferences dialog box

1. Use the Start Simulation and Stop Simulation sections to set the default configurations (for the default scenario) for the Simulation Start and Stop settings. For details regarding each of these parameters, see Modifying simulation settings for a scenario on page 344. Use the Currency section to set your default currency symbol.

2. From the Performer page, set the default settings for the performer types—Group, Individual, and System. For details regarding each of these settings, see Managing performers on page 355.

3. From the Process page, you can set the default simulation settings for added processes in the default scenario. For detailed information regarding each of these parameters, see Configuring simulation parameters for a process on page 349.

4. From the Workstep page, you can set the default worktime distribution method and other settings for supported workstep types in the default scenario. For details regarding each of these parameters, see For worksteps on page 351.

5. From the Resource page, you can set the default cost and count for consumable and non-consumable resources. For more information, see Managing resources on page 357.

6. From the UI page, you can set (or modify) your preferred settings for switching between simulation-related perspectives in Business Process Modeler. The perspective switches are listed as follows:
   - Open Progress OpenEdge BPM Simulation perspective on launching editor, when creating or opening a simulation project.
   - Open OpenEdge BPM Run Simulation perspective on simulation run, when running a process simulation.
   - Open Progress OpenEdge BPM Simulation perspective on simulation run exit, when exiting the OpenEdge BPM Run Simulation perspective after running a process simulation.
The available options for each of the above switches are **Always** (to launch the respective perspective without any user prompt), **Never** (to always perform the requested operation in the current perspective without switching the perspective), or **Prompt** (the default option, which prompts you to switch to the respective perspective).

You can also choose the color settings for status bars and progress bars used in the Simulation perspective. To choose another color, click the bar and then the adjoining color button to open the **Color** dialog box, where you can select another color. The available bars are listed in **Table 24 on page 96**

**Table 24: Appearance Settings**

<table>
<thead>
<tr>
<th>Bar</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>For Status Bar</strong></td>
<td></td>
</tr>
<tr>
<td>Inactive</td>
<td>Indicates the Simulation and Process Status bars before the simulation run.</td>
</tr>
<tr>
<td>Required</td>
<td>Indicates the total count of required process instances in the Processes Runtime View during the simulation run. It also indicates the time limit for the simulation project in the Simulation Status section of the Simulation Control Editor, during the simulation run.</td>
</tr>
<tr>
<td>Created</td>
<td>Indicates the progress status of created process instances in the Processes Runtime View during the simulation run.</td>
</tr>
<tr>
<td>Completed</td>
<td>Indicates the progress status of completed process instances in the Processes Runtime View during the simulation run. Indicates the progress status of simulation project in the Simulation Status section of the Simulation Control Editor, during the simulation run.</td>
</tr>
<tr>
<td><strong>For Probability Bar (For details of the Probability bar used in the Probability tab of the Properties view in Simulation Perspective; for details, see Setting the probabilities on page 354)</strong></td>
<td></td>
</tr>
<tr>
<td>Bar Fill Color FG/BG</td>
<td>Indicates the graded colors (smooth shading of one color to another color) used for the filled portion of the probability bar.</td>
</tr>
<tr>
<td>Bar Empty Color FG/BG</td>
<td>Indicates the graded colors (smooth shading of one color to another color) used for the empty portion of the probability bar.</td>
</tr>
<tr>
<td>Text</td>
<td>Indicates the color of the text displayed on the probability bar.</td>
</tr>
<tr>
<td>Shadow</td>
<td>Indicates the color of shadow bar accompanying the probability bar.</td>
</tr>
<tr>
<td>Color when selected</td>
<td>Indicates the color of the probability bar, when selected.</td>
</tr>
<tr>
<td>Border</td>
<td>Indicates the border color of the probability bar.</td>
</tr>
</tbody>
</table>
Using dataslots

You can use dataslots to manage information flow in a business process. Dataslots are global variables in business processes that enable the process to support exchange of data across worksteps. Typically, a dataslot is defined as the output of one workstep and the input of the successor workstep. Information—in this case, the value of the dataslot—is therefore passed from one workstep to the next workstep.

This chapter describes how to use dataslots in Business Process Modeler, and covers the following features:

- Business Process Modeler provides a set of predefined system dataslots for Process Model projects.
- Business Process Modeler supports a wide range of dataslot types.
- Business Process Modeler enables you to define customized dataslots that are specific to your requirements and can be used in process templates.
- Business Process Modeler supports importing and exporting dataslots singly or in bulk.

Note: You can assign dataslots to individual worksteps in the process, using the Fields tab in the workstep's Properties view. For more information, see Setting workstep properties on page 217.
For details, see the following topics:

- Reviewing the Dataslots tab
- Supported dataslot types
- Using system dataslots
- Creating user-defined dataslots
- Exporting and importing dataslots

### Reviewing the Dataslots tab

You can review existing dataslots in a process and define new dataslots in the Dataslots tab of the Business Process Modeler interface (Figure 2 on page 21). If you are adding a new process template for a process model, then only predefined system dataslots are listed, as shown in Figure 46 on page 98.

**Figure 46: Dataslots tab**

If you are modifying an existing process template, then predefined dataslots as well as dataslots that you have created for this process template are listed.

**Note:** Web applications do not include any predefined system dataslots.

Click the Dataslots tab for a new BPM process, to view the All Dataslots section, which contains all the dataslots available to the process template.

You can perform the following operations in the Dataslots tab:

**Table 25: Dataslots tab operations**

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a user-defined dataslot</td>
<td>See Creating user-defined dataslots on page 101.</td>
</tr>
<tr>
<td>Remove a dataslot</td>
<td>Select the dataslot to be removed, then click Remove. You cannot remove any of the system dataslots. You cannot remove any user-defined dataslot assigned to a workstep.</td>
</tr>
</tbody>
</table>
**Operation** | **Description**
--- | ---
Modify existing dataslots | Select the dataslot to be modified, then modify its properties in the respective sections of the **Dataslots** tab. You cannot modify the properties of any system dataslot.

Import or export dataslots | See [Exporting and importing dataslots](#) on page 111.

Filter the dataslot list | Enter text in the **type filter text** box to filter the dataslot list and display all dataslots whose names contain the matching letters.

Group the dataslot list by type (or by tag) | By default, all dataslots are grouped under dataslot type category, as discussed in [Supported dataslot types](#) on page 99. To group the dataslot list by tag namely, **System** (for system dataslots) or **User** (for user-defined dataslots), click the **Group by Tags** `( )` icon in the **All Dataslots** section. To group the dataslot list by type, click the **Group by Type** `( )` icon in the **All Dataslots** section.

Collapse the dataslot list | Click the **Collapse All** `( )` icon.

---

### Supported dataslot types

Table 26 on page 99 lists the supported ABL dataslot types for both Process Model projects and Web applications.

For more information on ABL datatypes and their default values, see the [OpenEdge Development: ABL Reference](#) guide.

**Table 26: ABL Dataslot Types Supported by Business Process Modeler**

<table>
<thead>
<tr>
<th>Dataslot Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHARACTER</td>
<td>A dataslot containing a text string. CHARACTER is the most common type of dataslot, and is selected by default. CHARACTER dataslot supports three display formats, namely Text field (default), Text area, and Combo box.</td>
</tr>
<tr>
<td>INTEGER</td>
<td>A dataslot containing 32-bit data (whole numbers).</td>
</tr>
<tr>
<td>INT64</td>
<td>A dataslot containing 64-bit data (whole numbers).</td>
</tr>
<tr>
<td>LOGICAL</td>
<td>A dataslot containing logical data that is evaluated as TRUE or FALSE.</td>
</tr>
<tr>
<td>DECIMAL</td>
<td>A dataslot containing decimal numbers up to 50 digits in length, including up to 10 digits to the right of the decimal point.</td>
</tr>
</tbody>
</table>
## Chapter 10: Using dataslots

<table>
<thead>
<tr>
<th>Dataslot Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATETIMETZ</td>
<td>A dataslot containing dates. DATETIMETZ datatype has three parts: date, time, and an integer representing the time zone offset from Coordinated Universal Time (UTC). The unit of time is milliseconds from midnight. The unit of time zone offset is minutes. Supports the following DATETIME-TZ formats in OpenEdge:</td>
</tr>
<tr>
<td></td>
<td>• yyyy-MM-dd'T'HH:mm:ss.SSS(+/-)zz:zz or yyyy-MM-dd HH:mm:ss.SSS(+/-)zz:zz</td>
</tr>
<tr>
<td></td>
<td>• yyyy-MM-dd'T'HH:mm:ss.SSS or yyyy-MM-dd HH:mm:ss.SSS</td>
</tr>
<tr>
<td></td>
<td>• yyyy-MM-dd'T'HH:mm:ss or yyyy-MM-dd HH:mm:ss</td>
</tr>
<tr>
<td></td>
<td>• yyyy-MM-dd</td>
</tr>
<tr>
<td></td>
<td>For example, 1995-01-17T12:12:12.234+05:30 or 1995-01-17 12:12:12.234+05:30.</td>
</tr>
<tr>
<td>Handle</td>
<td>A dataslot containing a handle which is a pointer to an ABL handle-based object. This can be a static object defined during compilation or a dynamic object defined at runtime.</td>
</tr>
<tr>
<td>Rowid</td>
<td>A dataslot containing a unique internal identifier for a row within a single database storage area.</td>
</tr>
<tr>
<td>Memptr</td>
<td>A dataslot containing a sequence of bytes in memory. You can use this dataslot type to manipulate the contents of a BLOB database or temp-table field in ABL language.</td>
</tr>
<tr>
<td>Raw</td>
<td>A dataslot containing any data, including data from databases other than OpenEdge.</td>
</tr>
<tr>
<td>Dataset</td>
<td>A dataslot containing the values of ABL ProDataset. This dataslot can be mapped to a DataSet or DataSet-Handle in ABL.</td>
</tr>
<tr>
<td>Table</td>
<td>A dataslot containing the values of ABL temp-table. This dataslot can be mapped to a Table or Table-Handle in ABL.</td>
</tr>
<tr>
<td>List</td>
<td>Treats a collection of related data items as a single entity. For example, a List dataslot can be used to hold choices for CHARACTER dataslots where the choices are dynamically determined by querying a database.</td>
</tr>
<tr>
<td>Map</td>
<td>A dataslot which enables you to select from a list of choices that can change dynamically. It also stores the choices that are made and these choices can be viewed and used in other worksteps.</td>
</tr>
<tr>
<td>Number</td>
<td>A dataslot containing a numeric value. The Number dataslot contains Double as a subtype. The numeric value is displayed as a decimal; for example, 2.2.</td>
</tr>
<tr>
<td>Object</td>
<td>A dataslot containing a serializable Java object. Typical Object dataslot types include Vector and Array which are passed as input or output dataslots to Adapters.</td>
</tr>
<tr>
<td>Business Object</td>
<td>A dataslot which encapsulates all the data and behavior associated with an entity of a business process. This dataslot type can retrieve data from internal and external database sources.</td>
</tr>
</tbody>
</table>

For more information on ABL datatypes, see the *Openedge Development: ABL Reference* guide.
Using system dataslots

Business Process Modeler provides predefined system dataslots for a Process Model project, which are displayed by default in the Dataslots tab in the System category (see Figure 46 on page 98) and are designed to extend the functionality of user-defined dataslots. Predefined system dataslots are read-only and their values are assigned at runtime.

**Note:** You cannot modify, remove, or export a System dataslot.

The following table summarizes the predefined system dataslots.

**Table 27: Predefined System Dataslots**

<table>
<thead>
<tr>
<th>Dataslot Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllDataslots</td>
<td>Contains (at runtime) all the user-defined dataslots and the associated values for the user-defined dataslots in a Hashtable. The Hashtable key is the dataslot name.</td>
</tr>
<tr>
<td>Creator</td>
<td>Contains the creator (at runtime) of the process instance. The value of this box can be the login name of a Business Process Portal user.</td>
</tr>
<tr>
<td>Priority</td>
<td>Contains the priority (at runtime) of the process (not the workstep priority). This box appears in all automatically generated HTML forms. Restrict priority values to Low</td>
</tr>
<tr>
<td>ProcessInstanceId</td>
<td>Contains the instance ID of the running process at runtime. This value is a unique identifier of each instance of this process.</td>
</tr>
<tr>
<td>ProcessName</td>
<td>Contains the process name (at runtime). The process name is a string formed by concatenating the process template name with a unique number.</td>
</tr>
<tr>
<td>ProcessTemplateName</td>
<td>Contains the process template name (SPT).</td>
</tr>
<tr>
<td>StartTime</td>
<td>Contains the date (at runtime) when the process instance was started.</td>
</tr>
<tr>
<td>WorkstepName</td>
<td>Contains the name (at runtime) of the workstep the process is currently executing.</td>
</tr>
</tbody>
</table>

**Note:** Business Process Modeler processes can only read the values of predefined system dataslots. The system dataslots are only available as input dataslots to the process.

Creating user-defined dataslots

You can create dataslots, which are specific to a process. You can create a dataslot from any of the supported dataslot types (see Supported dataslot types on page 99) as described in the following procedures.

**To define dataslots:**

1. From the Dataslots tab, click Add.
The first page of the **New Dataslot** wizard appears.

**Figure 47: New Dataslot wizard: Page 1**

2. Enter a unique name for the dataslot in the **Enter a unique ID** box. Valid dataslot names must start with a letter and can contain only include alphanumeric characters and the underscore (_). The maximum length of a dataslot is 28 characters.

The following restrictions apply to dataslot names:

   • Although Business Process Modeler accepts reserved Java words (or Java identifiers) as dataslot names, do not use them as dataslot names or the application fails to publish. For example, do not use “LOGICAL” or “object” as a dataslot name, however, you can use “myboolean” or “myobject”.

3. Select the dataslot type from the **Select a type** list. The options are: CHARCATER, INTEGER, INT64, LOGICAL, DATETIMETZ, Number, List, Map, Object, and Business Object. For more information, see Table 26 on page 99.

4. Specify the dataslot tags in the **Enter tags...** box. Process Modeler provides a default tag named “User” for all user-defined dataslots. You can either retain this default tag or type your own tag in the **Enter tags...** box. You can then assign these tags to new and existing dataslots.

5. Click **Next**. You can use the **Value** page to set the initial value of the dataslot. The appearance of the **Value** page depends on the type of dataslot that you have selected. For detailed information about setting the initial value, see the respective sub-section for the dataslot type in Dataslot properties on page 103.

6. Click **Next**. In the **Description** page, enter a brief description of the dataslot.

7. Click **Finish** to create the new dataslot, and add the defined dataslot to the **Dataslots** tab.
Dataslot properties

After defining a dataslot, you can configure the dataslot properties in the respective sections of the **Dataslots** tab. The variations in the dataslot properties are described in the following sub-sections.

For a CHARACTER dataslot

You can configure the properties of a CHARACTER dataslot in the following sections of the **Dataslots** tab:

<table>
<thead>
<tr>
<th>Table 28: CHARACTER dataslot properties</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Section</strong></td>
</tr>
<tr>
<td>-------------</td>
</tr>
<tr>
<td>Dataslot</td>
</tr>
<tr>
<td>Description</td>
</tr>
<tr>
<td>Initial Value</td>
</tr>
<tr>
<td><strong>Note:</strong> When you select Combo as the format type for a CHARACTER dataslot from the <strong>Default Format</strong> section and then define a choice list, you must assign the CHARACTER dataslot’s initial value from the defined list. Otherwise, the process fails to publish.</td>
</tr>
<tr>
<td>The CHARACTER dataslot’s initial value should not contain the bar (“</td>
</tr>
<tr>
<td>Default Format</td>
</tr>
<tr>
<td>Usage</td>
</tr>
</tbody>
</table>

For a INTEGER and INT64 dataslot

You can configure the properties of a INTEGER and INT64 dataslot in the respective sections (same sections as that for CHARACTER dataslot) of the **Dataslots** tab. For information on CHARACTER dataslot properties, see *For a CHARACTER dataslot* on page 103. The only difference is in the **Initial Value** section, which is described below.
You can use the **Enter an initial value** box to specify the initial value of the dataslot.

### For a LOGICAL dataslot

You can configure the properties of a LOGICAL dataslot in the respective sections (same sections as that for CHARACTER dataslot) of the **Dataslots** tab. For information on CHARACTER dataslot properties, see *For a CHARACTER dataslot* on page 103. The only difference is in the **Initial Value** section, where you can select *true* or *false* (default) option as the initial value of the dataslot.

### For a DATETIMETZ dataslot

You can configure the properties of a DATETIMETZ dataslot in the respective sections (same sections as that for CHARACTER dataslot) of the **Dataslots** tab. For information on CHARACTER dataslot properties, see *For a CHARACTER dataslot* on page 103. The only difference is in the **Initial Value** section, which is described below.

You can use the **Initial Value** section to specify the initial value of the dataslot.

**Figure 48: Initial Value section for a DATETIMETZ dataslot**

Select the **Date** option to use the calendar to set the date. You can modify the default date by clicking the ☁️ icon. You can also enter the specific time in hour, minute, and second, which you want to use for the dataslot. This dataslot converts string to date-time formats and vice versa.

Alternatively, select the **Start Time** option to use the “@STARTTIME” dataslot value as the starting time for the DATETIMETZ dataslot.

### For an List dataslot

You can configure the properties of an List dataslot in the respective sections (same sections as that for CHARACTER dataslot) of the **Dataslots** tab. For information on CHARACTER dataslot properties, see *For a CHARACTER dataslot* on page 103. The only difference is in the **Initial Value** section, which is described below.

You can use the **Initial Value** section to specify the initial value of the dataslot.

**Figure 49: Initial Value section for a List dataslot**

- Click **Add** to add a list element to the **Value** table.

To remove a value from the dataslot, select the value in the **Value** table and click **Remove**.
To change the order of the values, select the value in the panel and click **Move Up** or **Move Down** to move it up or down a row.

Values in a List dataslot are now read from a `<Values>` tag. The `<Value>` tag is compatible with the the `<DataValue>` tag used in earlier releases is also read.

**Note:** For more information, see About List Dataslots in the Application Developer’s Guide.

---

### For a Map dataslot

You can configure the properties of a Map dataslot in the respective sections (same sections as that for CHARACTER dataslot) of the Dataslots tab. For information on CHARACTER dataslot properties, see For a CHARACTER dataslot on page 103. The only difference is in the **Initial Value** section, which is described below.

You can use the **Initial Value** section to specify the initial value of the dataslot.

**Figure 50: Initial Value section for a Map dataslot**

To add a name/value pair to the table:

1. To add name/value pairs as choices in a multi-select list, click **Add** to open the **Add** dialog box (right image, Figure 50 on page 105), in which you can define your name/value pair.

   The Name of the option you enter will be displayed as its label. The Value you enter can be “true” or “false”. If “true”, it is selected by default when displayed; if “false”, it is not selected.

2. Click **OK** to add the defined name/value pair to the **Initial Value** table.

You can also change the options displayed for a Map dataslot or provide localized values for the options. In the Figure 50 on page 105, the name “coke” will be displayed as an option when the application is published on the Business Process Portal. To specify another drink in the case of a different locale or context, say “soft drink” and manually edit the properties file in the properties folder under the application by adding `coke=soft drink`; or open the **Label Properties** dialog box in the Form Editor and enter “coke” as the Key and “Soft-drink” as the Value.

---

### For a Number dataslot

You can configure the properties of a Number dataslot in the respective sections (same sections as that for CHARACTER dataslot) of the Dataslots tab. For information on CHARACTER dataslot properties, see For a CHARACTER dataslot on page 103. The only difference is in the **Initial Value** section.

You can use the **Enter an initial value** box to specify the initial value of the dataslot.
For an Object dataslot

An object dataslot contains a serializable Java object of a class available on the BP Server server. You can use a project-specific class file (JAR) or common JAR files which are available on publishing a Common Resource project. The class object is instantiated when a BP Server application instance is created or when a Web application is accessed.

You can configure the properties of an Object dataslot in the respective sections (same sections as that for CHARACTER dataslot) of the Dataslots tab. For information on CHARACTER dataslot properties, see For a CHARACTER dataslot on page 103. The only difference is in the Initial Value section, which is described below.

You can use the Initial Value section to specify the initial value of the dataslot.

**Figure 51: Initial Value section for an Object dataslot**

1. In the **Java Class Name** box, enter a unique name for the serializable Java object. Specify the Class Name (including the package name).
   - Classes, which are project-specific, are stored in `<project_name>\lib` folder (for BPM processes) or in `<project_name>\WEB-INF\lib` folder (for Web applications).
   - Classes, which are available on the BP Server server after you publish a Common Resource project, are stored in `OEBPS_Home\ebmsapps\common\classes` folder (for a BPM process) or in `WebApp_Home\WEB-INF\classes` folder (for Web application) wherein the `WebApp_Home` is application-server-specific and is the value of the `oebps.webappdir` property in `OEBPS_Home\conf\oebps.conf` file.

2. Use the **Constructor Parameters** section to define the parameters of the class constructor, which is invoked when the process instance is created. The object variables are initialized by invoking the proper constructor with the appropriate parameter values.
   a) Define the parameters by clicking **Add**, to open the **Add** dialog box (right image, Figure 51 on page 106).
   b) Specify the Java class name in the **Type** box and enter the value of the parameter.
      You can use Progress Developer Studio for OpenEdge to specify the methods for this class object in the Before Activation and When Completed tabs of the Advanced tab in workstep properties. For more information regarding using these tabs, see Progress Developer Studio for OpenEdge User's Guide.

3. Click **OK** to add the constructor parameter in the **Constructor Parameters** section (left image, Figure 51 on page 106).
For a Business Object dataslot

You can configure the properties of a Business Object dataslot in the respective sections (same sections as that for CHARACTER dataslot) of the **Dataslots** tab. For information on CHARACTER dataslot properties, see For a CHARACTER dataslot on page 103. The only difference is in the **Initial Value** section, which is described below.

The figure below shows how to create a business object dataslot.

**Figure 52: Initial Value section for a Business Object dataslot**

![Initial Value section for a Business Object dataslot](image)

**For a Business Object dataslot:**

1. Click **Browse** to open the **Business Object** dialog box (right image, Figure 52 on page 107).
2. Select a java file and click **OK**, and a valid business object is displayed in the **Initial Value** box. These values are derived from business objects that were previously created. For more information, see Managing business objects on page 397.

**Defining the dataslot description**

You can add a description (optional) for a dataslot by selecting the dataslot and entering the description text in the **Description** section of the **Dataslots** tab. The Description text area can accept any character or symbol, and has no restrictions on the amount of characters you can enter.

**Defining the dataslot format**

You can use the **Default Format** section of the **Dataslots** tab to set properties such as the presentation format and the label of the dataslot in Business Process Portal, validation (if required). You can also specify whether the dataslot is editable and required.

**Note:** The **Default Format** tab is not available for the Business Object dataslot.

Table 29 on page 108 lists the format properties, their descriptions, and the dataslot types for which they are available.
Table 29: Default Format for Dataslots

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Supported Dataslot Types</th>
</tr>
</thead>
</table>
| Type    | Determines how the dataslot should be presented in the interface page. | • As a text box, text area, or combo box for a CHARACTER dataslot.  
• Additionally, as a radio button, checkbox and label for CHARACTER dataslot in case of web applications.  
• “Date and Time” or “Date Only” for DATETIME-TZ dataslot. |
| Label   | Defines the label of the dataslot in the interface. | For all dataslots. |
| Editable| Specifies whether the dataslot is editable or read-only in the interface page. The **Editable** check box is selected, by default. | For all dataslots. |
| Required| Specifies whether the dataslot is mandatory or not. | Not available for LOGICAL dataslot. |
| Validation | Configures validation settings. | Available only for CHARACTER, INTEGER, INT64, and DATETIME-TZ dataslots. For more information, see *Specifying validation at the dataslot level* on page 110. |

You can modify the format of a dataslot at the workstep level, allowing each workstep in a process to have a different format. For example, you can set the format of a dataslot as a combo box with the label “Please enter a choice” in one workstep; and as a text box with the label “You selected:” in the next workstep. To modify the format of a dataslot at the workstep level, see the Dataslot Properties section for the workstep type; for instance, for an Activity workstep, see *Defining properties of Activity worksteps* on page 224.
Presentation formats for the CHARACTER dataslot

Business Process Modeler enables you to display a CHARACTER dataslot as a text box (single line of text), a text area (multiple lines of text), or a Combo box. In the case of Web applications, the CHARACTER dataslot has three additional formats, namely Radio button, Check box, and Label (read-only text). Each of these formats are described in the following bulleted list.

- **Text Field** format: Select this format and then do the following:

  **Figure 53: Text Field format for CHARACTER Dataslot**

  1. Enter data in the **Size** and **Length** box (only for Web applications) to define the dimensions of the dataslot as it appears in the interface.
  2. Select the **Password** checkbox to present text entered in this field as asterisks, same as the way passwords are commonly displayed.

- **Text Area** format: Select this format and enter the width and height of the text area in the respective fields.

- **Combo box** format: Select this format and then do the following:

  **Figure 54: Combo Box format for CHARACTER Dataslot**

  1. To add a new choice for the combo box, click **Add** to open the **Choice** dialog box.
  2. Enter a Value and a Label in the **Choice** dialog box, (right image, **Figure 54** on page 109). A choice can only include alphanumeric characters and underscores (_).
  3. Click **OK** to add the choice you created to the Choices table.

  To modify an entry, select the choice in the Choices section and click **Modify**. To remove a choice, select the choice and click **Remove**. To change the order of choice entries, select the choice and click **Move Up** or **Move Down**.
Radio button and Checkbox formats (only for Web application): The procedure for defining the options for these controls are same as those for defining Combo box.

Label format (only for Web applications): You can enter only a label for the dataslot.

Changing the presentation format at the workstep level does not affect the presentation at the process level.

Specifying validation at the dataslot level

Business Process Modeler introduces a new validation framework that enables you to specify validation at the dataslot level. You can now associate validation rules with a CHARACTER, an INTEGER, or a DATETIMETZ dataslot type.

Table 30 on page 110 lists and describes the dataslot validation rules and the dataslot type for which they are available.

Table 30: Validation Rules for User-Defined Dataslots

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>Available only for CHARACTER dataslots. Enter a Minimum Length and a Maximum Length for the string text.</td>
</tr>
<tr>
<td>Number validation</td>
<td>Available for CHARACTER and INTEGER dataslots. Select the Number type (Integer or Float) and enter the minimum value and the maximum value in the respective text boxes.</td>
</tr>
<tr>
<td>Decimal validation</td>
<td>Available for CHARACTER, DECIMAL, and Number dataslots. Enter the number of digits to the left and to the right of the decimal point in the respective text boxes.</td>
</tr>
<tr>
<td>Decimal validation (advanced)</td>
<td>Available for CHARACTER, DECIMAL, and Number dataslots. In addition to the Decimal validation rule, this rule enables you to set a range of digits allowed to the left and to the right of the decimal point.</td>
</tr>
<tr>
<td>IP Address</td>
<td>Available only for CHARACTER dataslots. Define the Minimum Port Allowed or the Maximum Port Allowed. Default values are 0 for Minimum and 65535 for Maximum.</td>
</tr>
<tr>
<td>U.S. Social Security Number</td>
<td>Available only for CHARACTER dataslots. The predefined format (at runtime) is xxx-xx-xxxx.</td>
</tr>
<tr>
<td>Currency</td>
<td>Available only for CHARACTER dataslots. You can modify the Currency Character (default is &quot;$&quot;), Grouping Character (default is &quot;,&quot;) and Decimal character (default is &quot;.&quot;).</td>
</tr>
<tr>
<td>U.S. Zip Code</td>
<td>Available only for CHARACTER dataslots. You can modify the Separator (default is &quot;-&quot;). You need to enter a 5-digit or 9-digit zip code at runtime. For the 9-digit zip code, the predefined format (at runtime) is xxxxx-xxxx.</td>
</tr>
</tbody>
</table>
| U.S. Style PhoneNo.         | Available only for CHARACTER dataslots. Select either Loose or Strict format.  
                                * For Strict format, you need to enter the phone number (at runtime) as (xxx) xxx-xxxx or (xxx) xxx xxxx.  
                                * For Loose format, you can enter a phone number (at runtime) without the brackets or hyphen or without both. |
| Email Address               | Available only for CHARACTER dataslots. The format is predefined. |
To assign a validation rule to a dataslot:

1. Select the CHARACTER, INTEGER, INT64, or DATETIMETZ dataslot in the All Dataslots section.
2. From the Default Format section, click Browse beside the Validation box.

The Validation dialog box appears.

Figure 55: Validation dialog box

3. Select any of the available rules from the Validation Rule drop-down list. Depending on the validation rule you selected, you may have to enter additional information, as described in Table 30 on page 110.

Note: If you want to apply any of the decimal validation rules to a INTEGER dataslot using the Decimal validation option in the Validation Rule drop-down list, make sure that you have selected either the Double or Decimal options in the Storage section.

4. Click OK to save the changes you made to the dataslot properties.

Exporting and importing dataslots

Business Process Modeler enables you to import or export dataslots, either singly or in bulk, to a file, which is only available to you or to a repository, which is available to all Process Modeler users.

Exporting dataslots

You can export one or more user-defined dataslots to an existing file or to a file, which you specifically create for the dataslots.
To export dataslots:

1. From the Dataslots tab, click the Export (EXPORT) icon. The Export Dataslots wizard appears.

Figure 56: Export Dataslots: Page 1

2. From the first page of the Export Dataslots wizard, select any of the following options, then click Next:
   - CSV (Comma-delimited) File: to export the dataslots to a CSV (comma-delimited) file.
   - XML File: to export the dataslots to an XML file.

3. If you select the CSV (Comma-delimited) File option, the CSV (Comma-delimited) File page of the Export Dataslots wizard appears.
   a) If you want to export the dataslots to an existing CSV file, select the Overwrite existing file without warning checkbox to replace the contents of the existing file without any user warning.
   b) To create a new dataslot file, click Browse File System.
   c) Enter a name in the File name box and click Save. Alternatively, to export the dataslots to an existing file, select the file and click Save.

   The dataslot file and its location is displayed in the text box provided in the CSV (Comma-delimited) File page.

4. If you select the XML File option, the XML File page of the Export Dataslots wizard appears.
   a) If you want to export the dataslots to an existing XML file, select the Overwrite existing file without warning check box to replace the contents of the existing file without any user warning.
   b) To create a new dataslot file, click Browse File System.
   c) Enter a name in the File name box and click Save. Alternatively, to export the dataslots to an existing file, select the file and click Save.

   The dataslot file and its location is displayed in the text box provided in the XML File page.

5. Click Next to display the Dataslots page of the Export Dataslots wizard.
You can use the **Dataslots** page to select the dataslots that you want to export. The **Dataslots** page displays all user-defined dataslots in your process.

**Figure 57: Export Dataslots: Dataslots page**

![Dataslots page](image)

a) Select the checkbox of each dataslot or the dataslot type (for example, "CHARACTER") check box (to select all dataslots of that type). Alternatively, click **Select All** to select all of the listed dataslots. To deselect the dataslots, click **Deselect All**.

6. Click **Finish** to export the dataslots. If you are exporting the dataslots to a file and have not selected the **Overwrite existing file without warning** checkbox, you will be prompted to click **Yes** to replace the content of the file with the exported dataslots.

### Importing dataslots

You can import user-defined dataslots from an existing dataslot XML (or CSV) file or from an existing process template file.

**To import dataslots:**

1. From the **Dataslots** tab, click the **Import** icon.
The **Import Dataslots** wizard appears.

**Figure 58: Import Dataslots: Page 1**

2. From the first page of the **Import Dataslots** wizard, select any of the following options, then click **Next**:
   - **CSV (Comma-delimited) File**: to import the dataslots from a CSV (comma-delimited) file.
   - **XML File**: to import the dataslots from an XML file.

3. If you select the **CSV (Comma-delimited) File** option, the **CSV (Comma-delimited) File** page of the **Import Dataslots** wizard appears.
   a) To select the CSV dataslot file, click **Browse File System**.
   b) Select the CSV file and click **Open**.
      The selected file and its location is displayed in the text box provided in the **CSV (Comma-delimited) File** page.

4. If you select the **XML File** option, the **XML File** page of the **Import Dataslots** wizard appears.
   a) Click **Browse File System** to browse to an file.
   b) Select the file (XML, SPT, or SWT) and click **Open**.
      The dataslot file and its location is displayed in the text box provided in the **XML File** page.

5. Click **Next** to display the **Dataslots** page of the **Import Dataslots** wizard.
   
   You can use the **Dataslots** page to select the dataslots that you want to import.
   
   a) Select the checkbox of each dataslot or the dataslot type (for example, "CHARACTER") check box (to select all dataslots of that type). Alternatively, click **Select All** to select all of the listed dataslots. To deselect the dataslots, click **Deselect All**.

6. Click **Finish** to import the dataslots.
You can define performers and assign them to worksteps. Business Process Modeler supports three types of performers—**User** (human performers), **Adapter** (a Java class as the performer), and **Subprocess** (another process, which can be assigned to a workstep).

**Note:** Web applications do not support user-type performers.

This chapter describes how to define and assign performers, as well as use swim lanes.
For details, see the following topics:

- Reviewing the Performers tab
- Defining performers
- Assigning performers
- Using swim lanes

### Reviewing the Performers tab

You can review existing performers and define new performers in the Performers tab of the Business Process Modeler interface.

![Performers tab](image)

Click the Performers tab for a new Process Model project, to view the All Performer section, which contains all the performers available to the process template.

You can perform the following operations in the Performers tab:

#### Table 31: Operations in Performers tab

<table>
<thead>
<tr>
<th>Operations</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a performer</td>
<td>See Defining performers on page 118.</td>
</tr>
<tr>
<td>Remove a performer</td>
<td>Select the performer to be removed, then click Remove. You cannot remove the default performer, @CREATOR.</td>
</tr>
<tr>
<td>Modify the performer details</td>
<td>Modify the performer details. You cannot modify the details of the default performer, @CREATOR.</td>
</tr>
<tr>
<td>Filter the performer list</td>
<td>Enter text in the type filter text box to filter the performer list and display all performers whose names contain the matching letters.</td>
</tr>
<tr>
<td>Operations</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Group (or ungroup) the performer list</td>
<td>By default, all performers are grouped under performer type categories, namely, <strong>User</strong> (for human performer), <strong>Adapter</strong> (for adapter performers), or <strong>Sub-Process</strong> (for subprocess performer). To ungroup the performer list, click the <strong>No Grouping</strong> icon in the <strong>All Performer</strong> section. To group the performer list under categories, click the <strong>Group by Type</strong> icon in the <strong>All Performer</strong> section.</td>
</tr>
<tr>
<td>Collapse the performer list</td>
<td>Click the <strong>Collapse All</strong> icon.</td>
</tr>
</tbody>
</table>

You can assign any performer defined in the **All Performer** section to a workstep. The Performer page stores all performers defined for the process template, whether they are assigned to a workstep or are not assigned.

### Default performers

In addition to the **Performers** tab, you can view existing performers by clicking the **Assign Participants** link in the Tasks pane.

**Figure 60: Performers in Tasks pane**

Expand each Performer type to see the available performers for that type. Business Process Modeler provides the following default performers:

1. For Process Models, expand the **Users** folder to display two subfolders, namely:
   - **Process**, which stores user-type performers defined for the process template. This includes a default user named "Creator." "Creator" is a dynamic dataslot value, which specifies the
creator of the process instance: value can change from instance to instance and can reflect changes made at runtime.

- **Organization**, which contains users defined using the User Management tool (see Working with User Management tool on page 381) under three subcategories namely, Users, Groups, and Queues.

2. **Managed adapters** enable you to define configuration and mapping between dataslots in a process template and inputs/outputs from a group of out-of-the-box adapters at process design time. These predefined managed adapters provide examples of specialized adapters that are commonly used in managing business processes. For more information, see Using managed adapters on page 121.

### Defining performers

You can define performers of any type from the **Performers** tab (see Figure 59 on page 116). Alternatively, you can define and assign a performer to an Activity workstep using the **Assign Participants** link in the Tasks pane (see Using the Tasks pane on page 126).

This section describes how to define performers in the **Performers** tab.

### Defining a user as a performer

You can add a new user (human performer), group (group of human performers), or a queue for a Process Model.

**To define a user as a performer:**

1. From the **Performers** tab, click **Add**.
The first page of the **New Performer** wizard appears.

**Figure 61: New Performer wizard: Page 1**

2. Select the **User** type of performer from the available options, then click **Next**. Options for Process Models include User, Adapter, and Sub-Process.

**Figure 62: New Performer wizard: User Performer page**

3. Enter an appropriate name for the User performer in the **Enter a name** box. Alternatively, for performer type, "User," you can link the performer to a CHARACTER dataslot. Type '@' in the **Enter a name** box to display the list of CHARACTER dataslots in the process. Select the dataslot to be linked to this performer.

**Note:** If your process contains a business object dataslot (for example, "bo1"), you can also create an individual user mapped to the business object attribute of type, String. Enter the performer name as @<dataslot_name>. (in this example, @bo1.) and then enter the attribute name (only of type, String) of the business object. If the business object contains another business object (example, "bo2") as an attribute, you can use a nested expression (in this example, @bo1.bo2.) to enter the attribute name (only of type, String) of business object, "bo2."

Alternatively, click **Browse** to select a user, group, or queue defined in the User Management tool (as discussed in Working with User Management tool on page 381). The dialog box that appears depends on the performer type that you select in the **Select a type** section; for example, if you select "User" as the performer type, the **User** dialog box listing all users defined in the User Management tool appears. Select a user (or group or queue) from those listed, then click **OK** to display the name in the **Enter a name** box.
4. Select the user type in the Select a type section to indicate whether the human performer is an individual user, queue, or a group.

Table 32 on page 120 describes each of the human performer types.

<table>
<thead>
<tr>
<th>Performer Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individual user</td>
<td>A single human performer.</td>
</tr>
<tr>
<td>Group of users</td>
<td>A group of human performers. Select <strong>Any member(s) of the group</strong> if only one member from the group is required to complete a task OR select <strong>All</strong> if each performer from the group is required to complete the task. You can filter the members of a selected group by entering a valid role (as defined in the Administration module) in the <strong>Role</strong> box. For example, if <strong>All</strong> option is selected, and Manager is entered in the <strong>Role</strong> box, then all managers from this group are required to complete the workstep task.</td>
</tr>
<tr>
<td>Queue</td>
<td>A queue is defined as a group of users with the <strong>Any member(s) of the group</strong> option selected. At this time, queues created in Progress Developer Studio for OpenEdge do not automatically appear in Business Process Portal, but the same queue name must be created again in Business Process Portal. Also, queues created in the Administration module of Business Process Portal do not appear in the list of performers in the <strong>Performer</strong> tab.</td>
</tr>
</tbody>
</table>

**Note:** When you assign a User performer type to a workstep, Business Process Modeler checks if the performer is an Individual User, then checks if it is a Group of users, then if it is a Queue. If it is none of these, then the task is assigned to Manager. For this reason, names for User, Group, and Queue must be unique.

5. Click **Finish** to add the new performer to the Performers tab.

**Note:** To modify the user performer details, select the performer to be modified and make your changes in the **User** section of the Performers tab.

---

### Defining an adapter as a performer

An adapter (previously called an external performer or EP) is a Java class that provides a mechanism that integrates external third-party classes and actions with Business Process Modeler; for instance, an adapter can automate some functions and tasks performed by a server or an external system.

Adapters are the "performers" of Adapter worksteps. Depending on your application requirements, you can use a predefined managed adapter provided by Business Process Modeler; or you can develop your own adapters to carry out specific tasks. Business Process Modeler provides a series of managed adapters that have complex, specific features. Advanced users can find out more about Managed Adapters in the Managed Adapters Guide.
To define an adapter and add it as a performer:

1. From the Performers tab, click Add.
2. From the first page of the New Performer wizard (Figure 61 on page 119), select the Adapter type of performer, then click Next.

Figure 63: New Performer wizard: Adapter Performer page

3. Type the adapter name in the Enter a name box. Specify the Java class and method in the appropriate boxes.
4. Select the Generate Java code checkbox if you want to generate a Java file for the Adapter workstep.
5. Click Next. In the Description page, enter a brief description of the custom adapter.
6. Click Finish to create the custom adapter and add it to the Performers tab.

Note: To modify the adapter performer details, select the performer to be modified and make your changes in the Adapter and Description sections of the Performers tab.

Using managed adapters

Business Process Modeler provides Managed adapters that enable you to define configuration and mapping between dataslots in a Process Model and inputs/outputs from a group of out-of-the-box adapters at process design time. Managed adapters are compatible with Business Process Modeler's adapter configuration and mapping framework.

Process Modeler provides the following “out-of-the-box” Managed adapters for Process Models:

- **DB Adapter**: Provides connections to one or more specified databases, and enables you to quickly access SQL statements and parameters.
- **Email Adapter**: Enables Business Process Modeler to generate and send e-mail messages.
- **File Adapter**: Reads from and writes to a variety of file formats.
- **FTP Adapter**: Enables you to transfer files from and to your local file system, or an FTP server.
- **JMS Adapter**: Enables you to use Java Message Service (JMS) to communicate asynchronously with external systems and facilitates data exchange between processes and external applications.
• **OpenEdge Adapter**: Enables you to integrate Business Process Server applications with Progress® OpenEdge® solutions.

• **Web Service Adapter**: Provides a connection to a specified Web Service, and enables you to map Web service methods to a dataslot in a Modeler process.

**To specify a managed adapter as the performer of a workstep:**

1. Click the **Assign Participants** link in the Tasks pane and expand the **Adapters > Managed** folder to display the managed adapters folders available by default.
2. Expand the respective managed adapter folder, then drag the adapter to an existing workstep (or a blank area of the Content pane) to add the adapter workstep.

The managed adapter is now assigned as a performer of an Adapter workstep.

**Defining a subprocess performer**

A Subprocess is a process template that can be included into another process template by using a subprocess workstep. In this case, the performer of the subprocess workstep is the subprocess itself.

You can define a subprocess in a local or remote server as a subprocess performer in your process template without exiting Business Process Modeler by performing the following steps.

1. From the **Performers** tab, click **Add**.
2. From the first page of the **New Performer** wizard (Figure 61 on page 119), select the **Sub-Process** type of performer, then click **Next**.

   ![Figure 64: New Performer wizard: Sub-Process Performer page](image)

3. Enter a name for the subprocess in the **Enter a name** box.
4. You can use an alias to appear as the subprocess instance name in Business Process Portal instead of the lengthy subprocess name that appears by default. In the **Enter an alias** box (available only for Process Model), enter the alias name. Alternatively, you can use a CHARACTER dataslot as the alias. Type ‘@’ in the **Enter a name** box to display the list of CHARACTER dataslots in the process. Select the dataslot to be used as the alias. When you use an alias, the subprocess name is in a format (<alias>#<sub-piid>) that controls the length of the subprocess name.
Note: To use aliases in multiple subprocesses, the Sub-Process box should contain a mapped dataslot for multiple processes and the Alias box must use dataslots of LIST type. In this way, each index can specify the alias for the respective subprocess. For more information, see “Subprocess Alias” in Chapter 16 of the BP Server Developer’s Guide, or the “Subprocess Alias” example in BM_Home\BP_Server\examples.

5. In the Select a process box, enter a valid name for the process you want to use as a subprocess. Alternatively, you can select one of the following options:

- Assign a dataslot as a dynamic performer of the subprocess. Type '@' in the Select a process box to display the LIST of CHARACTER dataslots in the process. Select the dataslot to be used as the subprocess performer.
- Click Browse to select a specific process template file to be assigned as a static subprocess performer.

Placing an existing process template in the subprocess box converts that process template to a subprocess.

6. Use the Enter a server URL box (available only for process models) to connect to a remote server (provided your Business Process Server administrator has already set up this connection). Enter a valid path to the external server using the following format: jndi://$RemoteMachine><jndiName>. Alternatively, type '@' in the Enter a server URL box to assign a CHARACTER dataslot as the dynamic performer on the remote server.

7. Select the Indexed checkbox (available only for process models) if you want to pass data from a LIST dataslot in a parent process to multiple subprocesses, with each subprocess containing a Primitive dataslot that asynchronously receives specific information from, and sends specific information back to, the LIST dataslot. You can map dataslots in the parent process to dataslots in each subprocess; or, if the dataslot name in each subprocess matches the dataslot name in the parent process, then no mapping is necessary.

This Indexed feature ensures that dataslot values in a subprocess are not overwritten, and that the correct value from a INTEGER dataslot is returned to its specific slot in the LIST dataslot.

If you are using a LIST dataslot as the subprocess performer and the Indexed check box is not selected, then each subprocess that is completed and submits data to the parent process may overwrite data from subprocesses that were completed earlier. Therefore, use a non-Indexed subprocess only when no data is to be returned or when overwritten of data does not matter. For more information about using a LIST dataslot to invoke multiple subprocesses, see Chapter 4, “Using a LIST Dataslot to Invoke Multiple Subprocesses” in the Application Developer’s Guide.

8. Click Next. In the Description page, enter a brief description of the subprocess.

9. Click Finish to create a subprocess performer and add it to the Performers tab.

Note: To modify the subprocess performer details, select the subprocess performer and make your changes in the Sub-Process and Description sections of the Performers tab.
Defining an inline subprocess

The Inline Subprocess is a self-contained subprocess that contains the same dataslots and other data variables as its parent process, and opens within the same File tab as its parent. By creating a process and subprocesses that share the same data variables, you can avoid using external processes as subprocesses to the active project. You may also avoid creating or assigning duplicate dataslots for these external processes, and performing complex dataslot mapping. Inline Subprocesses also reduce overheads by avoiding the checking out of multiple external subprocesses that might be part of other Process Model projects.

**Note:** The Inline Subprocess is not available for Web applications.

Note the following restrictions on the use of Inline Subprocesses:

- You cannot convert an inline subprocess to a conventional subprocess.
- The Rollback function is not supported for Inline Subprocesses.

You can use one of the following three methods to create an inline subprocess:

- Use the New Embedded Subprocess icon from the Tasks pane (see Using the New Embedded Subprocess icon on page 124).
- Use the Collapse functionality to collapse a group of worksteps having a single input link and a single output link to form an inline subprocess (see Using the Collapse functionality on page 125).
- Use a process model from the Repository Browser into the diagram (see Using a process model on page 126).

Using the New Embedded Subprocess icon

To define an embedded subprocess:

1. From the Tasks pane, click the Assign Participants link, then drag the New Embedded Sub-Process icon into the process template diagram. Notice that the Embedded Subprocess is indicated by the icon.
2. Open the subprocess workstep’s Properties view and define workstep properties as required.
3. Right-click the subprocess workstep and click Open to open a blank interface in which you can add shapes, connectors, and other data to create an inline subprocess.
4. To return to the parent process or to a higher level subprocess, in the Business Process Modeler toolbar, click the One Level Up ( ) icon or select the parent process or higher subprocess from the adjacent drop-down list.
Using the Collapse functionality

You can define an Inline Subprocess using the **Collapse** feature by quickly selecting several related worksteps and creating a self-contained inline subprocess, which is represented in the diagram as a single workstep.

1. Use the **Select** tool ( ) from the palette bar to select a workstep or a group of worksteps. The **Collapse** feature is enabled only if all these worksteps match the criteria of a single point of entry and a single point of exit.
2. Right-click the selected worksteps, then select the **Collapse** option to convert the selected workstep or group of worksteps to an inline subprocess.
3. The inline subprocess workstep is indicated by an Expandable ( ) icon (Step Figure 65 on page 125). Open the workstep’s **Properties** view and assign a Label, Priority, and other properties, as required.

**Figure 65: Inline Subprocess Workstep in Process Template**

4. To view the inline subprocess, right-click the inline subprocess workstep and then select **Open**. The grouped worksteps are displayed, as shown in Step Figure 65 on page 125. The current subprocess, Subprocess 1 in this example, is contained under the tab of the parent process and is displayed in the drop-down list adjacent to the **Up One Level** icon ( ).

**Figure 66: Inline Subprocess Opened in the Parent Tab**
5. To move up one level and, in this example, return to the parent process, click \( \text{△} \). You can use \( \text{△} \) to move up the chain of processes, or you can select a process from those listed in the adjacent box. The current subprocess always appears as the last name in the list of parent process and subprocesses; therefore, the combo box is disabled when the parent process is active.

6. If the worksteps you have grouped include Subprocess worksteps, then you can drill down further into the subprocesses by right-clicking the workstep and selecting \( \text{Open} \). To move up the chain, click \( \text{△} \) or select from the parent process/subprocesses list, which displays all visited inline subprocesses.

Using a process model

You can also create an Inline Subprocess by dragging a process model from the Repository Browser into the process template diagram. For details, see Adding an application to an existing process.

All dataslots present in the parent process are added to the inline subprocess and any dataslots present in the inline subprocess are merged with the parent's dataslots.

Assigning performers

After defining a performer, you must assign the performer to a workstep. By default, when you drag an Activity shape \( \text{Activity} \) from the Tasks pane to the content pane, you create an activity with no performer assigned. In certain cases, you may want to add this type of activity with an unspecified performer to your process template.

You can then perform any of the procedures, described in the following sections, to assign a performer:

Using the Tasks pane

You can use the Assign Participants link in the Tasks pane to define a new performer and then add the appropriate workstep (depending on the performer type) to the process template.

The new performer icons provided by Business Process Modeler for Process Model projects are listed in Table 33 on page 126.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Purpose</th>
<th>Navigation path</th>
</tr>
</thead>
<tbody>
<tr>
<td>New User</td>
<td>To create a new user type performer.</td>
<td>Expand Users &gt; Process.</td>
</tr>
<tr>
<td>New Adapter</td>
<td>To create a custom adapter performer.</td>
<td>Expand Adapters &gt; Custom.</td>
</tr>
<tr>
<td>New Sub-Process</td>
<td>To create a subprocess performer.</td>
<td>Expand Sub-Process &gt; External References.</td>
</tr>
<tr>
<td>New Embedded Sub-Process</td>
<td>To create an embedded subprocess (not available for web applications).</td>
<td>Expand Sub-Process.</td>
</tr>
</tbody>
</table>
You can drag any of these performers into the process template diagram to open the respective dialog boxes for a User, an Adapter or a Subprocess. After you have created the performer, a workstep is automatically inserted in the process template diagram. This workstep is performed by the performer you just created. This new performer is also added to those listed in the Tasks pane under the respective Users, Adapters, or Subprocesses folder.

1. Click the **Assign Participants** link in the Tasks pane to open a list of performer types.
2. Expand the folder of the Performer type you want to assign to the activity. Select a specific new performer icon and drag it on an existing activity shape or on the blank template.

**Note:** To assign a performer created using the User Management tool, expand **Users > Organization**, then drag the performer from the respective subcategory to an existing activity or on the blank template. The assigned performer is also added to the performer list in **Users > Process** and to the **Performers** tab.

Table 34 on page 127 illustrates the appearance of the activity shape after a performer is assigned to it.

### Table 34: Activity Shape with assigned performer

<table>
<thead>
<tr>
<th>Activity Shape Appearance</th>
<th>Performer Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Activity 1" /></td>
<td>For an individual performer or a group member (with <em>Any</em> option selected).</td>
</tr>
<tr>
<td><img src="image2.png" alt="Activity 2" /></td>
<td>For a group member (with <em>All</em> members option selected).</td>
</tr>
<tr>
<td><img src="image3.png" alt="Activity 3" /></td>
<td>For a custom adapter.</td>
</tr>
<tr>
<td><img src="image4.png" alt="Activity 4" /></td>
<td>For a managed adapter. The shape varies according to the managed adapter used. In this case, it is an e-mail adapter.</td>
</tr>
<tr>
<td><img src="image5.png" alt="Activity 5" /></td>
<td>For a subprocess performer.</td>
</tr>
</tbody>
</table>
Changing the workstep performer

You can change the assigned performer of an activity workstep to any other performer or performer type.

1. Right-click an Activity, Adapter, or Subprocess workstep and then select the Change Performer option to open the Change Performer dialog box.

   Figure 67: Change Performer

2. From any of the listed performer types, select a specific performer and click OK.

You can also change the performer through the Activity or Subprocess workstep’s Properties view. This procedure, however, only enables you to change to a performer of the same performer type.

Using swim lanes

A swim lane is a horizontal colored lane across the process template, which displays the worksteps performed by a specific user-type performer. Process template designers can use swim lanes to associate human performers with the worksteps they perform. You can also add a swim lane that has no specific performer, and any workstep added to this blank swim lane retains its original performer. Performers of Activity worksteps are automatically updated when an Activity workstep is added to a swim lane or moved from one swim lane to another. Other workstep types can be included in a swim lane, but, unlike Activity worksteps, their performers are not automatically updated.

Note: Swim lanes are not supported for Web applications.
Creating swim lanes

To add a new swim lane:

1. From the Tasks pane, click the Create Phases and Swim Lanes link in the Design Tasks panel, and then click the Swim Lane ( ) icon to open the New Lane dialog box (Figure 68 on page 129).

Figure 68: New Lane dialog box

2. Enter the ID and Name of the new swim lane in the respective boxes.

3. To assign a performer to the swim lane, click the ellipsis button beside the Performer box and then select the Select a performer option to open the Select a Performer dialog box, which displays a list of available performers that you can assign to the swim lane.

Note: You can use the Clear performer option to create a swim lane without assigning a performer to it.

4. Click the Appearance tab to apply a unique color to the swim lane.

Figure 69: Applying Swim lane Color

5. Select an option from the colors listed. You can also click More to display more color options. You can also change the color of multiple swim lanes by selecting the lanes and clicking the Color Chooser ( ) icon in the Modeler Palette to open the color palette, where you can select one of the listed colors, or click More to display more color options.

6. Click OK to add a new swim lane for the specified performer to the process template diagram. Move the workstep that this user is performing into its swim lane.
Swim lanes have two headers. The primary Pool Header contains all the swim lanes and displays the name of the Process Model. The Lane Header displays the Name, ID, or Performer of each lane, as determined by selecting one of these options from the View menu in the Lane Popup Menu. The Pool Header remains as a black and white header, but you can define a color for each Swim Lane.

By default, Business Process Modeler adds a new swim lane below existing swim lanes. You can also insert a swim lane between existing lanes.

**Figure 70: Process Template with Swim Lanes**

![Figure 70: Process Template with Swim Lanes](image)

You can assign the same performer to different swim lanes. You need not add all worksteps that have the same performer to the performer’s swim lane; these worksteps can also be placed in that part of the diagram outside the swim lanes.

**Resizing swim lanes**

You may need to resize a swim lane either to add a shape to the swim lane or to quickly move a shape into (or out of) this swim lane.

- To increase the width of a swim lane, select the **Move shapes when resizing swim lanes** checkbox, before resizing the swim lane. Shapes, adjacent to the swim lane, move as you resize the swim lane.

- To make the performer of the adjacent worksteps to that of this swim lane or to remove a shape from this swim lane, clear the **Move shapes when resizing swim lanes** checkbox, before resizing the swim lane. Shapes, adjacent to the swim lane, do not move as you resize the swim lane.

You can resize a swim lane by passing your cursor over the space between two swim lane headers and using the slider to adjust the swim lane’s size up or down.
Performing additional swim lane operations

You can perform additional operations on swim lane by right-clicking the swim lane header. Additional operations include:

- Changing the swim lane header display.
- Inserting a new lane between existing swim lanes.
- Removing a swim lane
- Modifying existing lane properties.

To change the swim lane display header:

- From the View menu (Figure 68 on page 129), select either Name, ID, or Performer to view your selection in the header of each swim lane.

To remove a swim lane:

- Click Remove Lane, after removing all worksteps from the swim lane.

To change the properties of a swim lane:

- Click Properties to display the current swim lane properties (as defined in the New Lane dialog box) in the Properties view. You can modify each property, as described in Creating swim lanes on page 129.
Introducing Managed Adapters

In a typical business scenario, business processes may span across multiple applications having diverse data structures, formats, and protocols. Adapters help applications exchange information and participate in distributed business processes by translating the data from one application to a format that the other application understands.

Adapters offer high flexibility to exploit fast changing technology while continuing to exchange information with legacy applications. Adapters also eliminate custom coding and provide quick, efficient access to business information across an organization.
For details, see the following topics:

- About Managed Adapters
- Managed adapter features
- Understanding the Managed Adapter framework
- Database Managed Adapter
- Email Managed Adapter
- File Managed Adapter
- FTP Managed Adapter
- JMS Managed Adapter
- OpenEdge Managed Adapter
- Web Service Managed Adapter

## About Managed Adapters

Business Process Server provide the following predefined Managed Adapters.

### Table 35: Predefined Managed Adapters

<table>
<thead>
<tr>
<th>Adapter category</th>
<th>Adapter name</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBAdapters</td>
<td>GenericDBAdapter</td>
</tr>
<tr>
<td>EmailAdapters</td>
<td>EmailAdapters</td>
</tr>
<tr>
<td>FileAdapters</td>
<td>GenericFileAdapter</td>
</tr>
<tr>
<td>FTPAdapters</td>
<td>GenericFTPAdapter</td>
</tr>
<tr>
<td>JMSAdapters</td>
<td>GenericJMSAdapter</td>
</tr>
<tr>
<td>OEAdapters</td>
<td>GenericOEAdapter</td>
</tr>
<tr>
<td>WebServiceAdapters</td>
<td>GenericWebServiceAdapter</td>
</tr>
</tbody>
</table>

The Managed Adapters in the above table are generic, "out-of-the-box" adapters that you do not have to install and are not configured for a specific task. You can, however, create an instance of a predefined Managed Adapter and use the Adapter Configurator and Map Configurator to configure and map it to your specifications, or create your own Managed Adapters with predefined configuration, using the Managed Adapter Browser in Progress Developer Studio for OpenEdge (for more information, see "Using the Managed Adapter Browser" in the OpenEdge Getting Started: Developing BPM Applications with Developer Studio) or the Map Configurator.
Managed adapter features

Managed Adapters support the following features:

• Adapter Framework
  • Configuration and mapping framework.
  • Better and extendable mapping.
  • Customizable GUI.
  • Ability to define complex mapping.
  For more information, refer to Understanding the Managed Adapter framework on page 136

• Database Adapter
  • Seamless integration with Progress Developer Studio for OpenEdge.
  • Sequential execution of multiple SQL statements.
  • Support for Business Process Portal customization.
  For more information, refer to Database Managed Adapter on page 141

• E-mail Adapter
  • Sending multipart MIME e-mail messages.
  • Generating messages that contain parts in all formats supported by the standard format plug-ins.
  • Sending messages to multiple recipients in a single workstep, while customizing the message content for each recipient.
  • Using the plug-in development kit to customize formats.
  For more information, refer to Email Managed Adapter on page 160

• File Adapter
  • Facilitates reading from and writing to a file.
  • Supported file formats - text, delimited text, and XML.
  For more information refer to File Managed Adapter on page 166

• FTP Adapter
  • Transfers files between an FTP server, a local file system.
  For more information, see FTP Managed Adapter on page 179

• JMS Adapter
  • Publishes and subscribes to JMS messages from messaging systems including MQ Series.
  • Supported modes - Send-only, Receive-only, Send / Receive.
  For more information refer to JMS Managed Adapter on page 183.
• OpenEdge Adapter
  • Integrates Business Process Server applications with Progress OpenEdge® solutions.
  • Interacts with OpenEdge procedures using BIZOE (Progress Interface Definition language) files.

For more information refer to OpenEdge Managed Adapter on page 190.

• Web Service Adapter
  • Enables you to find (or subscribe to) the services of other businesses on the Web.

Note: You can also access iWay adapters through the Web Service Adapter.

For more information, see Web Service Managed Adapter on page 203.

For more information on common business process- and adapter-related terms, see the Glossary section. For the complete list of Business Process Manager terminology, see the OpenEdge Business Process Server: Terminology Guide.

Understanding the Managed Adapter framework

Progress Developer Studio for OpenEdge and Business Process Modeler provides a group of predefined Managed Adapters that you can use to exchange information between your business processes and other popular applications. A Managed Adapter is a translating module that converts the process-specific protocol to another application-specific protocol.

When a Managed Adapter is used in a workstep, the Application Developer can define complex mapping between your process dataslots and adapter inputs/outputs in an external application. At run time, when the given workstep is executed, the Managed Adapter uses a mapper to interpret the mapping and configuration information that was defined at design time. This run-time mapper sets the adapter inputs and configuration, and maps the outputs to the appropriate output dataslots after the adapter execution.

The Configuration and Mapping Framework enables you to carry out the configuration and dataslot mapping for each Managed Adapter. The framework offers:

• Improved, extendable mapping.
• Customizable GUI for adapter configuration and dataslot mapping.
• Ability to define complex mapping during process design.

The next section describes this framework. The subsequent section, Using the Adapter Configurator and Map Configurator on page 136, provides an example in using the framework to configure and map a managed adapter.

Using the Adapter Configurator and Map Configurator

When a Managed Adapter is used in a process, you must define the mapping from input dataslots in your process to the adapter inputs in an external application and from the adapter outputs to your output dataslots. The MapConfigurator is the GUI-based component used to define this mapping.
Progress Developer Studio for OpenEdge (or Business Process Modeler) provide a default MapConfigurator that can be used for all out-of-the-box adapters. However, the MapConfigurator can also be defined separately for each adapter.

The MapConfigurator is started, when a Managed Adapter is used as the workstep performer.

In the following example, the Email Adapter is used to illustrate mapping. The following procedure assumes you have already created a preconfigured instance of this adapter.

1. Create a Managed Adapter workstep and assign Email Adapter as its performer. Double-click the workstep to open the workstep's Properties dialog box.
2. From the Properties view of this workstep, open the Configuration tab.

   You can specify other properties in the rest of tabs in the Properties view, same as that for custom adapters. For information regarding configuring properties for custom adapter workstep, refer to the "Defining Properties of Adapter Worksteps" section of the OpenEdge Getting Started: Developing BPM Applications with Developer Studio.

3. To configure the Managed Adapter, click Configure..., to open the AdapterConfigurator for the Email Managed Adapter.
Note: You can also open the Email Adapter Configurator using the new Managed Adapter Browser (available only from Progress Developer Studio for OpenEdge) functionality. In addition to configuring, you can use the Managed Adapter Browser to perform functions like creating a copy, renaming, deleting, as well as importing and exporting the configuration information. For more information, refer to the "Using the Managed Adapter Browser" section of the OpenEdge Getting Started: Developing BPM Applications with Developer Studio.

a) In the Adapter Configurator, you can enter new, or modify existing, information that are implemented only for the adapter instance. As shown in the following figure, open the Message tab and enter information in the From field and the Subject field. We leave the To field blank for now because it is mapped to a dataslot in Step 4.

Figure 71: Email adapter configurator - Message tab

b) Open the Message Body tab and enter information as shown in the following figure. Note that we use the $ symbol to create variables that are mapped to dataslots. Click Extract Variables to display the variables in the text area below.

Figure 72: Email adapter configurator - Message Body tab
c) Open the **Configuration** tab to configure the Mail Host. In this example, we use the notation 
"${oebps.smtp.host}" to tell the adapter to read the Mail Host name from the *oebps.smtp.host* property in the *oebps.conf* configuration file.

*Figure 73: Email adapter configurator - Configuration tab*

![Configuration Tab](image)

---

d) When you have finished configuring the Email Adapter, click **OK**, to open the **Dataslot Mapping** dialog box (or the Map Configurator). You can also open the Map Configurator by clicking the **Change mapping** button.
4. To modify the dataslot mapping, click **Change mapping** from the **Configuration** tab of the workstep’s **Properties** view, to open the **Dataslot Mapping** dialog box of the map configurator for the adapter. This example demonstrates how to send a message to all employees on a mailing list stored in a property file.

**Note:** The dataslot mapping dialog box are different for each adapter and each adapter configuration.

a) In the **Message** tab of the **Dataslot Mapping** dialog box, you can define the mapping between the input/output dataslots and the adapter inputs/outputs by either:

- Selecting one from the drop-down list. In the figure below, the Cc parameter is mapped to one or more managers that are in the “mgr” dataslot.

- Using the “GroupMapper” feature by entering a notation that refers to email addresses, such as an employee mailing list, that are stored in a property file. In this example, the To parameter is mapped to the notation “${mail.prop:empMailingList}”, where an employee mailing list, "empMailingList", is stored in the “mail.prop” property file. This notation instructs the adapter to get the “empMailingList” property from the “mail.prop” file and use the property value as the email destination.

b) Open the **Velocity Template** tab to map dataslots to each of the variables we used in the message body. Map the “meetingRoom” dataslot to the “roomNumber” variable to substitute the value of this dataslot for the $roomNumber field in the message text; and similarly map the “schedTime” dataslot to the “time” variable, as shown in the following figure.

c) Click **OK** to return to the **Configuration** tab of the workstep’s **Properties** view.

5. To edit the **mapping.prop** file to change the settings for mapping (for example, log level), click **Advanced properties** from the **Configuration** tab of the workstep’s **Properties** view, to open the **Advanced Adapter Properties** dialog box.

a) Edit the **mapping.prop** file displayed in the **Advanced Adapter Properties** dialog box appropriately as per your requirement.

b) Click **OK** to return to the **Configuration** tab of the workstep’s **Properties** view.

c) Click **OK** to save the changes.

The **Dataslot Mapping** dialog box lists all input and output dataslots for the workstep, and lists all the visible adapter parameters, grouped according to the config.xml. You can enter or select the value of the given dataslot or parameter. Simple syntax distinguishes constant values from values to be copied from dataslots or from adapter parameters. The following table explains the syntax in detail.
Table 36: Mapping syntax

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>@&lt;dataslot&gt;</td>
<td>The value of the corresponding adapter input comes from:</td>
</tr>
<tr>
<td></td>
<td>• The given dataslot.</td>
</tr>
<tr>
<td></td>
<td>OR</td>
</tr>
<tr>
<td></td>
<td>• The value of the output dataslot is transferred from the given adapter</td>
</tr>
<tr>
<td></td>
<td>output.</td>
</tr>
</tbody>
</table>

Usually you select the dataslot from the drop-down list, making it unnecessary to manually enter the @ symbol. Once a valid dataslot is selected, the @ symbol is not displayed, but an icon representing the dataslot type is shown.

<table>
<thead>
<tr>
<th>Begins with any other character</th>
<th>Any character other than &quot;@&quot; at the beginning of a string signals that the value is a constant. In case the constant itself begins with &quot;@&quot;, precede the &quot;@&quot; with &quot;&quot;; or if the constant begins with a &quot;&quot;, use &quot;&quot;. The &quot; and &quot;@&quot; symbols are significant only at the beginning of the string, and have no special meaning anywhere else in the string.</th>
</tr>
</thead>
</table>
| ${propertyFile: propertyName} | If an adapter input is mapped to a constant (see the previous row in this table), you can use this "GroupMapper" notation to extract values from property files on the server. The "propertyFile:" prefix is optional — if no property file is defined, "oebps.conf" are read. For example, "${oebps.home}" is substituted with the value of the "oebps.home" property from the "oebps.conf" file. If you use your own property files, make sure they are located in the server classpath (EJB Server or Portal Server, depending on where your application is running).

Custom property files may contain references to another file by including "alias=fileName" property. For example, if "file1.prop" includes "alias=file2.prop", the property values from "file2.prop" are read too. In case a property is present in both files, higher precedence are given to the value in the "file2.prop" file. |

Note: Even when the input and output have exactly same format, copying a value from an input dataslot directly to an output dataslot is not allowed.

Note that some of the adapter parameters have read-only combo boxes when:

• The parameter is output-only.
• The parameter is a constant.

When map configuration is completed, the MapConfigurator creates the input and output map files in the OpenEdge\workspace directory under <Process_Name>\maps\<Workstep_Name>.

Database Managed Adapter

The Database Adapter is one of the Managed Adapters conforming to the Adapter Configuration and Mapping Framework. For more information on this framework, refer to Understanding the Managed Adapter framework on page 136.
The Database Adapter provides:

- A Configuration GUI to specify database connections, SQL statements, variables, and parameters.
- A stand-alone mapping tool, as well as one embedded within Progress Developer Studio for OpenEdge to define mapping between dataslots and database values.
- The Database Adapter class, which connects to the specified databases, executes the specified SQL statements, and calls the Run-time Mapper to map dataslots and database values.

The Database Adapter helps application developers to quickly develop applications that access databases without additional programming efforts.

Database Adapter features

Business Process Server applications need to access a database to retrieve and save information. For this purpose, Application Developers design an adapter workstep and use JDBC programming in the adapter class. However, writing JDBC code from scratch is required every time, which can be streamlined using the Database Adapter.

The Database Adapter includes the following features:

- An easy-to-use adapter which is seamlessly integrated with the Progress Developer Studio for OpenEdge.
- A pre-built adapter class that loads information dynamically from XML and carries out the execution and mapping automatically, without any additional coding.
- Sequential execution of multiple SQL statements and connecting to multiple databases.
- The DB Adapter uses the default MapConfigurator GUI as described in Understanding the Managed Adapter framework on page 136

Working with the Database Adapter

The following section describes how to configure the Database Adapter.

Configuring the Database Adapter

The Database Adapter needs three types of information:

- Database connection parameters
- SQL statements
- Input and output variables of the adapter

The Database AdapterConfigurator GUI is designed to get the above information, and save it in the Adapter Configuration File.
Defining variables for the Database Adapter

To define variables:

1. After assigning the DBAdapter Managed Adapter as performer of a workstep, click the workstep to open its Properties view.

2. Open the Configuration tab and click the Configure button to display the Database Adapter Configurator GUI (You can alternatively use the DBAdapter.cmd|.sh script from the OEBPS_HOME/bin folder to invoke the Database Adapter Configurator GUI). The DBAdapter Configurator dialog box is displayed as shown in the following figure.

   Note: You can also open the DBAdapter Configurator using the Managed Adapter Browser dialog box (available only from Progress Developer Studio for OpenEdge) functionality from the Tools > Managed Adapters menu. In addition to configuring, you can use the Managed Adapter Browser to perform functions like creating a copy, renaming, deleting, as well as importing and exporting the configuration information. For more information, refer to the "Using the Managed Adapter Browser" section of the OpenEdge Getting Started: Developing BPM Applications with Developer Studio.

3. Click Variables to open the Variable Definitions dialog box, as shown in the following figure.
You can Add, Modify, and Remove variable definitions from this window.

4. Click **Add** to open the **Variable Editor** dialog box. You can also select a variable from the **Variable Definition** dialog box and click **Modify** to open the same box.

**Figure 76: Database Adapter Configurator - Variable Editor**

![Variable Editor dialog box]

a) Enter the Variable Name.
b) Select the **Type** and **Access** from the drop-down list.

- The supported Java data types are String, Integer, Long, Boolean, Date, Object, and List.
- The supported access types are Input, Output, Input/Output, and Local.

c) You can optionally provide a Default value and Description for the variable. For Date datatype, the supported default value formats are "dd/mm/yy", "mm/dd/yy", "dd/mm/yy hh:mm", and "mm/dd/yy hh:mm." For Boolean datatype, if the default value is empty, it is considered as "false." Any other value, except "true" (non case-sensitive), is also considered as "false."
d) Click **OK** to return to the **Variables Definitions** dialog box. The added variable is now displayed in the variable list.

5. Click **OK** to return to the **DBAdapter Configurator** dialog box.

**Defining Database connection**

You can use the DBAdapter Configurator dialog box (Figure 77 on page 145) to add, modify, and remove SQL statements. While defining or editing SQL statements, you can specify the database connection details.
To define database connection:

1. To create an SQL statement, click Add to open the SQL Statement dialog box. To modify an existing SQL statement, select an SQL statement and click Modify.

Figure 77: Database Adapter Configurator - SQL Statement - Connection tab

The SQL Statement dialog box provides four tabs: Connection, SQL Statement, Input Parameters, and Output Parameters. Initially, only the Connection tab is enabled.

2. Enter the SQL Statement Name.

3. To define Database Parameters, click Choose a Database to open the Database Definitions dialog box that lists available databases, as shown in the following figure.

Figure 78: Database Adapter Configurator - Database Definitions list

You can add, modify, and remove Database definitions.
4. To define a database definition, click **Add** to open the **Database Parameters** dialog box. To modify an existing database definition, select a Database name and click **Modify**.

**Figure 79: Database Adapter Configurator - Database Parameters**

![Database Parameters dialog box]

- a) Enter a name in the **Database Name** box.
- b) Select a database type from the **Database Type** drop-down list. Supported Database types include OpenEdge.
- c) Enter the **Login User name** and **Password**.
- d) Select the Database type to populate data in the **JNDI Name** field and in the **Driver** and **URL** fields in the Non-J2EE Datasource parameters panel. You need to modify this data with the proper database information.
- e) Click **OK** to return to the **Database Definitions** dialog box. The added database definition now displays in the definition list.

5. Select a database and click **OK** to return to the **SQL Statement** dialog box.

6. Click **Connect** to connect to the selected database.
The Status panel displays whether the connection is successful. If it is successful, the message "Successfully connected to database: <database_name> appears in the Status panel. The SQL Statement tab is also enabled so you can specify the SQL statement.

Figure 80: Database Adapter Configurator - SQL Statement - Successfully connected

7. Click the SQL Statement tab to define the SQL statements.

Defining SQL statements

In the SQL Statement tab, you can construct an SQL statement.

Note: For MS SQL Server, ensure that you make use of alias names for database functions in the SQL query to avoid runtime exception and suspension of the adapter workstep. For example, select UPPER(USER_NAME) AS USERNAME from UMUSER where USERNAME is the alias name for the database function, UPPER(USER_NAME).

To define SQL statements:

1. Choose the SQL Statement type from the following options: Select, Insert, Update, Delete, or Stored Procedure. These options are used to define an SQL Statement, and are described below:
   - Select. Enables you to select an existing record (or row) from a table to define a new SQL Statement.
   - Insert. Allows you to insert a record into a table.
   - Update. Enables you to update one or more selected records in a table.
   - Delete. Enables you to delete one or more selected records in a table.
• **Stored Procedure.** Enables you to choose a procedure from ones you have defined previously and stored, and use it as a new SQL Statement.

2. Depending on the type you choose, one or more corresponding buttons are displayed in the Clauses pane. In the example shown in the following figure, click **Select**, and the Clauses pane includes an enabled **Choose Tables** button.

Figure 81: Database Adapter Configurator - SQL Statement definition - SQL Statement tab

3. Click **Choose Tables.** The list of Available Tables is displayed in the **Choose Table(s)** dialog box.
   a) Choose the required tables. Based on the selection and the statement type, the configuration tool fills in one of the following into the statement:
      • Select from [tables]
      • Insert into [table]
      • Update [table]
      • Delete from [tables]
   b) Click **OK** to return to the **SQL Statement** tab.
4. When your SQL statement type is Select and you have chosen a table, the Choose Columns button is enabled in the Clauses pane. Click Choose Columns to display the list of available columns for the selected table.
   a) Choose the required columns for the syntax:
      
      Select [columns] from [tables]
   
   b) Click OK to return to the SQL Statement tab.

5. When your SQL statement type is Insert or Update and you select a Table, the Set Values button is enabled in the Clauses pane. Click Set Values to open the Select Columns & Set Their Values dialog box, which presents a list of Available Column.
   a) Choose the required columns, enter its Value directly, or click in the Use Variable column to open the Variable Definitions dialog box, where you can take a value from an input variable. The configuration tool fills in the following syntax:
      
      Insert into [table] (columns) (values)
      Update [table] set [column1=value1, ...]
   
   b) Click OK to return to the SQL Statement tab.

   The input parameters for the SQL statements are usually provided through input or internal variables. The column value, however, may be a constant or another SQL statement.

6. When your SQL statement type is Select, Update, or Delete, the Add Conditions button is available in the Clauses pane. Click Add Conditions, to open the Compose a Condition Clause dialog box.

   The condition clause is composed of expressions in the "column-operator-value" format.

   a) Select a column from the Database Columns drop-down list.
   b) Click Append a Column to append it to the condition clause.
   c) Select an operator from the SQL Operators drop-down list.
   d) Click Append an Operator to append it to the condition clause.
   e) When you need to use an input variable, click Append a Variable to open the Variable Definitions dialog box, where you can select and append one of the listed variables. For more information on adding variables, see Step 3.

   The SQL statement is displayed in the SQL Statement Preview pane.

   f) Click OK to return to the SQL Statement tab as shown in the left image of the above figure.

7. If the above controls are not adequate for your SQL statement, you can select the Expert Mode button to directly modify the where clause. In this case, you need to be careful with the syntax, as SQL syntax parsing and verification is not carried out in the Expert mode. For more information, see Using the Expert mode on page 156.

8. When your SQL statement type is Select, you can click Get Result MetaData to open the Query Tester dialog box where you can verify your statements.

   The upper Input Sample Values pane displays the parameters embedded in the SQL statement and the input variables to which they map.

   a) Enter actual values for parameters in the Value column. Alternately, select the Use Default Values check box to take the parameter values from input variable default values.
   b) Click Execute to test the query. The Query Result pane shows the test results.
   c) Click OK to return to the SQL Statement tab.
Note: When comparing ‘char’ database columns to variables, make sure that one of the following is true: 1) If an ‘equal to’ comparison is done and if the size in the variable is less than the size of the ‘char’ column, pad the variable value with spaces until the size of the ‘char’ column is matched. Only then does the ‘equal to’ comparison work. 2) Use a database-specific trim function to trim the value selected from the ‘char’ column before running the comparison with a variable value. This results in trimming down additional spaces from the column value before comparison.

Defining input and output variables

After successfully testing the Select statement, the Output Parameters tab is enabled to allow you to map values from ResultSet to output variables.
To define input and output variables:

1. Open the **Input Parameters** tab to display parameters embedded in the SQL statement, and the input variables to which they map.

   **Figure 82: Database Adapter Configurator - SQL Statement definition - Input Parameters tab**

   ![Input Parameters Tab](image)

2. Click the **Output Parameters** tab to display the output parameters.

   **Figure 83: Database Adapter Configurator - SQL Statement definition - Output Parameters tab**

   ![Output Parameters Tab](image)
A database query may return one single row or multiple rows.

3. Select the option **Expect Single Row** if you expect one single row. Select the option **Expect Columns as List** if you expect a list of values that you map to a List variable. Select the option **Expect Multiple Rows** if you expect multiple rows.

For multiple rows, you can only map the whole `ResultSet` to an Object type output variable. The Object dataslot in your process must be of type, `com.sun.rowset.CachedRowSetImpl`. For information regarding creating an Object dataslot, see Chapter 12, "Using Dataslots" in *OpenEdge Getting Started: Developing BPM Applications with Developer Studio*. The `CachedRowSetImpl` object can be processed by a custom adapter, or can be displayed as a table as described in the following Displaying multiple rows selection section.

For a single row, you can specify the output variable to which a column maps.

4. Click **OK** to complete the Database Adapter configuration.

To configure the installed Database Adapter, use the `DBAdapter.cmd|.sh` script and save the adapter configuration in the `OEBPS_HOME\managedadapters\DB\DB1` directory with the name `config.xml`.

**Using stored procedures**

You can execute a stored procedure that you have defined previously and stored, and use it as a new SQL Statement.
To use a stored procedure:

1. From the Configuration tab of the DBAdapter Properties dialog box, click Configure to open the DBAdapter Configurator dialog box.
   
   It displays the SQL Statements along with their types. You can add a new statement, or modify or delete selected statements. You can click Move Up and Move Down to change the order of the statements.

2. Click Add to open the SQL Statement dialog box.

3. Connect to a database and click the SQL Statement tab, which displays the available SQL statements.

Figure 84: DBAdapterConfigurator - Stored Procedures SQL Statements
4. Click **Stored Procedure** as the type, then click **Choose Procedure** to open the **Stored Procedures** dialog box, which displays the list of available stored procedures.

**Figure 85: DBAdapterConfigurator - Stored Procedures list**

![Stored Procedures dialog box](image)

5. From the list of procedures in the **Stored Procedures** dialog box, select a procedure and click **Procedure Info** to display the details about the selected procedure.

**Figure 86: DBAdapterConfigurator - Stored Procedures info**

![Procedure Info dialog box](image)
6. In the **DBAdapter Configurator** dialog box, click **Variables** to open the **Variable Definitions** dialog box. Once a stored procedure is selected, the input, output and I/O variables for that procedure get created and added to the existing list of variables. The input and output parameters of the stored procedure also get mapped to the corresponding variables.

**Figure 87: DBAdapterConfigurator - Stored Procedures variables**

![Variable Definitions dialog box]

You can add, modify or remove variables. If you change any variable, you must change the corresponding SQL statement appropriately.

7. From the **SQL Statements** dialog box, use the **Input Parameters** tab to view the defined input parameters.

8. Click the **Output Parameters** tab to view the defined output parameters.

**Figure 88: Database Adapter Configurator - Stored Procedures Output Parameters**

![Output Parameters dialog box]

**Note:** Result of a stored procedure execution is always a single row. Hence, by default, Expect Single Row option is selected, and all other options are disabled.
9. Click **OK** to return to the **DBAdapter Configurator** dialog box. Click **OK** to return to the **Properties** view.

10. Click **Change Mapping**.

**Figure 89: Database Adapter Configurator - Stored Procedures mapping**

For input parameters, only Source mapping can be defined. For output and return parameters only Target mapping can be defined. For I/O parameters, both Source as well as Target mapping can be defined. Note that this dialog box shows variables for all the stored procedures configured for a specific workstep.

**Using the Expert mode**

The Database Adapter Configurator provides the Expert mode, which enables you to enter SQL Statements directly.
To switch to Expert mode:

1. From the SQL Statement tab in the Database Adapter Configurator, click Expert mode (Figure 90 on page 157) to open the Confirm prompt.

   **Figure 90: Expert Mode Confirmation**

   ![Confirm](image)

   Once Expert mode is turned on, the SQL statement may get discarded or restored to previous state when you switch back to Wizard mode. Do you want to continue?

   Yes  No

2. Click Yes to leave the default Wizard mode and enter the Expert mode, and the following SQL Statement dialog box appears, with the Wizard Mode button now enabled.

   **Figure 91: SQL Statement in Expert mode**

   ![SQL Statement](image)

   3. You can manually modify the SQL Statement as required. You can still use variables for column values, but remember to precede the variable name with @. To leave the Expert mode, and return to the default Wizard mode, click the Wizard Mode button.

   4. When you have completed entering the statement, click Variables in the Clauses panel to open the Variable Definitions dialog box, where you can automatically define all the variables specified in the statement (if those are not already defined). You must still specify the data type for each variable. In this example, the variable "@IDds" is used and this variable is displayed in the Variable Definitions dialog box.

   5. You must now define the variable type. Select the variable, click Modify and define the data type in the Variable Editor dialog box as required.
Using Dynamic SQL statements

There are situations when the SQL statement cannot be defined completely at configuration time. In Expert mode, select the **Dynamic SQL** checkbox to switch to dynamic SQL mode, where parts of the statement (or even the whole statement) are supplied by dataslot values.

Dynamic SQL can be useful in many situations. As an example, consider a case where customer information is stored in a simple database table, as shown below:

**Table 37: Dynamic SQL statement example**

<table>
<thead>
<tr>
<th>ID (INTEGER)</th>
<th>Name (CHARACTER)</th>
<th>Phone (CHARACTER)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>George Smith</td>
<td>(111) 555-1111</td>
</tr>
<tr>
<td>101</td>
<td>Mike Green</td>
<td>(222) 555-2222</td>
</tr>
<tr>
<td>102</td>
<td>Jeff Black</td>
<td>(333) 555-3333</td>
</tr>
<tr>
<td>103</td>
<td>Dennis Gray</td>
<td>(444) 555-4444</td>
</tr>
</tbody>
</table>

Table 37 on page 158 provides a list of customer ID-numbers; the task is to use the DB Adapter to select the data about all the customers in the list.

**Note:** If you would like to try the Database Adapter with the example discussed here, instructions about creating the customer database table are given in Step 1 of Tutorial 2, Using the Database Adapter.

The SQL statement for getting the customer data from an ID-list would be:

```
SELECT * FROM CUSTOMER_INFO WHERE ID IN (100, 103, 104)
```

In the above example, 100, 103, and 104 are the customer IDs to select. Since the number of IDs in this list can vary, we cannot use prepared SQL statements (where the number of parameters is fixed), and are forced to switch to dynamically constructing the SQL.

To create a dynamic SQL statement:

1. Open the **Properties** view of a DBAdapter workstep. Open the **Configuration** tab and click **Configure**, displaying the **Database Adapter Configurator** dialog box.
2. From the **SQL Statement** tab, click **Expert mode**. Select the **Dynamic SQL** checkbox. Click **Select** as the type. In the SQL Statement panel, enter the following:

   ```
   select * FROM CUSTOMER_INFO WHERE ID IN (@IDLIST)
   ```
Note that we used the variable @IDLIST in place of the list of customer ID numbers.

Figure 92: SQL Statement tab - Dynamic SQL Statement

3. Click **Variables** to specify the type for the @IDLIST variable.

   Although an individual Customer ID is a number, a comma-separated list of IDs is not—that is the reason why we must select "CHARACTER" as the variable type.

   When you have defined the @IDLIST variable, click **OK** to return to the SQL Statement dialog box.

4. Click **Get Result MetaData** in the SQL Statement dialog box to invoke the Query Tester dialog box. Enter a list of values for the @IDLIST variable, and click in the row or press TAB to enable the values. Click **Execute** to show the list of selected customers in the QueryResult panel.

Replacing the variable values in the SQL statement is literal—meaning that you should be very careful when dynamically generating statements. For example, if the list we were providing consisted of Strings, and not numbers, the SQL syntax would require that the values are enclosed in single-quotiation marks:

'abc','def','xyz'

You must make sure that the string values you provide to the Database Adapter conform to the SQL syntax.

In the example above, we used a variable to substitute only a small part of the SQL statement—the list of ID numbers. It is also possible to replace more parts of the statement, or even to have the whole statement defined with a single variable, and to provide the complete SQL in a CHARACTER dataslot.
Additional caution must be exercised when using dynamic SQL. Ensuring that a dynamic statement is syntactically and semantically correct is each user’s responsibility.

Switching to dynamic SQL does not impact Database Adapter operations including defining the Input and Output parameters and dataslot mapping.

Email Managed Adapter

The Email Adapter enables you to generate and send e-mail messages from your business process. With the Email Adapter, you can create messages that can contain multiple MIME parts with text, HTML, XML, and other types of content, including attachments from files.

Email Adapter features

The features of the Email Adapter include:

- Sending multipart MIME e-mail messages.
- Generating messages that contain parts in all formats supported by the standard format plug-ins. For information on format plug-ins, see Using the Text Format plug-in on page 169 and other plug-in descriptions in File Managed Adapter on page 166.
- Generating messages that contain attachments from files.
- Sending messages to multiple recipients in a single workstep, while customizing the message content for each recipient.
- Using the plug-in development kit to customize formats, as described in Writing custom format plug-ins.
- Providing a custom Adapter Configurator GUI.

Working with the Email Adapter

The Email Adapter is a Managed Adapter and its usage is standardized along the same lines as the other Managed Adapters. For more information, see the adapter configuration and mapping framework described in Understanding the Managed Adapter framework on page 136.

Configuring the Email Adapter

You can assign the Email Adapter to an Adapter workstep in your process by performing the following procedures:

1. Click the Assign Participants link in the Tasks pane and then open the Adapters > Managed > EmailAdapters folder.
2. Drag the GenericEmailAdapter icon to the Content pane.

Alternatively, you can right-click the Adapter workstep and select Change Performer from the pop-up menu. Select GenericEmailAdapter from those listed in the Change Performer dialog box and click OK.
3. From the **Properties** view for the workstep, click the **Configuration** tab. This tab includes buttons for defining the adapter configuration, mapping, and specifying advanced properties.

**Figure 93: Properties dialog box for Email Adapter - Configuration tab**

4. Click **Configure** to open the **Email Adapter Configurator** dialog box.

5. Enter information in the fields displayed in the **Message** tab, as indicated in the following table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>From</td>
<td>Type the Sender e-mail address you want your messages to originate from. Make sure you enter a valid e-mail address, so you are able to receive any user replies or system messages (for instance, Delivery failure notifications). If you leave this field empty, you are prompted later to map it to a dataslot.</td>
</tr>
<tr>
<td>To</td>
<td>You can type one or more Recipient e-mail addresses, to whom you want to deliver the email message. If you enter more than one address, use commas as separators, example: &quot;<a href="mailto:user1@company.com">user1@company.com</a>, <a href="mailto:user2@company.com">user2@company.com</a>, …&quot;. If you enter more than one address here, a number of separate messages are sent—one to each of the recipients. If you leave the “To:” field empty, you are prompted later to map it to a dataslot. You can map this value to either a CHARACTER dataslot, containing one or more e-mail addresses in the format specified above, or to a LIST dataslot.</td>
</tr>
<tr>
<td>Cc / Bcc</td>
<td>You can type a list of e-mail addresses to deliver a carbon copy (Cc) or blind carbon copy (Bcc) of the message. Note that if you are sending a message to a number of To-recipient, the Cc- and Bcc-recipient receives one message per every one To-recipient. The reason for this is that the message content may be customized on a pre-recipient basis and may vary for every To-recipient.</td>
</tr>
<tr>
<td>Field</td>
<td>Information</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>Subject</td>
<td>Type the message subject here. If you leave this field empty, later you are prompted to map it to a dataslot. You can also use variables to customize your subject. Each variable can be mapped to a dataslot when the Adapter dataslot mapping is defined. For example, if you set the Subject to: Hello $customerName, an adapter input named &quot;customerName&quot; is created. This input is then be available for mapping from a dataslot. Note, however, that if you leave the Subject empty at this stage, and later map it to a dataslot, using variables in the mapped value is not allowed. In other words, if you map the subject to a CHARACTER dataslot, containing the value &quot;Hello $customerName&quot;, the variable &quot;customerName&quot; are not recognized.</td>
</tr>
<tr>
<td>Body</td>
<td>The table in the Body: field details the content of the message. Use the buttons on the right to define a list of MIME parts (see Defining the message body and configuring the format plug-ins). The content of each part are generated by a format plug-in, configured separately in the &quot;Mime Body&quot; tab. Plug-in configuration is discussed in detail in Preconfiguring the Email Adapter. To initially configure, just select the appropriate plugin, corresponding to the data format you want the MIME part to contain, and specify the appropriate mime-type in the &quot;Data Type&quot; column.</td>
</tr>
</tbody>
</table>

6. The information you enter in the **Message Body** tab varies, depending on the format plug-ins you selected in the Body text area in the **Message** tab.

7. The **Configuration** tab includes parameters specifying the mail server information. If you have already preconfigured the adapter, you do not have to modify the connection data in this tab. Refer to Preconfiguring the Email Adapter for information on how to use the **Configuration** tab and how to specify server defaults. If you need to have special configurations for the given business process you are designing, you can modify the parameters in the **Configuration** tab. The changes, however, are limited to your current business process.

**Defining mapping**

After configuring the Email Adapter, you must define the mapping of your process dataslots to adapter inputs/outputs. When you click **OK** in the Email Adapter Configurator, the **Dataslot Mapping** dialog box appears, as shown in the following figure. The number of tabs displayed in this dialog depends on the number of adapter components (or parts) that were defined during configuration. In this example, we defined three parts during in the Email Adapter Configurator, and there are three tabs—**Part 1: Velocity Template**, **Part 2: XML** and **Part 3: Attachment**—displayed in addition to the **Message** tab.
To define mapping:

1. In the **Message** tab, map the dataslot to the To parameter to provide the recipient's e-mail address. You can also map dataslots to the Cc and Bcc fields — either to a CHARACTER dataslot or LIST dataslot, or to a constant that is defined by directly typing a comma-separated list of addresses in each of the fields.

   **Figure 94: Email Adapter Dataslot Mapping - Message tab**

2. In the **Part 1: Velocity Template** tab, map the variables defined during the configuration of Velocity Template (see **Figure 94** on page 163) to relevant dataslots.

   **Figure 95: Email Adapter Dataslot Mapping - Part 1: Velocity Template tab**
3. In the **Part 2: XML** tab, map dataslots to previously defined parameters — the values from the dataslots are used to generate the XML part of the message; that is, the order form. This mapping is shown in the following figure.

**Figure 96: Email Adapter Dataslot Mapping - Part 2: XML tab**

4. In the **Part 3: Attachment** tab, map the source to the parameter. For more information, see Using the Attachment plug-in on page 165.

5. With dataslot mapping completed, click **OK** to complete the Email Adapter definition. The Email Adapter is now ready to use.
If the "Cc" or "Bcc" field is mapped to a null CHARACTER dataslot, the Email Adapter ignores it. However, you can change this default behaviour. To do so, on the Configuration tab of the Properties view, click Configure to open the Email Adapter Configurator window. In this window, click the Configuration tab. On this tab in the Extra Properties text box, by default, the following options are set to ‘true’, set these options to ‘false’.

allow.null.cc=true
allow.null.bcc=true

**Note:** If any or both of these options are set to ‘false’ the adapter is suspended if a non-initialized CHARACTER dataslot is mapped to the Cc: and/or Bcc: fields.

---

**Using the Attachment plug-in**

You can use the Attachment plug-in available in the Email Managed Adapter to attach a file to the e-mail message. The Attachment plugin allows the following mappings:

**Table 38: Mappings Supported by the Attachment plug-in**

<table>
<thead>
<tr>
<th>Mapping type</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHARACTER dataslot</td>
<td>The CHARACTER dataslot should contain the location of the file on the local (appserver) file system. The file is loaded and attached to the email message.</td>
</tr>
<tr>
<td>Constant</td>
<td>If every message has the same attachment, and the attachment is present on the local (appserver) file system, you can directly type in the map configurator dialog the fully-qualified name of the file</td>
</tr>
</tbody>
</table>

If you want to attach more than one attachment, you can use the "Attachment" plugin several times in the same message.
To send an attachment with the e-mail message:

1. Double-click an Email Adapter workstep to open its Properties view. Open the Configuration tab and click Configure to open the Email Adapter Configurator dialog box.

2. Click Add to add a new part to the message. Select Attachment from the options in the enabled cell in the Format Plugin column, as seen in the following figure.

![Email Adapter Configurator - defining the attachment plug-in](image)

3. Each plugin has a default MIME type, which is displayed in the Data Type column. The default data type for an Attachment format plug-in is a ZIP archive. If you know in advance what kind of file the attachment contains, click the cell in the Data Type column and enter the corresponding MIME type. For description of the data type used for each MIME type, refer to Table 1.

The Attachment plugin does not require any more configuration at this stage, so no sub-tab for the Attachment part appears in the Message Body tab. However, when the dataslot is mapped, you must define an adapter input (example, named "Attachment") that can be mapped to a dataslot or a constant.

**File Managed Adapter**

The File Adapter reads from and writes to a variety of file formats. It offers the benefits of:

- Support to popular file formats such as text files.
- Custom plug-in development for other file formats.
File Adapter features

The File Adapter features include:

- Supported file formats:
  - Text, for text files
  - Delimited, for delimited-text files (for example, comma-separated values)
  - XML, for extracting data from XML files or generating XML.
  - Velocity Templates, for Text, XML, HTML, PostScript and other formats, all generated with the Velocity template engine.

---

**Note:** The text files generated by the File Adapter use the 0x0A end-of-line character. Certain DOS and older Windows programs and utilities may not recognize the new line character, since they may require the 0x0D;0x0A character sequence for an end-of-line separator. Make sure you use the appropriate utility to manipulate such files. For example, if you encounter problems using the Notepad editor, use Wordpad to edit or view text files generated by the File Adapter.

- Custom formats allowed with a plug-in development kit as described in Writing custom format plug-ins.
- The File Adapter uses the default MapConfigurator GUI as described in Understanding the Managed Adapter framework on page 136

---

Working with the File Adapter

The File Adapter is a Managed Adapter and its usage is standardized along the same lines as the other Managed Adapters. For more information, see the adapter configuration and mapping framework described in Understanding the Managed Adapter framework on page 136

Configuring the File Adapter

You can configure the File Adapter in your process by performing the following procedures.

1. Click the **Assign Participants** link in the **Tasks** pane. From **Adapters > Managed > FileAdapters**, drag the **File Adapter** to the Content pane.
2. Double-click the File Adapter workstep to open its **Properties** view.
3. From the Properties view for the workstep, click the Configuration tab. This tab includes the Configure, Change mapping, and Advanced properties buttons for defining the adapter configuration and mapping.

4. Click Configure to open the File Adapter Configurator dialog box, as shown in the following figure.

Figure 98: File Adapter Configurator - Default setting

![Figure 98: File Adapter Configurator - Default setting](image)

**Note:** You can also open the File Adapter Configurator using the new Managed Adapter Browser (available only from Progress Developer Studio for OpenEdge) functionality. In addition to configuring, you can use the Managed Adapter Browser to perform functions like creating a copy, renaming, deleting, as well as importing and exporting the configuration information. For more information, refer to the "Using the Managed Adapter Browser" section of the Progress Developer Studio for OpenEdge User's Guide.

You can enter the basic configuration for the File Adapter:

- **Mode.** Defines the File Adapter behavior - whether it should Read a file, Write a file (overwriting the previous data, if any), or Append to an already existing file.

- **File Name and Path Prefix.** These fields allow you to specify the full path to the file you are going to process. You can enter values for either of them, or you can leave the fields empty. If these fields are empty, you are allowed to map any of them to a dataslot later.

**Note:** Generally, the File Name field stores just the name of the file, while the directory path is given in the Path Prefix field. This method allows you to dynamically map the file name from a process dataslot, while the directory path remains constant.

- **Data Format.** Specifies the format of the file. Options include Text (a simple text file), Velocity Template (complex text using the Velocity Template engine syntax), Delimited (a comma-separated list of values), or XML (an XML file). You can select one of the available plug-ins, including any custom Format Plug-ins you may have developed.
Note: For more information on data types supported by the Managed Adapter Framework, see Supported data types.

Once you select the plug-in, the **File Format** tab is enabled to display the corresponding plug-in GUI configurator.

Note that all plug-ins may not support all three modes - Read, Write, and Append. If you select a mode that is not acceptable for the given plug-in, you get a warning message and the mode is switched to the default plug-in mode.

The next sections deal with the configuration of each of the default plug-ins.

### Using the Text Format plug-in

The Text Format plug-in can be used for the Email Managed Adapter and the File Managed Adapter.

**To configure and map the Text Format plug-in:**

1. For a File Managed Adapter, open the **File Adapter Configurator** dialog box (refer to Figure 98 on page 168). Select the **Mode** as Write or Append, and enter the file name, preferably pointing to your temporary directory. The Text Format plug-in does not have any extra configuration and the format-specific tab (**File Format** tab) is not enabled.

   For an Email Managed Adapter, open the **Email Adapter Configurator** dialog box (refer to Configuring the Email Adapter on page 160). Click **Add** to add a new part to the message. Select **Text** from the options in the enabled cell in the Format Plugin column.

2. Click **OK**. The **Dataslot Mapping** box is displayed.

   Alternately, from the **Configuration** tab in the **Properties** view, click **Change mapping** to open the **Dataslot Mapping** dialog box, as shown in the following figure.

**Figure 99: File Adapter - Dataslot mapping**
3. In the **Source** drop-down list, select a dataslot where your text should come from. The value of this dataslot will be stored in your output file. Note that depending on the File Adapter configuration, the mapper window may look different or may contain additional fields.

4. Click **OK**. The File Adapter is now fully configured for text files.

### Using the Velocity Template plug-in

The Velocity Template plug-in can be used to generate complex text using the Velocity template engine syntax. The templates can contain loops, if-then-else blocks, and other expressions defined in the Velocity Template Language (VTL). For more information on Velocity and VTL, refer to the Velocity documentation, available at the following URL address: [http://velocity.apache.org/](http://velocity.apache.org/)

A typical use of the Velocity plug-in is to substitute variables in a template. The syntax used by Velocity for a variable is `$VAR` or `${VAR}`. For example, if you want to use the Velocity plug-in to generate a message to a user, you can define the template as follows:

```
Hello ${USER}: Your account was activated on ${DATE}.
```

You can click the **Extract Variables** button (in the **Velocity Template** tab of the **File Adapter Configurator** dialog box) to display a list of variables used in the template. If necessary, you can edit this list. Each variable will then be represented as an adapter input, available for mapping to a dataslot, or as a constant.

You can use the Velocity Template plugin for File Managed Adapters and Email Managed Adapters:

**To configure Velocity Template plug-in:**

1. For a File Managed Adapter, open the **File Adapter Configurator** dialog box (Figure 98 on page 168), select the **Mode** as Write or Append and enter the file name. Select the **Data Format** as Velocity Template. After you have selected Velocity Template, the generic **File Format** tab is enabled and renamed to a format-specific tab called **Velocity Template**.
For an Email Managed Adapter, open the **Email Adapter Configurator** dialog box (refer to Configuring the Email Adapter on page 160). Click **Add** to add a new part to the message. Select **Velocity Template** from the options in the enabled cell in the Format Plugin column. Open the **Message Body** tab.

2. Click the **Velocity Template** tab. The **File Adapter Configurator** dialog box for the Velocity Template plug-in appears:

*Figure 100: File Adapter Configurator - Velocity Template tab*
3. In the top text area, enter the text of the template. Refer to the Velocity documentation for help on VTL. If you already have a file with the prepared template, you can load it by clicking **Load Template**. An example template is shown in the following figure:

**Figure 101: File Adapter Configurator - Velocity Template tab with template**

![File Adapter Configurator - Velocity Template tab with template](image)

This template generates a file in the following format:

```
Product list for <CompanyName>:

--> <Product1>
--> <Product2>
--> ...
```

We recommend using the `${varname}` notation for the Velocity plugin. The company name and the list of products are defined in the variables `${company}` and `${products}` respectively. The `${company}` is a scalar variable, while `${products}` is a list.

4. The Velocity Template plug-in needs a list of variables, whose values will be set from dataslots. You can either enter the names of all such variables in the bottom text area, or click **Extract Variables** to build such a list automatically. In our example, clicking **Extract Variables** generates the list of variables shown below:

The Velocity plug-in configurator listed all the variables present in the template. Note, however, that the `${product}` variable is used only to iterate through the `${products}` list, so it should not receive its value from a dataslot as an adapter input. Edit the variable list, removing the `${product}` variable.
Depending on the complexity of your template and the types of variables you are using, you may need to add or remove more variables.

5. The configuration of the Velocity Template plug-in is now complete. Click OK to close the File Adapter Configurator, which also opens the Dataslot Mapping dialog box for the File Adapter workstep.

**Figure 102: File Adapter Dataslot Mapping in Velocity Template plugin**

You have one input parameter for each Velocity variable you listed. Map the "company" input to a CHARACTER dataslot, containing the company name ("@CompanyName"). Map the "products" input to a LIST dataslot, containing the list of products for the given company ("@CompanyProducts").

This process is ready to be published and executed. A sample output of the Velocity Template plug-in is shown below:

```
Product list for Fruit Co:
    --> Apples
    --> Oranges
    --> Peaches
```

You can use the Velocity Template plug-in to also generate XML, HTML, Postscript, and other files. The example above shows only a small part of the Velocity template engine functionality. For more information on Velocity and the VTL syntax, refer to the Velocity documentation.

**Using the Delimited Text plug-in**

The Delimited Text plug-in provides support for files containing a sequence of records, each one a comma-separated list of values. Although comma is the default, you may select any other string (except new line) as field separator. The record separator is always the new line character.

**Note:** The separator used in the Delimited Text plug-in must not be used as a part of the text.

**To configure Delimited text plug-in:**

1. In the File Adapter Configurator dialog box (refer to Figure 98 on page 168), select the Mode as Read, Write or Append, and enter the file name - preferably pointing to your temporary directory.

2. Select Data Format as Delimited. After you have selected Delimited, the generic File Format tab is enabled and renamed to a format-specific tab called Delimited.
3. Click the **Delimited** tab. The following figure shows the tab details.

   **Figure 103: file adapter configurator - delimited text plug-in**

   ![File Adapter Configurator](image1)

   In this tab you can enter the name and type of each field in your record.

4. Click **Add** to define a new record. The **Parameter Editor** dialog box appears, as shown in the following figure.

   **Figure 104: File Adapter Configurator - Parameter Editor**

   ![Parameter Editor](image2)

   a) Define a **Name** for the field.
   b) Define the **Type** of the value it will contain.
   c) Click **OK** to return to the configurator box as shown in **Figure 103** on page 174.

5. Continue adding more fields as appropriate.

   For example, if your file contains records in the format:
   
   `<user_id_number>;<user_name>;<address>;<phone_number>;<email>`
Then, the configurator definition will be as shown in the following figure.

**Figure 105: File Adapter Configurator - with records**

6. Click **Modify** to modify the field definition.
   Click **Remove** to delete a field.
   Click **Move Up** or **Move Down** to move the field to match the desired record structure.

7. Click **Advanced** to define a field separator other than the default ("","."). The **Advanced Parameters** dialog box is displayed.
   a) Select the **Separator String** from the drop-down list or enter your custom character/string.
      For Write or Append mode, the separator is simply a string that will be inserted between the fields. The only special characters are "space" and "tab" that are presented as \s and \t in the pull-down list, so they can be distinguished visually.
      In Read mode, the separator is not a simple string, but a regular expression (regex) that will be used to split the record into fields. You should be aware of this, especially if the separator you are using has a special regex meaning. For example, if you want the fields to be separated with ",", in Read mode you should define the separator regex as ".
      Using regular expressions to split the fields allows for more complex separators to be defined. For example, if the fields in your file are delimited with one or more blank (space or tab) characters, you can define the separator as \s+. For additional information on regular expressions syntax, check the API documentation for java.util.regex.Pattern, available online from Sun Microsystems.
   b) The **Line Number** field is enabled when you set the mode to Read. In this case, if the delimited-text file you are reading has more than one record, you can specify which record you want to read. The default is to read the first record (1). You can enter any record number you want, or select **Last** for the last record of the file.
You can also select **Mapped** indicating that the record number should come from a dataslot. In the last case, the record number field will be visible when you define the mapping for the adapter, allowing you to map the value from the dataslot, containing the record number.

c) Click **OK** to return to the **Configurator** dialog box.

8. Click **OK** to return to the **Properties** view.

9. Click **Change mapping** to open the **Dataslot Mapping** dialog box. Open the **Data** tab, as shown in the following figure.

**Figure 106: File Adapter Dataslot Mapping - Delimited Text plug-in in the Read mode**

10. For Write or Append mode, select a dataslot under **Source** where the fields in your record should come from. The value of this dataslot will be stored in your output file. For Read mode (as shown in the figure above), select a dataslot in the **Target** drop-down list to which the fields in your record should store. Note that depending on the File Adapter configuration, the mapper window may look different or may contain additional fields.

11. Click **OK**. The File Adapter is now fully configured for delimited text files.

**Using the XML Format plug-in**

The XML Format plug-in allows you to extract data from XML files, or to generate XML data from the BP Server dataslots.

You can use the XML plugin for File Managed Adapters and Email Managed Adapters.

**To configure XML format plug-in:**

1. For a File Managed Adapter, open the **File Adapter Configurator** dialog box (refer to **Figure 98** on page 168), select the **Mode** as Read or Write and enter the file name. Select the **XML** option for the Data Format. After you have selected XML, the generic **File Format** tab is enabled and renamed to a format-specific **XML** tab.
For an Email Managed Adapter, open the **Email Adapter Configurator** dialog box (refer to Configuring the Email Adapter on page 160). Click **Add** to add a new part to the message. Select **XML** from the options in the enabled cell in the Format Plugin column.

2. Open the **XML** tab.

**Figure 107: File Adapter Configurator - XML plugin**

![Image of File Adapter Configurator - XML plugin]

In this tab you can enter the Name, XPath, and Type for each field in your record. Each of the parameters defined here corresponds to a node in the XML file. The location of each node in the XML tree is specified by an XPath expression.

At this step, you need to define a template XML as explained in the section Defining the template XML.

3. Click **Template** to open the **Select XML Template** dialog box.
   a) Select the template file. If you do not want to use the pre-configured template, or if your XML template comes from a dataslot, click **Clear** to clear the file name.
   b) Click **OK** to return to the **File Adapter Configurator**.
4. Click **Add** to define a new record. The **Parameter Editor** dialog box appears, as shown in the following figure.

**Figure 108: File Adapter Configurator - XML plugin - Parameter Editor for XML file**

- Define a **Name** for the field.
- Define the **Type** of the value it will contain.
- Enter the XPath expression identifying the node directly, or click ![XPath Browser](image) in the XPath field to open the **XPath Browser** dialog box.
- Select the node. The XPath expression is displayed in the text field at the bottom. If required, you can manually edit this expression. For example, if you are looking for a particular record, one of a sequence of records, in a XML file - in this case you may specify the record number or search condition. For more information, refer to Using variables and conditions in XPath.
- Click **OK** to return to the **Parameter Editor** dialog box.
- Click **OK** to return to the **File Adapter Configurator**, as shown in the following figure.

**Figure 109: File Adapter Configurator - XML plugin - with records**

5. Click **Modify** to modify the field definition.

Click **Remove** to delete a field.

Click **Up** or **Down** to move the field to match the desired record structure.

6. Click **OK** to open the **Dataslot Mapping** dialog box, as shown in the following figure.
Alternatively, you can open the **Dataslot Mapping** dialog box by clicking **Change mapping** in the **Properties** view.

**Figure 110: File Adapter Dataslot Mapping for XML files**

7. Under **Source**, select a dataslot where the fields in your record should come from. The value of this dataslot will be stored in your output file. Note that depending on the File Adapter configuration, the mapper window may look different or may contain additional fields.

   For more complex mapping of XML files, refer to the section **Using variables and conditions in XPath**.

8. Click **OK**. The File Adapter is now fully configured for XML files.

**FTP Managed Adapter**

The FTP Managed Adapter allows you to transfer files between an FTP server and a local file system in your business processes.

**FTP Adapter features**

The FTP Adapter features include:

- Transfer of files from and to FTP servers and local file system.
- Support of FTP transfer in TEXT/BINARY mode.
- Support of Active/Passive FTP.
- Conformance of the FTP Adapter configuration and mapping to the specifications of our Managed Adapter Configuration and Mapping Framework (see **Understanding the Managed Adapter framework** on page 136)

**Working with the FTP Adapter**

The FTP Adapter is a managed adapter and its usage and framework is standardized along the same lines as the other Managed Adapters provided with the Business Process Server installation.
Configuring the FTP Adapter

You can assign and configure the FTP Adapter to an Adapter workstep in your process by performing the following procedures.

To configure the FTP adapter:

1. Click the **Assign Participants** link in the **Tasks** pane. Expand **Adapters > Managed > FTP Adapter** folder and drag the default GenericFTPAdapter (or a previously configured FTP adapter instance) from the Tasks pane to the Content pane, creating an FTPAdapter workstep.

2. Double-click the workstep to open its **Properties** view, and click the **Configuration** tab. This tab includes the **Configure** and **Change mapping** buttons for defining the adapter configuration and mapping respectively.

3. Click **Configure** to open the configurator box of the FTP Adapter Configurator.

   **Note:** You can also open the FTP Adapter Configurator using the new Managed Adapter Browser (available only from Progress Developer Studio for OpenEdge) functionality. In addition to configuring, you can use the Managed Adapter Browser to perform functions like creating a copy, renaming, deleting, as well as importing and exporting the configuration information. For more information, refer to the “Using the Managed Adapter Browser” section of the **OpenEdge Getting Started: Developing BPM Applications with Developer Studio**.

4. The FTP Adapter Configurator dialog box includes two tabs, **Source Location** and **Target Location**. Use the tabs to specify the Source and Target locations, respectively. The FTP Adapter Configurator also includes the following three file location options:

   - **Local Filesystem.** The file is loaded from (or saved to) a location on the Business Process Server’s file system. For more information, see **Accessing files on the Local FileSystem** on page 180
   - **FTP server.** The file is downloaded from (or uploaded to) a remote FTP server. For more information, see **Accessing files on a remote FTP server** on page 182.

   Any combination of source and target locations is allowed.

**Accessing files on the Local File System**

You can configure the FTP Adapter to load or save a file on the local (Business Process Server) file system.
To access files on the local filesystem:

1. Select the Local Filesystem option in the Source Location or Target Location tab.
2. You can now enter the Path Prefix and File Name. The values of those parameters will be concatenated to specify the full path to your file. For example, entering C:\temp for Path Prefix, and sample.txt in File Name will point to the C:\temp\sample.txt file. You can also leave either (or both) of these parameters empty.

Figure 111: FTP Adapter Configurator — Local FileSystem option

3. When you click OK, the Dataslot Mapping dialog box appears. If you have filled out the Path Prefix and File Name fields, the dialog box does not display the Source tab as the source location has been defined.

4. Assuming that only the Path Prefix has been entered at configuration file, the Dataslot Mapping dialog box appears.

Figure 112: FTP Adapter Configurator — Mapping dataslots for the Local FileSystem
5. As shown above, we map the dsxml dataslot to provide the name of the file. With the path prefix defined in the FTP Adapter Configurator (Figure 112 on page 181), the full path of the file would be: C:\temp\@dsxml, where @dsxml is the value of the dsxml dataslot.

6. Click OK to complete the mapping. If you have left the File Name field blank, a Warning appears that informs you to provide a File Name. If you have left both fields blank, you are warned to provide values for both missing parameters. The parameter value can be a constant (specified directly in the Map Configurator dialog), or may come from a CHARACTER dataslot.

When using the FTP Adapter to store a file on the local filesystem, the target file name is not mandatory. If you do not specify a name for the target file, the file name would be the same as the source file.

**Accessing files on a remote FTP server**

You can map an input or output to a file located on a remote FTP server.

**To access files on a remote FTP server:**

1. Select the FTP Server option in either the Source Location or Target Location tab to map the corresponding input or output to a file located on a remote FTP server.

   **Figure 113: FTP Adapter Configurator — FTP server option**

   ![FTP Adapter Configurator](image)

   2. Specify the FTP Server’s host name, optionally the port name (if different than the default, 21), the Username and Password, the complete path of the directory and the name of the file located in the remote Business Process Server. If you want to use the default FTP directory, enter "." for the Directory parameter.
3. Additionally, you can select the **ASCII Mode** check box, if you want your file to be transferred as a text document, or the **Passive** checkbox if you want to connect to the FTP server in Passive mode (for instance, in order to bypass a firewall).

4. With the configuration shown in the **Figure 113** on page 182, the **Dataslot Mapping** dialog box appears. Click **OK** to complete the mapping. If you do not supply any of the server parameters in the FTP Adapter Configurator, a prompt to map the missing values to dataslots appears.

**Figure 114: FTP Adapter Configurator — Mapping dataslots for the FTP server**

![Dataslot Mapping](image)

Again, if you do not provide a file name for a target FTP destination, the file name will be the same as the name of the source file. Note, however, that the Username, Password and Server name are required in all cases.

**JMS Managed Adapter**

The JMS Adapter enables you to use Java Message Service (JMS) to communicate asynchronously with external systems and facilitates data exchange between your business processes and external applications.
JMS Adapter features

The JMS Adapter can operate in the following modes:

- **Send-Only mode.** In this mode, the adapter (shown in the figure below as the Send workstep) sends a message to the external system.

  **Figure 115: Send-Only mode**

  ![](image)

  After the message is sent, the workstep completes and process execution continues without delay.

- **Send/Receive mode.** In this mode, the adapter (shown in Figure 116 on page 184 as the SendReceive workstep) first sends a JMS message to the external system. After the message is sent, the workstep remains activated until a response is delivered.

  **Figure 116: Send/Receive mode**

  ![](image)
When the response is finally received, the adapter extracts the necessary information from the JMS message, the workstep completes and the process execution continues.

- **Send-Only and Receive-Only mode.** This mode is always used in combination of a Send-Only workstep and a subsequent Receive-Only workstep:

  **Figure 117: Send-Only and Receive-Only workstep**

  The following procedures are typically implemented for Send-Only / Receive-Only modes.

  - First, the adapter in the Send workstep composes and sends a message to the external system.
  - After the message is sent, the workstep completes immediately and the process execution continues.
  - At some point later in the process, the workflow reaches the workstep (in this case, Receive) that contains an adapter configured in Receive-only mode. If the response from the EIS has been received at the time when this workstep is activated, the adapter extracts the relevant information from the message and the workstep completes immediately. If the response is not yet available at the time the workstep is activated, the Receive workstep will remain in Active state until the message is received.

- **Receive-Only mode.** In this mode, the adapter only receives a message from the external system.

  **Figure 118: Receive-Only mode**

  After the message is received, the workstep completes and process execution continues without delay.
Working with the JMS Adapter

The JMS Adapter is a Managed Adapter and its usage is standardized along the same lines as the other Managed Adapters. For more information, see the adapter configuration and mapping framework described in Understanding the Managed Adapter framework on page 136.

Assigning JMS Adapter as workstep performer

To assign the JMS Adapter as the performer of an Adapter workstep:

1. Click the Assign Participants link in the Tasks pane.
2. Expand Adapters > Managed > JMS Adapters.
3. Choose one of the following:
   • Drag the JMS Managed Adapter from the Tasks pane to the Content pane.
   • Right-click the Adapter workstep and select Change Performer from the pop-up menu. Select JMS Adapter from those listed in the Change Performer dialog box.

Note: As with any other Managed Adapter, there may be several preconfigured instances for the JMS Adapter, each selected by choosing different adapter name.

4. Double-click the Adapter workstep to open the Properties view, where the JMS Adapter is now listed as the performer of the workstep.

Configuring the JMS Adapter

You can use a JMS Adapter in a BP Server application in any of the three modes namely, Send-Only, Send/Receive, and Receive-Only. However, you can use a JMS Adapter in a Web application only in Send-Only mode; it cannot be used in Send/Receive mode and Receive-Only modes.

The Configuration tab in the Adapter workstep Properties view includes the following options:

• Configure
• Changemapping
• Advanced properties

To perform the JMS Adapter configuration follow the steps:

1. Click Configure, to open the JMS Adapter Configurator, shown below.

Note: You can also open the JMS Adapter Configurator using the new Managed Adapter Browser (available only from Progress Developer Studio for OpenEdge) functionality. In addition to configuring, you can use the Managed Adapter Browser to perform functions like creating a copy, renaming, deleting, as well as importing and exporting the configuration information. For more information, refer to the "Using the Managed Adapter Browser” section of the Progress Developer Studio for OpenEdge User’s Guide.
The **JMS Adapter Configurator** has three tabs:

2. In the **Configuration** tab, specify general configuration parameters for the adapter, such as connection mode for the Adapter (Send, Send/Receive, and Receive options in the Adapter Mode list), connection factory, and destination (queue) name. Depending on the selected connection mode, the tabs Outgoing Message and Incoming Message may become disabled.

![JMS Adapter Configurator - Configuration tab](image)

3. Click the **Outgoing Message** tab to define the format of the outgoing message. Currently, the JMS Adapter supports JMS Map message format, in which the message content is a set of name/value pairs. The Outgoing Message tab in turn includes two tabs:
   - **Payload.** Helps define the content of the message payload (body).
   - **Properties.** Lists message header properties that have to be set by the adapter.

![JMS Adapter Configurator - Outgoing Message tab - Payload tab](image)

4. Click the **Payload** tab to view a list of fields, each field associated with a parameter name and data type. To add a new parameter, click **Add**, to open the **Parameter Editor** dialog box (right image, Figure 120 on page 187). To modify existing parameters, select a parameter from those listed, click **Modify**, and make the required changes in the **Parameter Editor** dialog box.

**Note:** The JMS Adapter supports several numerical data types that Business Process Server does not support, and these include: Integer, Byte, Float, and Short. There is the possibility of data loss for numerical data types if a Business Process Server data type is mapped to one of the JMS Adapter data types that is not supported in Business Process Server; and if the incoming data value for the Business Process Server data is more than the boundary value of the mapped JMS Adapter data type.
5. Click the **Properties** tab to view a list of properties that have to be explicitly set by the JMS Adapter in the outgoing message’s header.

**Figure 121: JMS Adapter Configurator - Outgoing Message tab - Properties tab**

You can edit the header properties using the **Property Editor** dialog box (same as right image, **Figure 120 on page 187**).

6. Click the **Incoming Message** tab (enabled when the Adapter Mode is set to Send/Receive or Receive) to define the format of the incoming message (the response expected from the external system), as shown in the following figure.

**Figure 122: JMS Adapter Configurator - Incoming Message tab - Payload**

Currently, the JMS Adapter requires the response message to be a JMS Map message, where the message body is a set of name/value pairs. Here, the list in the **Payload** tab defines a number of parameters that have to be extracted from the message body of the response. Similarly, the **Properties** tab includes the list of message properties that have to be extracted from the header.

**Figure 123: JMS Adapter Configurator - Incoming Message tab - Properties**
Note that while any property name can be manually entered, the name can be also selected from a list of predefined properties (the Name field is an edit-enabled combo box that provides a listing of various system properties). Selecting a predefined property also automatically adjusts the property type.

**Note:** The list of property names is provided for your convenience only. Keep in mind that some message properties are read-only. The Property Editor dialog does not enforce correct property usage.

### Defining mapping with the Mapping Configurator

When the JMS Adapter configuration, inputs, and outputs are defined, the mapping to actual dataslots is performed through the Mapping Configurator. The Mapping Configurator may look differently, depending on the way you configured the JMS Adapter. Generally, for a send/receive configuration, the Mapping Configurator has an **Incoming Message** tab, an **Outgoing Message** tab, and a **Runtime Setup** tab that allows you to map such additional parameters as Correlation ID and, optionally, other parameters that had been left empty during JMS Adapter configuration.

The following three figures show the Mapping Configurator in a send/receive configuration for the JMS Adapter defined in the previous sections.

**Figure 124** on page 189 displays the content of the **Outgoing Message** mapping tab. The tab includes two tabs used to map dataslots or constant values into the outgoing message body (payload) fields and header. Note that in order to generate the message successfully, you must map all of the fields in the **Outgoing Message** tab. Otherwise, the message cannot be constructed properly and you may get an error message while executing the JMS Adapter.

**Figure 124: Mapping Configurator for Outgoing Messages - Payload and Properties tabs**

Similarly, the **Incoming Message** mapping tab (see the following figure) also includes two tabs, where data from the response payload and header is mapped back to dataslots.
The last tab—**Runtime Setup**—includes any additional run-time configuration that may be needed:

**Figure 125: Mapping Configurator for runtime setup**

![Runtime Setup Configuration](image)

One such parameter is the Correlation ID. In certain situations (as described in the next section), the Correlation ID may need to be obtained from a dataslot, or the automatically generated ID may need to be stored in a dataslot for future use. You can define this mapping in the **Runtime Setup** tab.

---

**OpenEdge Managed Adapter**

The OpenEdge Adapter enables you to integrate Business Process Server applications with OpenEdge solutions. You can use OpenEdge Adapter as a participant to perform tasks in a workstep.

The OpenEdge Adapter is one of the Managed Adapters conforming to the Adapter Configuration and Mapping Framework. For more information regarding the managed adapter framework, see [Understanding the Managed Adapter framework](#) on page 136.

The OpenEdge managed adapter workstep in your process enables you to request OpenEdge business logic to be executed on the OpenEdge application server. This allows you to link a specific business process step to the execution of specific OpenEdge business logic.

**OpenEdge Adapter features**

The OpenEdge Adapter enables you to:

- Use OpenEdge procedures with other processes in a BPM Workflow.
- Interact with OpenEdge procedures using BIZOE (Progress Interface Definition Language) files.
- Assign procedure file parameters to dataslots.
You can customize the OpenEdge adapter workstep to:

- Connect and log on to the OpenEdge application server.
- Run an external procedure in a state-free application server.
- Execute a single-run for an external procedure in a state-managed or state-free application server.
- Define a dataslot with ABL-specific data type to support "null" value.
- Map dataslots (ABL-specific and native) with parameters of type, input, output, and input/output.
- Support a return value for an external (or internal) procedure and a user-defined function.

Working with the OpenEdge Adapter

The OpenEdge adapter facilitates the use of OpenEdge procedures in your process workflow. You can use a OpenEdge procedure file or a generated BIZOE file for that procedure, in the added OpenEdge adapter workstep in your process.

Note: If you have installed Progress Developer Studio for OpenEdge, you can create a OpenEdge procedure file and generate the BIZOE file. For information regarding these operations, see the Progress Developer Studio for OpenEdge Guide.

You can configure and map the OpenEdge adapter to identify the OpenEdge application server, including the connection information, the procedure call, and mapping of the assigned dataslots with the parameters of the procedure.

Note: You can view information of the OpenEdge adapter including its configuration details, application server connection settings, and runtime properties in the spring.xml file (available after building the project). You can view this file in the Project Explorer view in the <Process_name>/maps/<OpenEdge_adapter_workstep_name> folder.

To execute the OpenEdge adapter, you need to perform the following operations:

1. Add and configure the OpenEdge adapter workstep, as discussed in Configuring the OpenEdge Adapter on page 191.
2. Define the dataslot mapping, as discussed in Defining the dataslot mapping on page 198.

Configuring the OpenEdge Adapter

You can add and configure the OpenEdge adapter workstep in your process by performing the following procedures.

To configure the OpenEdge adapter:

1. Click the Assign Participants link in the Tasks pane. Expand the Adapters > Managed > OEAdapters folder and drag the default GenericOEAdapter (or a previously configured OpenEdge adapter instance) to your Content pane, creating a OpenEdge adapter workstep.
2. Double-click the workstep to open its Properties view, and click the Configuration tab. This tab includes the Configure button and the Change mapping button for defining the adapter configuration and mapping respectively.
3. Click **Configure** to open the configurator box of the **OpenEdge Adapter Configurator**.

![OpenEdge Adapter Configurator](image)

**Note:** You can also open the OpenEdge Adapter Configurator using the Managed Adapter Browser (available only from Progress Developer Studio for OpenEdge) functionality. In addition to configuring, you can use the Managed Adapter Browser to perform functions like creating a copy, renaming, deleting, as well as importing and exporting the configuration information. For more information, refer to the "Using the Managed Adapter Browser" section of the *OpenEdge Getting Started: Developing BPM Applications with Developer Studio*.

4. You can use the **Input file** box to load a BPM invocation (BIZOE) or an R-code (*.r) file.

**Note:** You can also load an OpenEdge procedure (*.p) file if you have installed Progress Developer Studio for OpenEdge.

- To load a file from a file directory (local or remote machine), click **File System** and select the file to be loaded as the input file.
- To load a file from the project workspace, click **Workspace** and select the file available in the project workspace. This functionality is available only if you have installed Progress Developer Studio for OpenEdge with BPM modules.
- After loading a file, if you modify the specified file, click **Reload** to reload the updated file.

5. You can use the **Service Connection Details** section to set (or modify) the connection details (as described in the following table) to the OpenEdge application server.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host name</td>
<td>To specify the host machine name (local or remote) of the OpenEdge application server.</td>
</tr>
<tr>
<td>Service name/Port</td>
<td>To specify the port address or service name of the host machine.</td>
</tr>
<tr>
<td>Application service</td>
<td>Modify (if required) the name of the application service.</td>
</tr>
</tbody>
</table>
To enter the user name and password to connect to the specified OpenEdge application server. These fields are enabled only if you select the **Session-managed** option in the **Session mode** drop-down list.

**User name/Password**

This box is enabled only if you select the **Session-managed** option in the **Session mode** drop-down list. Specify a user-defined character string to be passed to the AppServer connect procedure, which is executed each time a client connects to the AppServer.

**AppServer Info**

This is a read-only box, whose value is generated dynamically on basis on the server connection details.

**AppServer URL**

Available options are **Session-free** and **Session-managed**.

**Session model**

Note: For information regarding operating models and session model types in OpenEdge application server, see *OpenEdge Application Server: Developing AppServer applications*.

**Direct connect**

Select this checkbox if the `-H` and `-S` parameters are interpreted as the network address and TCP/IP port number of the AppServer connection.

6. Click **Runtime Properties** to open the **Runtime Properties** dialog box, which display the runtime properties with their default values. The available properties vary depending on the session model selected in the **Session model** drop-down list. Modify the default values, if required. For information regarding the runtime properties, see **OpenEdge runtime properties** on page 193.

7. In the **Procedure Details** section, you can view the contents of the selected input file. The **Show members without annotations** and the **Generate annotations** checkboxes are enabled only if you have selected a procedure (.p) file as the input file.

   - Select the **Show members without annotations** checkbox to list all the procedures and functions without annotations in the right pane of the **Procedure Details** section. If the main procedure file contains internal procedures and functions, the left pane of the **Procedure Details** section lists these elements in a tree structure.
   - Select the **Generate annotations** checkbox to generate annotations for a procedure or function in the right pane of the **Procedure Details** section. The procedure or function files are automatically updated with the annotations. If you generate annotations for a procedure or a function that is open in the Form Editor, the annotations are reflected when you save the open procedure or function file. If you select the **Show members without annotations** checkbox, the **Generate annotations** checkbox is also selected.

8. Click **OK** to save the save your changes in **OpenEdge Adapter Configurator** dialog box. The **Dataslot Mapping** dialog box appears.

**OpenEdge runtime properties**

Table 39 on page 194 details the OpenEdge runtime properties including its name, label, default value, and description.
## Table 39: OpenEdge Runtime properties

<table>
<thead>
<tr>
<th>Property label</th>
<th>Property name</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stale object timeout</td>
<td>stale04GLOObject Timeout</td>
<td>0</td>
<td>The maximum duration (in seconds) that a service object (AppObject, SubAppObject, or ProcObject) can be idle before it is released. As part of managing certain service objects with OpenEdge, clients explicitly create them using factory methods before invoking other methods on them. When the client no longer requires the object, it has the responsibility to release the object from the service runtime context. However, if this time-out expires before the client releases the object, the OpenEdge Adapter assumes that the client application no longer requires access to the object, and deletes it from the service run-time context automatically. In effect, the adapter uses this time-out to provide garbage collection on service objects that client applications stop referencing and fail to release in the specified period of time. Any subsequent attempt by a client to access this object returns an error from the adapter.</td>
</tr>
</tbody>
</table>
| Request wait timeout | requestWait Timeout | -1 | Determines how the adapter handles requests when the service connection pool is full. The connection pool is full when the number of active sessions is equal to the value of the maxSessions property and all sessions are currently running requests.  
  - For value equal to -1, the OpenEdge adapter queues the request indefinitely till an AppServer session is available.  
  - For value equal to 0, the OpenEdge adapter rejects the request and returns an error message to the client indicating that there are too many concurrent requests.  
  - For value greater than 0, the OpenEdge adapter queues the request for the maximum number of seconds specified by the value till an AppServer session is available. If no session is made available in that time, the adapter returns an error to the client. |
<p>| NameServer client min port | nsClientMinPort | 0 | The minimum value for the adapter to specify for the UDP port number used to communicate with the NameServer. The value must be less than or equal to the value of the nsClientMaxPort property. For value equal to 0, the OpenEdge adapter chooses the NameServer client port number randomly. This property applies only to services that use a NameServer to access application services (an AppServer). |</p>
<table>
<thead>
<tr>
<th>Property label</th>
<th>Property name</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NameServer client max port</td>
<td>nsClientMaxPort</td>
<td>0</td>
<td>The maximum value for the adapter to specify for the UDP port number used to communicate with the NameServer. The value must be less than or equal to the value of the nsClientMaxPort property. For value equal to 0, the OpenEdge adapter chooses the NameServer client port number randomly. This property applies only to services that use a NameServer to access application services (an AppServer).</td>
</tr>
<tr>
<td>NameServer client port retry</td>
<td>nsClientPortRetry</td>
<td>3</td>
<td>The maximum number of requests that the adapter makes for a valid local UDP port number when attempting to communicate with the NameServer. This property applies only to services that use a NameServer to access application services (an AppServer).</td>
</tr>
<tr>
<td>NameServer client port retry interval</td>
<td>nsClientPortRetry Interval</td>
<td>200</td>
<td>The interval (in milliseconds) that the adapters waits between requests to get a valid UDP port number when attempting to communicate with the NameServer. This property applies only to services that use a NameServer to access application services (an AppServer).</td>
</tr>
<tr>
<td>Service logging level</td>
<td>serviceLogging Level</td>
<td>2</td>
<td>The amount and type of information written by the adapter to the service log for each log entry. Logging levels range from 1 to 4.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• For logging level equal to 1, the OpenEdge adapter logs the errors internally detected by the adapter only.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• For logging level equal to 2, the OpenEdge adapter logs additional errors that result in SOAP faults returned to the client.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• For logging level equal to 3, the OpenEdge adapter logs additional debugging information useful to the publisher.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• For logging level equal to 4, the OpenEdge adapter logs additional debugging information useful for Progress support services and engineering.</td>
</tr>
<tr>
<td>Service logging entry types</td>
<td>serviceLogging EntryTypes</td>
<td>-</td>
<td>Specifies the type of information recorded when performing diagnostic logging. The value is a comma-separated list of valid entry types. It is also possible to specify a different logging level for each entry type. For example: &quot;ENTRY:2&quot;, where the number following the colon is the level. Levels for types must be greater than 1.</td>
</tr>
<tr>
<td>Service fault level</td>
<td>serviceFaultLevel</td>
<td>2</td>
<td>The amount (level) of information returned to the client for a SOAP Fault as determined by an integer value. A level of 2 returns basic information in the &lt;FaultCode&gt; and &lt;FaultString&gt; elements for each SOAP Fault message, which is suitable for normal production environments. A level of 3 returns more detailed information that is suitable for development environments. Other values provide varying levels of diagnostic information, and are reserved for use by Progress Technical Support and Engineering.</td>
</tr>
<tr>
<td>Property label</td>
<td>Property name</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Wait if service is busy</td>
<td>waitIfBusy</td>
<td>0</td>
<td>An integer value that determines how to handle client requests to a service that is busy processing a prior request. If the value is 1, the OpenEdge Adapter queues multiple requests for this service and executes them one at a time until the queue is empty. If the value is 0 and the adapter is executing a prior request for the service, each subsequent request for the same service fails until the adapter completes the request it is currently executing.</td>
</tr>
<tr>
<td>Connection lifetime</td>
<td>connectionLifetime</td>
<td>0</td>
<td>The maximum lifetime (in seconds) of AppServer connections in the connection pool for this service. If the value is 0, then the lifetime of these connections is unlimited, unless they are disconnected according to the requirements of other property settings, such as idleSessionTimeout. If the value is greater than 0, then the OpenEdge adapter maintains any AppServer connections for this service for the specified number of seconds. Thus, when the idleSessionTimeout interval expires, the adapter trims the connections in the connection pool beginning with those whose connectionLifetime interval has expired. However, the adapter maintains the connections for all services whose connectionLifetime interval has not yet expired regardless of other property settings.</td>
</tr>
<tr>
<td>No host verification</td>
<td>noHostVerify</td>
<td>0</td>
<td>If the value is set to 1, turns off host verification for an SSL Web service connection (specified by the appServiceProtocol property). If cleared, the adapter compares the host name of the connecting AppServer with the Common Name specified in the server digital certificate, and raises a Web service error if they do not match. With this parameter specified, the Web service never raises the error.</td>
</tr>
<tr>
<td>No session reuse</td>
<td>noSessionReuse</td>
<td>0</td>
<td>If set to 1, the Web service connection does not reuse the SSL session ID when reconnecting to the same AppServer for an SSL Web service connection (specified by the appServiceProtocol property).</td>
</tr>
<tr>
<td>AppServer keep alive</td>
<td>appServer</td>
<td>denyClient</td>
<td>Indicates if the client would employ the AppServer Keepalive protocol on this connection, if supported and enabled by the AppServer. To enable the protocol, specify the value as allowServerASK. To disable the protocol, specify the value as denyServerASK. The absence of this property indicates that the default value for the ServerASK protocol is used on this connection.</td>
</tr>
<tr>
<td></td>
<td>KeepAlive</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Client ASK activity timeout</td>
<td>clientASKActivity</td>
<td>60</td>
<td>Determines the time interval the AppServer takes active steps to determine if the client is still connected. The ClientASK protocol denotes that the direction of the keepalive messages is from the client to the servers.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property label</th>
<th>Property name</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client ASK response timeout</td>
<td>clientASKResponse Timeout</td>
<td>60</td>
<td>Determines the time interval the AppServer can receive messages from the server. The ClientASK protocol denotes that the direction of the keepalive messages is from the client to the servers.</td>
</tr>
</tbody>
</table>

**Properties available only for "Session-free" session model**

<table>
<thead>
<tr>
<th>Property label</th>
<th>Property name</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum number of sessions</td>
<td>minSessions</td>
<td>1</td>
<td>The minimum number of connected sessions allowed in the connection pool. The OpenEdge adapter attempts to keep at least this many sessions connected to the application service (AppServer).</td>
</tr>
<tr>
<td>Maximum number of sessions</td>
<td>maxSessions</td>
<td>0</td>
<td>The maximum number of connected sessions allowed in the service connection pool. Once the number of sessions in the pool reaches this limit, the OpenEdge adapter creates no additional sessions for this service, and handles all requests for this service according to the requestWaitTimeout property setting. A value of 0 indicates that the size of the connection pool is unlimited.</td>
</tr>
<tr>
<td>Initial number of sessions</td>
<td>initialSessions</td>
<td>1</td>
<td>The number of network sessions to be created (and shared by all clients) when the service connection pool is initialized by the adapter. This value must be between the value of the minSessions property and the maxSessions property, inclusive, unless maxSessions is set to 0. If maxSessions is set to 0, the initialSessions value must only be greater than or equal to minSessions.</td>
</tr>
<tr>
<td>Idle session timeout</td>
<td>idleSessionTimeout</td>
<td>0</td>
<td>The duration (in seconds) between attempts by the adapter to shut down extra network connections to the AppServer, based on the client demand. The adapter monitors the maximum number of sessions needed since the last time-out, then disconnects any connections in excess of that number. A value of 0 indicates that the adapter will never disconnect idle sessions unless the connectionLifetime interval has expired.</td>
</tr>
<tr>
<td>NameServer client pick list size</td>
<td>nsClientPicklistSize</td>
<td>8</td>
<td>The number of available AppServer options (the broker pick list) that the adapter requests from the NameServer each time it looks up a given application service name. This property applies only to services that use a NameServer to access application services (an AppServer).</td>
</tr>
<tr>
<td>Property label</td>
<td>Property name</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>--------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>NameServer client pick list expiration</td>
<td>nsClientPicklistExpiration</td>
<td>300</td>
<td>The maximum duration (in seconds) that the adapter retains a list of AppServer options (pick list) for an idle application service. A value of 0 indicates that the pick list never expires. This property applies only to services that use a NameServer to access application services (an AppServer).</td>
</tr>
<tr>
<td>Minimum idle connections</td>
<td>minIdleConnections</td>
<td>0</td>
<td>If set to 1, turns off host verification for an SSL Web service connection (specified by the appServiceProtocol property). If cleared, the adapter compares the host name of the connecting AppServer with the Common Name specified in the server digital certificate, and raises a Web service error if they do not match. With this parameter specified, the Web service never raises the error.</td>
</tr>
</tbody>
</table>

### Defining the dataslot mapping

After configuring the OpenEdge Adapter workstep, you must define the mapping of your process dataslots (source and target) to the parameters in the selected OpenEdge file. The **Dataslot Mapping** dialog box (Figure 127 on page 198) displays the parameters depending on the input file selected in the **OpenEdge Adapter Configurator** dialog box (Figure 126 on page 192).

**Figure 127: OpenEdge Adapter Dataslot Mapping**

To define the dataslot mapping for an OpenEdge adapter:

1. After adding and configuring the OpenEdge adapter workstep, double-click the workstep to open its **Properties** view.
2. From the **Configuration** tab, click **Change mapping** to open the **Dataslot Mapping** dialog box, as shown in (left image, Figure 127 on page 198).
3. In the **Source** drop-down list in the **Type Mapping** tab, select the dataslot you want to associate with the parameter listed in the **Parameter** column. The value of this parameter is set from the value of the dataslot selected as the source.
**Note:** For information regarding the supported dataslot types (native or ABL) that can be mapped to each ABL parameter type, see Supported ABL dataslots for OpenEdge parameters on page 199.

4. In the **Target** drop-down list in the **Type Mapping** tab, select the dataslot you want to associate with the parameter listed in the **Parameter** column. The value of this parameter is stored in the dataslot selected as the target.

5. The **Advanced** tab (right image, Figure 127 on page 198) is available only if you select the **Session-managed** option for session model in the **OpenEdge Adapter Configurator** dialog box. You can use this tab to specify dataslot values for AppServer parameters, namely, Username, Password, AppServer Info, and Return Value, which are used to connect to the AppServer.
   a) In the **Source** drop-down list in the **Advanced** tab, select the dataslots you want to associate with the Username, Password, and AppServer Info parameters.
   b) In the **Target** drop-down list in the **Advanced** tab, select the dataslot you want to associate with the Return Value parameter.

   At runtime, the value of the dataslot that you specify in **Advanced** tab replaces the character string value that you specify for the corresponding field in the OpenEdge adapter configuration dialog box.

6. Click **OK** to complete the dataslot mapping for the OpenEdge adapter workstep.

**Supported ABL dataslots for OpenEdge parameters**

You can use the **Dataslot Mapping** dialog box (Figure 127 on page 198) to map the ABL dataslot type as source, target, or both to the OpenEdge parameter types.

**Table 40** on page 200 details the data conversion matrix between source dataslot types (native or ABL) and its supported OpenEdge parameter types used in ABL procedures.
Table 40: Conversion Matrix of source dataslot type to OpenEdge parameter type

<table>
<thead>
<tr>
<th>OpenEdge parameter type</th>
<th>Supported dataslot types</th>
<th>Level of support</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHARACTER</td>
<td>CHARACTER</td>
<td>Fully supported. Supported for a maximum of 30000 characters.</td>
</tr>
<tr>
<td></td>
<td>INTEGER</td>
<td>Supported with string value of the dataslot passed to CHARACTER.</td>
</tr>
<tr>
<td></td>
<td>INT64</td>
<td>Supported with string value of the dataslot passed to CHARACTER.</td>
</tr>
<tr>
<td></td>
<td>DECIMAL</td>
<td>Supported with string value of the dataslot passed to CHARACTER.</td>
</tr>
<tr>
<td></td>
<td>DATETIMETZ</td>
<td>Supported with string value of the dataslot passed to CHARACTER.</td>
</tr>
<tr>
<td></td>
<td>LOGICAL</td>
<td>Supported with string value of the dataslot passed to CHARACTER.</td>
</tr>
<tr>
<td>CHARACTER LIST</td>
<td>LIST</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>LONGCHAR</td>
<td>CHARACTER</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>DATE</td>
<td>DATETIMETZ</td>
<td>Supported for passing the date part.</td>
</tr>
<tr>
<td>DATETIME</td>
<td>DATETIMETZ</td>
<td>Supported for passing the date and time part.</td>
</tr>
<tr>
<td>DATETIME-TZ</td>
<td>DATETIMETZ</td>
<td>Fully supported.</td>
</tr>
</tbody>
</table>

Supports the following formats to map a constant value with a DATETIME-TZ parameter in the OpenEdge Adapter Dataslot Mapping dialog:

- "yyyy-MM-dd'T'HH:mm:ss.SSS(+/-)zz:zz" or "yyyy-MM-dd HH:mm:ss.SSS(+/-)zz:zz"
- "yyyy-MM-dd'T'HH:mm:ss.SSS" or "yyyy-MM-dd HH:mm:ss.SSS"
- "yyyy-MM-dd'T'HH:mm:ss" or "yyyy-MM-dd HH:mm:ss"
- "yyyy-MM-dd"

For example, 1995-01-17T12:12:12.234+05:30 or 1995-01-17 12:12:12.234+05:30.

Note: The list of supported date formats specified above is applicable to these three date parameters: DATE, DATETIME, and DATETIME-TZ.

---

2 The UNKNOWN value is not supported for some dataslot types.
<table>
<thead>
<tr>
<th>OpenEdge parameter type</th>
<th>Supported dataslot types</th>
<th>Level of support</th>
</tr>
</thead>
<tbody>
<tr>
<td>DECIMAL</td>
<td>DECIMAL</td>
<td>Supported for numbers with a maximum of 50 digits with up to 10 digits after the decimal point. ²</td>
</tr>
<tr>
<td></td>
<td>DECIMAL</td>
<td>Fully supported.</td>
</tr>
<tr>
<td></td>
<td>INT64</td>
<td>Fully supported.</td>
</tr>
<tr>
<td></td>
<td>INTEGER</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>HANDLE</td>
<td>Handle</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>INT64</td>
<td>INT64</td>
<td>Fully supported.</td>
</tr>
<tr>
<td></td>
<td>INTEGER</td>
<td>Fully supported. ²</td>
</tr>
<tr>
<td>INTEGER</td>
<td>INTEGER</td>
<td>Fully supported. ²</td>
</tr>
<tr>
<td>LOGICAL</td>
<td>LOGICAL</td>
<td>Fully supported. ²</td>
</tr>
<tr>
<td>MEMPTR</td>
<td>Memptr</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>RAW</td>
<td>Raw</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>ROWID</td>
<td>Rowid</td>
<td>Fully supported.</td>
</tr>
</tbody>
</table>

Table 41 on page 202 details the data conversion matrix between OpenEdge parameter types used in ABL procedures and the supported dataslot types (native or ABL) specified as target.
## Table 41: Conversion Matrix of OpenEdge parameter type to target dataslot types

<table>
<thead>
<tr>
<th>OpenEdge parameter type</th>
<th>Supported dataslot types</th>
<th>Level of support</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHARACTER</td>
<td>CHARACTER</td>
<td>Fully supported. Supported for a maximum of 30000 characters.³</td>
</tr>
<tr>
<td></td>
<td>INTEGER</td>
<td>Supported with conversion into numeric format.</td>
</tr>
<tr>
<td></td>
<td>INT64</td>
<td>Supported with conversion into numeric format.</td>
</tr>
<tr>
<td></td>
<td>DECIMAL</td>
<td>Supported with conversion into numeric format.</td>
</tr>
<tr>
<td></td>
<td>DATETIMETZ</td>
<td>Supported with conversion into numeric format.</td>
</tr>
<tr>
<td></td>
<td>LOGICAL</td>
<td>Supported with conversion into numeric format.</td>
</tr>
<tr>
<td>CHARACTER LIST</td>
<td>LIST</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>LONGCHAR</td>
<td>CHARACTER</td>
<td>Supported for a maximum of 256 characters for a STRING of CHAR type and 2000 characters for a STRING of VARCHAR type.³</td>
</tr>
<tr>
<td>DATE</td>
<td>DATETIMETZ</td>
<td>Fully supported.³</td>
</tr>
<tr>
<td>DATETIME</td>
<td>DATETIMETZ</td>
<td>Fully supported.³</td>
</tr>
<tr>
<td>DATETIME-TZ</td>
<td>DATETIMETZ</td>
<td>Supported for passing the date and the time part.⁵</td>
</tr>
<tr>
<td>DECIMAL</td>
<td>DECIMAL</td>
<td>Supported for numbers with a maximum of 50 digits with up to 10 digits after the decimal point.³</td>
</tr>
<tr>
<td>HANDLE</td>
<td>Handle</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>DECIMAL</td>
<td>DECIMAL</td>
<td>Fully supported.³</td>
</tr>
<tr>
<td>INT64</td>
<td>INT64</td>
<td>Fully supported.</td>
</tr>
<tr>
<td></td>
<td>DECIMAL</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>INTEGER</td>
<td>INTEGER</td>
<td>Fully supported.³</td>
</tr>
<tr>
<td></td>
<td>DECIMAL</td>
<td>Fully supported.¹</td>
</tr>
<tr>
<td></td>
<td>INT64</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>LOGICAL</td>
<td>LOGICAL</td>
<td>Fully supported.³</td>
</tr>
<tr>
<td>MEMPTR</td>
<td>Memptr</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>RAW</td>
<td>Raw</td>
<td>Fully supported.</td>
</tr>
<tr>
<td>ROWID</td>
<td>Rowid</td>
<td>Fully supported.</td>
</tr>
</tbody>
</table>

³ The UNKNOWN value is not supported for some dataslot types.
Web Service Managed Adapter

The Web Service Managed Adapter enables businesses to provide (or publish) their services on the Web and to find (or subscribe to) services of other businesses on the Web. Businesses using Web Services can act as either a service provider that, for example, publishes one of your business applications as a Web service, or as a service requester that helps you to locate required Web services.

Introducing Web Service Adapter

The Web Service Adapter allows you to invoke Web services. The adapter can be configured and used in an entirely point-and-click fashion, without having to write any program code.

Figure 128: Web Service Adapter example

A company has automated its order shipping process. Figure 128 on page 203 displays a portion of the "Shipping" application. After an inventory check, the address of delivery must be validated before the order is packaged and shipped. Another company provides an address verification Web Service.

The "CheckAddress" workstep is handled by the Web Service managed adapter, who generates the SOAP request, sends it to the postal address verification service and receives the response, containing the address status and validated address.

Note: In previous releases, the integration with adapters from the iWay Software Universal Adapter Suite was done through an iWay Managed Adapter that we provided. Currently, iWay Software provides a Web Service layer for interaction with their adapter suites. As a result, we recommend that you access iWay Adapters through the Web Service Managed Adapter. For more information about creating iWay adapters through Web Services, refer to the iWay Software documentation at: http://www.iwaysoftware.com/products/webservicesadapters.

Configuring the Web Service Adapter

To configure Web Service Adapter in your process, follow the steps:

1. Open the process template diagram for which you have to add the Web service adapter.
2. From the Tasks pane, click the Assign Participants link.
3. Expand Adapters > Managed > Web ServiceAdapters.
4. Drag the adapter icon to the Content pane. It is displayed as a Web Services Adapter workstep.
5. Double-click the workstep to open its **Properties** view.
6. From the Configuration tab, click Configure. The Web Service Adapter Configurator appears (Figure 129 on page 204).

**Figure 129: Web Service Adapter Configurator**

The configurator dialog has two main areas: Web Service Configuration and WSDL Detail.

In the Web Service Configuration area, you can enter the URL of the WSDL file, containing the Web Service definition. For WSDL URL, you can paste the URL or type it directly. You can also use the UDDI Browser to search and select a service, by clicking Search UDDI. The use of the UDDI Browser is covered in Using the UDDI Browser on page 205 below.

**Note:** You can also open the WebService Adapter Configurator using the new Managed Adapter Browser (available only from Progress Developer Studio for OpenEdge) functionality. In addition to configuring, you can use the Managed Adapter Browser to perform functions like creating a copy, renaming, deleting, as well as importing and exporting the configuration information. For more information, refer to the "Using the Managed Adapter Browser" section the OpenEdge Getting Started: Developing BPM Applications with Developer Studio.
If obtaining the WSDL requires authentication or you want to define other security settings, expand the Security Settings panel. For more information on defining security settings, see Defining the security settings on page 206.

Using the UDDI Browser

Use the default UDDI Browser to locate a valid WSDL file in a UDDI registry.

To open the UDDI Browser and locate a WSDL file in a UDDI registry, follow the steps:

1. Click Search UDDI in the Web Service Adapter Configurator dialog box. The UDDI Browser dialog appears.
2. Specify your search criteria.
3. Select a UDDI registry from those listed in the UDDI Registry URL drop-down list.

Figure 130: UDDI Browser

4. Enter criteria in the Search For box. If you enter * or %, the UDDI Browser finds all available services.
5. Select ways to sort your results in the Sort results by drop-down list. Options in this list include the following:
   • Alphabetical (Ascending)
   • Alphabetical (Descending)
   • Date (Ascending)
   • Date (Descending)

Note: If either of the date options is selected in the above list, the search result sorts the list of services on basis of the published date.

6. Select the maximum number of results you seek from the drop-down list.
7. Select the **Exact match results only** checkbox to obtain only exact matches to your search criteria.

8. Select the **Case sensitive search** checkbox to search for specified case sensitive criteria.

9. Click **Find**. The Messages dialog box appears and states the number of services that match your search criteria.

10. Click **OK** to view matching WSDL files in the **Search Results** area on the right side of the UDDI Browser, as shown in Figure 130 on page 205.

11. Valid WSDL files are indicated by an enabled font; that is, the file name is not greyed out. Select a valid WSDL file and information about it is displayed below.

12. Click **Select**. The URL address of the WSDL file appears in the **WSDL**: box in the Web Service Adapter Configurator dialog box.

### Defining the security settings

The Web Service adapter uses Apache WSS4J to provide support for Web Service security specifications that include: UserName Tokens, Timestamps, SAML Tokens, Digital signatures, and Message encryption.

Expand the Security Settings panel to display all the Security Settings features. By default, this panel is open, and the User Name and Password boxes are blank. However, if the Web service you are invoking requires your authentication to access the service or for run-time http authentication, enter the User Name and Password to your Web server in the WSDL HTTP Authentication section or the Runtime HTTP Authentication section, as shown in the following figure.

**Figure 131: Web Service Adapter Configurator - Security Settings Panel**
You can also choose one of the following options for run-time http authentication:

- **UserName Token**: You must provide a username.wsdd file that is available in the system classpath.

- **SAML Token**: SAML, or Security Assertion Markup Language, is a XML-based framework that ensures communications transmitted over Web Services are secure. You must provide a SAML file with a *.wsdd extension that is available in the system classpath.

If the Web Service you invoke expects SAML Tokens, provide the following client WSSD (Web Service Deployment Descriptor), which meets Apache WSS4J requirements. This client-deploy.wsdd file has the following format:

```xml
<service name="STPing">
  <requestFlow>
    <handler type="java:org.apache.ws.axis.security.WSdoAllSender">
      <parameter name="action" value="Timestamp SAMLTokenUnsigned"/>
      <parameter name="samlPropFile" value="/keys/saml.properties"/>
    </handler>
  </requestFlow>
</service>
```

The name of the client-deploy.wsdd file must be unique across the Business Process Server installation and available in the application server classpath. A OEBPS_HOME\WebService\config folder is added to the system classpath of the Portal and EJB servers. All *.wsdd files are maintained in this folder.

- **Other**: For advanced users who want to make use of SAML, digital signatures, and encryption. You must use an actual file name with a *.wsdd extension that is compliant with wss4j requirements. This *.wsdd file must be available in the system classpath.

### Supporting passing of session tokens to Web Services

If a Web Service is protected by SiteMinder or a similar type of security and requires you to pass cookie information in addition to the security measures of entering your user name and password, the Web Service Adapter provides session token support. By default, the session support is disabled and hidden.

**Follow the steps to enable session token support, as follows:**

1. Configure the Web Service Managed Adapter workstep (without session token), and click **OK** to close the Map Configurator.
2. Save the project.
3. From the Window menu, select **Show View > Other > Navigator**. The Navigator pane appears on the left.
4. Right-click the project node in the Navigator pane and select **Build Project**. The maps directory appears in the Navigator pane. You may have to refresh the Navigator pane to see the directory.
5. From the **maps** directory, select the workstep for which you want to enable session cookies and open the config.xml file.
6. Change the value of the "SESSION_TOKEN_SUPPORT" parameter to "true", as shown in bold in the following code sample:

```xml
<configform synchronous="true" version="1.2">
  <block title="Inputs" description="Inputs" hidden="false">
    <param name="SESSION_TOKEN_SUPPORT" access="enable" description="SESSION_TOKEN_SUPPORT" type="java.lang.String" value="true"/>
  </block>
</configform>
```
7. **Save and close** config.xml.
Setting session token cookies from dataslot

Follow the steps to set session token cookie from dataslot as follows:

1. Open the Adapter Configurator again. (After performing the preceding procedure.)

   When the session support is enabled, The Set session token cookie from dataslot checkbox will appear in the Advanced tab of the Web Service Adapter Configurator.

2. Select the Set session token cookie from dataslot check box if your Web services use SiteMinder to provide authentication. If you select this checkbox, an additional input (named "session_token") will be provided for the Web service.

3. Map this input to a dataslot that contains the SiteMinder session token.

Reviewing information on WSDL details

After you enter the WSDL Location, either manually or using the UDDI Browser, in the WSDL: box, click Go to load the WSDL and display information about the services in the "WSDL Detail" area.

In order to illustrate the operation of the adapter configurator, we use the sample "MathService" service, available at the following URL:


Follow the steps to understand a sample service:

1. Enter the MathService URL in the WSDL field.
2. Click Go. The relevant Web Service information is displayed.

   Figure 132: Web Service Adapter Configurator with Web Service information
As shown in the left panel, MathService has the "Add" arithmetic operations.

3. Click the Add tree node to update the Information tab to show relevant information.

**Figure 133: Web Service Adapter Configurator with Web Service Operation information**

Part of the information, displayed when an operation is selected, is the list of operations inputs and outputs, as well as the data type, associated with each input or output. Once the adapter is configured, those parameters become inputs and outputs to the adapter and can be mapped to dataslots.

**Using the Web Services Adapter Configurator dialog box**

The adapter configurator provide three other tabs in addition to the Information tab. The Types and Mapping tab is used to define mapping for complex and/or custom data types, and is described in *Dealing with complex data types* on page 210. The Advanced tab is used to define timeout parameters (see *Using the Advanced tab* on page 210). The Test tab, where a configuration can be tested actually invoking the service, is described in the following section.

Open the Test tab to display the following screen.

**Figure 134: Web Service Adapter Configurator - Test tab**
Endpoint field displays the URL as in WSDL. Here you can also optionally specify an alternative target endpoint URL for testing. This alternative URL is used only during adapter configuration; it does not have any effect when the adapter is invoked at runtime.

The Test tab also lists all the inputs and SOAP Response outputs, and lets you specify values for the input parameters. It also displays a box where for the "Add" operation of MathService, there are two inputs, "num1" and "num2", both integer numbers. The single output is named "AddResult", and also is an integer. Note that you can also enter Array type inputs, in the format java.lang.Integer[2; 3; 4].

**Verifying the Add web service**

**Follow the steps to verify the Add service example:**

1. Click Reset to clear the values in the Test tab.
2. In the Value column, enter numerical values for the 'num1' and 'num2' inputs.
3. Click Invoke to invoke the operation and it passes the values you specified as parameters. The response message contains the sum of the "num1" and "num2" values, in the "AddResult" parameter.

**Note:** This example only applies to simple data. For Web Services where the outputs are complex types see the example in Figure 135 on page 211. For cases where both inputs and outputs are complex data, testing is not supported. For more information, see Dealing with complex data types on page 210. For information on using XML output, see Using XML output for complex data types on page 213.

**Using the Advanced tab**

Open the Advanced tab to define timeout settings. If required, you can enter the number of seconds you want the workstep to wait to complete the web services invocation. If the defined timeout is exceeded, the workstep is suspended. By default, the timeout is set to "-1", or for an infinite time.

The Advanced tab also can enable you to provide session token support. By default, the session support is disabled and hidden. If you want to enable it, see the procedures in Supporting passing of session tokens to Web Services on page 207.

**Dealing with complex data types**

Most Web Services have input and output parameters that are of basic data types, such as String, Integer, or Decimal. Some services, however, use complex data types to represent entities such as employee information, product order, and so on.

If the Web Service you are going to use does not contain complex types, you may skip this section and continue to Defining the dataslot mapping on page 214.

When a complex object is either passed as an input parameter to a Web Service, or is returned from the service as an output, the mapping between the CHARACTER type and the corresponding Java class must be defined in the Types and Mapping tab.

**Follow the steps to understand a sample service:**

2. Click Go. The following screen is displayed.
Note: Regardless of the public Web Service you use, the following section describes how to use complex data types in the Web Service Adapter.

**Figure 135: Web Service Adapter Configurator - Types and Mapping tab**

The Web Service Adapter Configurator makes its best effort to simplify your work, and it automatically defines the type mapping for every complex type it encounters. The automatically generated definition assumes the following:

- Java class name is the same as the XML type localpart name
- Java class is a JavaBean type object, whose fields can be manipulated with a set of set/get methods

**Using Java input and XML output**

The following steps guide you to generate XML output from Java input:

1. Select the Java Input and Soap Message Output option.
2. Click **Edit**. The Mapping Editor dialog box appears as follows. Modify the type mapping definition, only if any of the preceding assumptions is not true.

**Figure 136: Web Service Adapter Configurator - Mapping Editor**

3. Provide the required information and Click **OK**.
Use the Mapping Editor to enter values (or edit the already present values) for the XML-type, fully qualified class name for the Java object that represents the data type. Sometimes a complex object type may contain another complex object as one of its fields. In this case, you may have to manually define the XML mapping for the nested type(s). If needed, you can also define your custom serializer and deserializer factory that are used to convert the Java object to XML form and back.

Generally, when using a Web Service, the provider can give you enough information about what each of these fields must contain. Additional information about writing custom serialization and deserialization factories can be obtained from Apache Axis (http://ws.apache.org/axis/).

Note: For more information about complex data types, see BP Server examples or use the example provided under OEBPS_HOME\BP Server\examples\WebserviceComplexType.zip.

Finally, note that Web Services containing complex type inputs cannot be tested through the Web Services Adapter Configurator. In this case, make sure to properly test your configuration by running the process from Business Process Portal.

Note: The classes for the complex types are not reloadable. If there is any change in the binary, stop the application Server and restart them.

Using XML Input and Output

The WebService managed adapter can construct the input XML on the fly by using the XML format plugin and mapping dataslots to certain XML elements. Use Soap Message as Input and Output option to have input and output in XML format.

To use XML Input and Output follow the steps:

1. Select Soap Message as Input and Output option, in the Types and Mapping tab.
2. Open the Test tab. The XML input is displayed on the SOAP Request text area.
3. Check Map Request fields. The Request Mapper option is available.
4. Click Request Mapper. The XPath Wizard appears.
5. Click Add to add a parameter. The Parameter Editor dialog appears.
   a) Enter the parameter name in the Name box.
   b) Browse and select the X path and Click OK.
   c) Select the type from the Type drop-down list.
   d) Click OK to return to the XPath wizard.
   e) Click OK to return to the WebService Adapter Configurator.

Figure 137: Web Service Adapter Configurator - XPath Wizard
6. Provide the values and click Invoke. The XML output appears in the lower text area, as shown in the following figure.

**Figure 138: Web Service Adapter Configurator - Test Tab - XML Response**

7. Click OK to complete the configuration, and open the Dataslot Mapping dialog window (see Defining the dataslot mapping on page 214).

**Using XML output for complex data types**

In case the Web Service you are using contains complex output and you do not want to use a custom Java class to handle it (see Dealing with complex data types on page 210), you can select the Java Input and Soap Message Output option that is displayed in the Types and Mapping tab and shown in Figure 135 on page 211.

Selecting the Java Input and Soap Message Output option indicates that you are expecting the result to be returned as an XML string and not as a Java object. All the complex data type outputs will disappear from the mapping table, as seen in Figure 135 on page 211.

If you use the Java Input and Soap Message Output option, all result data will be returned as a single output that, when you are mapping dataslots (see Defining the dataslot mapping on page 214), you can map to an CHARACTER dataslot. In some situations, returning XML may not be convenient or necessary; for instance, if you are interested in only a few values that are contained in the XML response. In this case, you can use the built-in XPath Editor to directly return the data of interest, instead of the complete response XML.

**To use XML output for complex data types:**

1. Select Java Input and Soap Message Output option in the Types and Mapping tab.
2. Open the Test tab.
3. Add values for each of the Java inputs listed.
4. Click Invoke, and the XML output appears in the lower text area, as shown in the following figure.

**Figure 139: Web Service Adapter Configurator - Test Tab -Java Input**

The Map Response fields option is available.

5. Check Map Respond fields. The Response Mapper option is available.

6. Click Respond Mapper. The XPath Wizard appears. (See Figure 137 on page 212)

7. Click Add to add a parameter. The Parameter Editor dialog appears.
   a) Enter the parameter name in the Name box.
   b) Browse and select the X path and Click OK.
   c) Select the type from the Type drop-down list.

8. Click OK to complete the configuration, and open the Dataslot Mapping dialog box (see Defining the dataslot mapping on page 214).

For information on the XML Plugin, see Using the XML Format plug-in on page 176. Also, note that you cannot use XML plugin internal variables with the Web Service Adapter (for more information on use of variables, see Using variables and conditions in XPath).

When using the XML plugin, all the selected fields are available for mapping to dataslots as adapter outputs. The response XML is not available as an output.

**Defining the dataslot mapping**

Once the Web Service Managed Adapter is configured, the Web Service inputs and outputs must be mapped to the dataslots that are present in your particular business process.

The Map Configurator dialog box appears automatically when the adapter’s configuration is complete. You can also manually invoke the Map Configurator by clicking Change Mapping… in the Configuration tab of the Properties view for the Web Service Adapter workstep.

The appearance of the Map Configurator dialog box depends on the inputs and outputs of your web service. The following sections describes each of the available tabs.
Using the Inputs tab

The Inputs tab includes the list of inputs for the web service.

For the MathService example (see Configuring the Web Service Adapter on page 203), the Map Configurator lists the "A" and "B" input parameters in its "Inputs" tab.

From the combo box beside each input parameter, select the dataslot that you want to be associated to the input. In this case, the value for the A input comes from the dataslot X, and the B input receives its value from the dataslot Y. You can also directly enter constant values in the combo box for each input — in this case, the given is used for each invocation of the adapter, and you don’t need to define a separate dataslot for it.

Using the Outputs tab

The Outputs tab contains the list of the output parameters from the Web Service, and allows you to specify target dataslots for each value. In the MathService case, the Outputs tab contains the "AddResult" parameter, containing the sum of the A and B parameters.

Here, we mapped the AddResult so the value from the addition is stored in the Z dataset.

Notice that the dataset data types are not mentioned anywhere in the mapper. In case the type of a dataset does not match the type of an input or output, the managed adapter framework converts the value appropriately. Not all such conversions may make sense in the context of a given Web Service or Business Process Server process, so be careful when mapping different types, and consult the information in the "Supported Types for Conversion" table in the "Developing Custom Managed Adapters" chapter in the Customization Guide.

Using the Advanced tab

The Advanced tab enables you to map the Web Service results at run time to another machine (Target Endpoint), if required. Enter the Target Endpoint Address (that is, the URL address of the WSDL Soap address location) in the Source combo box, or select a dataset that contains the value of the Target Endpoint Address location.

If you do not know the Target Endpoint Address, you can find it in the service name, WorkflowWSSEntity. The wsdlsoap:address location parameter is shown in bold in the following sample code.

```xml
- <wsdl:service name="WorkflowWSSEntity">
  - <wsdl:port binding="impl:BPServer1SoapBinding" name="BPServer1">
    <wsdlsoap:address location="http://10.1.5.169:18793/sbm/services/BPServer1" />
  </wsdl:port>
</wsdl:service>
```

```xml
</wsdl:definitions>```
Setting workstep properties

This chapter describes how to set properties of worksteps and other shapes that are used to create process templates in Process Modeler. Each process template must have one Start workstep, one or more intermediate worksteps, and one or more End worksteps, with the worksteps connected by links.
For details, see the following topics:

- About worksteps
- Defining properties of Start workstep
- Defining properties of Activity worksteps
- Defining properties of Adapter worksteps
- Defining properties of Subprocess worksteps
- Defining properties of Message worksteps
- Defining properties of End worksteps
- Defining properties of Decision gateways
- Defining properties of Or-Join gateways
- Defining properties of Exclusive Or-Join gateways
- Defining properties of AND gateways
- Defining link properties
- Defining a Rollback
- Defining a Timeout in worksteps
- Managing multiple worksteps

### About worksteps

Worksteps define the flow of business processes. In Business Process Modeler, you can create a workstep:

- In process models by dragging one of these shapes into the Content pane: Start, Activity, Decision, Or-Join, Exclusive Or-Join, AND Gateway, Message, and End.
- In Web applications using these shapes: Start, Activity, Decision, Or-Join, and End.

Step Table 42 on page 219 lists and describes the types of worksteps and how they appear in a process template.
### Table 42: Types of Worksteps

<table>
<thead>
<tr>
<th>Workstep Type</th>
<th>Appearance</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity</td>
<td><img src="image" alt="Activity" /></td>
<td>This workstep is performed by a User (human performer). In a Process Model, an activity can be performed by an individual or group user, or by a queue. In a Web application, this workstep is performed only by a single user. For details, see Defining properties of Activity worksteps on page 224.</td>
</tr>
<tr>
<td>Adapter</td>
<td><img src="image" alt="Adapter" /></td>
<td>This is an activity workstep, performed by an adapter. For details, see Defining properties of Adapter worksteps on page 230.</td>
</tr>
<tr>
<td>Subprocess</td>
<td><img src="image" alt="Subprocess" /></td>
<td>This workstep is a process by itself with its own process diagram and is external to the main process, including the subprocess workstep. For details, see Defining properties of Subprocess worksteps on page 233.</td>
</tr>
<tr>
<td>External Activity</td>
<td><img src="image" alt="External Activity" /></td>
<td>This is a monitoring workstep, performed by an external human performer. This workstep type is only available for process models. For details, see Introducing Monitoring process on page 249.</td>
</tr>
<tr>
<td>External Adapter</td>
<td><img src="image" alt="External Adapter" /></td>
<td>This is a monitoring workstep, performed by an external adapter performer. This workstep type is only available for process models. For details, see Introducing Monitoring process on page 249.</td>
</tr>
</tbody>
</table>

Although the properties of an Activity, Adapter, and Subprocess worksteps vary according to their performers, these worksteps have the following properties in common:

- **Name.** Business Process Modeler assigns each workstep a default name; for instance, the first Activity workstep added to a process template is displayed as Activity 1. You can replace the default name with a more descriptive name in the Properties view for the workstep.

- **Dataslots.** Dataslots are used to transfer information between worksteps. You can define editable inputs (information that flows into the workstep) and read-only outputs (information that is sent out to other worksteps) in the Dataslots tab of the Properties view for Adapter, Message and Subprocess worksteps; in the Fields tab of the Properties view for Start and Activity worksteps; and in the Configuration tab of the Properties view for Managed Adapter worksteps.

- **Performers.** Each of these worksteps can be assigned a performer. Performer types include: User, Adapter, and Subprocess.
Workstep properties

You can configure properties of a workstep, using the Properties view for that workstep.

Note: The Properties view is available for each process template element in Diagram and Overview tabs.

The Properties view for worksteps supports the following dynamic features:

- **General** tab: Allows you to define performers for Activity (see Specifying general properties of Activity workstep on page 224) and Adapter (see Specifying general properties of Adapter workstep on page 230) worksteps. You can also specify the workstep label (as it appears in BPM Portal) as well as define Overdue and Loop settings. For Start and Activity worksteps, you can also define the presentation format for the workstep.

- **Dataslots** tab (or in the Configuration tab for Managed Adapter worksteps or the Fields tab for Start or Activity worksteps): Allows you to assign dataslots to the workstep and modify them as they would appear on Business Process Portal. For more information on defining dataslots for a workstep type, see the Dataslot Properties description for that workstep type; for instance, for the Start workstep, see Using the Fields tab of Start workstep properties on page 222.

- **Collaboration** tab (only available for Activity worksteps): Allows you to select a level of collaboration and indicate the performers with whom you want to collaborate. You can also add such options to the workstep as reassigning tasks, adding Notes to a process, or sending Email or Instant Messaging.

- **Messaging** tab (only available for Start, Message, and End worksteps in Process Models): Allows you to select a message to initiate a process or a workstep; and to generate an outgoing message on the completion of a workstep or a process. Using this type of workstep provides a way to synchronize process events and to define actionable events at various stages of the process execution.

- **Alerts** tab (only available for Process Models): Allows you to add alerts to be associated with activation or completion of the workstep.

Note: You can also configure workstep properties using the Properties dialog box. You can open the Properties dialog box for any workstep by double-clicking the workstep or right-clicking the workstep and clicking Properties option.

Defining properties of Start workstep

The Start workstep (○) indicates the beginning of a Business Process Server process template. Each process must have one Start workstep. The Start workstep only contains output dataslots, which can be mapped to specific locations.

Note: The Properties view for worksteps in Process Models have more features than those for Web Applications. In the following sections, only Properties view for Start worksteps in Process Models are presented, which are applicable for Web applications as well, unless otherwise stated.
Click the Start workstep to view its properties in the **Properties** view. The **Properties** view includes five tabs (Start worksteps in Web applications include three tabs).

## Specifying general properties of Start workstep

**From the General tab of the Properties view:**

1. Identify the workstep by entering a name in the **Name** box. The workstep name appears in the process template diagram. Workstep names can only include alphanumeric characters and underscores (_).

   **Figure 140: Start Workstep Properties – General Tab**

2. Enter a label for the workstep in the **Label** box. Once you assign a label to a Start workstep, it always appears in the Start workstep in the process template diagram and in Business Process Portal—the Name of the Start workstep no longer appears.

3. To modify the default look and feel of the presentation pages for the Start workstep in Business Process Portal, click the **ellipsis** button beside the **Presentation** box (for more information, see *Defining workstep presentation format* on page 257).

**Note:** You can enter a description (optional) for the workstep in the **Description** tab. By default, the description can have up to 4095 characters.
Using the Fields tab of Start workstep properties

For Process Models, you can use the **Fields** tab (see Step Figure 141 on page 222) to assign and manage dataslots, which control the information flow from the Start workstep. In Web applications, the **Dataslots** tab replaces the **Fields** tab.

**Figure 141: Start Workstep Properties – Fields Tab**

![Image of the Fields tab](image)

The Header section displays the options—Instructions, Priority—to be displayed in the header section of Business Process Portal’s task details page (corresponding to this workstep). You can choose to show or hide such information as Instructions for the workstep in the **Header** section of the Start workstep.

**To do this:**

1. Select any or all of the checkboxes in the **Options** section, to display the corresponding fields in the user’s Business Process Portal page.

2. Enter instructions for the workstep in the **Instructions** box. Instructions can be up to 255 characters. These instructions appear in Business Process Portal.

   Alternatively, you can link the instructions to a specific CHARACTER dataslot. To do so, select the **Link instructions to Dataslot** checkbox, then select the dataslot (if any) from the adjoining drop-down list. To create a new dataslot, click the **ellipsis** button beside the **Link instructions to Dataslot** checkbox to open the **New Dataslot** dialog box (Creating user-defined dataslots on page 101).

   The user-defined dataslots assigned to the workstep are listed in the Fields section, which contains a table displaying the **Icon**, **Name**, **Type**, **Category**, **Label**, **Editable**, and **Required** columns. For information regarding these dataslot parameters, see Using dataslots on page 97.

**Important:** If the workstep has a Form presentation format, the **Add**, **Remove** and other functions (excluding Modify) in the Fields tab are disabled and a message states that “You can add or remove fields in the Form Editor”. Add or remove a dataslot from the Form Editor or select a different presentation form to enable these functions.
Adding dataslots

You can add user-defined dataslots to the **Fields** section in the **Fields** tab.

**To do so:**

1. Click **Add** to open the **Add** dialog box, which lists all the dataslots available to the process.

    ![Figure 142: Select Dataslots dialog box](image)

2. Select one or more dataslots and click **OK** to add them to the workstep.

You can modify a dataslot listed in the **Fields** section by selecting it and clicking **Modify**, displaying the **Field Properties** dialog box where you can edit the selected dataslot's properties.

To remove a dataslot listed in the **Fields** section, you can click **Remove**. Click **Move Up** or **Move Down** to change the sequencing of a selected dataslot.

**Important:** We recommend that you do not use dataslots with predefined values in a Start workstep.

**Note:** For Process Models, you can use the **Alerts** tab to associate an alert with a Start workstep when it is completed. For more information, see **Associating an alert with a workstep** on page 272.

Using the Messaging tab of Start workstep properties

In Business Process Modeler, you can open the **Messaging** tab, but then you can only set the **Enabled** value to “True” in the Start workstep’s **Properties** view. This action results in the Start workstep displaying a Message (✉) icon, and is for notational purposes only—you must open the process in Progress Developer Studio for OpenEdge to enable the Start workstep to receive a message. To remove the **Message** icon from the Start workstep, you must delete the workstep and create a new Start workstep.
Defining properties of Activity worksteps

Activity worksteps are the basic unit of work, and must be performed in their entirety. An Activity workstep in a Process Model is an activity assigned to an individual user or to a group of users. An Activity workstep in a Web application can only be assigned to an individual user.

Note: The Properties view for worksteps in Process Models have more features than those for Web applications. In the following sections, only Properties view for Activity worksteps in Process Models are described unless otherwise stated.

Click the Activity workstep to view its properties in the Properties view. As shown in Step Figure 143 on page 224, the Properties view contains five tabs (Activity worksteps in Web applications contains three tabs).

Specifying general properties of Activity workstep

From the General tab of the Properties view:

1. Identify the workstep by entering a name in the Name box. The name of the workstep is displayed in the process template diagram. Workstep names can only include alphanumeric characters and underscores (_).

   Figure 143: Activity Workstep Properties – General Tab

2. Enter a label for the workstep in the Label box. This label of the workstep appears in Business Process Portal.
3. The **Performer** box (available only for Process Models) displays the performer assigned to the Activity workstep (as described in Assigning performers on page 126). To assign another performer, click the **ellipsis** button beside the **Performer** box to open the **Performer** dialog box.

**Figure 144: Performer dialog box**

![Performer dialog box](image)

a) All the User performer types available in the process are listed in the **Performer** dialog box. If there are many performers listed, then you can search the performer using the Search box and any User name containing that string is displayed in the text area. For example, if you entered "mgr" in the Search box in the **Performer** dialog box, two users, mgrA and remoteMgr, are listed. You can create a performer by clicking **New** to open the **Performer** dialog box (Defining a user as a performer on page 118).

b) Click **OK**, to select the user as the workstep performer and to return to the **General** tab.

4. From the **Default Performer** box (available only for process models), specify the default performer, to whom the Activity workstep is assigned if the workstep performer is not available. The default performer can be an individual user or a dataslot. This is an optional field. To assign a default performer, click the **ellipsis** button beside the **Default Performer** box, then select the **Select a Performer** option to open the **Performer** dialog box, which lists all the performers shown in Figure 144 on page 225.

5. The **Presentation** box defines the look and feel of the presentation pages for the Activity workstep. To change the presentation format, click the **ellipsis** button beside the **Presentation** box to open the **Presentation** dialog box (for more information, see Defining workstep presentation format on page 257).

6. To set the priority of the workstep, select any of the predefined options from the **Priority** drop-down list (available only for process models). Predefined options include: **Low**, **Medium**, **High**, **Critical**, and **Use a dataslot**. If you select **Use a dataslot**, then you can use the **Select Dataslots** dialog box to select any of the listed dataslots of CHARACTER type.

7. The **Overdue in** box displays the default duration (2 hours) in which the workstep needs to be completed. Modify the time in which the workstep is overdue by clicking the **ellipsis** button beside the **Overdue in** box, and select either of the following options:

   - **Enter a duration** option to open the **Duration** dialog box (Step Figure 145 on page 225). Enter the number of days, hours, minutes, or seconds to specify the workstep’s duration. Click **OK**.

**Figure 145: Workstep duration**

![Duration dialog box](image)
• **Use a Dataslot** option to open the **Select Dataslots** dialog box, which lists the INTEGER dataslots available in the process. Select a INTEGER dataslot to specify a dynamic duration for the workstep, and click **OK**. You can use only INTEGER dataslots with integer values for duration.

**Note:** The value of this INTEGER dataslot must be in seconds, since this is the unit of time that BP Server expects for the duration until workstep completion. For example, if you want to set the workstep overdue duration to two minutes, then enter the dataslot value as 120 seconds.

8. Select the **Skippable** checkbox to allow an individual or member of a group to skip this work item (or task) at runtime. When the application is published on Business Process Portal, the task that represents the work item includes a **Skip** button that performers can click to skip the task. For example, if a work item is assigned to three performers and two of the performers choose to skip the work item, the third performer can perform the task. This feature is present only in Activity worksteps for Process Models.

9. Select the **Collect Work Time** check box if you want to force the performer to enter the time it took them to complete the task. By selecting this checkbox, you create a footer in Business Process Portal that requires the user to enter the Days, Hours and/or Minutes they needed to complete the task. This feature is present only in Activity worksteps for Process Models.

10. To make the Activity workstep execute multiple times, select the **Loop** checkbox. Follow the **Defining a Loop condition** on page 226 procedure. This feature is not available for Web applications.

**Defining a Loop condition**

A Loop occurs when the workflow is repeated through the current workstep until a specified loop condition is met. To track the number of times the workstep is executed, you can define a counter that increments every time the loop is executed.
To define a Loop condition:

1. From the General tab of an Activity workstep’s Properties view, select the Loop checkbox, enabling the field and ellipsis button.
2. Click the ellipsis button to open the Loop dialog box, in which you can define the Loop condition.

   **Figure 146: Activity Workstep Properties – Loop dialog box**

3. Enter a variable (note that the variable must be a INTEGER dataslot type) by clicking the ellipsis button beside the Variables box. From the Select Dataslots dialog box, select a INTEGER dataslot type from those listed. Each time the workstep executes the loop, this variable is incremented by one.

4. From the Condition section, build a condition by selecting a dataslot from the drop-down list on the left and an operator from the middle drop-down list, and then selecting or entering a value in the drop-down list on the right. To enter a value:
   a) From the drop-down list on the right, select Enter Value option to open the Enter Value dialog box.
   b) Enter the value and click OK.

5. Click OK after defining the loop condition for this workstep.

As long as the defined condition is “true”, the workstep executes a loop. When the condition reads “false”, the workflow passes to the next workstep.

The example shown in Step Figure 146 on page 227 above is a counter-controlled loop, where the first condition sets the workstep to iterate five times. To do this, set the count dataslot to “is less than or equal to”. On the sixth iteration, the workflow goes out of the loop and continue to the next workstep. This type of loop is useful if, for example, you wanted a user to enter data five times before proceeding to the next workstep.

The second condition indicates that the Performer dataslot must equal “Jones”: if the performer is not Jones, the condition is false and the workflow proceeds to the next workstep.

When a workstep is defined as a Looping workstep, the Loop (◯) icon appears within the workstep shape, indicating to other users that a loop occurs in this workstep.

**Note:** If a workstep has both a Skip condition and a Loop condition, then the Skip condition is evaluated first. If it evaluates to true, then the workstep is skipped; if false, then the workstep is
executed and the Loop condition is evaluated. If the Loop condition is evaluated as true, then the Skip condition is evaluated again.

**Note:** You can enter a description (optional) for the Activity workstep in the **Description** tab. A description can have up to 4095 characters.

---

### Using the Fields tab of Activity workstep properties

You can use the **Fields** tab to assign and manage dataslots that control the information flow into and out of an Activity workstep. You can manage existing dataslots and add dataslots using the **Fields** tab, as described for the Start workstep (see **Using the Fields tab of Start workstep properties** on page 222).

**Important:** If the workstep has a Form presentation format, the **Add**, **Remove**, and other functions (excluding **Modify**) in the **Fields** tab are disabled and a message states that “You can add or remove fields in the Form Editor”. You must add or remove a dataslot from the Form Editor or select a different presentation form to enable these functions.

---

### Defining collaboration tasks

The Collaboration feature is available only for Activity worksteps in Process Models. Collaboration allows you to set a number of options in Business Process Modeler, which displays the activity workstep as a collaborative task in Business Process Portal. Collaborative tasks allow application users to complete their work by collaborating with other users or groups (referred to as Collaborators). These options include the ability to select a level of collaboration as well as indicating the performers with whom you want to collaborate. You can also select other options such as reassigning tasks, adding Notes to a process, or sending Email or Instant Messaging.

**To define collaboration properties:**

1. Click the **Collaboration** tab in the **Properties** view for an Activity workstep.

**Figure 147: Activity Workstep Properties – Collaboration Tab**

![Collaboration Tab](image)

2. From the **Options** section, select one or more of the following options that you want to include in Business Process Portal.

   - **Collaborative Tasks.** Select the appropriate options in the drop-down list to indicate whether you want to allow Business Process Portal user to edit the dataslots associated with the workstep, or only view the dataslots that are now read-only, or see no dataslots.
• **Notes.** Specify whether you want the Business Process Portal user to see notes at the Instance Level (for all worksteps in the instance) or at the Workstep Level (for the current workstep). For more details regarding Notes, see the *Business Process Portal User's Guide.*

• **Reassign.** By default, the assigned performer of this workstep can reassign the task to another performer. To disable reassigning, clear this checkbox.

• **Email.** To allow the assigned performer of this workstep to communicate with any of the collaborators using e-mail.

• **Instant Messaging.** To allow the assigned performer of this workstep to chat with any of the collaborators using Instant Messaging.

3. From the **Collaborators** section, add the collaborators and also specify the operations the workstep performer can perform with each collaborator in BPM Portal. Note that there is a column added for each option that you select in the **Options** section.

![Figure 148: Activity Workstep Properties – Collaboration Tab, Collaborators tab](image)

a) To add a collaborator, click **Add** to open the **Select Users** dialog box, in which you can select one or more of the users or groups, and click **OK** to add them to the **Collaborators** list.

**Note:** You cannot add (or create) a queue as a collaborator. Further, you cannot add a group with ‘Any’ member selected as a collaborator or a group with a defined role.

b) For each of the specified collaboration options, select either “true” or “false” from the drop-down list.

**Note:** To delete a collaborator from the list, select the user or group and click **Remove.**

---

For worksteps using Form presentation, all the options selected in the **Options** section are automatically available in the Standard footer in Form Editor. For worksteps with default presentation, the options selected in the **Options** section are displayed in the Task Details page of the workstep and all the collaborators defined in the **Collaborators** section are available in the **Collaborators** list in the **Create Collaboration** page in Business Process Portal.

**Note:** For Process Models, you can use the **Alerts** tab to associate an alert with an Activity workstep when it is initially activated or completed. For more information, see Associating an alert with a workstep on page 272.
Defining properties of Adapter worksteps

An adapter workstep is an activity workstep performed by an adapter, which runs an external system, remote device, or an application that is on a remote server external to Process Modeler. You can create an adapter workstep using a custom adapter or any of the eight predefined managed adapters.

**Note:** Properties view for worksteps in Process Models have more features than those for Web applications. In the following sections, only Properties view for Custom Adapter worksteps in Process Models are described, unless otherwise stated.

Click the Adapter workstep to view its properties in the Properties view. The Properties view contains four tabs (Adapter worksteps in Web applications contains three tabs) as shown in Step Figure 149 on page 230.

**Specifying general properties of Adapter workstep**

Similar to that for an Activity workstep (Specifying general properties of Activity workstep on page 224), you can use the General tab to define the general properties of an Adapter workstep.

**Figure 149: Adapter Workstep Properties – General Tab**

1. Identify the workstep by entering a name in the Name box. Workstep names can only include alphanumeric characters and underscores (_).  

   **Note:** There is no default duration shown in the Overdue in box for Custom and Managed Adapters. There is no Performer box for managed adapters.

2. To define other general properties of an Adapter workstep that are identical to those already described for Activity worksteps, refer to Specifying general properties of Activity workstep on page 224.

   **Note:** You can enter a description (optional) for the adapter workstep in the Description tab. A description can be up to 4095 characters.

   **Note:** For Process Models, you can use the Alerts tab to associate an alert with the Adapter workstep when it is initially activated or completed. For more information, see Associating an alert with a workstep on page 272.
Using the Dataslots tab of Adapter workstep properties

You can use the **Dataslots** tab to control the information flow into and from a workstep performed by a custom (user-defined) adapter. From the **Dataslots** tab, you can assign dataslots to the adapter workstep and manage existing dataslots.

**Note:** For a managed adapter, the **Configuration** tab replaces the **Dataslots** tab. For information on configuring the managed adapter and mapping dataslots between Managed Adapter inputs/outputs and dataslots in processes, see the *Managed Adapters Guide*.

Figure 150: Adapter Workstep Properties – Dataslots Tab

Dataslots assigned to the adapter workstep are listed in a table in the **Dataslots** tab. You can add, modify, or remove the dataslots.

The **Name** column provides the name of the dataslot. The other two columns show whether the dataslot provides Input to the Adapter or Output from the Adapter. For information on mapping dataslots, see *Mapping dataslots in an Adapter workstep* on page 232.

**To add dataslots to those listed in the table:**

1. Click **Add** to open the **Select Dataslot** dialog box.
2. Select one or more dataslots and click **OK** to assign them to the adapter workstep.

You can modify the Input and Output adapter settings for the existing dataslots in the **Dataslots** tab. Click in the **Input to Adapter** or **Output to Adapter** column for the dataslot you want to modify, then select **true** or **false** in the drop-down list.

Figure 151: Field Properties of a Dataslot in an Adapter Workstep

**Note:** To remove a listed dataslot, select it and click **Remove**.
Mapping dataslots in an Adapter workstep

Mapping dataslots play an important role in adapter worksteps. Mappings enable you to pass data between dataslots in a process and methods in an adapter, and to automatically handle occasions when the names of the dataslot do not match the names of the adapter methods.

When you define a dataslot as an input to an adapter, BP Server invokes a corresponding set method in the adapter to assign the dataslot value to the variable in the set method (adapter). Similarly, to retrieve a variable from an adapter to a dataslot, BP Server invokes a get method at run time.

Note: For information on configuring and mapping any of the managed adapters, see the relevant chapters in the Managed Adapters Guide.

To map dataslots in an adapter workstep to a method in an external adapter:

1. From the adapter workstep Properties view, assign the dataslots for the adapter workstep, as described in Using the Dataslots tab of Adapter workstep properties on page 231.

The Fields view, which is displayed by default, shows dataslots as Input to the adapter, Output from the adapter, or both.

2. Select the Advanced option from the drop-down list.

The Advanced view automatically creates and displays the set methods used for Adapter inputs and the get methods used for Adapter outputs.

3. To edit an adapter input (set method), select an adapter input as displayed in the Advanced View and click Modify to open the Field properties dialog box.

4. Click the ellipsis button beside the Adapter input(s) box to open the Adapter input(s) dialog box.

5. Enter a new set method name in the text box provided and click Add, which adds it to the list in the Adapter input(s) dialog box.

Note: To delete a set method, select it from the list and click Remove. To change the sequence of listed set methods, use the Move up and Move down buttons.

6. After defining the set methods, click OK to return to the Field properties dialog box.

7. To add an adapter output (or get method), enter a name in the Adapter output box.

8. Click OK, returning to the Dataslots tab of the Properties view.
Defining properties of Subprocess worksteps

The Subprocess workstep is an activity workstep performed by another process. It allows you to nest a complete Business Process Modeler process into another process template.

Specifying general properties of Subprocess workstep

From the General tab of the Properties view:

1. Identify the workstep by entering a name in the Name box. The name of the workstep appears in the process template diagram. Workstep names can only include alphanumeric characters and underscores (_).

![Figure 153: Subprocess Workstep Properties – General Tab](image)

2. When specifying the Performer in a Subprocess workstep, you must select a process to serve as the subprocess in the workstep. You can specify a process installed either on your local server or on a remote server.

   To define a performer (that is, to select a process) for the Subprocess workstep, click the ellipsis button beside the Performer box to open the Performer dialog box.

   a) All the subprocesses available in the process are listed in the Performer dialog box. If many subprocesses are listed, then you can search the Subprocesses performer type using the Search box and any name containing that string is listed.

   b) After selecting a subprocess, click OK to return to the Subprocess workstep and returning to the Subprocess Properties view. The subprocess is defined as the performer of the Subprocess workstep.

3. To define other general properties of a Subprocess workstep, which are the same of those described for Activity worksteps, refer to Specifying general properties of Activity workstep on page 224.

   **Note:** You can enter a description (optional) for the subprocess workstep in the Description tab. A description can be up to 4095 characters.

   **Note:** For Process Models, you can use the Alerts tab to associate an alert with the Subprocess workstep when it is initially activated or completed. For more information, see Associating an alert with a workstep on page 272.
Using Dataslot tab of Subprocess workstep properties

You can use the **Dataslots** tab to assign and manage dataslots, which control the information flow into and from the Subprocess workstep.

**Note:** The **Dataslots** tab is not available for inline subprocesses.

---

**Figure 154: Subprocess Workstep Properties – Dataslots Tab**

You can add, modify, or remove dataslots assigned to the Subprocess workstep in the Dataslots tab, an easy-to-use interface where you can add new or modify existing dataslots.

By default, the **Fields** view is displayed, with dataslots listed in a table. The Name column lists the name of the dataslot. The other two columns indicate whether the dataslot is input or output for the subprocess.

You can add, modify, or remove dataslots, the same way as you manage dataslots for an Adapter workstep (as described in **Using the Dataslots tab of Adapter workstep properties** on page 231).

To assign a dataslot as input to or output from the **Fields** view, click in the **Input to Sub-Process** or **Output from Sub-Process** column for the dataslot you want to modify, then select **true** or **false** in the drop-down list.

**Mapping dataslots in a Subprocess workstep**

If the name and data type of the dataslots in a Business Process Model and the dataslots in a Web Application are identical, then default mapping occurs between the processes. If the name or data type differ, then default dataslot mapping does not work correctly.

**Note:** Default dataslot mapping is case-sensitive. For example, dataslots called “Budget” and “budget” are not mapped by default and must be mapped in the workstep Properties.

**To map dataslots in a Subprocess workstep:**

1. From the drop-down list under Dataslots, select **Advanced** option, which displays the **Set dataslot(s)** and **Get dataslot** columns in the table below that.

   The Advanced view enables you to pass dataslot values from the subprocess to an input (Set) dataslot or to an output (Get) dataslot in a parent process.

2. Select a dataslot and click **Modify**, to launch the **Field Properties** dialog box. You can enter multiple Set dataslots and one Get dataslot for the Subprocess dataslot.
3. To map a parent process dataslot to one or more inputs in the subprocess, click the **ellipsis** button beside the Set dataslot(s) box to open the Set Dataslots dialog box.

   a) In the text box, enter the name of the dataslot in the parent process that you want to map to the subprocess dataslot and click **Add**. The parent process dataslot is added to the dataslot list. Use the other command buttons to delete, or move a selected dataslot up or down.

   b) Click **OK** to return to the **Field Properties** dialog box.

4. In the **Field Properties** dialog box, enter an output dataslot name in the **Get dataslot** box to map a subprocess dataslot to an output dataslot in a parent process.

5. Click **OK**.

The dataslots to which you mapped the subprocess dataslots are displayed in the Set dataslot(s) and Get dataslot columns.

---

**Defining properties of Message worksteps**

In Business Process Modeler, the Message workstep ( ) is for notational purposes only—that is, you can only use this workstep as a symbol in the process template diagram. You cannot define the properties of a Message workstep in Business Process Modeler. You must open the process in Progress Developer Studio for OpenEdge to enable the **Properties** view for the Message workstep.

In Progress Developer Studio for OpenEdge, the Message workstep enables users to use an incoming event or message to initiate a process or a workstep; and conversely to generate an outgoing message with the completion of a workstep or a process. Using this type of workstep provides a way to synchronize process events and to define actionable events at various stages of the process execution.

---

**Defining properties of End worksteps**

End worksteps ( ) mark the end of the process flow. A process template must have at least one End workstep.

Click the End workstep to view its properties in the **Properties** view. The **Properties** view of the End workstep for process models contains four tabs. For Web applications, it contains three tabs.

**Figure 155: End Workstep Properties – General tab**

From the **General** tab, enter the label for the workstep in the **Label** box. Once you assign a label to an End workstep, it always appears in the process template diagram and in Business Process Portal. If no label is entered, then the End workstep appears blank.
**Note:** You can enter a description (optional) for the End workstep in the **Description** tab. A description can be up to 4095 characters.

**Note:** For Process Models, you can use the **Alerts** tab to associate an alert with an End workstep when it is activated or when it is completed. For more information, see **Associating an alert with a workstep** on page 272.

---

### Using Messaging tab of End workstep properties

The **Messaging** tab is only seen in End worksteps in Process Models—it is not present in End worksteps in Web applications.

**Figure 156: End Workstep Properties – Messaging Tab**

You can open the **Messaging** tab, but then you can only set the Enabled value to “true” in the End workstep’s **Properties** view. This action results in the End workstep displaying a Message (✉️) icon, and is for notational purposes only—you must open the process in Progress Developer Studio for OpenEdge to enable the End workstep to send a message upon process completion.

### Using the Dataslots tab for Web applications

The **Dataslots** tab is available for End worksteps in Web applications only. If you are planning to use the process as a subprocess, then you can use the Dataslots tab to assign dataslots to the End workstep that passes information onto the parent process.

You can add, modify, and remove dataslots assigned to the End workstep, similar to how you manage dataslots for other worksteps.

### Defining properties of Decision gateways

The Decision gateway (☐) represents a division of the control flow of the process. A Decision gateway can contain multiple incoming and outgoing links. For each Decision gateway, at least one of the outgoing links must have an empty Condition (the Default link), and the other links can each contain a condition. Business Process Modeler follows BPMN notation by automatically displaying the default link as a line with a slash (/) through it.

Alternatively, you can get the equivalent of a Decision gateway by adding multiple outgoing links to a successor Activity workstep, and assigning a specified condition to some of the links. With this type of link, any links with conditions are indicated by a small diamond (◇) at the start of the link, and the default link as a line with a slash (/) through it. Once the predecessor workstep is completed, one or more successor worksteps are activated if conditions in the links are fulfilled.
Note: It is possible that the number of outgoing links for a Decision gateway can be more than the number of incoming links. This can occur when a Decision gateway has more than two outgoing links and two or more of these links fall within the probability range and are activated, resulting in an apparent discrepancy in simulation data. For an Exclusive Decision (see Defining Exclusive Decisions on page 237), however, only one outgoing link is activated and the number of outgoing links is always equal to the number of incoming instances.

Click the Decision gateway to view its properties in the Properties view. The Properties view of a Decision gateway includes two tabs—General and Description.

Figure 157: Decision Gateway Properties

To define properties for the Decision gateway:

1. From the General tab, modify (if required) the name of the Decision gateway in the Name box. Enter the label for the gateway in the Label box. Once you assign a label to a Decision, it always appears in the process template diagram. If no label is entered, then the Decision is blank.

2. To make the Decision gateway an exclusive decision gateway, select the Exclusive checkbox (available for Process Models only). For details, see Defining Exclusive Decisions on page 237.

3. The Links section contains a table that lists all the multiple outgoing links (and conditions, if any) from this gateway. You can modify or remove any link by selecting it and then clicking Modify or Remove. For information on defining link properties, see Defining link properties on page 240.

Note: You can enter a description (optional) for the Decision gateway in the Description tab. A description can be up to 4095 characters.

Defining Exclusive Decisions

An Exclusive Decision indicates that the workflow can go through only one outgoing link, while the typical Decision executes all links that meet specified conditions.
To set the Decision gateway as an Exclusive Decision gateway, select the **Exclusive** check box. When you select the **Exclusive** checkbox, the **Move Up** and **Move Down** buttons are displayed (Figure 158 on page 238), allowing you to establish the sequence of the outgoing links.

**Figure 158: Exclusive Decision Gateway Properties**

This ordering is important because links are evaluated in sequence, and the first link that fulfills the specified condition is executed. The remaining links are not evaluated.

In an Exclusive Decision, there must be only one default link (that is, a link with no condition). This link is executed when conditions on the other links are not fulfilled. If there is more than one default link or if there is no default link, then an error message is displayed.

Once you have defined the Decision as an Exclusive Decision gateway, it is displayed as a Decision gateway with an ‘X’ through it ( ). According to BPMN conventions, the appearance of the Exclusive Decision icon is the same as that of the Exclusive Or-Join (XOR) icon. The Exclusive Decision differs in that it has a single predecessor workstep and multiple successor worksteps.

**Note:** The Exclusive Decision gateway is not available for Web applications.

### Defining properties of Or-Join gateways

You can use an Or-Join gateway to connect multiple predecessor worksteps to one successor workstep. The successor workstep is performed only when any one of the predecessor worksteps are completed. The most common use of an Or-Join gateway is with a loop back or to connect to an End workstep. Alternatively, you can get the equivalent of an Or-Join by adding multiple incoming links to an Activity workstep, and this workstep is activated when any of its predecessor worksteps are completed.
Defining properties of Exclusive Or-Join gateways

You can use an Exclusive Or-Join gateway (\(\bigoplus\)) to direct the process workflow to proceed only once from multiple predecessor worksteps to a successor workstep. When the Exclusive Or-Join (XOR) gateway is completed, any other predecessor human-performed worksteps that immediately precede the XOR are terminated. According to BPMN conventions, the Exclusive Or-Join (XOR) icon is the same as that of the Exclusive Decision icon. The Exclusive Or-Join (XOR) differs in that it has multiple predecessor worksteps and a single successor workstep.

Click the Exclusive Or-Join gateway to view its properties in the Properties view, which is similar in functionality to that for Or-Join gateway.

**Note:** The Exclusive Or-Join gateway is supported only in Process Models.

**Figure 160: Exclusive Or-Join Example**

*Figure 160* on page 239 illustrates an example of the Exclusive Or-Join (XOR), wherein three companies are scheduled to collect a package. When any of these companies pick up the package, the XOR gateway is activated and the predecessor worksteps are terminated. The XOR gateway then proceeds to activate the Confirmation workstep.

**Important:** The XOR gateway functions correctly (that is, terminates the remaining predecessor worksteps) only if the predecessor worksteps are all human-performed Activity worksteps.
The XOR gateway allows workflow to proceed from only one of the predecessor worksteps. This action differs from the Or-Join, which would allow workflow to continue from Activity 1, 2, or 3 as each of these Activity worksteps are completed.

Defining properties of AND gateways

You can use an AND gateway (俶) to connect multiple worksteps to one successor workstep, or a single predecessor workstep to multiple successor worksteps. The successor workstep is activated only if all the predecessor worksteps are completed. This feature enables you to synchronize the completion of multiple predecessor worksteps. When an AND gateway has multiple incoming links, it must have a single outgoing link:—in this case, the AND Gateway acts as an And-Join. When an AND Gateway has a single incoming link, it can have multiple outgoing links:—in this case, the AND Gateway acts as an And Fork (or Split). These variations in incoming and outgoing links for AND gateways are illustrated in Using multiple incoming/outgoing links with gateways on page 56. Click the AND gateway to view its properties in the Properties view, which is similar in functionality to that of the Or-Join gateway.

Note: The AND gateway is supported only in Process Models.

Defining link properties

You can use the Connect Shapes link in the Tasks pane (see Connecting worksteps on page 53) to define the workflow connection (or links) between process template elements. You can either use the Normal Flow to link worksteps in the workspace, or the Compensation Flow to define the workflow in the event of a rollback (see Defining a Rollback on page 244), or the Time-out Flow (see Defining a Timeout in worksteps on page 245).

You can define links with or without conditions. In Decision gateways (see Defining properties of Decision gateways on page 236), you can modify an existing link condition in the Properties view, as described in this section.
For process models

You can now assign a condition to a link in the process template diagram. In previous releases, you could assign a condition only to a link from a Decision gateway. Once you assign a condition to a link that does not have a Decision as a predecessor workstep, the link becomes the equivalent of a Decision gateway.

**Figure 161: Link Properties – Basic Setting**

To define the properties of a standard link in process models:

1. Click the link to view its properties in the **Properties** view (Figure 161 on page 241).
2. Assign a name to the link in the **Name** box and add a label in the **Label** box of the **General** tab.
3. To assign a condition to the link, clear the **Default** checkbox to view the **Basic** and **Advanced** settings. By default, the **Basic** setting is displayed. For information on **Advanced** settings, see Step 4 below.
4. Use the features of the **Basic** setting to build an expression based on existing conditions.
   a) Select a dataslot from the drop-down list on the left (the Basic mode only includes CHARACTER, INTEGER, INT64, LOGICAL, and DATETIME-TZ dataslot types; the Advanced mode includes all data types). Select an operator from the adjacent drop-down list. Operator values change, depending on the dataslot type: For example, a CHARACTER dataslot displays more options than an INTEGER dataslot.

   **Note:** If you want to evaluate a Null value for a CHARACTER dataslot, then you only need to select ‘is null’ from the Operator drop-down list.

   b) Select a dataslot from the drop-down list on the right, or select **Enter value** option to open the **Enter Value** dialog box, in which you can enter a static value.

   c) Click 📌 to add another condition, enabling the And/Or operand.
Chapter 13: Setting workstep properties

d) Select And or Or and add as many conditional expressions as required.
e) To remove a conditional expression, click ✗.

5. To configure the condition using Advanced settings, select the Advanced option to view a list of dataslots and a Condition text area (Figure 162 on page 242) wherein you can build more complex expressions or to cut and paste expression that have been defined elsewhere.

Figure 162: Link Properties – Advanced Setting

a) You can enter or paste an expression in the Condition text area, or drag a dataslot from the Dataslot list to add it to the advanced condition.
b) When you add an expression in the Advanced setting, it overwrites all expressions that you created in the Basic setting. Click Validate to verify if the expression is valid.
c) Select Basic option to delete the expressions in the Advanced setting and to return to the Basic setting.

Note: You can enter a description (optional) for the Link in the Description tab. A description can be up to 4095 characters.
For Web applications

To define the properties of a link from an Activity workstep in Web applications, click the link to view its properties in the Properties view.

Figure 163: Link Properties in Web applications

1. Assign a name to the link in the Name box and enter the label in the Label box.
2. Business Process Modeler supports three types of links for Web applications, namely Submit button, Hyperlink, and Image. Select the link type from the Type list.
   - Select the Submit button option to display a Submit button on the interface page. Data is passed on from this interface page to the next interface page. Enter text in the Label box to be displayed as the name of the button, replacing the default Submit label for the button.
   - Select the Hyperlink option to create a line of text on the interface page that functions as a hyperlink. The hyperlink connection is displayed as a dotted line (••••••). In the Label box, enter the hypertext label to be displayed on the image or text link seen on the workstep interface page in your browser. You can also open the dialog box in your browser to add a brief description that will appear when you move the mouse over the image.
   - Select the Image option to create an image link on the interface page. In the Label box, enter the hypertext label to be displayed on the image or text link seen on the workstep interface page in your browser. Specify the image file name to be displayed on the workstep interface in the Image box. For images stored outside the Business Process Modeler installation directories (for example, in a Web server folder), specify the full URL in the Image box. Select the Submit checkbox, to ensure that information from dataslots on the current interface page is passed on to the next interface page. If the Submit check box is cleared, then no data is passed on to the next interface page and the link in the process template diagram is displayed as a dashed line, not a solid line.

Note: You can enter a description (optional) for the Link in the Description tab. A description can be up to 4095 characters.

When using Links in Web applications, note that:

- You must select the Submit checkbox for Image types of links in order to pass on data to the next page.
- Activity worksteps that include output dataslots can use a link of the Link type. Image type links can be used only if the Submit check box is selected. Activity worksteps without output dataslots must contain links of the Image type or a Link type with a value.
- Selecting Submit button as the link type for an Activity workstep automatically creates a Submit button in the application interface. You can override the default “Submit” label by entering a label name in the Label box of the Properties view for the link.
Defining a Rollback

Business Process Modeler allows you to add a “rollback” operation to a workstep in a Process Model, which has already been completed earlier in a process; that is, the workflow is restarted from the designated rollback workstep in the process. This process is required to restore the original information and eliminate the numerous changes that can occur to the process between the designated rollback workstep and the currently activated workstep.

To define the Rollback, you must set a workstep as a rollback point and specify which dataslot values are saved. In addition, you have the option of:

- Writing JavaScript code to handle a rollback request.
- Configuring an Activate workstep that, after the rollback is performed, activates the given workstep and returns to the same state as when that workstep was first performed.

In a rollback, the workflow is restarted from the workstep selected as the rollback point in the process. You can only apply a rollback point to Activity, Adapter and Subprocess worksteps in Process Models.

You can define a rollback by using the Compensation Flow type of connector.

Using the Compensation Flow connector

You can use the Compensation Flow connector (see Connecting worksteps on page 53) to define the workflow in case of a rollback.

To define a compensation flow:

1. Click the Connect Shapes link in the Tasks pane, to view the three supported Flow types.
2. Click the Draw compensation flow connector icon ( ) and connect the workstep that initiates the rollback to the rollback point workstep from where the workflow is restarted.
An example of a rollback is shown in Figure 164 on page 245, in which Activity 5 initiates the rollback and Activity 1 is the Rollback Point.

Figure 164: Example of a Process with Rollback and Timeout

Inserting a Compensation Flow automatically makes the target workstep the rollback point. As seen in Figure 164 on page 245, the Rollback Point workstep is marked with a icon and the start of the Compensation Flow link is indicated by the icon.

Defining a Timeout in worksteps

You can use the Timeout feature to indicate the direction of a process workflow after Overdue Actions for the workstep have been executed and the Last Overdue action is completed.

To define a Timeout:

1. Click the Connect Shapes link in the Tasks pane: the three supported Flow types are displayed.
2. Click the Draw time-out flow connectors ( icon and connect the workstep that begins the timeout flow to the next workstep in the timeout workflow.

In the example shown in Figure 164 on page 245, Activity 2 initiates the timeout and Activity 3 is the next step in the Timeout flow. The source workstep of a Timeout Flow is marked with a .

To remove the Timeout Flow, right-click the Timeout Flow link and click Remove.
Managing multiple worksteps

You can perform the same operation on multiple worksteps in a process template using the Select icon in Modeler palette.

Modifying multiple workstep properties

You can simultaneously edit multiple worksteps of the same type. For example, you can select all Activity worksteps in a process template and assign the same performer to all, or change the Priority of all Adapter worksteps at the same time.

To modify multiple worksteps:

1. Click the Select icon in the Modeler palette and drag across the worksteps to be selected. Alternatively, you can press CTRL and then click the worksteps to be selected. The Properties view displays the name as “(multiple worksteps)” in a read-only box.

Figure 165: Modifying multiple worksteps of the same type

2. Modify properties such as the performer, priority, and duration, as required. For instance, change the priority to Critical.

Aligning and spacing worksteps

Business Process Modeler allows you to align selected worksteps and distribute selected worksteps at equal intervals.
To align worksteps or other components in a process template diagram:

1. Use the Select tool to select the worksteps and other components that you want to align.
2. Click the Select and Change Layout link in the Tasks pane, displaying the Layout options in the Layout Tasks panel.

**Figure 167: Layout in Tasks Pane**

![Layout in Tasks Pane](image)

Table 43 on page 247 lists the available layout options.

**Table 43: Layout options**

<table>
<thead>
<tr>
<th>Layout Operation</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Align shapes horizontally</td>
<td>To align the selected components horizontally.</td>
</tr>
<tr>
<td>Align shapes vertically</td>
<td>To align the selected components vertically.</td>
</tr>
<tr>
<td>Distribute shapes evenly on the horizontal</td>
<td>To evenly space the selected components horizontally. If you select more than two components, then only the components between the one on the far left and the one on the far right are moved.</td>
</tr>
<tr>
<td>Distribute shapes evenly on the vertical</td>
<td>To evenly space the selected components vertically. If you select more than two components, then the uppermost and lowermost components are not moved and the other components are evenly distributed in reference to the upper and lower components.</td>
</tr>
<tr>
<td>Auto layout</td>
<td>Generates an ideal layout for process template elements, thus eliminating the need for manual adjustment.</td>
</tr>
</tbody>
</table>
Introducing Monitoring process

Business Process Modeler enables you to design and configure a Monitoring process, which uses business events to monitor an external process on Business Process Server platform. This chapter describes how you can use Business Process Modeler to create a monitoring process.

Note: For detailed description of the working of monitoring process, see Chapter 29: "Working with Monitoring process" in BP Server Developer’s Guide.

For details, see the following topics:

• Designing a monitoring process

Designing a monitoring process

You can create a process model (as discussed in Creating a process model) that can be used as a monitoring process. The following restrictions apply to a monitoring process:

• It must include at least one monitoring workstep. A monitoring workstep can be of type, External activity or adapter, which receive external events for activation and completion.

• It must include a CHARACTER dataslot selected as the EIID, as discussed in Assigning EIID on page 253.

• Any dataslot, associated with a monitoring workstep, must be editable, so that its value can be modified by the external event. For more information, see Using dataslots on page 254.

Note: You cannot design a monitoring process using a Web application project.
Configuring monitoring workstep

Setting workstep properties on page 217 describes how you can add and configure activity, adapter and other workstep types in Business Process Modeler. Similarly, you can add and configure monitoring worksteps for tasks, performed by external users or adapters.

Business Process Modeler supports the following types of monitoring worksteps performed by external participants:

- **External Activity**: This workstep is performed by an external user, specified in the triggering event. In case the event does not include the performer's name, you can configure a default performer for this workstep.
  
  **Note**: The name of the external performer, as well as the default performer, need not be defined in your monitoring process template.

- **External Adapter**: This workstep is triggered and performed by an external system performer.

**To add a monitoring workstep:**

2. From the Business Process Modeler Tasks pane, click the **Assign Participants** link and expand the **External** folder.
   a) To add an external activity workstep, drag the **External User** icon.
   b) To add an external adapter workstep, drag the **External Adapter** icon.

**Figure 168: Adding monitoring workstep**
Similar to activity and adapter worksteps, you can use the **Properties** view to configure workstep properties for the external activity and adapter worksteps.

**Figure 169: External activity workstep properties – General tab**

Step **Figure 169** on page 251 displays the **General** tab of the **Properties** view for an external activity workstep. The Properties view is similar to that for an Activity workstep (as discussed in Defining properties of Activity worksteps on page 224) with the following exceptions:

- From the **General** tab, you cannot assign or change the performer in the **Performer** box. This is because the performer is from the external system and need not exist in the current process template.

- In addition to selecting an individual performer or dataslot, you can use the **Default performer** box to specify a performer, who does not exist in the current process template. For more information, see Default performer for monitoring workstep on page 252.

- If you select the **Wait For Activation** checkbox, then the external activity workstep waits and processes the ACTIVATE_WORKSTEP event before proceeding with the COMPLETE_WORKSTEP event, even if it has arrived. If the **Wait For Activation** checkbox is not selected, then the workflow processes the event which comes first. For example, if COMPLETE_WORKSTEP event comes before ACTIVATE_WORKSTEP event, then it is processed and the ACTIVATE_WORKSTEP event is moved to invalid event table.

- The **General** tab does not contain Priority, Presentation, Skippable, and Collect Work Time fields, as they are not applicable to a monitoring workstep.

- You can use the **Dataslots** tab to add dataslots. For more information, see Using dataslots on page 254.

- There is no **Collaboration** tab, because it is not required in a monitoring workstep.
Note: You can configure the remaining external activity properties, as discussed in Defining properties of Activity worksteps on page 224.

Figure 170: External adapter workstep properties – General tab

Step Figure 170 on page 252 displays the General tab of the Properties view for an external adapter workstep. The Properties view is similar to that for an external activity workstep. The only exception is that there is no default performer, because this field is not applicable for an adapter workstep.

Default performer for monitoring workstep

You can specify a default performer for an external activity workstep by clicking the ellipsis button beside the Default performer box in the General tab and then performing any of the following operations by selecting the appropriate option:

• To assign an existing performer:
  1. Select the Select a Performer option. The Performer dialog box appears, which lists all the performers in the current process.
  2. Select any of the performers listed and click OK to assign the selected user as the default performer.

• You can create and assign a new performer as a default performer. To create and assign a new performer:
  1. Select the Select a Performer option.
  2. From the Performer dialog box, click New to open the Performer dialog box.
  3. Enter the name of the performer in the Name box.
     Alternatively, click the ellipsis button beside the Name box, then select either of the following options.

Table 44: Performer options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select an Existing User</td>
<td>To add a user defined in the User Management tool (see Working with User Management tool on page 381).</td>
</tr>
<tr>
<td>Use a Dataslot</td>
<td>To select a CHARACTER dataslot as the performer.</td>
</tr>
</tbody>
</table>
4. Click OK to add the selected performer to the list of performers in the Performer dialog box.
5. Click OK to assign the new performer as the default performer.

- You can add a performer that does not exist in the process template.

1. Select the Enter a new value option
2. From the Default Performer dialog box that appears, type the name of the performer.
3. Click OK to assign the specified performer as the default performer.

To specify another performer, click the ellipsis button beside the Name box, and select the Modify value option. Modify the default performer name in the Default Performer dialog box.

To remove the selected default performer, click the ellipsis button beside the Default performer box, and select the Clear option.

Assigning EIID

You can assign an EIID to the monitoring process after adding a monitoring workstep. You can assign only one dataslot as the EIID.

The dataslot assigned as the EIID must satisfy the following requirements:

- Dataslot type must be CHARACTER.
- Value must be unique across all instances of the monitoring process.
- Storage type must be VARCHAR with size 128.

To assign the EIID to a monitoring process:

1. Click a blank area of the Diagram tab to open the Properties view. For details regarding process properties, see Setting process properties on page 89.

The Properties view displays the additional Monitoring tab.

Figure 171: Process Properties – Monitoring tab

2. Click the ellipsis button beside the External Instance Id Dataslot box and then select one of the following options:

- **Use a User Dataslot**: The Select Dataslot dialog box that appears lists the user-defined CHARACTER dataslots in this process. Select a dataslot, then click OK to assign it as the EIID. Alternatively, you can click New to create a CHARACTER dataslot and then assign it as the EIID.
- **Use ProcessInstanceId**: Assigns the system dataslot, “ProcessInstanceId,” as the EIID.

The assigned dataslot is displayed in the External Instance Id Dataslot box. To clear this selection, click the ellipsis button beside the External Instance Id Dataslot box and then select the Clear option.
Using a subprocess

In some cases, you may need to use multiple EIIIDs in the same monitoring process. For example, an “Order” process requires a unique Order ID during ordering, as well as Shipment ID during product shipment.

You can handle this requirement by using a subprocess. In the above example, the subprocess can handle the shipment process. For more information regarding subprocesses, see Defining a subprocess performer on page 122.

Note: You cannot use an inline subprocess to handle this requirement.

As the subprocess is an independent process model, it can have its own EIID (for instance, ShipmentID), which can be mapped to a dataslot from the parent process. If required, then the parent process can pass on its EIID (example, OrderID) to the subprocess at the time of its creation.

Important: Ensure that the value of the dataslot in the parent process, mapped to the EIID of the subprocess, is not null (or empty); otherwise, the subprocess instance is suspended.

Using dataslots

Table 45 on page 254 lists the dataslot types that you can assign to a monitoring workstep in the monitoring process. A business event only updates those dataslots assigned to monitoring worksteps.

Table 45: Supported dataslots

<table>
<thead>
<tr>
<th>Dataslot</th>
<th>Java Type</th>
<th>Comments (if any)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHARACTER</td>
<td>String</td>
<td>Any string length longer than the maximum length specified for the process template, is truncated.</td>
</tr>
<tr>
<td>INTEGER</td>
<td>Double, BigDecimal</td>
<td></td>
</tr>
<tr>
<td>INT64</td>
<td>Long</td>
<td></td>
</tr>
<tr>
<td>LOGICAL</td>
<td>Boolean</td>
<td></td>
</tr>
<tr>
<td>DATETIME-TZ</td>
<td>Long</td>
<td>Date must be specified in milliseconds.</td>
</tr>
<tr>
<td>LIST</td>
<td>List</td>
<td></td>
</tr>
<tr>
<td>Map</td>
<td>Map</td>
<td></td>
</tr>
</tbody>
</table>

Note: Dataslots of the type Object and Business Object are not supported for a monitoring process.

The following business events can modify the value of the dataslots in the monitoring process instance.

- UPDATE_DATASLOT: This event can update the value of any user-defined dataslot (of the type listed in Table 45 on page 254) in the monitoring process, including those dataslots not added to the monitoring workstep. This event identifies the process instance using the EIID
value, before updating the dataslot values. If the value of the dataslot passed on by the event is invalid, then the dataslot update fails.

- **COMPLETE_WORKSTEP**: This event can only update the values of the dataslots (of the type listed in Table 45 on page 254) added to the monitoring workstep. This event triggers the completion of the monitoring workstep, only after updating the dataslot values. If the value of the dataslot passed on by the event is invalid, then the dataslot update fails.

**Note**: The UPDATE_DATASLOT and COMPLETE_WORKSTEP events cannot update the EIID value.

**To add a dataslot to a monitoring workstep:**

1. From the **Properties** view of the workstep (external activity or adapter), click the **Dataslots** tab.

**Figure 172: Monitoring workstep properties – Dataslots tab**

2. Click **Add** to open the **Select Dataslots** dialog box, which lists all the user-defined dataslots in the monitoring process.

**Note**: You must add only editable dataslots to a monitoring workstep.

3. Select the dataslot to be added, then click **OK** to add the dataslot to the table provided in the **Dataslots** tab.
Defining workstep presentation format

Business Process Modeler allows you to change the presentation format of Start and Activity worksteps. The presentation formats available are:

- **Portal**:
  - **Auto-generated**: This is the default HTML presentation form, automatically generated at runtime in Business Process Portal.
  - **Form**: Enables you to design the presentation using the Form Editor wherein you can create more complex interfaces with multiple columns, radio buttons and other formatting, and bind these fields to specific dataslots. You can then load and save the new form.
  - **Custom**: Enables you to define a customized Java Server Page (JSP) which is generated from an XSL transformation.
  - **Flow**: Available only for BPM processes. Enables you to start and design a Web application from a BPM process.
  - **Flow (private)**: Available only for BPM processes. This is the same as the Flow format but can be used only in the process template in which it is placed.

- **Mobile**: Available only for BPM processes. Enables you to complete an activity workstep (in a BPM process) on a Mobile device.

- **GUI**: Available only for BPM processes. Enables you to complete an activity workstep from a GUI for .NET desktop application.

- **Tablet**: Available only for BPM processes. Enables you to complete an activity workstep on a Tablet.

- **Other**: Available only for BPM processes. You choose Other to complete an activity workstep using an interface other than the options listed above. For example, if you want to complete an
activity workstep on a iPad, you select Other as the presentation type for the activity in BPM process.

This chapter describes how to define presentation (or interface) forms for Process Models. Similar to Process Models, you can define the look and feel of the presentation pages for Activity worksteps in Web applications. For Web applications, you cannot change the presentation format for the Start workstep.

After you have developed a Web application, you can view it as a series of presentation pages (or interfaces) in your Web browser by entering the appropriate URL. For example, to view the BarChartDemo Web application, enter the following URL:

http://machine_name.<domain_name>:port_number/sbm/BPM Workflow/BarChartDemo/Start.jsp.

For details, see the following topics:

- **Defining workstep presentation**

---

**Defining workstep presentation**

You can change the look and feel of the presentation pages for Start or Activity worksteps in Business Process or Web application. OpenEdge BPM Designer allows you to add multiple presentation types such as Portal, Mobile, GUI, and Tablet to a single activity workstep. This enables you to complete an activity from a browser, a Mobile device, a GUI for .NET application, or a Tablet device.

**Note:** The presentation types Mobile, GUI, Tablet, and Other are available only for BPM processes.

**To define the presentation format:**

1. Click the Start or Activity workstep in the process template to display its properties in the Properties view.
2. From the General tab, do one of the following:
   - Click the ellipsis button beside the Presentation box for an activity in a Web application.
   - Click the Add button from the Presentation section for an activity in a BPM process.

   The Add Presentation dialog box appears.
3. Select the presentation type option from the Type drop-down list. The options are: Portal, Mobile, GUI, Tablet, and Other for BPM process models. For Web applications, the presentation type is selected as Portal in the Type drop-down list and is disabled.
4. Select the presentation format option from the Form Type drop-down list. The options are: Auto-generated, Form, Custom JSP, Flow, and Flow (private) for BPM process models. For Web applications, the options are: Auto-generated, Form, and Custom.

   **Note:** The Form Type drop-down is not available when you select the Mobile, GUI, Tablet, or Other from the Type drop-down list.

5. Modify (if required) the presentation name in the Name box. You cannot specify a name for the Auto-generated format.
Alternatively, you can map worksteps (with Custom JSP and Flow presentations) to CHARACTER dataslots to dynamically associate the workstep with a JSP file or a Web application respectively. You need to set the initial value of the CHARACTER dataslot to a Web application (*.BIZSOLO) or a valid JSP file (existing in the <project_name>/jsp folder). This feature is not available for Web applications.

a) From the Form Type drop-down list, select the appropriate option. The available options are Custom JSP, Flow, and Flow (private). The Form Type drop-down is not available for Mobile, GUI, Tablet, or Other as the presentation types.

b) Click the ellipsis button beside the Name or URL text field, then select Use a Dataslot option to display the Add Dataslots dialog box. Select the CHARACTER dataslot to be mapped to the workstep presentation, then click OK to add the dataslot in the Name or URL text field.

6. Click OK to return to the Properties view.

Note: You cannot add the same presentation type twice to one activity workstep. If you do, an error message appears stating that the Presentation Type already exists. Do you want to override? Click OK to override the presentation type.

For more details regarding each of the presentation formats available for Process Models, refer to Table 46 on page 259.

Table 46: Presentation Format for BPM project components

<table>
<thead>
<tr>
<th>Presentation Type</th>
<th>Presentation Format</th>
<th>See</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portal</td>
<td>Auto-generated</td>
<td>Defining Auto-generated HTML presentations on page 260.</td>
</tr>
<tr>
<td></td>
<td>Form</td>
<td>Using the Form presentation on page 263.</td>
</tr>
<tr>
<td></td>
<td>Custom</td>
<td>Defining custom presentations on page 260.</td>
</tr>
<tr>
<td></td>
<td>Flow</td>
<td>Using a Flow presentation on page 261.</td>
</tr>
<tr>
<td></td>
<td>Flow (private)</td>
<td>Using a Flow (private) presentation on page 262.</td>
</tr>
<tr>
<td>Mobile</td>
<td>-</td>
<td>Defining Mobile presentation on page 263.</td>
</tr>
<tr>
<td>GUI</td>
<td>-</td>
<td>Defining GUI presentation on page 264.</td>
</tr>
<tr>
<td>Tablet</td>
<td>-</td>
<td>Defining Tablet presentation on page 266.</td>
</tr>
<tr>
<td>Other</td>
<td>-</td>
<td>Defining Other presentation on page 267.</td>
</tr>
</tbody>
</table>
Defining Auto-generated HTML presentations

Business Process Modeler provides a default HTML presentation for Start and Activity worksteps, automatically generated at runtime in Business Process Portal. For instance, the Assignment sample application uses the auto-generated format.

Figure 173: Default Presentation Format for the Assignment Application, as seen in Business Process Portal

Business Process Portal uses the Header options and dataslots specified in the workstep's Properties view in Business Process Modeler. In addition to Header options that include the Instruction and Priority fields directly under the heading, Business Process Portal also displays fields corresponding to the read-only and editable dataslots specified in the Fields tab in the workstep's Properties view. In the example in Figure 173 on page 260, the Task name, Assignee, Priority, Due Date, and Instructions boxes are editable; that is, you can enter data in these fields and the data is passed on to the next workstep.

Defining custom presentations

The Custom presentation form for Process Models can be used with HTML forms to add custom JSP presentations. Like HTML presentations, the presentation of the JSP forms can be edited using an HTML editor that supports JSP tags. In addition, the logic part of the JSP (implemented in Java) can be modified to customize the content of the presentation using any text editor. For example, it is possible to add Java code to populate the content of a drop-down list from a database query.

If you select the Custom option in the Presentation dialog box in the Start or Activity worksteps properties, then Business Process Modeler creates a Java Server Page (JSP) workstep presentation that is generated from an XSL transformation. This type of JSP presentation enables you to change the interface for all worksteps using this option by making a single change in the XSL transformations.
Defining Flow presentations

You can use the Flow presentation (available only for Process Models) to execute a Web application without opening it from Business Process Portal. You need to only click on a link provided in the interface to launch a Web browser with this format. Business Process Modeler provides two Flow options in the Presentation dialog box:

- **Flow** starts a Web application from a workstep in the Process Model by redirecting the Web application to the BPM Workflow server (see Using a Flow presentation on page 261).

- **Flow (private)** creates a workstep that is restricted to the current process template and is not redirected to the BPM Workflow server. You must copy the files in the Web application to the webapp folder in Business Process Modeler. For more information, see Using a Flow (private) presentation on page 262.

Using a Flow presentation

You can define an Activity workstep (in a Process Model) with Flow presentation type to start a Web application. As this workstep runs in a Web browser, the Web application it calls must be published and installed.

To define and use a Flow presentation for a workstep:

1. From the General tab of the workstep’s Properties view, click the ellipsis button beside the Presentation box to open the Presentation dialog box.

   ![Figure 174: Setting the Format for a Workstep to a Flow Presentation](image)

2. Select the Flow option from the Type drop-down list.

3. You can modify the name of the default Flow interface in the Name box. The default name is the name of the current workstep followed by the *.BIZSOLO extension.

   The .BIZSOLO extension indicates to the servlet that it should invoke the appropriate methods of the BPM Workflow server to get the URL for the BPM Workflow server and redirect the Web application to that page. The BPM Workflow server implements a dispatcher that takes as input the Web application name and produces the URL of the Start workstep of the Web application.

4. Alternatively, you can associate a dataslot with the Flow workstep. Click the ellipsis button beside the Name box, then select the Use a dataslot option to select the dataslot from the Use a Dataslot dialog box. Click OK to add the dataslot.

5. Click OK to define the Flow presentation format for this workstep.

   After defining the presentation format of the workstep as Flow, you can now create the Web application.
6. Right-click the workstep with the Flow presentation, then click **Open** to launch the **Open to WebApp** dialog box.

**Figure 175: Opening the Flow presentation workstep**

7. Select either an existing web application (in **Existing** folder) or **Web Application Project**.
8. Click **OK**. If you select an existing web application (in **Existing** folder), the process template of this web application is displayed. Alternatively, if you select **Web Application Project**, the New Web application wizard (see Step Figure 175 on page 262) is launched, using which you create a Web application.

**Note:** You can also check in the Flow presentation associated with a workstep to the Process repository, as described in Saving a process to the process repository. After refreshing the Repository Browser, the process created from a Flow presentation is seen in the list of models under the Models tab.

### Using a Flow (private) presentation

Unlike the Flow presentation type, the Flow (private) presentation type does not require you to publish and install the Web application in order to execute properly in its process template. Additionally, upon subsequent modifications to the input or output dataslots of the workstep in a Process Model with a Flow (private) presentation, the dataslots of the private flow are automatically updated to reflect the input and output changes in the workstep. This results in the automatic dataslot synchronization between Process Models and Flow (private) applications. Removing dataslots from the Process Model does not remove the corresponding dataslots in the Flow (private) application, but is interpreted as disassociating the dataslots in the Process Model from the dataslots in the Flow (private) application.

**Important:** If an application is used as a Flow (private) application in Process Modeler, then you cannot use it as a stand-alone Web (BPM Workflow) application; similarly, an installed Web (BPM Workflow) application cannot be used as a Flow (private) workstep.

### To create a Flow (private) presentation for a workstep:

1. From the **General** tab of the workstep’s **Properties** view, click the **ellipsis** button beside the **Presentation** box to open the **Presentation** dialog box.
2. Select the **Flow (private)** option from the **Type** drop-down list.
3. Modify, if required, the default name in the **Name** box. The default name is the name of the current workstep.

   Alternatively, you can associate a dataslot with the Flow (private) workstep. To do so:
   a) Click the **ellipsis** button beside the **Name** box.
   b) Select the **Use a dataslot** option to select the dataslot from the **Use a Dataslot** dialog box.
   c) Click **OK** to add the dataslot.
4. Click OK to define the presentation format for this workstep.
5. Right-click the workstep and click Open to open a Web application with the same name as the Flow (private) workstep.

The Flow (private) workstep is now confined to the current process template.

**Using the Form presentation**

You can also use the Form presentation format for Process Models and Web applications to use the Form Editor, where you can create a customized interface form. The Form Editor option enables you to create a more complex presentation form for each Activity workstep (and for Start worksteps only in Process Models), including headers, multi-column tables, adding formatting (text fields, drop-down lists, images, and so on), and the ability to bind dataslots to a specific presentation item. For details regarding using Form Editor, see Using Form Editor on page 275.

**Defining Mobile presentation**

You can define an activity workstep (in a BPM process) with presentation type as Mobile to complete the activity using a Mobile device. As the activity runs in a Mobile device, you must install a mobile app in the device. Then, you must map the mobile app's URL to the activity workstep (in a BPM process) to access the activity.

**Note:** You can build and install mobile apps for Mobile using Progress OpenEdge Mobile Application Builder (Mobile App Builder). For more information, see the OpenEdge Development: Mobile Applications guide and Progress Developer Studio for OpenEdge help.

To define a Mobile presentation for an activity workstep:

1. Select an activity workstep in a BPM process for which you want to define the presentation type as Mobile.
2. From the Properties view, click the General tab to view the Presentation section.

**Figure 176: Presentation section**

<table>
<thead>
<tr>
<th>Type</th>
<th>Metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portal</td>
<td>Form (Activity1.jsp)</td>
</tr>
<tr>
<td>Mobile</td>
<td><a href="http://localhost:8080/mobile/activity.jsp">http://localhost:8080/mobile/activity.jsp</a></td>
</tr>
</tbody>
</table>

**Note:** By default, the Form (Activity_name.jsp) presentation type is defined for the activity workstep that is displayed in the Presentation section.
3. Click **Add** to open the **Add Presentation** dialog box.

   **Figure 177: Add Presentation dialog box**

   ![Add Presentation dialog box](image)

4. Select Mobile from the **Type** drop-down list.

5. Specify the mobile app file path or URL in the **URL** text field. For example, `https://<host>:<port>/mobile/activity.html`.

   Alternatively, you can map workstep with Mobile presentation format to CHARACTER dataslot to dynamically associate the workstep with a mobile app URL. You need to set the initial value of the CHARACTER dataslot to a valid mobile app URL.

   a) Click the **ellipsis** button beside the **URL** text field.
   b) Select **Use a dataslot** to open the **Add Dataslots** dialog box.
   c) Select a CHARACTER dataslot to be mapped to the workstep presentation.
   d) Click **OK** to add the dataslot and then close the **Add Dataslots** dialog box.

6. Click **OK** to add the Mobile presentation format for the activity workstep.

   **Note**: You cannot add the same presentation type twice to one activity workstep. If you do, an error message appears stating that the **Presentation Type already exists. Do you want to override?**. Click **OK** to override the presentation type.

The Mobile presentation format is mapped to the selected activity workstep. The defined Mobile presentation type is displayed in the **Presentation** section.

   **Note**: You can add multiple presentation types such as Portal, Mobile, GUI, and Tablet to a single activity workstep. This enables you to complete an activity from a browser, a Mobile device, a GUI for .NET application, or a Tablet device.

### Defining GUI presentation

You can define an activity workstep (in a BPM process) with presentation type as GUI to complete the activity using a GUI for .NET desktop application. As the activity runs a GUI for .NET desktop application, you must develop a OpenEdge GUI for .NET application. Then, you must map the GUI for .NET application name (ABL class .cls filename) to the activity workstep (in a BPM process) to access the activity.

   **Note**: For more information on developing OpenEdge GUI for .NET applications, see the *OpenEdge Getting Started: Introducing the Progress Developer Studio for OpenEdge Visual Designer* guide.
To define a GUI for .NET presentation for a workstep:

1. Select an activity workstep in a BPM process for which you want to define the presentation type as GUI.

2. From the Properties view, click the General tab to view the Presentation section.
   
   Figure 178: Presentation Types section

   Note: By default, Portal type with Form (Activity_name.jsp) presentation format is listed in the table on the Presentation section.

3. Click Add to open the Presentation Types dialog box.
   
   Figure 179: Add Presentation dialog box

4. Select GUI from the Type drop-down list.

5. Specify an ABL class (.cls) file in the Name field. For example, SampleApp.cls.
   
   Alternatively, you can map workstep with GUI presentation format to CHARACTER dataslot to dynamically associate the workstep with an ABL .cls filename. You need to set the initial value of the CHARACTER dataslot to a valid ABL .cls filename.
   
   a) Click the ellipsis button beside the Name text field.
   
   b) Select Use a dataslot to open the Add Dataslots dialog box.
   
   c) Select the CHARACTER dataslot from the Add Dataslots dialog box.
   
   d) Click OK to add the dataslot and then close the Add Dataslots dialog box.

6. Click OK to add the Mobile presentation format for the activity workstep.
   
   Note: You cannot add the same presentation type twice to one activity workstep. If you do, an error message appears stating that the Presentation Type already exists. Do you want to override? Click OK to override the presentation type.

The GUI presentation format is mapped to the selected activity workstep. The defined GUI presentation type is displayed in the Presentation section.

Note: You can add multiple presentation types such as Portal, Mobile, GUI, and Tablet to a single activity workstep. This enables you to complete an activity from a browser, a Mobile device, a GUI for .NET application, or a Tablet device.
Defining Tablet presentation

You can define an activity workstep (in a BPM process) with presentation type as Tablet to complete the activity using a Tablet device. As the activity runs in a Tablet device, you must install a mobile app in the device. Then, you must map the mobile app's URL to the activity workstep (in a BPM process) to access the activity.

**Note:** You can build and install mobile apps for Tablet using Progress OpenEdge Mobile Application Builder (Mobile App Builder). For more information, see the *OpenEdge Development: Mobile Applications* guide and *Progress Developer Studio for OpenEdge* help.

To define a Tablet presentation for an activity workstep:

1. Select an activity workstep in a BPM process for which you want to define the presentation type as Tablet.
2. From the **Properties** view, click the **General** tab to view the **Presentation** section.

   **Note:** By default, the Form (`Activity_name.jsp`) presentation type is defined for the activity workstep that is displayed in the **Presentation** section.

3. Click **Add** to open the **Add Presentation** dialog box.

   **Figure 180: Add Presentation dialog box**

   ![Add Presentation dialog box]

4. Select Tablet from the **Type** drop-down list.
5. Specify the mobile app file path or URL in the **URL** text field. For example, `https://<host>:<port>/tablet/activity.html`.
   Alternatively, you can map workstep with Tablet presentation format to CHARACTER dataslot to dynamically associate the workstep with a mobile app URL. You need to set the initial value of the CHARACTER dataslot to a valid mobile app URL.
   a) Click the **ellipsis** button beside the **URL** text field.
   b) Select **Use a dataslot** to open the **Add Dataslots** dialog box.
   c) Select a CHARACTER dataslot to be mapped to the workstep presentation.
   d) Click **OK** to add the dataslot and then close the **Add Dataslots** dialog box.
6. Click **OK** to add the Tablet presentation format for the activity workstep.

   **Note:** You cannot add the same presentation type twice to one activity workstep. If you do, an error message appears stating that the **Presentation Type already exists. Do you want to override?**. Click **OK** to override the presentation type.
The Tablet presentation format is mapped to the selected activity workstep. The defined Tablet presentation type is displayed in the **Presentation** section.

**Note:** You can add multiple presentation types such as Portal, Mobile, GUI, and Tablet to a single activity workstep. This enables you to complete an activity from a browser, a Mobile device, a GUI for .NET application, or a Tablet device.

---

### Defining Other presentation

You can define an activity workstep (in a BPM process) with presentation type as Other to complete the activity using a device other than the defined presentation types such as Portal, Mobile, GUI, or Tablet.

**To define a Tablet presentation for an activity workstep:**

1. Select an activity workstep in a BPM process for which you want to define the presentation type as Other.
2. From the **Properties** view, click the **General** tab to view the **Presentation** section.

   **Note:** By default, the Form (Activity_name.jsp) presentation type is defined for the activity workstep that is displayed in the **Presentation** section.

3. Click **Add** to open the **Add Presentation** dialog box.

   **Figure 181: Add Presentation dialog box**

4. Select **Other** from the **Type** drop-down list.
5. Specify the filename in the **Other** text field.

   Alternatively, you can map workstep with Other presentation format to CHARACTER dataslot to dynamically associate the workstep with a file. You need to set the initial value of the CHARACTER dataslot to a valid file that you want to use to complete the activity.

   a) Click the **ellipsis** button beside the **URL** text field.
   b) Select **Use a dataslot** to open the **Add Dataslots** dialog box.
   c) Select a CHARACTER dataslot to be mapped to the workstep presentation.
   d) Click **OK** to add the dataslot and then close the **Add Dataslots** dialog box.

6. Click **OK** to add the Other presentation format for the activity workstep.

**Note:** You cannot add the same presentation type twice to one activity workstep. If you do, an error message appears stating that the **Presentation Type already exists. Do you want to override?** Click **OK** to override the presentation type.
The Other presentation format is mapped to the selected activity workstep. The defined Other presentation type is displayed in the Presentation section.

**Note:** You can add multiple presentation types such as Portal, Mobile, GUI, and Tablet to a single activity workstep. This enables you to complete an activity from a browser, a Mobile device, a GUI for .NET application, or a Tablet device.
Using alerts

Business Process Modeler provides the Alerts functionality for process models, which you can define and associate with selected worksteps. Using Alerts, you can notify application users in BPM Portal of a predefined event, such as change in the status of a process instance or in case of an exceptional condition being satisfied. In Business Process Modeler, you can use the Alerts tab to define a pool of alerts that can be associated with selected worksteps. These alerts can be associated with selected worksteps using the Alerts tab in the workstep’s Properties view, at various transition points (such as, “On activation” and “When completed”) during execution of the process.

The associated alerts appear in Business Process Portal under the My Alerts tab when, for example, a specific workstep is activated or when it is completed. Each alert must contain a message. Alerts can be assigned to one or more specified recipients or they can be triggered by a particular condition. For information on managing alerts in Business Process Portal, see the Business Process Portal User’s Guide.
For details, see the following topics:

- Reviewing the Alerts tab
- Defining an alert
- Associating an alert with a workstep

## Reviewing the Alerts tab

You can view the "default" alert and define new alerts in the **Alerts** tab of the Modeler interface (Figure 2 on page 21).

### Figure 182: Alerts tab

Click the **Alerts** tab for a new process model, to view the **All Alerts** section, which contains all the alerts available to the process template.

You can perform the following operations in the **Alerts** tab:

### Table 47: Alerts tab operations

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define an alert</td>
<td>See Defining an alert on page 271.</td>
</tr>
<tr>
<td>Remove an alert</td>
<td>Select the alert to be removed, then click Remove. You cannot remove the &quot;default&quot; alert.</td>
</tr>
<tr>
<td>Modify existing alerts</td>
<td>Select the alert to be modified, then modify its properties in the respective sections of the <strong>Alerts</strong> tab.</td>
</tr>
<tr>
<td>Filter the alert list</td>
<td>Enter text in the type filter text box to filter the alert list and display all alerts whose names contain the matching letters.</td>
</tr>
<tr>
<td>Operation</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>Group (or ungroup) the alert list</td>
<td>By default, all alerts are grouped under alert severity categories, namely, Critical, High, Medium, and Low. To ungroup the alert list, click the No Grouping icon in the All Alerts section. To group the performer list under severity categories, click the Group by Severity icon in the All Alerts section.</td>
</tr>
<tr>
<td>Collapse the alert list</td>
<td>Click the Collapse All icon.</td>
</tr>
</tbody>
</table>

**Defining a new alert**

You can define new alerts in the Alerts tab.

**To define an alert:**

1. Load the Process Model (SPT) for which you want to define the alert.
2. From the Alerts tab (Figure 182 on page 270), click Add. The New Alert wizard appears.

**Figure 183: New Alert wizard: Page 1**

3. From the first page of the New Alert wizard, you can specify the alert name, severity, and description of the alert.
   a) Type the name of the alert in the Name box. The name can only include alphanumeric characters and underscore (“_”). Do not start the alert name with a numeric character. The name for an Alert should not exceed 64 characters.
   b) Select an option from the Severity drop-down list; options are: Critical, High, Medium, and Low. The Severity indicates to the alert recipients the urgency with which they should respond to the Alert.
   c) Enter a description (optional) of the alert in the Description box. This will enable other users of the application in Progress Developer Studio for OpenEdge to determine the intent of the alert.
4. Click **Next**. From the second page of the **New Alert** wizard, you can define a condition that activates the alert when the condition is satisfied.
   
a) To define an alert condition, click **Condition** to open the **Condition** dialog box.
b) Select a dataslot from the drop-down list on the left. Only CHARACTER, INTEGER, INT64, LOGICAL, and DATETIME-TZ dataslot types are listed.
c) Select an operator. To complete defining the condition, select a dataslot from the drop-down list on the right or enter a value.
d) To add another condition, click ![Condition](image) and select **And** or **Or** option from the drop-down list. To remove a condition, click ![Remove](image).
e) Click the **Advanced** option to go to the advanced mode where you can create a condition by adding specific dataslots or by entering (or copy-paste) a condition.
f) To verify the expression's validity, click **Validate** in basic or advanced mode, and the “Validation successful” message appears.

5. Click **Next**. You can use the third page of the **New Alert** wizard to specify the alert message.

   In the **Message** box, enter the content of the message that you want the alert recipients to see when they open the Alert. You can enter static or dynamic values in the message. For dynamic values, you can use dataslots (${dataslot_name} or ${@System.dataslot}). Press CTRL+SPACE to view the list of dataslots in the process. Select the dataslot to be added and press ENTER to add the dataslot in the **Message** box. For example, a message reading “Vacation request of ${days} has been filed by ${@CREATOR}” will use the days dataslot (a dataslot in the User category) and the @CREATOR dataslot (a System category dataslot) to obtain dynamic values for the number of days and the name of the person who filed the vacation request at runtime.

   **Note:** If you use a System dataslot, it must be prefixed with “@”.

6. Click **Next** to define the recipients of the alert.
   
a) Click **Add** to open the **Performer** dialog box, which lists performers of type, “User” and “Group” (group of sub-type, “All”) in the process.

   **Note:** You cannot add queues, groups of type “Any,” or comma-separated lists of multiple performers as alert recipients.

b) Select a single (or multiple) performer and click **OK** to assign your selected user or group as the alert recipient.

   To delete a recipient, select the recipient and click **Remove**.

7. Click **Finish** to add the defined alert to the **Alerts** tab.

**Associating an alert with a workstep**

After defining the alert in the **Alerts** tab, you must associate the alert with a workstep at various workstep transition points (such as, “On activation” and “When completed”) during execution of the process. You can associate an alert with a Start, End, Activity, Adapter, or Subprocess workstep. In the case of the Start workstep, you can associate the alert only on completion of the workstep.
To associate an alert with a workstep:

1. From the workstep's Properties view, open the Alerts tab.

The Alerts tab displays the On Activation and When Completed sections. For the Start workstep, the When Completed section is only displayed.

Figure 184: Properties – Alerts Tab

2. To associate the alert with the workstep when the workstep is initially activated, add the alert to the table in the On Activation section.

To associate the alert when the workstep is completed, add the alert to the table in the When Completed section.

3. Click Add to display the Add dialog, which lists all the available Alerts.
   a) Select an Alert from the drop-down list provided.
   b) Select the Send SMS check box, if you want to send the alert message as an SMS to the alert recipients. The alert message is sent to the recipient's phone number, as registered on Business Process Portal.

   Note: To enable using SMS for sending alerts, the Business Process Server property oebps.sms.enable must be set to TRUE in the oebps.conf configuration file. In addition, you can set the oebps.sms.failonerror property to TRUE for suspending a workstep when SMS sending fails. The oebps.conf file is located in the OEBPS_HOME\conf directory.

   Before you use the SMS option, you must update the oebpssms.properties file in the OEBPS_HOME\conf directory with the SMS gateway provider details.

   The Alert is added to the table in the Properties view.
Using Form Editor

Business Process Modeler provides a Form Editor, which enables you to design a customized Form interface right from scratch. This option is available for all Activity worksteps and for Start workstep (only in case of Process Models). You can use Form Editor to develop customized forms with:

- Layout components including multi-column tables, tabbed panes, and dividers.
- Standard HTML controls including text fields, drop-down lists, images, trees, grids, and buttons.

For a workstep with designed form interface, the corresponding task (at runtime) renders the designed form interface with all the added form elements.

This chapter describes how to use the Form Editor to design a form interface for worksteps with "Form" presentation format. For defining the presentation format of a workstep, see Defining workstep presentation format on page 257.

The instructions provided in this chapter are equally applicable to worksteps in BPM processes and to Web applications.
For details, see the following topics:

- Opening the Form Editor
- Using a form template
- Using the Layout tab
- Defining the form’s layout
- Defining the form’s controls
- Defining the form’s flow
- Defining the form’s data sources
- Using form fragments
- Other Form Editor operations
- Using the Overview tab
- Configuring actions
- Using the Graphical Event Logic Editor
- Using Script tab
- Managing forms
- Invoking server side methods

Opening the Form Editor

To open the Form Editor, you must enable the Form interface for a workstep by selecting the Form presentation format from the Presentation box (Defining workstep presentation on page 258).

1. From the General tab in the Properties view of an Activity workstep (or for a Start workstep in a Business Process Model), click the ellipsis or Add button beside the Presentation box to open the Add Presentation dialog box.
2. Select the Portal option from the Type drop-down list.

Note: For a Web Application, the Portal option is selected by default in the Type drop-down list and is disabled.
3. Select the **Form** option from the **Form Type** drop-down list, and modify the default name (if required) for the form in the **Name** box, then click **OK**. If you do not enter a name, then the form is assigned the name of the workstep (as in *Activity1.jsp*).

The name appears in the Presentation box of the **Properties** view.

4. Right-click the workstep to which you have enabled the Form Editor option, then click **Open Form** to open a new file tab with the workstep name.

**Figure 185: Form Editor**

![Form Editor interface](#)

*Figure 185* on page 277 identifies the commonly used panes and components in the Form Editor interface, which are further described in the following table.

**Table 48: Form Editor components**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Form Editor Toolbar</td>
<td>Contains icons providing shortcuts to the following commonly used functions:</td>
</tr>
<tr>
<td></td>
<td>- <strong>New Form</strong> (icon) to create a new form or to open an existing form template from the current process or from the Process Repository (see Using a form template on page 278).</td>
</tr>
<tr>
<td></td>
<td>- <strong>Preview</strong> (icon) to open a window that provides a preview of the current form or of all your forms. For more information, see Previewing forms on page 335.</td>
</tr>
<tr>
<td></td>
<td>- Text-formatting icons, that enable you to: define text font, size and style (Bold, Italics, Underline); align text to the right, left or center; and a <strong>Font Color</strong> icon to define the color of the text.</td>
</tr>
<tr>
<td>Content pane</td>
<td>Contains the main work space where you can add the various elements of the form, including dataslots, tables, command buttons, and form controls (such as text box, drop-down list, combo box, and link).</td>
</tr>
</tbody>
</table>
You can save the form with an SFT (OpenEdge Form Template) extension or as an XML file. If you want to reuse the saved form as a template, click File > Save As to save the form to the Forms Library. By default, the form is saved to the Workspace_Home\.com\.savvion\.studio\forms\bizlogic\library folder (for a BPM Process) and Workspace_Home\.com\.savvion\.studio\forms\bizsolo\library folder (for a Web application). Once saved, the form then appears in the File tab, and is available to all your Business Process Modeler applications.

### Using a form template

You can use the default form (as shown in Figure 186 on page 279) or use a previously designed form as a template. Business Process Modeler provides pre-designed forms that you can use to replace the default form.
To use a form template:

- From the Form Editor toolbar, click the **New Form** icon to open the **New Form** dialog box.

**Figure 186: New Form dialog box – File Tab**

The **New Form** dialog box contains the following two tabs:

**Table 49: New form tabs**

<table>
<thead>
<tr>
<th>Tab name</th>
<th>Description</th>
</tr>
</thead>
</table>
| **File** | Click **Browse** to open the library folder (right image, Figure 186 on page 279), which displays three predefined forms for— **Header and Footer** to open a form with default header and footer, **New Blank Form** to open a blank form, and **Purchase Order** to display a sample purchase order template—that are available to all Form-enabled worksteps in Business Process Modeler. Additionally, it may contain any forms that you design and save as templates. Click the *.*.SFT file to be used, then click **Open**. **Note:** The location of the predefined forms is 

```
Workspace_Home\.com.savvion.studio\forms\bizlogic\library folder (for a BPM Process) and 
Workspace_Home\.com.savvion.studio\forms\bizsolo\library folder (for a Web application).
```
| **Process** | Contains forms (if any) that you have developed for other worksteps in your current process template. You can use these forms only in the current process. |
Setting form properties

You can use the Properties view to set (and modify) properties of the form.

1. Open the form whose properties you want to define.
2. Click a blank area of the Content pane, to open the Properties view.

   By default, it opens the Attributes tab.

   Figure 187: Form Properties view

3. You can use the Class attribute to select a CSS class style from the list of predefined styles. To do so, click the Style row and then click the ellipsis button in the Value column.

   The Properties dialog box appears.

   Figure 188: Style Properties

   a) From the Class drop-down list, select any predefined style. Click OK.

4. The default form header and footer is not visible at runtime. To change this setting, set the Visible attribute value in the Header and Footer sections to true.

5. Use the Events tab to add events associated with the form. The available events for form are onLoad, onUnload, and onSubmit. For information regarding the usage of the Events tab, see Configuring actions on page 315.
Using the Layout tab

The **Layout** tab in the Form Editor interface is the default tab, where you can design the form elements using the following Tasks pane links:

- **Layout** to add layout controls including multi-column tables and tabbed panes. For more information, see Defining the form’s layout on page 281.
- **Controls** to add standard form controls such as text boxes, text areas, images, and sliders. For more information, see Defining the form’s controls.
- **Flow** to add command buttons and links that connect form pages. For more information, see Defining the form’s flow on page 303.
- **Data Sources**, to which you can add and view dataslots and adaplets. For more information, see Defining the form’s data sources on page 305.
- **Form Fragments**, which lists all form fragments. For more information, see Using form fragments on page 309.

You can use drag-and-drop operation to move form elements to another location in your form.

Defining the form’s layout

You can design the layout of your form using the **Layout** link in the Tasks pane. The Layout section in the Form Editor displays the following layout controls: **One-**, **Two-** or **Three-Column Table**, **Custom Table**, **Tabbed Pane**, **Field Set**, **Divider**, **Panel**, and **Include**, each of which you can click to add to the Content pane. Each of these layout controls are described in the following sections.

Adding a table

You can insert a single column, multi-column, and a custom table to your Content pane. You can click any of the following table type options to add the corresponding type of table to the Content pane.

**Table 50: Table types**

<table>
<thead>
<tr>
<th>Table Options In Layout section</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>One-Column Table</td>
<td>Inserts a single-column table.</td>
</tr>
<tr>
<td>Two-Column Table</td>
<td>Inserts a table with two columns.</td>
</tr>
<tr>
<td>Three-Column Table</td>
<td>Inserts a table with three columns.</td>
</tr>
<tr>
<td>Custom Table</td>
<td>Displays the <strong>Custom Table</strong> dialog box, where you can specify the number of rows and columns, as well as the number of header and footer rows. Clicking <strong>OK</strong> inserts the custom table.</td>
</tr>
</tbody>
</table>

You can enter text in the table cells and apply formatting to it using the font type, size and style tools in the Form Editor Toolbar. You can also align the text in each cell by using the aligning tool in the toolbar. Additionally, you can perform the following common table operations.
Table 51: Other Table Operations

<table>
<thead>
<tr>
<th>To...</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert a row or column</td>
<td>Right-click the table, point to Insert, then select the <strong>Insert Columns to the Left</strong>, <strong>Insert Columns to the Right</strong>, <strong>Insert Rows Above</strong>, or <strong>Insert Rows Below</strong> options.</td>
</tr>
<tr>
<td>Remove a row or column</td>
<td>Right-click any cell of the row or column to be removed, point to <strong>Delete</strong>, then click <strong>Delete Rows</strong> or <strong>Delete Columns</strong> options.</td>
</tr>
<tr>
<td>Split a cell</td>
<td>Right-click the cell, then select <strong>Split Cells</strong> option.</td>
</tr>
<tr>
<td>Merge Cells</td>
<td>Select the cells to be merged. Right-click the selection, then select <strong>Merge Cells</strong> option.</td>
</tr>
<tr>
<td>Define Table Properties</td>
<td>Select the table and define the Table properties in the <strong>Properties</strong> view. For details, see <strong>Defining table properties</strong> on page 282.</td>
</tr>
<tr>
<td>Define Table Cell Properties</td>
<td>Click in the table cell and define the cell properties in the <strong>Properties</strong> view. For details, see <strong>Defining table cell properties</strong> on page 283.</td>
</tr>
<tr>
<td>Delete the table</td>
<td>Select the table and press <strong>Delete</strong>. Alternatively, right-click the table and select the <strong>Delete</strong> option.</td>
</tr>
</tbody>
</table>

**Defining table properties**

**Figure 189: Form Editor – Table Properties**

To configure table properties:

1. From the default **Attributes** tab, define the Table Alignment (Left, Right, Center, or Justify) and Table Width (auto, in %, or in pixels) in the respective fields. You can also adjust the Table Border and the Padding and Spacing of the Table cells in the respective boxes.

2. You can use the **Class** row to select a class from the list of predefined styles. To do so, click the **ellipsis** button in the **Value** column, then select the style from the **Properties** dialog box.

3. Enter a name in the **ID** box. This name is in a `<div>` tag that you can refer to when generating JavaScript in the Script tab. This name must be unique. You can refer to the table’s name to hide (or show) it as required by entering JavaScript in the Script tab. For example, `sbm.util.hide('name_Mytable');`

4. Use the **Events** tab to add events associated with the layout. For information regarding the usage of the **Events** tab, see **Configuring actions** on page 315.

**Note:** You can use your customized table in other forms by clicking **File > Save As** and saving it to the appropriate Form fragments folder. After you save the table and refresh Business Process Modeler, the customized table can be retrieved from the Form Fragments section in the Tasks pane. For more information, see **Using form fragments** on page 309.
Defining table cell properties

You can define the properties of each cell of a table added to your form.

**Figure 190: Form Editor – Cell Properties**

![Form Editor – Cell Properties](image)

To configure cell properties in the Properties view:

1. Click in the table cell to be configured.
2. From the **Attributes** tab, define the following properties in the **Appearance** section:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alignment</td>
<td>Specifies the horizontal alignment of the text in the cell. Available options are: left, right, center, justify, and char.</td>
</tr>
<tr>
<td>Class</td>
<td>Specifies the CSS class style used for the table cell. To select another style, click the <strong>ellipsis</strong> button in the <strong>Value</strong> column, then select the style from the <strong>Properties</strong> dialog box.</td>
</tr>
<tr>
<td>Vertical Alignment</td>
<td>Specifies the vertical alignment of the text in the cell. Available options are: top, middle, bottom, and baseline.</td>
</tr>
<tr>
<td>Width</td>
<td>Specifies the width of the cell (in pixels or %).</td>
</tr>
</tbody>
</table>

3. Enter a name in the **ID** box. The name (or its Field ID) must be a unique designation because JavaScript uses it to identify the cell. As this designation must be unique, Business Process Modeler does not allow the duplication of the cell name.

Adding a tabbed pane

You can use a tabbed pane to categorize form information across multiple tabs, thus reducing complexity. You can add text, tables, fields controls, dataslots and even another tabbed pane to a tabbed pane.

By default, a Tabbed Pane in Form Editor displays only one page with the default label, “Page 1.” You can perform the following operations to manage a tabbed pane.
Table 52: Managing a tabbed pane

<table>
<thead>
<tr>
<th>To</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a page</td>
<td>Right-click a page label, point to Add, then select the Add Page option. This displays the Add Page dialog box, where you can type the unique page ID and the page label in the respective boxes. Click OK to add a page.</td>
</tr>
<tr>
<td>Remove a page</td>
<td>Right-click a page label, point to Delete, then select the Delete Page option.</td>
</tr>
<tr>
<td>Define Pane Properties</td>
<td>Click the tabbed pane to view the pane properties in the Properties view. From the General tab, you can enter the name. The name (or its Field ID) must be unique because JavaScript uses it to identify the field. As this designation must be unique, Business Process Modeler does not allow the duplication of the field name. Use the Events tab to add events associated with the layout. For information regarding the usage of the Events tab, see Configuring actions on page 315.</td>
</tr>
</tbody>
</table>

An example of a tabbed pane is displayed in Figure 191.

Figure 191: Form with added tabbed pane

![Figure 191: Form with added tabbed pane](image)

Note: You can save and use your customized tabbed pane in other forms by clicking File > Save As and save it to the appropriate Form fragments folder. After you save the tabbed pane and refresh Business Process Modeler, the customized tabbed pane can be retrieved from the Form Fragments section in the Tasks pane. For more information, see Using form fragments on page 309.

To assign a specific style sheet to a Tabbed Pane, you must move it into a Divider. You can then assign a style sheet to the Divider, and this style is also assumed by the Tabbed Pane that is within the Divider. For more information, see Adding a divider on page 285.

Adding a Field Set

You can use a Field Set to insert a group of related field components. A Field Set is represented by an outlined box. You can add dataslots, form fragment, and other elements to a Field Set box.
To define properties for the added field set, select the field set and define the Field Set properties in the Properties view.

**Figure 192: Form Editor – Field Set**

1. Enter the field set name in the ID value box. The name (or its Field ID) must be a unique designation because JavaScript uses it to identify the field. As this designation must be unique, Business Process Modeler does not allow the duplication of the field name.
2. Select a Class from the list of predefined styles. To do so, click the ellipsis button in the Value column, then select the style from the Style dialog box.
3. Enter the title of the field set in the Legend box.
4. Use the Events tab to add events associated with the layout. For information regarding the usage of the Events tab, see Configuring actions on page 315.

**Note:** You can use your customized field set in other forms by clicking File > Save As and save it to the appropriate Form fragments folder. After you save the field set and refresh Business Process Modeler, the customized field set can be retrieved from the Form Fragments section in the Tasks pane. For more information, see Using form fragments on page 309.

To assign a specific style sheet to a Field Set, you must move it into a Divider. You can then assign a style sheet to the Divider, and this style is also assumed by the Field Set that is within the Divider. For more information, see Adding a divider on page 285.

### Adding a divider

You can use the Dividers function to add an HTML divider to the form. The Divider sets the boundary of a `<div>` tag. A typical use of the Divider option is to associate a business object or a form Layout type with a set of css styles associated with the `<div>` tag. Any item added to a Divider—for example, a dataslot, a table, a tabbed pane or a field set, assumes the style assigned to the Divider.

**To define properties for the divider:**

1. Select the Divider layout and define the Divider properties in the Properties view (Figure 193 on page 286).
2. Enter the divider name in the ID value box. The name (or its Field ID) must be a unique designation because JavaScript uses it to identify the field. As this designation must be unique, Business Process Modeler does not allow the duplication of the field name.
3. Select a Class from the list of predefined styles. To do so, click the ellipsis button beside the Class box, then select the style from the Style dialog box.

You can also open the Advanced tab to associate a particular style to the selected class.

4. Use the Style to associate a particular style to the selected class. To do so, click the ellipsis button in the Value column, then select the style from the Style dialog box.
5. Use the Events tab to add events associated with the layout. For information regarding the usage of the Events tab, see Configuring actions on page 315.
The figure below shows the Divider properties. This class, "ApBody", is applied to any form component added to the Divider.

**Figure 193: Form Editor – Divider**

![Divider Properties](image)

*Note:* You can save and use a customized divider in other forms by clicking **File > Save As** and save it in the appropriate Form fragments folder. After you save the divider and refresh Business Process Modeler, the customized divider can be retrieved from the Form Fragments section in the Tasks pane. For more information, see *Using form fragments* on page 309.

---

**Adding a Panel**

You can use the Panel layout to embed form components. A panel functions as a Divider, and supports attributes including title, width, height, and collapsibility.

To define properties for a Panel layout, select the panel layout and define the panel properties in the **Properties** view.

**Figure 194: Form Editor – Panel**

![Panel Properties](image)

1. Enter the panel name in the **ID** value box. The panel name (or its Field ID) must be a unique designation because JavaScript uses it to identify the field. As this designation must be unique, Business Process Modeler does not allow the duplication of the field name.
2. Enter the panel title in the **Title** value box. The panel title is displayed in the form interface.
3. From the **Collapsible** value box, enable (or disable) the collapsibility by specifying true or false.
4. From the **Width** and **Height** value boxes, set the width and height (in pixels) of the panel.
5. Use the **Style** to associate a particular style to the selected class. To do so, click the **ellipsis** button in the **Value** column, then select the style from the **Style** dialog box.

Select a Class from the list of predefined styles. To do so, click the **ellipsis** button in the **Value** column, then select the style from the **Style** dialog box.

**Note:** The **Events** tab is not supported for Panel layout.

### Adding an Include layout

You can use an Include section to insert an inline frame within a form or a block of text. It can act as a target for other links, including URLs, HTML documents, and external objects. The Include section can act as a Client-side Include that uses an iFrame layout option; or as a Server-side Include, where the content of the Include section is generated on the server and is rendered on the Form Editor as HTML markup (equivalent to using a `<jsp:include>` tag in JSP forms).

To define properties for an Include section, select the Include layout and define the properties in the **Properties** view.

**Figure 195: Form Editor – Include Properties**

1. Enter a name for the Include layout in the **Name** value box. The name (or its Field ID) must be a unique designation because JavaScript uses it to identify the field. As this designation must be unique, Business Process Modeler does not allow the duplication of the field name.
2. For **Type** box, select either the **Client-side Include** or **Server-side Include** option.
3. For a Client-side Include, enter a valid URL in the **URL** box.

   The URL parameter for a client-side Include section can be a Relative or an Absolute URL. In addition, the value can be dynamically assigned by using a tag such as: `<bizsolo:value name='myDataslot'/>`. In this example, the Include option has its source set dynamically based on the value of myDataslot, which must be a CHARACTER dataslot with a valid URL as its value.

4. For a Server-side Include, enter a URL in the **URL** box, or click the **ellipsis** button beside the **URL** box to select a JSP file published on the server. In this case, the content of the Include section is generated on the server and rendered on the form as HTML markup. The Server-side URL can be a Relative URL that starts with “…”, or an Absolute URL that starts with “http://”, or Relative to the Web application root that starts with “${request.contextPath}”.

**Note:** The **Events** tab is not supported for Include layout.
Assigning a style

You can use the **Style Sheet Links** function to associate a specific style sheet (or *.css file) with the main form, a table or table cell (or group of tables or cells); for example, you can change the background color for a column in a table or define a border style around fields.

To associate a style sheet with a form element:

1. Select a form, table or a table cell (or a group of each type).
2. From the **Tools** menu, click **Style Sheet Links**, opening the **Style Sheet Links** dialog box.

![Figure 196: Form Editor – Style Sheet Links Dialog](image)

3. To access the *.css files you want, click **New** to open the **Style Sheet Link** dialog box. Click the **ellipsis** button beside the **URL** box and select either of the following options.
   - **Use default style sheet** to use the default
   - **Select from file** to browse to a location for the *.css files. Select the file, then click **Open** to return to the **Style Sheet Link** dialog box.

4. Click **OK** to add it to the **Style Sheet Links** dialog box. You can also enter or paste a valid URL address (with http prefix) to access the required *.css files.
5. Select a style sheet from the table in the **Style Sheet Links** dialog box and click **OK** to assign it to the main form, a selected file or selected table cell.

Defining the form’s controls

The Form Editor supports controls (or widgets) accessible from the **Controls** panel of the **Controls** link in the Form Editor. Form controls provide the following benefits:

- General attributes including appearance and effects, as described in **Setting attributes** on page 290.
- Can be bound to a service, dataslot, or have a static value. For details, see **Using data binding** on page 293.
Defining the form’s controls

- Advanced validation framework, as described in Applying validation on page 301.
- Advanced event handling.

Figure 197: Form Editors – Controls section

You can drag each of the following controls to your form, or click the control to insert the control at the insertion point.

Table 53: List of Controls

<table>
<thead>
<tr>
<th>Control Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TextField</td>
<td>A Text field control</td>
</tr>
<tr>
<td>TextArea</td>
<td>A Text Area control</td>
</tr>
<tr>
<td>Combobox</td>
<td>A Combo box control</td>
</tr>
<tr>
<td>Checkbox</td>
<td>A checkbox control</td>
</tr>
<tr>
<td>Radio</td>
<td>A Radio button control</td>
</tr>
<tr>
<td>List</td>
<td>A List control, consisting of static, multiple data</td>
</tr>
<tr>
<td>Tree</td>
<td>A tree control, consisting of nodes and subitems</td>
</tr>
<tr>
<td>Grid</td>
<td>A Grid control</td>
</tr>
<tr>
<td>Menu</td>
<td>A menu control containing an array of menu items</td>
</tr>
<tr>
<td>Slider</td>
<td>A Slider control that enables you to adjust values over a finite range</td>
</tr>
<tr>
<td>Date Time</td>
<td>A data and time control that allows you to choose a date or time</td>
</tr>
<tr>
<td>Button</td>
<td>A generic command button typically used to trigger a event</td>
</tr>
<tr>
<td>Image</td>
<td>An image control</td>
</tr>
<tr>
<td>Label</td>
<td>A text label. For more information, see Adding a label on page 311.</td>
</tr>
</tbody>
</table>

In addition to these widgets, you can add widgets available on Business Process Server server (located under <Application_Server>\sbm\resources folder).
To add the widget to Form Editor:

1. Copy the widget folder (available on Business Process Server server) to Workspace_Home\.com.savvion.studio\forms\widgets folder.

2. Edit the widgets.xml file (located in Workspace_Home\.com.savvion.studio\forms folder) with the entry for the widget to be added.
   - Add the entry for image (*.png) to any of the available sections (for instance, <sbm>) as <widget_name icon="<image_name>.png">. For example, <textfield icon="textfield_icon.png">.

3. Start Business Process Modeler to view the control in the Controls panel in the Controls link of the Form Editor Tasks pane.

Setting attributes

You can configure control attributes including appearance, effects, and general properties for each control from the Attributes tab of control’s Properties view. Figure 198 on page 290 displays the Attributes tab for a TextField widget.

Figure 198: Control Properties of TextField widget – Attributes tab

Table 54 on page 290 describes the common attributes and the advanced control they are used for.

Table 54: List of attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Control used for</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disabled</td>
<td>TextField, Combobox, Checkbox, Radio, TextArea, List</td>
<td>A Boolean attribute to disable the control or not.</td>
</tr>
<tr>
<td>Read Only</td>
<td>TextField, Combobox, Checkbox, Radio, TextArea, List</td>
<td>Specifies whether the control is read-only or not.</td>
</tr>
<tr>
<td>Required</td>
<td>TextField, Combobox, Checkbox, Radio, TextArea, List</td>
<td>Specifies whether the control is mandatory or not.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Control used for</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Size</td>
<td>TextField, List</td>
<td>An integer attribute to specify the number of visible characters in the control. The default value varies according to the control used.</td>
</tr>
<tr>
<td>Layout</td>
<td>Checkbox, Radio</td>
<td>Specifies the layout for the control. Values are “Vertical” and “Horizontal.”</td>
</tr>
<tr>
<td>ID</td>
<td>For all controls</td>
<td>The control name (or its Field ID) must be a unique designation because JavaScript uses it to identify the field. Accept the default name assigned to the control or enter a unique name. As this designation must be unique, Form Editor does not allow the duplication of a control name. <strong>Note:</strong> The control names are not case-sensitive, for example, you cannot add controls named “textfield1” and “TextField1” in the same form.</td>
</tr>
<tr>
<td>Description</td>
<td>For all controls</td>
<td>(Optional) Used to describe the control.</td>
</tr>
<tr>
<td>Tab Order</td>
<td>For all controls</td>
<td>Used to set the sequence in which the various controls appear in the presentation by entering a sequence of numbers; example, 1, 2, 3, 4, 5, 6.</td>
</tr>
</tbody>
</table>

**Table 55** on page 291 describes the attributes, which are specific to each advanced control.

**Table 55: Control-specific attributes**

<table>
<thead>
<tr>
<th>Control</th>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TextField</td>
<td>Type</td>
<td>Indicates the type of text field control. Available options are Text, Password, and Label.</td>
</tr>
<tr>
<td>Max Length</td>
<td>An integer attribute to specify the maximum number of characters allowed in the control. The default value is 50.</td>
<td></td>
</tr>
<tr>
<td>Effects</td>
<td>Used to apply visual enhancement to the control. The available effects are Fade, Slide, Blind, Grow, Shake, and Highlight. To apply an effect, click the <strong>ellipsis</strong> button in the Value column. You can modify the default duration of the effect and set the initial and final opacity (in pixels or percentage) of the effect.</td>
<td></td>
</tr>
<tr>
<td>Combobox</td>
<td>Dropdown Size</td>
<td>Used to specify the size of the dropdown list of the Combobox control.</td>
</tr>
<tr>
<td></td>
<td>Cascade / Level</td>
<td>Used for creating a cascading control.</td>
</tr>
<tr>
<td>Control</td>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| TextArea  | Wrap      | To enable or disable word wrap. Select any of the following available options:  
|           |           | • **Soft**: to move the entire word being truncated to the next line.  
|           |           | • **Hard**: to split the word.  
|           |           | • **Off**: to disable word wrap. In this case, the entered text appears on a same line.  
|           | Rows      | Indicates the rows of text displayed in the Text Area.  
|           | Cols      | Indicates the appropriate width of the control.  
| Image     | alt       | Used to specify the text that displays as an alternative to the image that does not display at runtime.  
|           | border    | Used to apply a border to the image.  
|           | width/height | Used to specify the size (in pixels) of the image.  
| Slider    | animate   | A Boolean attribute to set the slider into motion without user interaction.  
|           | initialValue | Indicates the starting point (and value) of the slider.  
|           | maxValue  | Indicates the maximum value (or point) of the slider.  
|           | minValue  | Indicates the minimum value (or point) of the slider.  
|           | sizeInPixels | Used to specify the size (in pixels) of the slider.  
|           | sliderType | Sets the slider orientation to Horizontal (H) or Vertical (V).  
|           | tickSize  | Specifies the size of each movement of the slider.  
| Date Time | Display Format | Used to set the display format of the date. The default format is MM/dd/yyyy.  
| Button    | type      | Specifies the type of button. Available options are: button, complete, cancel, reset, save, and reassign.  
|           | label     | Specifies the button label. Default label is “Push me.”  
|           | checked   | A Boolean attribute to indicate if the control is selected or not.  
|           | href      | Specifies the target URL.  
|           | duallist  | A Boolean attribute to enable this button to move (or copy) items from the sourcelist attribute to the targetlist attribute. Typically, this is used in conjunction with two list boxes.  
|           | sourcelist | Specifies the Source widget from where you retrieve form values.  
|           | targetlist | Specifies the Target widget where you store form values.  

**Note:** For information on attributes specific to Label control, see Adding a label on page 311.

To enter an attribute value, click the **Value** column for that attribute and type the value. Alternatively, to select an attribute value, double-click the **Value** column for that attribute and select the value from the drop-down list that appears.
Using data binding

Business Process Modeler supports multiple dataslot types for binding to a form control, for example, the “Date Time” control can be bound to a CHARACTER or DATETIME-TZ dataslot.

You can define the data binding using the Data Binding tab of control’s Properties view. Figure 199 on page 293 displays the Data Binding tab for a TextField control.

Figure 199: Control Properties of TextField widget – Data Binding tab

Depending on the type of control, the Data Binding tab contains the following properties:

- **Data Source** property (not available for List, Combobox, Radio, and Checkbox controls), using which you can select the data source to which you can bind the control. To configure a data source, click the ellipsis button in the Value column for the Data Source property. The Data Source Binding Wizard appears.

Figure 200: Data Source Binding wizard
You can select any of the following options for populating the control:

- **Service**, as discussed in Using a defined service on page 295.
- **DataSlot**, as discussed in Using dataslot on page 296.
- **Static** (default), as discussed in Specifying static data on page 299.

- **Choices** property (available only for multi-select controls including List, Combobox, Radio, and Checkbox controls), using which you can select the choice option for populating the control. To configure a choice, click the ellipsis button in the Value column for the Choices property. The Choices Data Binding Wizard appears.

Figure 201: Choices Data Binding wizard

![Choices Data Binding Wizard](image)

You can select any of the following options for data source:

- **Service**, as discussed in Using a defined service on page 295.
- **DataSlot**, as discussed in Using dataslot on page 296.
- **Static** (default), as discussed in Specifying static data on page 299.

- **Choices** property (available only for multi-select controls including List, Combobox, Radio, and Checkbox controls), using which you can select the choice option for populating the control. To configure a choice, click the ellipsis button in the Value column for the Choices property. The Choices Data Binding Wizard appears.

Figure 201: Choices Data Binding wizard

![Choices Data Binding Wizard](image)

You can select any of the following options for populating the control:

- **Service**, as discussed in Using a defined service on page 295.
- **DataSlot**, as discussed in Using dataslot on page 296.
- **Adaplet**, as discussed in Using adaplets on page 299.
- **Static**, as discussed in Specifying static data on page 299.

- **Data Target** property (not available for List, Combobox, Radio, and Checkbox controls), using which you can select the dataslot to store the control data when the form is submitted. If you have specified a dataslot as the source dataslot in the Data Source property, then the same
Dataslot must be used in the **Data Target** property. To configure a data target, click the **ellipsis** button in the **Value** column for the **Data Target** property. The **Data Binding Wizard** appears.

**Figure 202: Data Binding wizard**

![Data Binding wizard](image)

Select the **DataSlot** option for the data target. For more information, see *Using dataslots* on page 254.

- **Selection Source/Target** property (available only for List, Combobox, Radio, and Checkbox controls), where you can select the same dataslot to be used as the source (for data retrieving) and target (for data storage). To configure this property, click the **ellipsis** button in the **Value** column for the **Selection Source/Target** property. The **Data Binding Wizard** appears, which is similar to Figure 202 on page 295. Select the **DataSlot** option for the source and target. For more information, see *Using dataslot* on page 296.

**Note:** The **Data Binding** tab is not available for the Button control.

---

**Using a defined service**

You can bind controls to a defined service (*.sjs) to be used either as a data source or as a choice for populating Combobox, List, Checkbox, and Radio controls.

**To do so:**

1. From the **Data Source Binding Wizard** (Using data binding on page 293) or **Choices Data Binding Wizard**, click the **Service** option, then click **Next**.
2. From the second page of the respective wizard, select either of the following data sources:
   - Specify the JSON file. To do so, click the **ellipsis** button and select the JSON file (*.sjs) located in the `<Project_Name>/jsp` folder.
• Specify the URL to a location that returns JSON style data. You can use this option for dynamic services that connect to the datasource at runtime and return data in proper JSON format. For an example of dynamic services, refer to the "WidgetDemo" project, which contains the Grid control bound to the data1.jsp service. At runtime, this service uses Hibernate to query the database and returns data in the grid JSON format.

Figure 203: Data Source Binding wizard, Page 2

3. Click Finish to add the specified service in the Details column and specify the value as "Service" for the respective property.

Using dataslot

You can bind controls to supported dataslots to be used as a data source, data target (or both), or as a choice for populating Combobox, List, Checkbox, and Radio controls. You cannot bind Tree and Grid controls to dataslots.
To bind the control to a dataslot:

1. From the **Data Source Binding Wizard**, **Choices Data Binding Wizard**, or **Data Binding Wizard**, click the **DataSlot** option, then click **Next**.

2. The second page of the respective wizard displays the list of system-defined (not available for Data Target) and user-defined dataslots.

   **Figure 204: Selecting a dataslot**

   ![Figure 204: Selecting a dataslot](image)

   a) Select the dataslot to populate the control.

   b) Click **Finish** to add the specified dataslot and value (if set) in the **Details** column and specify the value as "DataSlot" for the respective property.

If you select a dataslot in the **Data Source** property for the control, the **Data Target** property is disabled as the same dataslot specified in the **Data Source** property is used to store the output data. For the other selections, you can select the dataslot in the **Data Target** property similar to as described in the preceding steps.

**Note:** For the Slider control, you can only specify a dataslot in the **Data Target** property.

For Combobox, List, Checkbox, and Radio controls, you can use the **Selection Source/Target** property to specify the dataslot to be used as the source and target.

**Note:** You can also bind a control to a dataslot by dragging the valid dataslot from the **Data Sources** section of the Tasks pane (as described in **Binding to a data source** on page 306) on the control.

**Table 56** on page 298 provides the list of supported dataslot types for each of the controls.
Table 56: Supported Dataslot Types for Controls

<table>
<thead>
<tr>
<th>Control Name</th>
<th>Dataslot Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>TextField</td>
<td>• CHARACTER</td>
</tr>
<tr>
<td></td>
<td>• INTEGER, INT64</td>
</tr>
<tr>
<td></td>
<td>• String and Number attributes of Business Object dataslot</td>
</tr>
<tr>
<td>TextArea</td>
<td>• CHARACTER</td>
</tr>
<tr>
<td></td>
<td>• String attribute of Business Object dataslot</td>
</tr>
<tr>
<td>Combobox</td>
<td>• CHARACTER, LIST</td>
</tr>
<tr>
<td></td>
<td>• MAP for choice dataslot only</td>
</tr>
<tr>
<td></td>
<td>• String attribute of Business Object dataslot</td>
</tr>
<tr>
<td>Checkbox</td>
<td>• CHARACTER, LOGICAL, LIST</td>
</tr>
<tr>
<td></td>
<td>• MAP for choice dataslot only</td>
</tr>
<tr>
<td></td>
<td>• String and Boolean attributes of Business Object dataslot</td>
</tr>
<tr>
<td>Radio</td>
<td>• CHARACTER, LIST</td>
</tr>
<tr>
<td></td>
<td>• MAP for choice dataslot only</td>
</tr>
<tr>
<td></td>
<td>• String attribute of Business Object dataslot</td>
</tr>
<tr>
<td>List</td>
<td>• LIST</td>
</tr>
<tr>
<td></td>
<td>• CHARACTER and MAP for choice dataslot only.</td>
</tr>
<tr>
<td></td>
<td>• String attribute of Business Object dataslot (for choice dataslot only)</td>
</tr>
<tr>
<td>Image</td>
<td>Dataslots not supported.</td>
</tr>
<tr>
<td>Tree</td>
<td>Dataslots not supported.</td>
</tr>
<tr>
<td>Grid</td>
<td>Dataslots not supported.</td>
</tr>
</tbody>
</table>
### Defining the form’s controls

<table>
<thead>
<tr>
<th>Control Name</th>
<th>Dataslot Types</th>
</tr>
</thead>
</table>
| Menu         | • CHARACTER  
               • String attribute of Business Object dataslot |
| Slider       | Supported only for target dataslot:  
               • CHARACTER  
               • String attribute of Business Object dataslot |
| Date Time    | • CHARACTER, DATETIME-TZ  
               • String and Timestamp attributes of Business Object dataslot |

### Using adaplets

Adaplets are preconfigured managed adapters used in forms. An adaplet represents a managed adapter with a predefined output format. For more information regarding adaplets, see Adding adaplets to a form on page 308.

You can bind Combobox, List, Checkbox, and Radio controls to adaplets, which returns a List object to the control.

**To bind a control to an adaplet:**

1. From the Choices Data Binding Wizard, click the Adaplet option, then click Next.
2. From the second page of the Choices Data Binding Wizard, click the ellipsis button to open the Select Choices Source dialog box, which lists the defined adaplets under each configured adapter. For example, DBAdapter_1.
3. Select the adaplet, then click OK to bind the adaplet to the control. Ensure that you select an adaplet that returns a List object to the control.

### Specifying static data

You can specify a static value for controls.

**To enter static data for the control:**

1. From the Data Source Binding Wizard or Choices Data Binding Wizard, click the Static option, then click Next.
2. The second page of the respective wizard displays two tabs, namely, Editor and Source. The appearance of the Editor and Source tabs differs for each control, as described in Table 57 on page 300.
### Table 57: Entering static values for controls

<table>
<thead>
<tr>
<th>Controls</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TextField, TextArea, Image, Slider, Date Time</td>
<td>Enter the initial value in the box provided in the Editor tab.</td>
</tr>
<tr>
<td>Checkbox, Combobox, List, Radio</td>
<td>Enter the label/value pair by clicking New and entering the appropriate information. Repeat for additional label/value pairs. You can also modify, remove, or change the position of the list entries.</td>
</tr>
<tr>
<td>Tree</td>
<td>• To edit any of the existing nodes, select the node and click Edit Node. Modify the label and value in the appropriate boxes.</td>
</tr>
<tr>
<td></td>
<td>• To add a node, select the root node and click Add Node. Enter the label and value in the appropriate boxes.</td>
</tr>
<tr>
<td></td>
<td>• You can also remove and change the node position.</td>
</tr>
<tr>
<td>Menu</td>
<td>Design a menu.</td>
</tr>
<tr>
<td></td>
<td>• To add a node, select the root node and click Add Node. Enter the label and the menu action to be performed. You can also specify a style (*.CSS) and disable the node.</td>
</tr>
<tr>
<td></td>
<td>• To edit any of the existing nodes, select the node and click Edit Node. Modify the menu properties, as required.</td>
</tr>
<tr>
<td></td>
<td>• You can also remove and change the node position.</td>
</tr>
<tr>
<td>Grid</td>
<td>Design a grid. You can edit the table contents and column name, as well as add and remove rows/columns. You can also move up (or down) the inserted rows.</td>
</tr>
</tbody>
</table>

3. Click **Finish** to add the static value in the Details column and specify the value as "Static" for the respective property.
Applying validation

You can use an advanced validation framework to validate advanced controls from the Validation tab of control’s Properties view. You can configure validation for the TextField, CheckBox, Combobox, List, Radio, and TextArea controls. Figure 205 on page 301 displays the Validation tab for a TextField widget.

Figure 205: Control Properties of TextField widget – Validation tab

Table 58 on page 301 lists the common validation rules for all controls.

Table 58: Common validation rules

<table>
<thead>
<tr>
<th>Validation Rule</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Validate On</td>
<td>Specifies when to validate the control. Available options are “[blur]” (when the control loses focus), “[change]” (when the data is entered or modified in the control), and when both these events occur.</td>
</tr>
<tr>
<td>IsRequired</td>
<td>Specifies if the control is a required field.</td>
</tr>
<tr>
<td>Use Character Masking</td>
<td>(Available for TextField and TextArea) Enables character masking (or wildcard entry) for the controls. Not available for TextField types “custom,” “Email,” and “URL.”</td>
</tr>
<tr>
<td>Minimum Characters</td>
<td>(Available for TextField and TextArea) Specifies the minimum number of characters to be entered in the control. In the case of TextField, this field is available only for types, “Email” and “URL.”</td>
</tr>
<tr>
<td>Maximum Characters</td>
<td>(Available for TextField and TextArea) Specifies the maximum number of characters which can be entered in the control. In the case of TextField, this field is available only for types, “Email” and “URL.”</td>
</tr>
<tr>
<td>Invalid Value</td>
<td>(Available for Combobox and List) Specifies the value that cannot be entered in the control. Default value is -1.</td>
</tr>
</tbody>
</table>

Table 59 on page 301 lists the widget-specific validation rules.

Table 59: Widget-Specific Validation Rules

<table>
<thead>
<tr>
<th>Control</th>
<th>Validation Rule</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Checkbox</td>
<td>Minimum Selections</td>
<td>Used to specify the minimum number of checkboxes in a group to be selected.</td>
</tr>
<tr>
<td></td>
<td>Maximum Selections</td>
<td>Used to specify the maximum number of checkboxes in a group to be selected.</td>
</tr>
<tr>
<td>Control</td>
<td>Validation Rule</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>TextField</td>
<td>Credit Card Format</td>
<td>(Available for type “Credit Card”) Used for validating format of credit card number. Available formats are Visa, Mastercard, Amex, Discover, and Dinersclub.</td>
</tr>
<tr>
<td></td>
<td>Minimum / Maximum Value</td>
<td>(Available for types “Currency,” “Date,” “Integer,” “Real,” and “Time”) Used to specify a range of values that can be entered.</td>
</tr>
<tr>
<td></td>
<td>Pattern</td>
<td>(Available for types “Custom,” “Social Security Number,” “Phone Number,” and “Zip Code”) Used for setting the pattern of the entered value. Patterns include checking for whole numbers and case-sensitivity.</td>
</tr>
<tr>
<td></td>
<td>Date Format</td>
<td>(Available for type “Date”) Used for validating format of the entered date. Available formats are mm/dd/yy, mm/dd/yyyy, dd/mm/yy, dd/mm/yyyy, yy/mm/dd, yyyy/mm/dd, mm-dd-yy, dd-mm-yy, yyyy-mm-dd, mm.mm.yy, and dd.mm.yyy.</td>
</tr>
<tr>
<td></td>
<td>IP format</td>
<td>(Available for type “IP”) Used for validating format of the entered IP address. Available formats are IPv4, Dot Comma, and IPv4, IPv6.</td>
</tr>
<tr>
<td></td>
<td>Phone Format</td>
<td>(Available for type “Phone Number”) Used for customizing format of the phone number. Available formats are Custom Phone Number (default) and Phone Number. You can customize the phone number format using the Pattern box.</td>
</tr>
<tr>
<td></td>
<td>Time Format</td>
<td>(Available for type “Time”) Used for validating format of the entered time. Available formats are HH:mm, HH:mm:ss, hh:mm tt, hh:mm:ss tt, hh:mm tt, and hh:mm:ss tt.</td>
</tr>
<tr>
<td>TextArea</td>
<td>Textarea Counter Type</td>
<td>Checks the number of characters entered. Available types are “Characters Remaining,” which deducts the number of entered characters from the value in the Maximum Characters box, and “Character Count” which counts the number of entered characters.</td>
</tr>
<tr>
<td></td>
<td>Textarea Hint</td>
<td>Indicates the type of data that the user needs to enter in the Text Area.</td>
</tr>
</tbody>
</table>

Table 60 on page 303 lists the error messages for each control in case of validation failure.
Table 60: Validation messages

<table>
<thead>
<tr>
<th>Control</th>
<th>Validation Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TextField</td>
<td>Textfield Required Message</td>
<td>Used to specify a common message for any validation failure.</td>
</tr>
<tr>
<td>Checkbox</td>
<td>Checkbox Required Message</td>
<td>Used to specify the message in case a required checkbox is not selected.</td>
</tr>
<tr>
<td></td>
<td>Checkbox Minimum Selection Message</td>
<td>Used to specify the message in case the minimum number of checkboxes (as specified in the Minimum Selections box) are not selected.</td>
</tr>
<tr>
<td></td>
<td>Checkbox Maximum Selection Message</td>
<td>Used to specify the message in case the maximum number of checkboxes (as specified in the Maximum Selections box) are exceeded.</td>
</tr>
<tr>
<td>TextArea</td>
<td>Textarea Validation Message</td>
<td>Used to specify the message in case an invalid value is entered.</td>
</tr>
<tr>
<td></td>
<td>Validation Minimum Characters Message</td>
<td>Used to specify the message in case the minimum number of characters (as specified in the Minimum Characters box) are not entered.</td>
</tr>
<tr>
<td></td>
<td>Validation Maximum Characters Message</td>
<td>Used to specify the message in case the maximum number of characters (as specified in the Maximum Characters box) are exceeded.</td>
</tr>
</tbody>
</table>

**Note:** You can use the Events tab to add events associated with the control. For information regarding the usage of the Events tab, see Configuring actions on page 315.

---

**Defining the form’s flow**

You can add navigation tools to your form using the Flow link in the Tasks pane. Using this link, you can add a button or link that helps you to proceed to the next workstep. The Flow section in the Form Editor displays the Button and Link flow controls, each of which you can click to add to the Content pane.

**Adding a button**

You can use the Button icon to add a button to your form. By default, an added button is displayed as a Complete button (Complete). For example, you can add the Complete button to a customized footer in your form. For a form workstep in a Web application, you can use the Complete button to activate the next page.
To define the button properties, click the button to display its Control Properties in the **Properties** view. By default, the Properties view opens to the **General** tab.

**Figure 206: Control Properties – Button**

1. Accept the default name assigned to the control or enter a unique name in the **ID** box. The control name (or its Field ID) must be a unique designation because JavaScript uses it to identify the field. As this designation must be unique, Business Process Modeler does not allow the duplication of a Control name.
2. Use the **Tab Order** box to set the sequence in which the various controls appear in the presentation by entering a sequence of numbers; example, 1, 2, 3, 4, 5, 6.
3. Select the button type from the **Type** drop-down list.
   
   [Table 61 on page 304 lists the available button types.]

   **Table 61: Button types**

<table>
<thead>
<tr>
<th>Button Type</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete</td>
<td>To complete the workstep.</td>
</tr>
<tr>
<td>Save</td>
<td>To save form details.</td>
</tr>
<tr>
<td>Reset</td>
<td>To reset the form field details to its original values.</td>
</tr>
<tr>
<td>Cancel</td>
<td>To cancel the changes made, and exit the form.</td>
</tr>
<tr>
<td>Reassign</td>
<td>To reassign the workstep to another performer.</td>
</tr>
<tr>
<td>Button</td>
<td>A configurable command button to perform any other function.</td>
</tr>
</tbody>
</table>

4. Enter a label for the button in the **Label** box. The Label overwrites the name of the selected type.
5. Use the **Description** tab (optional) to enter a description.
6. Use the **Events** tab to add events associated with the control. For information regarding the usage of the **Events** tab, see Configuring actions on page 315.

### Adding a link

You can use the **Link** icon to add a link to your form. You can also specify the target URL of the link and indicate how the target URL opens in Business Process Portal. In case of a link, you need to specify the link properties when you click the Link icon in the Tasks pane.
To configure the link properties:

1. From the **Link Properties** dialog box, enter the link text to be displayed, in the **Text** box, and the target URL in the **URL** box.
2. Use the **Advanced** tab (right image, Figure 207 on page 305) to indicate how the link appears in Business Process Portal after it is published. Enter a tooltip for the link in the **Tool Tip** box that displays when you place your cursor on the link.
3. From the **Target** drop-down list, choose where you want the target URL to be displayed. Table 62 on page 305 lists the URL target options.

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>_self</td>
<td>Opens the target URL in the same window.</td>
</tr>
<tr>
<td>_top</td>
<td>Navigates to the top of the current window.</td>
</tr>
<tr>
<td>_blank</td>
<td>Opens the target URL in a new window.</td>
</tr>
<tr>
<td>_parent</td>
<td>Navigates to the parent window.</td>
</tr>
</tbody>
</table>

4. To execute a JavaScript command through this link, enter the script in the **On Click** box (right image, Figure 207 on page 305).

For Web applications only, you can also direct the link to the next workstep instead of a URL. From the **Link Properties** dialog box, enter the name of the connector to the next workstep, in the **URL** box. For instance, use the protocol: “follow:Connection2”, where Connection2 is the name of the link to the next workstep.

**Defining the form’s data sources**

You can use the **Data Sources** link in the Tasks pane to bind form elements to a Dataslot or Adaptlet. The Data Sources function in the Tasks pane displays all the current process-level dataslots and adaplets.
You can expand the **Dataslots** and **Adaplets** folder to view and select the item you want to bind to a Form Editor element.

- The **Dataslots** folder contains two subfolders, namely, **System Dataslots** (predefined dataslots displayed only for forms in BPM processes) and **User Dataslots** (user-defined dataslots).
- The **Adaplets** folder contains DBAdapters and WebServiceAdapters adaplet types.

**Figure 208: Data Sources section in the Form Editor**

You can add more user-defined dataslots to the existing list. Right-click the **User Dataslots** folder, then select **Add** option to open the **New Dataslot** dialog box where you can add a new dataslot. For more information, see **Creating user-defined dataslots** on page 101.

You can add all the listed dataslots (system and user) in bulk to your form.

1. Expand the **Dataslots** folder, then drag the **System Dataslots** folder (to add all the predefined system dataslots, available in a Process Model) or **User Dataslots** folder (to add all the user-defined dataslots) to your form.

   The Form Editor displays the following options:
   - **Controls Table**, to display the added dataslots in a two-column table.
   - **Controls**, to display the added dataslots as separate fields.

2. Click the appropriate option to add the dataslots in bulk.

   **Note**: You can also add business objects with "Collection" attribute by dragging them to your form. For more information, see **Using business objects in a form** on page 405.

### Binding to a data source

You can drag an individual dataslot to a blank portion of your form to create a data source field. You can also drag a data source to an existing table column, text field or other control element, in order to bind it to the Form Editor element.
Figure 209 on page 307 illustrates an example of form elements added by dragging the corresponding dataslot.

**Figure 209: Form Editor – Data Sources option**

You can edit the label of the bound form element, by right-clicking the label (displayed in orange typeface), then selecting the **Label Properties** option.

You can define properties for the form element bound to a dataslot, similar to defining properties for an unbound element. To define the properties for a field bound to a dataslot, click the control to view its Control Properties (as described in **Defining the form’s controls** section) in the **Properties** view.

The **Properties** view for each dataslot type (bound to the control) is described below:

- For controls bound to CHARACTER or INTEGER dataslot types, the **Properties** view is the same as that for the Text Field control.

- For controls bound to DATETIME-TZ dataslot type, the **Properties** view is the same as that for the Date Time control.

- For controls bound to LOGICAL dataslot type, the **Properties** view is the same as that for the Checkbox control.

- For controls bound to LIST dataslot type, the **Properties** view is the same as that for the List control.

- For controls bound to Map dataslot type, the **Properties** view is the same as that for the Combobox control.

- For controls bound to an Object dataslot, the **Properties** view includes the **Renderer** box with the default custom dataslot. To use a Custom Dataslot in a form, retain Default as the Renderer. You can use the default renderer if its class implements iHTML (or inline HTML) and Serializable.

**Figure 210: Form Editor – Control Properties for Object Dataslots**
Adding adaplets to a form

Adaplets contain the same functions that Managed Adapters have, but are more compact—for example, they can be included in a form without adding a separate Adapter workstep or additional dataslots. Using an adaplet enables you to add a configured adapter to a workstep's presentation form and associate it with a form element. When you open that workstep as a task in Business Process Portal, the adaplet controls are displayed when a specified event, such as loading the form or on clicking a button, occurs.

You can review the group of adaplets (generic and preconfigured) in the Adaplets folder in the Data Sources section of the Tasks pane. For instance, WebServiceAdapters displays its generic and preconfigured (if any) adaplets.

You can only use adaplets that have been preconfigured. You may not use generic adaplets in the Form Editor unless you configure them first. To configure a generic adaplet, right-click the adaplet and select Properties option.The Configurator for that adaplet opens; for example, selecting Properties for the DBAdapter opens the DBAdapter Configurator dialog box. For information on configuring a specific adaplet type, see the relevant chapter in the Managed Adapters Guide. You can also use this method to re-configure an existing preconfigured adaplet.

**Note:** The preconfigured adaplet can only be used for the current form.

Alternatively, you can configure an adaplet from the Managed Adapter Browser by making a copy of a generic adapter, configuring it, and exporting it to the repository for future re-use. For more information on using this browser, see Using managed adapters on page 121. The adaplet configured using the Managed Adapter Browser can be saved to, and imported from, the repository, and can be used by all Business Process Modeler users.

**Note:** If you add or configure an adaplet in the Managed Adapter Browser, you can use the Refresh functionality to reflect the changes in the adaplet list in your form. To do so, right-click the adaplet folder in the Data Sources link and select the Refresh option.

You can drag the configured adaplet into the form. You can also expand a configured adaplet and drag and drop one or more of its input/output parameters in the form.

You can invoke an adaplet from an added form control using the Graphical Event Logic (GEL) editor, as described in Adding Expressions on page 318. For example, to copy the dataslot values when the adaplet is invoked, use a callback function as described in the following steps:

1. Create the preconfigured adaplet, "GetCustomerInfo" for the DBAdapter in your process.
2. Add the button, "button1" in your form and add the onClick event to this button.
3. Using Graphical Event Logic (GEL) Editor, add the "Invoke adaplet with callback" expression for this event with the <activity_name>/DBAdapters/GetCustomerInfo as the adaplet name and updateFields() as the callback function.

   In this case, the callback function, updateFields(), is invoked following completion of the GetCustomerInfo adaplet.

4. Add the following JavaScript for the updateFields() method in the Script tab (Using Script tab on page 332) of the Form Editor.

   ```javascript
   function updateFields() {
     sbm.util.setValue('custid_ds', sbm.util.getValue('CustomerID'));
     sbm.util.setValue('custname_ds', sbm.util.getValue('CustomerName'));
     sbm.util.setValue('custaddr_ds', sbm.util.getValue('CustomerAddress'));
   }
   ```
When you open this form in Business Process Portal, the input/output parameters (including Customer ID and Customer Address) of the DBAdapter will be displayed and available for users to view or enter input. Click the Submit button (“button1”) to open the Web services Adapter.

Using form fragments

You can add customized form elements (for example, a header, footer, divider, table or other Layout fragment) to your form, using the Form Fragments link in the Tasks pane. The Form Fragments section displays individual form elements that are not associated with any specific functionality. You can define form fragments for the following categories in Form Editor:

- **Local**: to be accessed by forms in current process only. You must save the local form fragments in the
  
  `Workspace_Home\<project_folder>\<application_folder>\resources\forms\fragments`  
  
  folder.

- **Workspace**: to be accessed by forms in current and other processes. You must save the workspace form fragments in the
  
  `Workspace_Home\.com.savvion.studio\forms\bizlogic (or bizsolo)\fragments`  
  
  folder.

The new form fragment appears in the respective category (Local or Workspace) under Form Fragments section in the Tasks pane (as shown in Figure 211 on page 309). You can then drag a specific form fragment into the Content pane and use it as a predefined element in other forms also.

You can easily develop a form using available form fragments. In the example shown in Figure 211 on page 309, the form header was created from the “Custom” type header and the footer from the “Work Time” footer in the Local form fragment category.

**Figure 211: Form Editor – Using form fragments**

![Form Editor – Using form fragments](image)

Defining header and footer fragments

By default, the Form Fragments section displays two subfolders, namely, **Headers** and **Footers**, in the Local and Workspace category folders. The Headers and Footers folders contains the Custom form fragment.
To define a custom header or footer:

1. From the Form Fragments section, expand Local (or Workspace) folder, and then Headers (or Footers) subfolder, and drag the Custom fragment to the Content pane. This inserts a custom header (or footer) table.

2. You can customize the custom (or footer) table and the controls added in the footer, according to your requirement. Click the form element to be customized, and modify the properties in the Properties view.

3. To use the customized Header (or Footer) in other forms, click File > Save As and save it to the Workspace_Home\com.savvion.studio\forms\bizlogic\fragments\Headers (or Footers) folder. After you save the form and refresh Business Process Modeler, the modified Header (or Footer) can be retrieved from the Workspace category in the Form Fragments section.

Defining other form fragments

In addition to header and footer fragment, you can customize layout controls such as table, divider, field set, and tabbed pane and add them to the form fragment section.

**Note:** You cannot add a form fragment within another form fragment.

To create a custom form fragment:

1. Open a blank Content pane in the Form Editor.
2. Add a Divider layout to your form, as described in Adding a divider on page 285.
3. Within the divider, add a table, divider, or other layout options to be used as form fragments. For information, see Defining the form’s layout on page 281.
4. Click File > Save As, and save the fragment (*.SFT) file as a local form fragment (in the Workspace_Home\<application_folder>\resources\forms\fragments folder) or as a workspace form fragment (in the Workspace_Home\com.savvion.studio\forms\bizlogic (or bizsolo)\fragments folder).
5. Restart Progress Developer Studio for OpenEdge, and the saved fragment appears in the respective category in the Form Fragments section.

Using form fragments by reference

You can use form fragments by reference for new and existing forms. The key benefit of using form fragments by reference is that you can edit a form fragment (local and workspace category) in Business Process Modeler, independent of the form in which it is used. Once saved, all the forms in the process that contains this form fragment are updated to reflect the change.

**To update a form fragment by reference:**

1. Create a form using Form Editor and add a form fragment (for example, Custom footer) to this form.
2. Save and close the form.
3. Open the form fragment for editing as follows:
• For local form fragments, from the Project Explorer, double-click the associated *.SFT located in <project_name>\resources\forms\fragments folder (for example, Custom.sft located in <project_name>\...\fragments\Footers folder) to display the selected form fragment in the Form Editor.

• For workspace form fragments, click File > Open, then navigate to the Workspace_Home\.com.savvion.studio\forms\bizlogic\ fragments\Headers (or Footers) folder. Double-click the associated *.SFT file to display the selected form fragment in the Form Editor.

4. Edit the form fragment (*.SFT) using Form Editor functionalities. For example, drag a user-defined dataslot to the Custom footer.

5. Save and close the Form Fragment file.

6. Open the form containing this form fragment.

The form now displays the updated form fragment.

Other Form Editor operations

This section describes other Form Editor operations such as adding a label and inserting a dataslot value.

Adding a label

You can add a label with a key/value pair (or as a static value) to your form. For localizing purpose, it is easier to use the key/value pair for inserted labels in your form.

You can add a label to your form using either of the following methods:

• By dragging the Label control from the Controls panel in the Form Editor tasks pane (Figure 197 on page 289). You can use this method to add a label with static text.

• By right-clicking a blank portion of your form and select the Insert > Insert Label option. The Insert Label dialog box appears, which lists the keys defined in the application-specific properties file. Select a key, then click OK to insert the label with a key. At runtime, the label is replaced with the key value, specified in the properties file.

Figure 212: Insert Label
Labels with static text are displayed on your form with an orange typeface (for example, ManagerName). Labels with keys are displayed on your form with a grey typeface (for example, %process.label). Similar to other form elements, you can drag the labels to other locations in your form and configure its properties in the Properties view.

To define properties for the added label, select the label and define the label properties in the Properties view.

**Figure 213: Form Editor – Label**

1. From the Label For property, select the control ID associated with this label. Click in the Value column box to select the control.

2. Enter the label name in the ID value box. This name is in a <div> tag that you can refer to when generating JavaScript in the Script tab. This name must be unique. You can refer to the label’s name to hide (or show) it as required by entering JavaScript in the Script tab; for example: sbm.util.hide('name_MyCustAddress');

3. In the Text box, enter the label text that you will see in your form. You can enter a static value for the label (for example, "Enter name:"). Alternatively, you can link the label value to a dataslot value. Click in the Value column box and type '@' to display the list of dataslots in the process. Select the dataslot to be linked to this label.

To modify the label text to a property key, type '%' in the Value column box to display the list of property keys (as listed in Figure 212 on page 311). Select the key to be used as the label value.

**Note:** You cannot edit the text of an inserted label directly on the form. You can only modify the label using the Properties view.

The added label is saved in the <App_Name>.properties file under the properties folder of the application. Any key/value pairs imported from other properties files are merged with the application’s properties file and saved as properties\<App_Name>.properties. You can localize the key/value pairs by manually copying the <App_Name>.properties file and renaming it as a localizable file; for instance, <App_Name_ja>.properties. You can then open this file with a text editor and change its key/value settings to labels in Japanese. For a language that does not use a Latin 1 compatible character set, you can use the Java tool native2ascii (see the appropriate Java documentation) to more easily edit non-Latin 1 *.properties files.
Inserting a dataslot value

To insert a dataslot value to your form:

1. Right-click a blank portion of your form and select the Insert > Insert Dataslot Value option to open the Dataslot Value dialog box. Use this dialog box to select a dataslot and include its value as a label that is visible in BPM Portal.

2. Select the dataslot whose value you want to insert into your form.

   To add a new dataslot, click New to open the New Dataslot dialog box; or select a dataslot from those listed and click Modify to modify it using the Modify Dataslot dialog box.

   a) Enter a value, if required, and make any other required changes.
   b) Click OK, to return to the Dataslot Value dialog box.

3. Click OK to add the Dataslot value to the Form Editor interface.

   The name of the Dataslot containing the assigned value is displayed in the Form Editor in a blue typeface, as shown in this example: newAssigncc.

Using the Overview tab

The Overview tab in the Form Editor interface provides a list of the form elements added in the Layout tab. The Overview tab provides the following benefits:

• You can filter the list of form elements on basis of the element type (for example, TextField).

• You can search a form element using its unique ID (or name). For example, if you enter the search text as "list0," the search result displays List controls with ID, "list0" and "list01." Similarly, if you search for "01," all form elements with "01" in their ID (for example, "textField01," "checkbox01, "fieldSet01") are listed.

• You can configure the properties of each form element in the Properties view without switching to the Layout tab.

• Any form element changes that you make in the Overview tab are reflected in the Layout tab as both tabs are synchronized. You can also select a form element in either tab and switch to the other tab to view the selection.

• It is easier to review the form design from the Overview tab. You can view all the form elements, including the controls inside a layout and check the dataslot binding, choice binding, validations, and events for each control.
The **Overview** tab is useful for complex forms with multiple form elements. The figure below displays a sample **Overview** tab.

**Figure 214: Form Editor - Overview tab**

The **Overview** tab displays the following default information about each form element in a tabular format:

**Table 63: Information in Overview tab**

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elements</td>
<td>Displays the name (or Field ID) of the form element. To change the name, click the row of the form element and modify the name using the <strong>Properties</strong> view.</td>
</tr>
<tr>
<td>Type</td>
<td>Indicates the type of form element. For example, Image, TextField.</td>
</tr>
<tr>
<td>Source</td>
<td>Displays the data source (if any) bound to the control. You can also set (or modify) the data source in the <strong>Data Source Binding Wizard</strong>, which appears when you click the <strong>ellipsis</strong> (...) button for the control in this column.</td>
</tr>
<tr>
<td>Target</td>
<td>Displays the data target (if any) bound to the control. You can also set (or modify) the data target in the <strong>Data Binding Wizard</strong>, which appears when you click the <strong>ellipsis</strong> (...) button for the control in this column.</td>
</tr>
<tr>
<td>Choices</td>
<td>Displays the source of choices (if any) bound to multi-select controls. You can also set (or modify) the choice source in the <strong>Choices Data Source Binding Wizard</strong>, which appears when you click the <strong>ellipsis</strong> (...) button for the control in this column.</td>
</tr>
<tr>
<td>Validation</td>
<td>Displays any validation rule configured for the control. For details on applying validation, see <strong>Applying validation</strong> on page 301.</td>
</tr>
<tr>
<td>Events</td>
<td>Displays events (if any) added to this control. For more information, see <strong>Configuring actions</strong> on page 315.</td>
</tr>
</tbody>
</table>

The table below describes the operations that you can perform in the **Overview** tab.
Table 64: Overview tab Operations

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expand or collapse the list</td>
<td>By default, all form elements are collapsed (hidden). You can expand the entire list in order to view all form elements, including the form controls inside a layout control (for example, &quot;userID&quot; and &quot;decision&quot; in Table layout). To expand the list, click the Expand all (.expand_more) icon. To collapse the list, click the Collapse all (.expand_less) icon.</td>
</tr>
<tr>
<td>Filter the list</td>
<td>Select the option from the Filter by type drop-down list. The available options are Button, Checkbox, Combobox, Date Time, Grid, Image, List, Menu, Radio, Slider, TextArea, TextField, and Tree. To remove the filtering, select the &quot;None&quot; option.</td>
</tr>
<tr>
<td>Search a form element</td>
<td>Type the text in the Search box.</td>
</tr>
<tr>
<td>Define (or modify) properties</td>
<td>Select any form element in the list to view and modify its properties in the Properties view.</td>
</tr>
</tbody>
</table>

**Configuring actions**

The Form Editor provides the **Events** tab in the **Properties** view of your form and for each added element in your form.

**Note:** The Events tab is not supported for Grid and Menu controls.

**To configure actions for your added control or layout:**

1. Add a form element (example, a Text field) to your form.
2. Select the added element, then open the Events tab in the Properties view. The list of events in the Events tab vary according to the selected element.
3. Select an event (in this case, onClick), then click Add to open the Event Name dialog box, in which you can modify the default event name, if required.

**Figure 215: Adding an Event**

![Adding an Event](image)

4. Click OK to add the event name to the Events tab.

The Graphical Event Logic (GEL) editor appears with the added event. For more information, see Using the Graphical Event Logic Editor on page 316.

- To remove the event, click Remove.
- Alternatively, to open the Graphical Event Logic (GEL) editor for the added event, click View Action.
Using the Graphical Event Logic Editor

The Form Editor provides the Graphical Event Logic (GEL) editor in the Logic tab. The Logic tab provides the list of configured actions for form elements, as discussed in Configuring actions on page 315.

**Note:** For a tutorial on using the Graphical Event Logic (GEL) editor, see Tutorial: Using the Graphical Event Logic (GEL) Editor on page 461.

**Figure 216: GEL Editor — Logic tab**

The Graphical Event Logic editor provides the following benefits:

- Provides a centralized location to configure multiple actions.
- Easy user interaction using action templates, which can be simply dragged to the editor. Additionally, you can easily rearrange and remove actions.
- Allows nested actions using the IF-THEN or IF-THEN-ELSE constructs.
- Develop custom script templates to be used as action templates.

The table below displays the GEL editor interface, whose components are described in the following table.
Table 65: GEL Editor Components

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Toolbar</td>
<td>Contains icons providing shortcuts to the following commonly used functions:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Expand all</strong> (.expand) icon, to display the details of all added actions.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Collapse all</strong> (contract) icon, to hide the details and only display the list of actions.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> For each action, you can click the toggle button [+] or [-] to collapse or expand the action respectively.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Show only functions for selection</strong> (filter) icon, to filter the list of displayed actions based on the form element you select in the <strong>Layout</strong> tab. To activate this function, select this icon and click the form element (for example, “TextField1”) in the <strong>Layout</strong> tab. The <strong>Logic</strong> tab displays only those actions for events configured for “TextField1” control.</td>
</tr>
<tr>
<td></td>
<td>• The <strong>Search</strong> box, which can be used to search action name, field ID, or any constant value. Type the text to be searched in the <strong>Search</strong> box.</td>
</tr>
<tr>
<td>Logic tab</td>
<td>Displays the list of actions (for example, “list1OnChange” and “onLoad”) added for all form elements. You can configure actions for multiple form elements from this tab. To configure an action, drag any expression template from the <strong>Expressions</strong> pane to the added action.</td>
</tr>
<tr>
<td>Expressions pane</td>
<td>Displays the list of expressions that you can drag to a defined event in the <strong>Logic</strong> tab. For information on expressions, see <strong>Adding Expressions</strong> on page 318.</td>
</tr>
<tr>
<td>Conditions pane</td>
<td>Displays the list of conditions that you can drag to an IF-THEN or IF-THEN-ELSE expression construct in the <strong>Logic</strong> tab. For information on conditions, see <strong>Adding Conditions</strong> on page 322.</td>
</tr>
</tbody>
</table>
Adding Expressions

You can use the **Expressions** pane in the Graphical Event Logic (GEL) editor to add expressions to individual actions in the **Logic** tab.

**Figure 217: Expressions pane**
You can drag and configure the following actions for events generated for each added control in your form.

- **Data Operations**: Associated with TextField, TextArea, and Combobox controls. Supports the following actions:
  - **Set widget value**: Sets the value of a control (TextField, TextArea, or Combobox) to a constant value.
    
    **Figure 218: Example**

    ![Figure 218: Example](image)

  - **Copy widget value**: Copies the value of one control (TextField or TextArea) to another control (TextField or TextArea).
    
    **Figure 219: Example**

    ![Figure 219: Example](image)

- **Effects**: Applies the selected effect (or visual enhancement) to form elements. The supported effects are:
  - **Show widget**: Shows a hidden form element. Applicable to all form elements.
    
    **Figure 220: Example**

    ![Figure 220: Example](image)

  - **Hide widget**: Hides the selected form element. Applicable to all form elements.
    
    **Figure 221: Example**

    ![Figure 221: Example](image)

  - **Enable widget**: Enables the selected form element. Applicable to all form elements.
    
    **Figure 222: Example**

    ![Figure 222: Example](image)

  - **Disable widget**: Disables the selected form element. Applicable to all form elements.
    
    **Figure 223: Example**

    ![Figure 223: Example](image)
• **Messages**: Displays a message to the user in either of the following formats:
  
  • **Popup message**: Displays the specified message in a popup message.
  
  **Figure 224: Example**

  ![Example of a popup message]

  • **Inline message**: Displays the specified message as an inline message within the form. The inline message is displayed in the specified Divider layout.
  
  **Figure 225: Example**

  ![Example of an inline message]

• **Control Structures**: Supports the following actions:
  
  • **If-Then and If-Then-Else**: Used to add an IF-THEN and IF-THEN-ELSE constructs. For the IF block, you must add a condition, as discussed in *Adding Conditions* on page 322. Depending
on your requirement, you can also create nested actions. For information on usage and examples, see Adding Conditions on page 322.

- **Stop script execution**: Used to stop the current script execution.
- **Return value**: Used to return the specified value.

**Figure 226: Example**

![Example](image1)

- **Other**: Used to add customized JavaScript and invoke adaplet for your control. The supported actions are:
  - **Custom script (one line)**: Used to add a single line of JavaScript code.
    **Figure 227: Example**
    ![Example](image2)
  - **Custom script (multi-line)**: Used to add multiple lines of JavaScript code.
    **Figure 228: Example**
    ![Example](image3)
  - **Invoke adaplet**: Used to invoke the specified adaplet.
    **Figure 229: Example**
    ![Example](image4)
  - **Invoke adaplet with callback**: Used to invoke the specified adaplet along with the callback method.
    **Figure 230: Example**
    ![Example](image5)
Adding Conditions

You can use the **Conditions** pane in the Graphical Event Logic (GEL) editor to add conditions to the IF block of an IF-THEN or IF-THEN-ELSE construct.

**Figure 231: Conditions pane**

The figure below displays an IF-THEN-ELSE construct.

**Figure 232: IF-THEN-ELSE action construct**
You can drag and configure the following conditions for the IF block in your IF-THEN and IF-THEN-ELSE actions.

- **Comparisons**: Used to compare value of an added control with static values or value in another added control, and also to check the length of the entered value. The supported conditions are:
  - **Compare with number**: Compares the value of a control (TextField, TextArea, Combobox, Checkbox, Radio, and List) to a number.
  - **Compare with string**: Compares the value of a control (TextField, TextArea, Combobox, Checkbox, Radio, and List) with a string.

**Figure 233: Example**

![Example of Compare with number](image1)

**Figure 234: Example**

![Example of Compare with string](image2)
• **Empty/Nonempty string**: Used to check if a control (TextField, TextArea, Combobox, Checkbox, Radio, and List) contains a value or not.

**Figure 235: Example**

![Example](image1)

• **String length**: Used to check the length of a string (or number of characters entered) in a control (TextField, TextArea, Combobox, Checkbox, Radio, and List).

**Figure 236: Example**

![Example](image2)
• **Compare two widgets (numeric):** Compares the numeric value of a control to the numeric value of another control. Applicable to TextField, TextArea, Combobox, Checkbox, Radio, and List controls.

Figure 237: Example

![Example](image1.png)

• **Compare two widgets (string):** Compares the string value of a control to the string value of another control. Applicable to TextField, TextArea, Combobox, Checkbox, Radio, and List controls.

Figure 238: Example

![Example](image2.png)

• **Validations:** Used to apply validation rules to all controls. The supported conditions are:
  - **Standard validation:** Used to validate if the numeric value entered in the control is a valid integer, decimal number, phone number, credit card number, US Zip code, or US state abbreviation.

Figure 239: Example

![Example](image3.png)

• **Regex validation:** Used as a regular expression (regex) validator.
Figure 240: Example

- **Other**: Used to add JavaScript conditions. The supported conditions are:
  - **Complex condition**: Used to add multiple (or cascading) conditions. You can configure this condition to perform the action if all (or any) of the conditions are satisfied.

  ![Complex condition example](image)

  **Figure 241: Example**

- **Custom condition (one line)**: Used to add a single line of JavaScript condition.

  ![Custom condition (one line) example](image)

  **Figure 242: Example**

- **Custom condition (multi-line)**: Used to add multiple lines of JavaScript condition. Depending on your requirement, you can build your JavaScript code block that returns true (or false) value.

  ![Custom condition (multi-line) example](image)

Depending on your requirement, you can create nested conditions using the IF-THEN-ELSE construct. For example,

```javascript
Condition1
---Start Block---
```
The figure below shows an example of an IF-THEN construct nested within an IF-THEN-ELSE construct.

**Figure 243: Example**

Note: For the form_onSubmit event, ensure that the associated function returns a boolean value. If the return value is "true," the form is submitted and the form workstep is completed. The form submission fails if the return value is "false." You can use the "Return value" expression with "true" or "false" as values.

### Customizing action script templates

Business Process Modeler stores all out-of-the-box action script templates (listed in the **Expressions** and **Conditions** pane of the Graphical Event Logic editor) as XML files in the `Workspace_Home\.com.savvion.studio\scripting\forms` folder.

The list of template files are:

- **main.palette**, which is the initial file to be loaded and which contains the list of template files used along with the overall structure including the categories and the labels.

- Template files used in Expression category:
  - `data_operations.palette`
  - `effects.palette`
  - `messages.palette`
• control_structures.palette
• other_expressions.palette

• Template files used in Conditions category:
  • comparisons.palette
  • validations.palette
  • other_conditions.palette

The main.palette file provides the following content to define the categories and the list of template files:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<palette version="1.0">
  ...
  <category label="$Scripting.category.expressions" path="expressions" icon="fldr_obj.gif"/>
  <category label="$Scripting.category.conditions" path="boolean" icon="fldr_obj.gif"/>

  <!-- Define tree order -->
  <category path="expressions/data_operations"/>
  <category path="expressions/effects"/>
  <category path="expressions/messages"/>
  <category path="expressions/control_structures"/>
  <category path="expressions/other"/>
  <category path="boolean/comparisons"/>
  <category path="boolean/validations"/>
  <category path="boolean/other"/>

</palette>
```

Each `<category>` tag defines the category label: expressions or conditions, as well as the path of this category in the tree. For the path of subcategories (for example, "data_operations"), you must prefix it with the category name (in this case, "expressions/data_operations"). The order of the subcategories in this file is important, as this same order is displayed in the tree.

You can create your own categories and custom action templates. You must include the same in the main.palette file so that the Graphical Event Logic editor (when launched later) displays your custom categories and action templates.

Each of the action template files (for example, data_operations.palette) includes the list of action items (in this case, "Set widget value" and "Copy widget value"). The content of the data_operations.palette file is provided below:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<palette version="1.0">
  <category label="$Scripting.category.data_operations" path="expressions/data_operations" icon="fldr_obj.gif"/>

  <item label="$Scripting.item.set_widget_value" path="expressions/data_operations" icon="template.gif">
    <grid content="EXPRESSION" collapsible="true" description="$Scripting.item.set_widget_value">
      <template>Bm.util.setValue('${widget}', '${string}');</template>
      <widgets>
        <label row="0" col="0" rowspan="1" colspan="1" id="widget" type="WIDGET">$Scripting.label.set_the_value_of</label>
        <choiceWidget col="1" id="widget" type="COMBO_BOX,TEXT_AREA,TEXT_FIELD"/>
      </widgets>
    </grid>
  </item>
</palette>
```
Each <item> tag includes the following attributes:

- **label** of the action item that is displayed in the tree (for example, “Set widget value”).
- **path**, which defines the location in the tree where the action item appears (in this case, “expressions/data_operations”).

The **grid** item within the <item> tag is used to organize the widgets. The figure below (with the highlighted widgets) is for the “Set widget value” action item.

**Figure 244: Conditions pane**

The **grid** item for the “Set widget value” action item includes the following attributes:

- **label** to add the label, "Set the value of:"
- **choiceWidget** to add a combo box. The **type=WIDGET** attribute is used to list all available (or those specified in the **filter** tag) widgets on your form.
- **label** to add the "to" label.
- **entryWidget** to add a text box or text area. The **presentation** attribute is used to specify either a text box or a text area.

**Note:** For detailed information on all tags and attributes that can be used in the .PALETTE files, see **Action script tags** on page 329.

---

**Action script tags**

You can use the following tags and attributes in the .PALETTE (out-of-the-box and custom) action script templates:

- **<palette version=1.0>**
  This is the root tag of each .PALETTE file. You must ensure that the **version** attribute value is the same in your custom template.

- **<category label="" path="" icon=""/>**
  Defines the category label (for example, "data operations") and category path (for example, "expressions/data_operations").

- **<item label="" path="" icon=""/>**
Defines the label of the action item (for example, “Set widget value”) and the path of the category under which this item is listed (for example, “expressions/data_operations”).

• <grid>

The grid container tag with the <widgets> tag, which contains the list of nested widgets.

• <template>

Located in the body of the widget and specifies the script generated by the widget. Uses ${widgetID} to refer to the value of the widget.

• <choiceWidget type="" filter="">

This is a combo box widget, which allows you to select a form control, adaplet, or a constant value. The type attribute is used to set the type of values (WIDGET, ADAPLET, or CONSTANT). In the case of the type="WIDGET" attribute value, you can add the filter attribute to specify a comma-separated list of widget types to be displayed. The available widget types are BUTTON, CHECK_BOX, COMBO_BOX, CUSTOM, DATETIME, IMAGE, LINK, LIST, NONE, OBJECT, RADIO_BUTTON, TEXT_AREA, TEXT_FIELD, XML, and DIVIDER. Alternatively, you can use the predefined widget type lists namely, *ALL (for all widgets), *NUMBER_WIDGETS (for widgets with number values), or *STRING_WIDGETS (for widgets with string values). You can use the following tags within the <choiceWidget> block:

• <choice type="" presentation="">value</choice>

You can add multiple <choice> tags to list the choices in the combo box widget. The type attribute value must match the value of the type attribute in the <choiceWidget> tag. The presentation attribute is used only for type="CONSTANT" attribute value and value is displayed in place of the value specified.

• <selection type="" presentation="">value</selection>

Used to specify the default selection among the list of choices in the combo box widget. The type attribute value must match the value of the type attribute in the <choiceWidget> tag. The presentation attribute is used only for type="CONSTANT" attribute value and value is displayed in place of the value specified.

• <string id="" value="">

You can use this tag to predefine a list of widgets that can replace the comma-separated value of the filter attribute in choiceWidget tag. This is useful when the list of widgets in the filter attribute is long and also when you can reuse this string ID at multiple locations. The id attribute is used to set a unique name (for example, "STRING_WIDGETS") and the value attribute is the comma-separated list of widgets.

• To define the <string> tag for use in a single .PALETTE file, you must add the <string> tag definition once before the <choiceWidget> tags in which you want to use it. For example, refer the <string id="MY_WIDGETS" value="CHECK_BOX,COMBO_BOX,LIST,TEXT_AREA,TEXT_FIELD"/> in any <choiceWidget> tag as <choiceWidget type="WIDGET" filter="*MY_WIDGETS"/>

by using the *ID syntax.

• Alternatively, to define the <string> tag for use in all .PALETTE files, you must add your <string> tag definition in the main.palette file. The main.palette file contains predefined <string> tags, namely, ALL, NUMBER_WIDGETS, and STRING_WIDGETS. You can refer the predefined and custom <string> tags in the <choiceWidget> tags by using the *ID syntax.
• <entryWidget presentation="" validation=""/>

This is a text box (or text area) widget. The presentation attribute is used to specify if the widget is a text box (value, "FIELD") or a text area (value, "AREA"). The validation attribute is optional and is used to limit the type of values that you can enter. The available values are FLOAT, INTEGER, or NONE. You can use the following tag within the <entryWidget> block:

• <reference type="" encoding="">value</reference>

Used to specify how to manage the value entered in the widget. The type attribute is used to set the type of value (WIDGET, ADAPLET, or CONSTANT) that you can enter in the widget. You must enter any widget ID (in your form) for type="WIDGET" attribute value, or adaplet (in your form) name for type="ADAPLET" attribute value, or any constant value for type="CONSTANT" attribute value. The encoding attribute (generally used only for constant values) is used to encode the entered string during script generation. Supported values are NONE (no encoding is performed), JAVA_STRING (to encode the entered value as a Java string), HTML (to replace characters (for example, ">") with its HTML code (in this case, "&gt;")), and HTML_JAVA_STRING (to combine HTML and JAVA_STRING encodings).

• <list label="" prompt="" allowedContent="" max=""/>

This is a list container widget used in IF-THEN and IF-THEN-ELSE constructs. The label attribute is optional and used to specify the label (for example, "If," "then," or "else"). The prompt attribute (optional) is displayed in the list body when it is empty (for example, "drop an expression here."). The allowedContent attribute is used to specify the type of templates that can be dragged on this list. Supported values are EXPRESSION (to drag expression templates) and BOOLEAN (to drag conditions that return true or false values). The max attribute value is used to specify the maximum number of action templates that can be dropped on the list.

• <operationJoin allowedContent="" label="" prompt=""/>

This is similar to the list container widget and is used for complex nested conditions joined with an operator. All attributes used in this tag are used for the same purpose as the <list> tag attributes. You must use the following tag within the <operationJoin> block:

• <operation>

Used to define the widget to select the type of join operation ("all" or "any") of the list.

• <label>

This is the label widget used to specify text.

You can use the following attributes in all widgets and the <grid> tag.

• id=""

The widget ID value, which is referred using the ${widgetID} syntax from the <template> tag.

• content=""

Used to specify the type of content returned by the widget. Supported values are EXPRESSION (to return a script expression (such as, alert('Hello');)) and BOOLEAN (to return a boolean value (true or false)).

• row="" col="" rowspan="" colspan="" fill="true|false"
The row and col attributes are used to specify the position of the widget in the grid. For example, `row="0" col="0"` places the widget in row 1 and column 1 of the grid. The rowspan and colspan attributes are optional and are used for widgets, which span more than cell in the grid. The fill attribute is optional and used to control the layout of the widget in the grid. If set to true (default), the widget displays as expanded and fills the allocated space. If set to false, the widget does not expand in the allocated space.

- collapsible="true|false" collapsed="true|false"

The collapsible attribute is used to specify if the widget can be collapsed (true) or not (false). For collapsible widgets, the collapsed attribute is used to specify the default state of the widget. The widget displays the toggle expand button (for collapsed="true") and collapse button (for collapsed=false).

- description=""

Provides the description text for a widget block. This is useful when the complete widget block is not displayed (for example, when the block is collapsed).

- searchable="true|false"

If set to true (default), the content of the widget is indexed and included when you search in the Graphical Event Logic (GEL) Editor. If set to false, the contents of this widget cannot be searched.

## Using Script tab

You can use the Script tab in the Form Editor interface, which provides the JavaScript editor to add Java or JavaScript code to your form.

### Adding script

You can add Java and JavaScript methods using the JavaScript editor in the Script tab. The JavaScript editor displays the default AutoComplete options. To populate OEBPS options, the Progress Developer Studio for OpenEdge Script Library is added with the BPM project.

**Note:** The JavaScript editor does not support parameter level AutoComplete.

1. From the Script tab, enter the functions that you want to use.
Alternatively, use the AutoComplete feature by entering **sbm.** to activate a series of JavaScript packages containing methods and utilities.

**Figure 245: Form Editor – Script tab**

2. Depending on the function selected, build your JavaScript method or utility, as required.
3. Press **ENTER** to start a new line and add more JavaScript code, if needed.

**Referencing JavaScript file**

You can insert a reference to a specific JavaScript (or *.JS file) from your local file system (Relative) or from remote files (Absolute) to your form. The JavaScript commands in the referenced file are executed when the form is loaded in BPM Portal.

To insert a reference:

1. From the **Tools** menu, click **Scripts** to open the **Scripts** dialog box (left image, **Figure 246** on page 333).

2. To insert a source *.JS file, click **New** to open the **Script** dialog box (right image, **Figure 246** on page 333). To reference a local file, click the **ellipsis** button provided to select the *.js file you require and click **Open**, returning to the **Script** dialog box. Click **OK** to add it to the list in the **Scripts** dialog box.

   To insert a reference to a remote file, enter or paste a valid URL in the **Source** box. Click **OK** to add it to the **Scripts** dialog box.
3. Click OK to insert the reference.

Adding code

You can also insert a code snippet to your form:

1. Right-click a blank portion of your form and select the Insert > Insert Code option to open the Code dialog box.
2. Enter custom code (either HTML or JSP) and click OK.

The code is displayed in the Form Editor in a distinctive background and font color with angle brackets, as in `<??>`.

To edit the code snippet, select `<??>`. Right-click the selection, then select Insert > Insert Code option to open the Code dialog box, in which you can edit the code.

Managing forms

When you publish the process, the *.XML file you have created in the Form Editor (for instance, Activity1.xml) is transformed to a *.JSP file, which is jsp folder under that application (for instance, Approval\jsp\Activity1.jsp), and published onto the server. After publishing, you can delete the generated *.JSP forms, which is typical behavior for most business users who have completed all the customization of the form in the Form Editor.

As an advanced user, however, you may want to further customize the generated JSPs before they are published onto the server. You can create an interface in the Form Editor, then save it as a custom JSP form. You can then open the JSP pages and further customize the interface to your specifications.

You can also save and reuse any forms you have created in the Form Editor.

To save a form to your Form Library:

1. From the File menu, click Save As to open the Save As dialog box.
2. Type a file name and click Save to save the form in the forms folder in Business Process Modeler.

Note: Alternatively, you can save a form to the repository, using the Save to Repository menu option from the File menu. For more information, see Saving form to repository.

This form is now displayed in the File tab of the New Form dialog box (Figure 186 on page 279) and is available to all processes in your Business Process Modeler installation.

3. To reuse a Form Editor form in another workstep, click the New Form button from the Form Editor toolbar. Select the form you want to reuse. For more information, see Using a form template on page 278.
Previewing forms

After designing a form using Form Editor, you can preview its appearance in your Web browser.

To preview a form, click the Preview icon in the Form Editor toolbar. The Form Editor renders the form preview using a generated JSP page, which is the same as the page rendered by Business Process Portal to display the form. You can interact with the form similar to how you interact with a rendered form using Business Process Portal. Additionally, clicking the Preview icon prompts you to save the form (only if your form has unsaved changes). You can save the form to reflect the changes in the form preview.

The following figure illustrates the form preview of the "Load Form" form workstep in the FormsDemo sample process.

Figure 247: Sample Form Preview

To view the form in your default Web browser, click Go to open the specified URL in your browser.

Note: Form preview resembles the native Web browser of the operating system; for example, Internet Explorer for Windows operating system.

Form preview supports:

- Rendering of all form controls.
- Rendering of data of form controls bound to static data and services. Form controls bound to dataslots are rendered, but without the dataslot values.
- Rendering of all form layouts.

Note: Form preview fails if the port number specified in the Form page of the Preferences dialog box (Using the Form page on page 94) is not available. Ensure that this port number is available or modify the port number in the Form page. Form preview in multiple sessions of Business Process Modeler must be performed using different port numbers.
Form preview does not support:

- Access to runtime data using adaplets.
- Custom scripting.
- Rendering of any effects and event actions configured for form controls.

**Caution:** Do not attempt to use JavaScript invocation code when in the Preview mode. Preview can only be used to see how the form might look in Business Process Portal.

**Note:** To preview forms on a UNIX platform, you must modify the browser handler that is located in `Workspace_Home\.com.savvion.studio\conf\resources\designer\DesignerFileTypes.xml`. For all UNIX operating systems, the argument for command should not contain quotes (shown as "").

### Invoking server side methods

You can invoke server-side methods by using Direct Web Remoting (DWR).

Perform the following procedures if you have a Java class with a function that returns a value you want to use on the client (either a Collection or primitive value). In this example, you can expose the `getProcessTemplateWorksteps(String processTemplateName)` function from the Java class `com.savvion.sbm.bizmanage.api.BizManageBean`. This class returns the list of worksteps for a particular process template.

**To do this:**

1. Create the following entry for the class and function in the `dwr.xml` located in the `$webapp\sbm\WEB-INF` folder.

   ```xml
   <create creator="new" javascript="BMB" class="com.savvion.sbm.bizmanage.api.BizManageBean"><include method="getProcessTemplateWorksteps"/></create>
   ```

   Observe the following best practices:
   
   - The JavaScript attribute value above must be unique. It should not have the same value as any of the other `create` tags in the file.
   - The class attribute value must be the class name of the containing class, including the package hierarchy.
   - The method attribute value of the include tag must be the name of the method you want to expose. You may expose multiple methods per class, by adding another include tag for each method you want to expose.

2. Modify the `sbm.js` to include the JavaScript method that will access your newly exposed java method, as shown in the following sample entry for `getProcessTemplateWorksteps`:

   ```javascript
   function getProcessTemplateWorkstepList(processTemplateName, callback) {
       DWREngine._execute('/sbm/dwr', 'BMB', 'getProcessTemplateWorksteps', callback, processTemplateName);
   }
   ```

   The above sample code defines a new JavaScript function that takes two parameters: `callback`, which is the name of the JavaScript method the engine will appropriately call back after completing, and `processTemplateName` which will be the process template name.
3. You must also create your own `DWREngine._execute()` call. The first parameter, which is the URL to send the request to, must always be `/sbm/dwr`. The second parameter, `BMB`, must match the JavaScript attribute value of the create tag you made in the `dwr.xml` file. The third parameter is the name of the method you want to call. The fourth parameter is the name of the callback method. In this case, the name of the callback method is passed on. After passing on, all of the parameter values required by your java method are taken in the order it expects them. If your method has no parameters, your call will end after the callback method. In this example, the Java method takes one parameter—the process template name.

4. Call the newly created JavaScript method from your web page. The following sample provides a usage of the `getProcessTemplateWorkstepList` method defined above.

```javascript
function updateWorksteps()
{
    var appName = getValue('application');
    getProcessTemplateWorkstepList(appName, createList);
} function createList(WsList)
{
    DWRUtil.removeAllOptions("workstep");
    DWRUtil.addOptions("workstep", allOption);
    DWRUtil.addOptions("workstep", WsList);
}
```

The HTML code is not included, but the `onChange` event of the application combo box calls the `updateWorksteps()` method. The `getValue` method retrieves the value of the application combo box. Then, the `getProcessTemplateWorkstepList` method is called, passing the process template named retrieved from the combo box and the callback method `createList`. When the DWR engine has finished executing the Java method, it calls the `createList` function passing the list that the Java method returned as its return value. Then, use the enclosed utility functions to remove all the existing options in the workstep combo box, add an ‘All’ option, and add all of that process template’s worksteps to the workstep combo box.
Configuring and running simulation

Business Process Modeler provides the Simulation project (see Creating a simulation project on page 45), which you can use to configure and run simulation for multiple processes. The Simulation project is implemented using a new simulation engine and is detached from the process workflow. You can perform simulation activities like defining and configuring scenarios, execution, and generating reports and analysis, using the new simulation perspective in Business Process Modeler.

Business Process Modeler provides two perspectives namely, Progress OpenEdge BPM Simulation perspective (for configuring simulation parameters) and the OpenEdge BPM Run Simulation perspective (for monitoring the simulation run and then generating simulation reports).

The Simulation perspectives provides:

- **Faster Testing for Design Flaws**: Allows you to quickly uncover potential problems in a process design. Instead of using a testing environment that requires two servers—one for production and one for testing—to create and test instances of the process, you can run a process in the Simulation perspectives as many times as required.

- **Estimated Cost of Resources**: Enables you to estimate the cost of the process, or the number of resources that the process will consume. You can define consumable (or single usage) resources (such as, equipment, electricity, and paper) and non-consumable (or reusable) resources (such as, computer and printer) at the process level to calculate the total estimated cost of the process.

- **Utilization of Business Calendar**: Allows you to set performer availability as defined in the organization-level business calendar. For details regarding business calendar, see Working with Business Calendar on page 391.

- **Identification of Bottlenecks**: Assigns worksteps to performing resources; for example, users or user groups for Activity worksteps. The simulation engine locks a performing resource during the execution of the workstep and releases it when the workstep is completed. Bottlenecks occur when another workstep needs the same performing resource, and must wait for the performing resource to be released from the previous workstep. Until then, the task for the next
workstep is placed in a waiting queue. The simulation engine keeps track of the number and status of instances in the waiting queues and indicates when a bottleneck occurs, that is, when a workstep has to wait for a performing resource to be released.

- **Randomization**: Supports two types of randomization techniques: randomizing the intervals between process instances; or randomizing the duration of work time for a specific workstep in the process. Business Process Modeler provides four probability distribution types: Constant distribution, Negative Exponential distribution, Normal distribution, and Uniform distribution.

For details, see the following topics:

- Adding a process model
- Exploring the Progress OpenEdge BPM Simulation perspective
- Configuring a simulation scenario
- Performing other operations
- Running a process simulation
- After completion of simulation

### Adding a process model

You can add a completed process to a simulation project from the Business Process Modeler perspective.

1. Open the Process Model (*.SPT) to be added to the simulation project.
2. From the Modeler toolbar, click **Start Simulation** icon ().
3. If the current project has been added as a process to a simulation project, then the Simulate dialog box appears, which lists the simulation projects containing the Process Model.

For a Process Model that has not been added to a simulation project, the New Simulation Project wizard (Creating a simulation project on page 45) appears, using which you can create a new simulation project (as described in Creating a simulation project on page 45) and add the current project as a process.

Figure 248: Simulate dialog box

4. From the Simulate dialog box, select an existing simulation project or BPM Simulation Project option to add the Process Model to a new simulation project.

5. Click OK.
Exploring the Progress OpenEdge BPM Simulation perspective

The Progress OpenEdge BPM Simulation perspective in the Business Process Modeler interface provides a central location where you can view or edit all the simulation parameters for each of your simulation scenarios. This section describes the Progress OpenEdge BPM Simulation perspective user interface. Figure 249 on page 342 displays a typical simulation perspective interface created for configuring a Simulation project.

Figure 249: Progress OpenEdge Business Process Modeler BPM Simulation Perspective

Figure 249 on page 342 indicates the commonly used panes and views in the Progress OpenEdge BPM Simulation perspective, and these are further described in the following table.
Table 66: Progress OpenEdge BPM Simulation Perspective components

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Explorer View</td>
<td>The Project Explorer displays the contents of active projects, including the Simulation project. It consists of a hierarchical view of all the files that make up the active project.</td>
</tr>
</tbody>
</table>
| Simulation Control Editor (or Simulation controller) | The Simulation Control editor contains the following three sections.  
  • **Scenarios**, to add and remove scenarios. For details, see Adding a scenario on page 344.  
  • **Processes**, to add (or remove) processes. It also displays the current processes, performers, and resources in the project. You can view (and edit) the properties for added scenarios, as well for each individual process (and its worksteps), connector, path, performer, and resource in the **Properties** view.  
  • **Controls**, which contains controls to start, stop, pause, exit the simulation run, as well as set the speed of the simulation. For operations, see Running a process simulation on page 362. |
| Properties view                           | Allows you to view (an edit) the properties for a selected item in the Simulation Control Editor. You can define (or edit) properties for the current simulation project, as well as for each of the added processes, worksteps, paths, performers, and resources. You can define the objectives for each of these components (see Defining objectives on page 358) and manage resources (see Managing resources on page 357). |
| Processes view                            | Consists of a **Process** tab for each of the added processes. Each Process tab comprises of the:  
  • **Show Process** drop-down list, which lists the current process and inline subprocesses (if any). For information regarding inline subprocess, see Defining an inline subprocess on page 124.  
  • **Diagram** view, which displays the process template diagram. |

Configuring a simulation scenario

Business Process Modeler creates a single “default” scenario for each simulation project. You can add scenarios to the simulation project and configure simulation settings for each scenario independently. You can also import and export scenario information between simulation projects. After running simulation for each scenario, you can compare the results of the selected scenarios.

Configuring a simulation scenario consists of:

• **Simulation configuration**: includes setting the simulation start (and stop) date and time. For details, see Modifying simulation settings for a scenario on page 344.

• **Process configuration**: includes configuring simulation properties for each process (simulation start and stop date and time, number of instances, randomization method) and for each process workstep (specifying worktime with time deviation method, specifying the resources, specifying the probabilities in case of multiple outgoing links and for Decision gateway).
• **Performer and Resource configuration**: includes setting the cost of each workstep performer, as well as defining consumable and non-consumable resources and their properties.

• **Objectives definition**: includes setting simulation objectives for the process, workstep, path, and for the usage of performers and resources.

**Note:** Any process or resource that you add in the Simulation perspective, is independent of simulation scenarios.

The following sections describe how to add a scenario, configure simulation properties for individual scenarios, and import and export scenario information.

### Adding a scenario

Business Process Modeler enables you to create simulation scenarios in which you can change resources, the size of a group, the probabilities out of Decisions, and other simulation parameters.

**To define a new scenario:**

1. From the **Scenarios** section in the **Simulation Control Editor**, click the **Add** icon to open the **Add Scenario** dialog box.

   **Figure 250: Add Scenario**

   ![Add Scenario](image)

2. Enter a name for the scenario in the **Name** box. This is mandatory.

3. From the **Using scenario** drop-down list, select any of the existing scenarios to be used as a template for the new scenario. Resources, as well as processes and performers, defined in the scenario used are provided in the new scenario.

4. Click **OK** to add the new scenario to the Scenario drop-down list. Create as many additional scenarios as required.

   To remove the scenario displayed in the Scenario drop-down list, click **X**.

### Modifying simulation settings for a scenario

You can modify the simulation settings for the default scenario, as well as for any added scenarios. By default, any scenario created (see the preceding section) as a copy of the “default” scenario has the same simulation start and stop time as the default scenario.

**To modify the simulation settings for a particular scenario:**

1. From the **Scenarios** drop-down list in the **Scenarios** section (in the **Simulation Control Editor**), select the scenario to be configured.

2. From the **Processes** section, click the current simulation project (in this case, **TestSim**). In the **Properties** view, the **General** tab displays the details (name, label, and description) of the current simulation project.
3. Click the **Scenario** tab to view the simulation settings (including simulation start time and end time) for the selected scenario (as described in Step 3 in *Creating a simulation project* on page 45 section). Modify the settings, as required.

4. From the **Objectives** tab, add the objectives for the selected scenario. For details, see *Defining objectives* on page 358.

5. From the **Calendar** tab, you can view the project calendar settings configured for the current scenario (as described in Step 5 in *Creating a simulation project* on page 45 section). Modify the settings, as required.

**Importing and exporting a scenario**

You can import and export scenario information between simulation projects using the **Simulation Control Editor**. You can also import scenario real-time data from the server. Importing and exporting scenario information enables you to:

- Modify scenario information using any XML Editor, independent of Business Process Modeler.
- Replace the current scenario settings with the imported scenario information, thus eliminating the need to configure each parameter in the scenario.

**Exporting a scenario**

To export scenario information to a simulation scenario file (*SSF*):

1. From the **Scenarios** drop-down list in the **Scenarios** section (in the **Simulation Control Editor**), select the scenario to be exported.

2. Click the **Export current scenario information** (✍️) icon to open the **Export To** dialog box.

3. Modify the default file name, if required, then click **Save** to save the *SSF* file in the `Workspace_Home\<Application_Name>\exports` folder.
Importing a scenario

You can import scenario information to the current scenario:

1. From the Scenarios drop-down list in the Scenarios section (in the Simulation Control Editor), select the scenario to be replaced.

2. Click the Import information into current scenario (glyphicon glyphicon-import) icon to open the Import Scenario Data wizard.

   Figure 251: Import Scenario Data wizard – Page 1

   ![Import Scenario Data wizard](image)

3. From the first page of the Import Scenario Data wizard, use either of the following sources from which you want to import data.
• Select the **Local file** option to import scenario data from the local *.SSF* file, then click **Next**.

**Figure 252: Importing from Local file**

![Import Scenario Data](image1)

a) From the **Import file details** page, click the **ellipsis** button beside the **Select file** box, then select the *.SSF* file containing the required scenario information.

b) Click **Finish** to import the selected scenario information to your simulation project.

• Select the **Server (Real-Life)** option to import real-time scenario information from the server, then click **Next**.

**Figure 253: Specifying Server Connection Details**

![Import Scenario Data](image2)

a) Specify the server IP address and port number in the respective boxes.

b) Enter the user name and password required to connect to the server.

c) Modify (if required) the start date and end date in the **Start Date** and **End Date** boxes by clicking the adjoining **ellipsis** button. For details, see **Specifying a date** on page 361. Data of all the process instances that have been completed in this date range (between start and end date) are retrieved from the server.

d) Click **Connect** to verify the connection to the server.
e) Click **Next** to display the Scenario Difference Page.

**Figure 254: Scenario Difference page**

![Scenario Difference Page](image)

The left pane of the **Scenario Difference** page displays the simulation-related components of the current scenario including the added processes (and their individual worksteps), subprocesses (if any), performers, and resources. This structure is similar to the **Processes** section of the **Simulation Control Editor**.

f) Click any component to view the difference in the current and real-time data in the table provided in the right pane.

**Table 67 on page 348 describes each column in the real-time data table.**

**Table 67: Real-time data table**

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the simulation-related properties for the selected component. For example, instance count, duration, randomization type, and start time for a process (as shown in <strong>Figure 254</strong> on page 348).</td>
</tr>
<tr>
<td>Old Value</td>
<td>Displays the current value.</td>
</tr>
<tr>
<td>New Value</td>
<td>Displays the real-time value. Once imported, this value replaces the old value.</td>
</tr>
</tbody>
</table>

g) Click **Finish** to import the real-time scenario data to your current scenario.

**Note:** After running the simulation, you can also generate a real-time comparison report which compares the result of a simulation run with the server real-time data. For more information, see **Generating Real-time Comparison Report** on page 378.
Configuring simulation parameters for a process

You can use the **Processes** section in the **Simulation Control Editor** to add processes as well as configure simulation parameters for each of the added processes and its worksteps. You can also configure simulation properties for each of the workstep paths in the process.

**Figure 255: Processes section in Simulation Control Editor**

To add and configure simulation parameters for each process for a particular scenario:

1. From the **Processes** section, click the **Add** icon to open the **Required Project Selection** dialog box, which lists the current projects in Business Process Modeler.
2. Click the project to be added, then click **OK**.

   The selected process is added to the list of processes under the **Processes** folder (Figure 255 on page 349). To remove a process from the simulation project, select the process from the list of processes, then click **Remove**. You cannot remove a subprocess.

**Note:** In case you modify the process template after adding the process to the simulation project, select the process, then click the **Refresh** icon to reflect the changes made in the added process. You need to reconfigure the simulation settings for any process workstep, which has been impacted by the change.

3. From the **Scenarios** drop-down list in the **Scenarios** section (in the **Simulation Control Editor**), select the scenario to be configured.
4. Expand the **Processes** folder (Figure 255 on page 349), then click the process (for instance, Process1) whose simulation properties you want to configure. In the **Properties** view, the **General** tab displays the process details (name and label) of the selected process. You cannot modify any information in this tab.
5. Click the **Instances** tab to view the simulation settings (process start time, instances, and randomization type) for the selected process, as configured when adding the process in the **New Simulation Project** wizard (Creating a simulation project on page 45). Modify (if required) the settings, as described in Step b in **Creating a simulation project** on page 45 section.
**Note:** In a simulation project with multiple processes, if you do not want to include a particular process in the simulation run, enter “0” as the process instance count.

**Figure 256: Properties view for a Process – Instances tab**

![Properties view for a Process – Instances tab](image)

**Note:** The Instances tab does not appear for Subprocesses.

6. From the **Objectives** tab, add the objectives for the selected process. For details, see [Defining objectives](page 358).

7. Use the **Calendar** tab to include calendar restrictions for the current process. For details, see [Setting calendar](page 360).

Alternatively, you can view the entire list of added processes and subprocesses in a tabular format by clicking the **Processes** folder in the **Processes** section of the **Simulation Control Editor**.

**Figure 257: List of added Processes in the Properties View**

![List of added Processes in the Properties View](image)

**Figure 257** on page 350 displays the list of main processes in the **Master Processes** section table and subprocesses (if any) in the **Auxiliary Processes** section table. You can add, modify, or delete a main process by clicking the appropriate button.

You can also configure simulation properties for each workstep in the process and for each of the workstep paths and connectors, as described in the following sections.
For worksteps

You can configure simulation settings for individual worksteps in a process. Workstep configuration is applied to the selected scenario only.

1. From the Scenarios drop-down list in the Scenarios section (in the Simulation Control Editor), select the scenario to be configured.
2. Expand the Processes folder (Figure 255 on page 349), then expand the process (for instance, Process1). Expand the Worksteps folder to display the list of contained worksteps.
3. Click the workstep to be configured. Alternatively, you can click the workstep in the Diagram view.

In the Properties view, the General tab displays workstep details including name and label. You cannot modify any information in this tab. The rest of the subtabs vary depending on the type of the selected workstep, and are described in Step Table 68 on page 351.

Table 68: Configuring a workstep

<table>
<thead>
<tr>
<th>Workstep type</th>
<th>Tabs in the Properties view</th>
<th>Description</th>
</tr>
</thead>
</table>
| Activity/ Adapter/ Subprocess worksteps | Task | For an Activity or Adapter workstep, review the workstep performer in the Performer box. From the Time Distribution section (available only for Activity worksteps), you can modify randomization method for workstep instances (see Choosing a randomization method on page 361).

**Note:** You cannot configure the worktime distribution for an adapter workstep. By default, it is set to 1 second.

For an Activity workstep with a Timeout link, use the Timeout section to enable the workstep timeout functionality. You can specify the timeout duration in terms of hours, minutes, and seconds in the respective boxes.

For a Subprocess workstep, select the Using worktime option and specify the randomization method for the workstep instances in the Worktime Distribution section.

<table>
<thead>
<tr>
<th></th>
<th>Probability</th>
<th>Allows you to configure the probability setting for worksteps with multiple outgoing links. For details, see Setting the probabilities on page 354.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Objectives</td>
<td>Allows you to add objectives for the workstep (example, the total completion time for all instances of the workstep must not exceed 10 hours). For details, see Defining objectives on page 358.</td>
</tr>
<tr>
<td></td>
<td>Resources</td>
<td>Allows you to add resources (consumable and non-consumable) and the quantity likely to be consumed for this workstep. For non-consumable resources, you cannot specify a consumption count more than the available count.</td>
</tr>
<tr>
<td></td>
<td>Start/ End workstep</td>
<td>Objectives</td>
</tr>
</tbody>
</table>
### Workstep type | Tabs in the Properties view | Description
---|---|---
Decision gateway | Probability | Allows you to configure the probability setting for the default link and the conditional link. For details, see Setting the probabilities on page 354.
 | Objectives | Allows you to add objectives for the Decision gateway. For details, see Defining objectives on page 358.
Other Gateway types | Objectives | Allows you to add objectives for the particular gateway. For details, see Defining objectives on page 358.

Alternatively, you can configure (or modify) simulation settings for worksteps by clicking the Workstep folder for the selected process in the Processes section of the Simulation Control Editor.

**Figure 258: List of Worksteps in the Properties View**

Step Figure 258 on page 352 displays the Activity Worksteps section table (containing all the activity worksteps), Adapter Worksteps section table (containing adapter worksteps), Decision Worksteps section table (containing decision gateways), Inline Worksteps section table (containing inline subprocess worksteps), Subprocess Worksteps section table (containing subprocess worksteps), and Other Worksteps section table (containing additional worksteps such as Start and End). You can modify the simulation settings for any workstep by selecting it and then clicking the adjoining Modify button.

**For connectors**

You can use workstep connectors to configure a delay between the connecting worksteps. This introduces a time delay between workstep instance creation during the simulation run. This configuration is applied to the selected scenario only.

**To configure a connector delay:**

1. From the Scenarios drop-down list in the Scenarios section (in the Simulation Control Editor), select the scenario to be configured.
2. Expand the Processes folder (Figure 255 on page 349), then expand the process (for instance, Process1). Expand the Connectors folder to display the list of workstep connectors.
3. Click the connector to display the connector properties in the Properties view. In the Properties view, the General tab displays connector details including name, label, type (default or conditional) as well as the names of the connecting worksteps. You cannot modify any information in this tab.

4. Use the Delay tab to configure the connector delay.

**Figure 259: Properties view for a Connector – Delay tab**

- From the Delay Distribution section, choose a method of randomizing the duration of delay between workstep instances in the simulation. For information regarding randomizing methods, see Choosing a randomization method on page 361.

### For paths

A simple process consisting of a couple of Activity worksteps with Start and End worksteps, has an isolated path from start to completion. A complex process (for instance, a manufacturing system) can have a network of paths providing multiple routes between the same two locations. For such cases, Business Process Modeler provides path analysis functionality in order to determine the shortest path or the least expensive path.

You can define paths for individual processes in a simulation project. You can configure these added paths differently for different scenarios.

**Note:** A defined path must contain more than one Activity workstep.

#### To define a path:

1. Expand the Processes folder (Figure 255 on page 349), then expand the process (for instance, Process1).
2. Click Paths to display the Paths table in the Properties view.

   This table displays the list of defined paths for this project. You can add new paths, and modify or delete existing paths.

3. Click Add to open the Path dialog box.
4. Enter the name and a brief description of the path in the Name and Description boxes respectively.
5. The Available Choices section displays a list of all the elements contained in the process. Select the element that denotes the starting location of your path, then click the Move Left (/xml) icon to add the element to the Current Path section. The Successors section on the right, now displays the elements following the added element in the process.
6. Continue adding elements to build the path.
To delete a path element, select the element in the Current Path section, then click \textit{X}. You can only remove the last added element. To reset the path elements, click \textit{\textdegree}.

7. After building the path, click \textbf{OK} to add the path to the Paths table in the Properties view.

8. After you define a path, expand the Paths folder for this process to view the added path. Click the path to view the path properties in the Properties view, where you can modify the path properties, as well as define objectives. To define path objective for a particular scenario:
   a) Select a scenario from the Scenario drop-down list in the Simulation Control Editor.
   b) Click the added path to be configured.
   c) From the Objectives tab of the Properties view, add the objectives for the selected path. For details, see Defining objectives on page 358.

**Setting the probabilities**

You can use the Probability tab in the Properties view to review or change probability settings for:

- Activity worksteps with multiple outgoing links.
- Activity worksteps with a defined loop condition (see Defining a Loop condition on page 226).
- Decision gateways.

To configure the probability settings for any of these worksteps:

1. Select a scenario from the Scenario drop-down list in the Simulation Control Editor.
2. Click the Activity workstep or Decision gateway to be configured, then click the Probability tab in the Properties view.

**Figure 260: Setting Probability for a Simulation**

![Properties view](image)

| Note: | To perform this operation, you need to understand how to configure multiple links originating from a workstep. For details, see Using multiple links on page 55. |

The Probability tab displays the Default bar, which represents default links with no specified condition. Additionally, it may display the following bars:

- **Conditional** bars, which represent links with a specified condition.
- **Looping** bar (only for Activity worksteps), if the workstep contains a loop condition.
- **Compensation** bar (only for Activity worksteps), which represents links with a compensation flow.

The sum of the Probability values (on the right side) always equals 100%.
3. You can modify the probability value of all bars, except the Default bar. Use your cursor to move any of the probability bars to the right or left, or specify a value in the adjoining box. This automatically adjusts the value in the Default bar.

4. For worksteps with multiple conditional links, you can also add a **Multiple Condition** bar, which represents a condition when multiple conditional links are executed (that is, when the condition in all the links are satisfied).

**Note:** You cannot add a multiple condition bar for an Exclusive Decision gateway.

**Figure 261: Setting Probabilities for Workstep with multiple conditional links**

![Diagram showing setting probabilities for workstep with multiple conditional links]

- a) Click **Add** to open the **Conditional links** dialog box, which displays all the conditional links from the workstep.
- b) Click the conditional link to be added. Press **CTRL**, and then click the other conditional link to be added. You need to select at least two conditional links to enable the **OK** button.
- c) Click **OK** to add a multiple condition bar to the **Probability** tab. You can now adjust the probability value for this bar, as explained in Step 3.

To remove a multiple condition bar, select the bar and then click **Remove**. You cannot remove any of the other probability bars.

**Managing performers**

You can define the simulation properties for the assigned performers using the **Performers** section in the Simulation Control Editor. You can review (or modify) the settings for human performers (individual and group) and adapter performers. Performer configuration is applied to the selected scenario only.

**To configure simulation settings for a performer:**

1. From the **Scenarios** drop-down list in the **Scenarios** section (in the Simulation Control Editor), select the scenario to be configured.

2. From the **Processes** section in the Simulation Control Editor, expand **Performers** to display four subfolders namely, **Systems** (containing any adapter performers), **Individuals** (containing single human performers), **Groups** (containing a group of human performers), and **Queues** (containing any queue performers).

   In case you add (or modify) groups in the User Management tool after adding the process (containing the group performer) to the simulation project, select the **Groups** folder, then click the **Refresh** (✓) icon to reflect the changes made in the group members.

3. Expand these folders to view the respective performers assigned to the worksteps in the processes added to this project.

4. Click each performer to review the current performer information in the **Properties** view.
• For an Adapter performer, review the name of the performer in the General tab (see Figure 262 on page 356). Enter a number (greater than 0) in the Limit box to specify the maximum number of times you want to invoke the adapter. Enter an amount (greater than 0) in the Cost box that provides a value for the cost per adapter invocation. In this case, the adapter is invoked three times at the cost of $5.75 per invocation; any additional invocations must wait until one of the three invocations has completed its task. Use the Objectives tab to add objectives for the performer, as described in Defining objectives on page 358.

Figure 262: Configuring an Adapter Performer

• For an individual performer, review the name of the performer in the General tab (see Figure 263 on page 356). Enter the hourly cost for the performer in the Per Hour box. Use the Objectives tab to add objectives for the performer, as described in Defining objectives on page 358. Use the Calendar tab to add calendar restrictions for the performer, as described in Setting calendar on page 360.

Figure 263: Configuring an Individual Performer

• The Properties view is similar for group and queue performers. Review the name of the performer in the General tab. In the Members tab (see Figure 264 on page 356), select the From User Management option to configure the performer settings as defined in the User Management tool. Alternatively, select the Virtual option to enter a virtual group size in the Number of Members box and the hourly cost of each group member in the Cost (per Hour) box. You need to specify a minimum virtual group size more than or equal to 1.

Figure 264: Configuring a group or queue performer
Note: The From User Management option is only enabled for groups and queues existing in the User Management tool and containing group members. For more information, see Managing groups on page 387 and Managing queues on page 389.

Note: You can use the Calendar tab to add calendar restrictions for the group or queue performer, as described in Setting calendar on page 360.

Alternatively, you can configure (or modify) simulation settings for performers by clicking the Performers folder in the Processes section of the Simulation Control Editor.

Figure 265: List of performers in the Properties view

Figure 265 on page 357 displays the Individual Performers section table (containing individual performers), System Performers section table (containing adapter performers), Group Performers section table (containing group performers), and Queue Performers section table (containing queue performers). You can modify the simulation settings for any performer by selecting it and then clicking the adjoining Modify button.

Tip: You can also click any of the four subfolders under the Performers folder (in the Simulation Control Editor) to view and modify the simulation settings for the respective performer type.

Managing resources

You can define resources independent of any process or scenario. Resources are any expendable items, but do not include personnel. Business Process Modeler supports two types of resources, namely:

- **Consumable** resources: These are the resources that are used up during the simulation and may typically include either Services (such as electricity, gas) or Supplies (such as paper, staples). These resources are usually modeled either as a function of time or associated with an event, such as the completion of an operation.

- **Non Consumable** resources: These are shared resources with multiple activities waiting for the same resource. These type of resources are most likely to affect system performance, as they may cause bottlenecks. An example of a non-consumable resource is machinery.
To define a new resource (consumable or non-consumable):

1. From the Processes section in the Simulation Control Editor (Figure 255 on page 349), expand Resources to display the Consumables and NonConsumables folders.

2. Click each of the respective folders to display the Consumables or the Non Consumables table in the Properties view. This table displays the list of defined resources for this project. You can add new resources, modify existing resources, and delete resources from the respective Properties view.

3. Click Add in the respective Properties view to open the Consumable Resource dialog box, or the Non Consumable Resource dialog box.
   a) For a consumable resource, enter the appropriate data in the Name, Unit, and Cost boxes. The name of the resource cannot contain any blank spaces. Data entered in the Unit box must contain only alphabetic characters.
   b) For a non-consumable resource, enter the appropriate data in the Name, Availability, and Cost boxes. Enter a number in the Availability box, which indicates the quantity of the resource that can be shared.

4. Click OK to save the resource and add it to the respective table in the Properties view.

5. After you define a resource, expand the Consumables or NonConsumables folder (in the Simulation Control Editor) to view the added resources. Click the resource to view the resource properties in the Properties view.
   a) From the General tab, modify (if required) the respective resource properties.
   b) From the Objectives tab, add the objectives for the selected resource. For details, see the following Defining objectives on page 358 section.

Alternatively, you can view the entire list of added resources in a tabular format by clicking the Resources folder in the Processes section of the Simulation Control Editor.

**Figure 266: List of Resources in the Properties View**

Figure 266 on page 358 displays the Consumable Resources section table (containing list of consumable resources) and Non Consumable Resources section table (containing list of non-consumable resources). You can add, modify, or delete a consumable or non-consumable resource by clicking the appropriate button.

**Defining objectives**

Business Process Modeler allows you to define objectives, which helps achieve goals for optimum time, cost, and resource utilization. You can define objectives at process, workstep, and path levels, as well as for defined performers and resources. Any violation of these objectives during simulation execution are listed in the Violations view in the Simulation perspective.

Business Process Modeler provides a predefined set of properties for objectives. You can define a condition for these properties, whose values are set according to cost, count or a time duration.
To define an objective:

1. Type the name of the objective in the **Name** box.

2. From the **Condition** drop-down list, select the operator. The available options are: == (equal to), != (not equal to), < (less than), > (greater than), <= (less than or equal to), and >= (more than or equal to).

3. In the adjoining text box, enter the condition value in terms of a number (for count- or cost-based objectives) or in terms of hours, minutes, and seconds (for duration-based objectives).

Table 69 on page 359 provides the complete list of predefined properties and their description.

<table>
<thead>
<tr>
<th>Property name</th>
<th>Available for</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completion Duration</td>
<td>Simulation project only</td>
<td>Sets the duration for the complete simulation.</td>
</tr>
<tr>
<td>Created instances count</td>
<td>Process and workstep</td>
<td>Sets the number of instances created for a process or workstep.</td>
</tr>
<tr>
<td>Completed instances count</td>
<td>Process and workstep</td>
<td>Sets the number of completed instances for a process or workstep.</td>
</tr>
<tr>
<td>Terminated instance count</td>
<td>Process and workstep</td>
<td>Sets the number of instances terminated for a process or workstep.</td>
</tr>
<tr>
<td>Aggregate completion time</td>
<td>Process and workstep</td>
<td>Sets the total completion time of all instances of a process or workstep.</td>
</tr>
<tr>
<td>Average completion time</td>
<td>Process and workstep</td>
<td>Sets the average completion time of all instances of a process or workstep.</td>
</tr>
<tr>
<td>Total Task Count</td>
<td>Individual Performer</td>
<td>Sets the total count of tasks performed by the performer.</td>
</tr>
<tr>
<td>Total Task Duration</td>
<td>Individual Performer</td>
<td>Sets the total duration of tasks performed by the performer.</td>
</tr>
<tr>
<td>Total Cost</td>
<td>Performer and Resource</td>
<td>Sets the total cost of a performer or resource.</td>
</tr>
<tr>
<td>Average Cost per Task</td>
<td>Individual Performer</td>
<td>Sets the average cost of a performer for each completed task.</td>
</tr>
<tr>
<td>Total Utilization</td>
<td>Individual Performer and Resource (Non-consumable)</td>
<td>Sets the utilization value of a performer or a non-consumable resource.</td>
</tr>
<tr>
<td>Total Count</td>
<td>Resource (consumable)</td>
<td>Sets the total count for a consumable resource.</td>
</tr>
<tr>
<td>Average Cost</td>
<td>Resource (Non-consumable)</td>
<td>Sets the average cost of a non-consumable resource.</td>
</tr>
<tr>
<td>Total Invocations</td>
<td>Adapter Performer</td>
<td>Sets the number of adapter invocations.</td>
</tr>
<tr>
<td>Traversal Count</td>
<td>Path</td>
<td>Sets the total count for traversing a path.</td>
</tr>
<tr>
<td>Summed Traversal Time</td>
<td>Path</td>
<td>Sets the total time for traversing a path for all process instances.</td>
</tr>
<tr>
<td>Average Traversal Time</td>
<td>Path</td>
<td>Sets the average time for traversing a path for all process instances.</td>
</tr>
</tbody>
</table>

The defined objective is added to a table in the **Objectives** tab of the **Properties** view in the Simulation perspective. You can also modify or remove the objective from the table by clicking the respective buttons.
Setting calendar

Business Process Modeler enables you to set calendar restrictions for each project added to your simulation project, as well as for each human performer (individual and group) and queue performer in your simulation project.

You can use the Calendar tab in the Properties view for the process or performer to configure the calendar restriction. Figure 267 on page 360 displays the Calendar tab in the Properties view for a process.

Figure 267: Properties view for a Process – Calendar tab

You can select any of the following options to configure calendar settings:

• **None** (without any calendar restrictions)

• **User management** to include calendar restrictions as specified in the calendar created using the Business Calendar tool, as discussed in Working with Business Calendar on page 391.

• **Project** to include calendar restrictions as specified in the project calendar for the current simulation project.

• **Local** to configure process-specific calendar settings using the Workdays and Sessions sections provided. This operation is similar to configuring the project calendar, as discussed in Step in Creating a simulation project on page 45 section.

Performing other operations

This section describes commonly used simulation-related operations such as selecting a randomizing method and selecting a date.
Choosing a randomization method

Business Process Modeler supports two types of randomization in the Simulation perspective; namely, randomizing the time of intervals between process instances; or randomizing the duration of work time for a specific workstep in the process.

- To randomize the time interval between process instances, use the **Instances** tab of the **Properties** view for a process, as described in Step 5 in Configuring simulation parameters for a process on page 349 section.

- To randomize the work time for a specific workstep instance, use the **Task** tab of the **Properties** view for an Activity, Adapter, or Subprocess workstep, as described in Step 3 in For worksteps on page 351 section.

Business Process Modeler provides four probability distribution types, which are described below:

<table>
<thead>
<tr>
<th>Distribution Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant</td>
<td>Use this option to maintain the same interval between instances in the simulation. You can define the mean interval between instances in the adjoining boxes in terms of hours, minutes, and seconds. This is used when the duration is fixed, and does not change over time.</td>
</tr>
<tr>
<td>Negative Exponential</td>
<td>A continuous probability distribution that is often used to characterize the time between events or the durations of activities. It does not assume a predetermined pattern of distribution. A Negative Exponential distribution is used when the probability decreases over time.</td>
</tr>
<tr>
<td>Normal</td>
<td>Use this option to define the symmetrical, bell-shaped distribution pattern that is often used for a simulation. This distribution pattern is determined by its mean and the standard deviation. You can use the adjoining <strong>Mean</strong> and <strong>Std-dev</strong> boxes to set the mean and standard deviation in terms of hours, minutes, and seconds. In a typical normal distribution, 70% of results fall between one standard deviation above and one standard deviation below the mean. The standard deviation is a measure of how tightly items are clustered around the mean in a set of data. A low standard deviation means that the results are tightly clustered; a high number that the distribution is widely spread. <strong>Note:</strong> If the Standard Deviation setting is more than the Work Time value for a workstep and results in the range of the deviation being a negative value, then only the absolute value of the deviation (not the negative value) is used for running the simulation.</td>
</tr>
<tr>
<td>Uniform</td>
<td>A discrete uniform distribution option that represents a uniform distribution pattern within a finite set of possible values. The possible values are set within the minimum and the maximum value specified in hours, minutes, or seconds. This is used when the process instance count (or the worktime duration of a workstep instance) is fixed over a specified range of values.</td>
</tr>
</tbody>
</table>

Specifying a date

You can enter (or modify) a date (and time) in the **At time** box for a Date, for instance, when specifying the simulation start and stop time for a simulation project and for process worksteps.
To specify the date and time:

1. Select the **At time** option (if you are specifying a date for the first time) to display the current system date and time in the adjoining text box.

2. To specify another date (or time), click the **ellipsis** button beside the **At time** option to open the **Calendar** dialog box, which display the current month’s calendar. By default, the calendar highlights today's date and current time.

3. To select a date from the displayed month, click the desired date.

4. To display the calendar for another month, use the arrows provided beside the month. Click the arrow pointing right for the next month, or the arrow pointing left for the previous month. Alternatively, to select any other month, you can click on the current month and then select the desired month from the list.

5. Similarly, to display the calendar for another year, click the current year and then type the desired year. Alternatively, you can use the adjoining controls to increment (or decrement) the year by one. Click outside the year control to accept the changed year.

6. To change the default time from the **Time** box, click the hour, minute, and second boxes and type the respective values. Alternatively, you can use the adjoining controls to increment (or decrement) the respective value. You can also use these controls to switch between **AM** and **PM**.

7. Click **OK** to display the selected date and time in the text box beside the **At time** option.

**Running a process simulation**

After designing the process simulation in the **Progress OpenEdge BPM Simulation** perspective, you can run the simulation in the **OpenEdge BPM Run Simulation** perspective to discover potential design flaws. For example, you can now quickly identify such problems as loops that repeat too frequently or bottlenecks in the process.
Each simulation run represents a simulation for a set of resources. It could have single or multiple processes running in parallel.

**Figure 268: Business Process Modeler Simulation Runtime perspective**

![Simulation Runtime Perspective](image)

**Figure 268** on page 363 indicates the commonly used panes and runtime views in the OpenEdge BPM Run Simulation perspective, and these are further described in the following table.

**Table 71: OpenEdge BPM Run Simulation Perspective components**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| Simulation Control Editor (or Simulation controller) | The Simulation Control editor contains the following additional sections.  
  - **Simulation Status**, which displays the progress status of the simulation project.  
  - **Generate**, to view the results of a simulation run for a scenario and to generate a report, as described in *Generating a simulation report* on page 371. |
<p>| Individuals view | Allows you to view the runtime performance of the individual human performers (including group members) during the simulation run. Performance parameters include total task count and usage time, total cost, and cost per hour for the defined performers. |
| Systems view | Allows you to view the runtime performance of the system (or adapter) performers during the simulation run. Performance parameters include cost per invocation, number of invocations, and total cost for system performers. |
| Consumables view | Allows you to view the consumable resource usage during the simulation run. |</p>
<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NonConsumables view</td>
<td>Allows you to view the non-consumable resource usage during the simulation run.</td>
</tr>
<tr>
<td>Violations view</td>
<td>Allows you to view any violations in the defined objectives after completion of the simulation run.</td>
</tr>
<tr>
<td>Recommendations view</td>
<td>Lists any workstep bottlenecks due to non-availability of performers and resources after the completion of the simulation run, and recommends a possible solution for the same.</td>
</tr>
<tr>
<td>Processes view</td>
<td>Consists of a <strong>Process</strong> tab for each of the added processes. Each Process tab comprises of the:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Diagram</strong> view, which displays the process template diagram, and indicates the status of each workstep across all instances during the simulation run.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Details</strong> view: You need to click <strong>Details</strong> to display this view. For details, see Viewing process information on page 365.</td>
</tr>
</tbody>
</table>

**To run a process simulation:**

1. From the **Scenarios** drop-down list in the **Scenarios** section (in the Simulation Control Editor), select the scenario for which you want to run the simulation.

2. From the **Controls** section of the Simulation Control Editor, click **Play** to begin the process simulation.

3. If you are currently not in the OpenEdge BPM Run Simulation perspective, then Business Process Modeler prompts you to switch to this perspective. Click **Yes** to run the process simulation in the OpenEdge BPM Run Simulation perspective. Alternatively, click **No** to run the process simulation in your current perspective. If you select the **Remember my decision** checkbox, then Business Process Modeler does not display any prompt and executes your selected command the next time you run a process simulation.

**Note:** You can modify the above selected configuration for running the process simulation using the **Simulation > UI** page in the **Preferences** dialog box. For information, see Using the Simulation page on page 95.

During the simulation run, you can perform any of the following operations from the **Controls** section of the Simulation Control Editor:

**Table 72: Simulation controls**

<table>
<thead>
<tr>
<th>Control</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pause ( ⏪ )</td>
<td>To pause the simulation. Click <strong>Play</strong> to resume the simulation.</td>
</tr>
<tr>
<td>Stop ( ⏪ )</td>
<td>To stop the simulation and return to the start of the simulation.</td>
</tr>
<tr>
<td>Exit ( ✈️ )</td>
<td>To stop the simulation and exit from the <strong>OpenEdge BPM Run Simulation</strong> perspective. Process Modeler prompts you to switch to the Progress <strong>OpenEdge BPM Simulation</strong> perspective. You can click the appropriate button, similar to as described in Step 3 above.</td>
</tr>
<tr>
<td>Speed control</td>
<td>To control the simulation speed between low and high.</td>
</tr>
</tbody>
</table>
You can view the overall progress of the simulation project in the **Simulation Status** section.

**Figure 269: Simulation Status progress**

![Simulation Status](image)

The **Start** date and time displays the Simulation Start setting, as configured for this scenario in the **Scenario** tab of the **Properties** view (see Modifying simulation settings for a scenario on page 344). The **Current** date and time indicates the progress of the simulation in date and time values, with its final value calculated according to the Simulation Stop setting, as configured for this scenario in the **Scenario** tab of the **Properties** view.

During the simulation run, you can view the process-related information in the **Processes** view (as described in the following Viewing process information on page 365 section).

You can also monitor the usage of performers and resources, as well as review objective violations and performance bottlenecks (and recommendations) in their respective views. For details, see Viewing project information on page 367.

**Viewing process information**

The **Processes** View consists of a **Process** tab for each of the added process in the simulation project. Click each Process tab to view the simulation information for the process in the **Diagram** and **Details** views. By default, the **Details** view is hidden. To show (or hide) either of the views for each **Process** tab, click the respective button in the Processes view.

**Figure 270: Processes View in Simulation Runtime perspective**

![Processes View](image)

The **Details** view displays the progress status of the completed process instances during the simulation run. It also displays the instance count (required, completed, created, terminated), the workitem cost (total, average, minimum, maximum), and the duration (total, average, minimum, maximum) of the completed process instances. The adjoining table contains the:

- **Worksteps** tab, which lists the worksteps of the current process and the number of workstep instances, which have been created, activated, completed, terminated, and timed out, along with the total, average, maximum, and minimum duration taken to complete the instances.

- **Workitems** tab, which lists the workitem tasks for each instantiated workstep, along with number of workitems completed, the total worktime for each workitem, and the cost (total, minimum, maximum, average) for completing each workitem. Additionally, this tab displays the number
of tasks waiting in queue for the performer to become available, with the total, minimum, maximum, and average waiting time.

- **Connectors** tab, which lists the workstep connectors of the current process and the number of connector traversals, which have been created, completed, and terminated, along with the total completion and termination duration taken for each connector.

- **Consumables** tab, which indicates the consumption details of consumable resources for each assigned workstep. Resource details for each workstep include cost per unit, workitem count, consumption count, and total cost.

- **Non Consumables** tab, which indicates the consumption details of non-consumable resources for each assigned workstep. Resource details for each workstep include cost per hour, workitem count, requirement, usage time, and total cost.

- **Paths** tab, which displays information for each defined path in the process. Path information includes path name, the number of times the path was traversed, and the duration (total, minimum, maximum, average) of path traversal. Additionally, this tab displays the cost (total, minimum, maximum, average) of traversing a path. For details regarding paths, see Configuring simulation parameters for a process on page 349 > For paths on page 353.

Additionally, the **Details** view displays simulation information for the current process in the following sections:

Table 73: Details View information

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>Displays the same data as in Process Status section, but in numeric values.</td>
</tr>
<tr>
<td></td>
<td>Additionally, it displays the number of terminated process instances.</td>
</tr>
<tr>
<td>Cost</td>
<td>Displays the cost data for workitem tasks (or instantiated worksteps). Cost</td>
</tr>
</tbody>
</table>
|                       | includes the cost of workitem performers and resources (if any). The **Total**
|                       | cost indicates the total completion cost of all workitems, the **Average** cost|
|                       | indicates the average cost of workitems, the **Minimum** cost indicates the least|
|                       | expensive workitem, and the **Maximum** cost indicates the most expensive    |
|                       | workitem.                                                                   |
| Completed Instance Time| Displays the duration of completed process instances in terms of hours,     |
|                       | minutes, and seconds. The **Minimum** value indicates the shortest completion |
|                       | time, the **Maximum** box indicates the longest completion time, and the     |
|                       | **Average** box indicates the average completion time for process instances.  |
|                       | The **Total** value indicates the cumulative completion time for all instances.|

Additionally, each Process tab contains the following components:

- **Show Process** drop-down list, where you can also view the runtime data for an inline subprocess by selecting the process. For information regarding inline subprocess, see Defining an inline subprocess on page 124.

- Contains the **Status** progress bar, which indicates the total number of process instances required, along with the number of instances created and completed. The total number of process instances required are depicted in light green color, those created (and in progress) in olive green, and those completed in bluish green. You can also move your pointer over the progress bar, to view the same information in numeric terms.

In the **Diagram** view (which is displayed by default), each process workstep displays a number in parenthesis, which indicates the number of activated instances.
Each workstep (and associated path) displays a color-coded heatmap, which enables you to:

- Determine the number of activations for each workstep, relative to the rest of the worksteps in the workflow.
- Identify potential bottlenecks in the process.

Table 74 on page 367 describes the color coding used for heatmaps.

### Table 74: Heatmap color coding

<table>
<thead>
<tr>
<th>Color Gradient</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blue to White</td>
<td>For workstep activations from minimum to median value$^4$.</td>
</tr>
<tr>
<td>White to Red</td>
<td>For workstep activations from median$^a$ to maximum value.</td>
</tr>
</tbody>
</table>

A bottleneck can be identified by a red-colored workstep immediately followed by one or more white-colored worksteps.

**Tip:** When unexpected results occur during a simulation, such as the number of activations of a workstep not matching the number of instances, it may result in the presence of Inclusive Decisions. We recommend that you use Exclusive Decisions in this type of process.

## Viewing project information

The **OpenEdge BPM Run Simulation** Perspective provides additional views—**Individuals, Systems, Consumables, NonConsumables, Violations, and Recommendations**—where you view overall simulation information, independent of the processes. Click the **Individuals, Systems, Consumables, and NonConsumables** views after starting the simulation run. Click the **Violations** and **Recommendations** views after completing the simulation run (as described in **After completion of simulation** on page 370).

- **Individuals** view: indicates the utilization of individual (or human) performers in the process, and the current task (if any) for each performer.

### Figure 271: Individuals view

The **Individuals** view displays the following runtime information of individual performers.

---

$^4$ Median is calculated by dividing the sum total of all activated workstep instances by the number of worksteps
Table 75: Individual performer information

<table>
<thead>
<tr>
<th>Heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the name of the performer. A blue icon indicates that the performer is available. A grey icon indicates that the performer is unavailable.</td>
</tr>
<tr>
<td>Cost per Hour</td>
<td>Displays the cost per hour defined for each performer.</td>
</tr>
<tr>
<td>Available For</td>
<td>Displays the total duration (in terms of hours, minutes, seconds) for which this performer is available.</td>
</tr>
<tr>
<td>Total Cost</td>
<td>Displays the total cost of the performer, which is the product of the hourly cost of the performer and the total time (in hours) for which the performer is available.</td>
</tr>
<tr>
<td>Task Count</td>
<td>Displays the number of tasks currently assigned to this performer.</td>
</tr>
<tr>
<td>Total Usage Time</td>
<td>Displays the total work-time (in terms of hours, minutes, seconds) for this performer.</td>
</tr>
<tr>
<td>Utilization %</td>
<td>Displays the percentage of the total usage time of the performer, relative to the available time.</td>
</tr>
<tr>
<td>Current Tasks</td>
<td>Indicates the current task (if any) performed by this performer. The format used is: &lt;processname&gt;-&lt;taskname&gt;.</td>
</tr>
</tbody>
</table>

- **Systems** view: indicates the utilization of system (or adapter) performers in the process, and the current task (if any) for each performer.

Figure 272: Systems view

The **Systems** view displays the following runtime information of adapter performers.

Table 76: Adapter performer information

<table>
<thead>
<tr>
<th>Heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the name of the adapter performer. A blue icon indicates that the performer is available. A grey icon indicates that the performer is unavailable.</td>
</tr>
<tr>
<td>Cost per Invocation</td>
<td>Displays the cost per invocation as defined for each adapter.</td>
</tr>
<tr>
<td>Invocations</td>
<td>Displays the total number of invocations of this adapter.</td>
</tr>
</tbody>
</table>
Running a process simulation

<table>
<thead>
<tr>
<th>Heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Cost</td>
<td>Displays the total cost of the adapter, which is the product of the cost per invocation and the number of invocations.</td>
</tr>
<tr>
<td>Current Tasks</td>
<td>Indicates the current task (if any) performed by this adapter. The format used is: <code>&lt;processname&gt;-&gt;&lt;taskname&gt;</code>.</td>
</tr>
</tbody>
</table>

- **Consumables** view: displays the utilization of consumable resources in the simulation run for this scenario.

  **Figure 273: Consumables View**

  ![](image1.png)

  The **Consumables** view displays the following runtime information of consumable resources.

  **Table 77: Consumable resource information**

<table>
<thead>
<tr>
<th>Heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the name of the resource used.</td>
</tr>
<tr>
<td>Cost per Unit</td>
<td>Displays the cost of resource as defined for this scenario.</td>
</tr>
<tr>
<td>Consumer Count</td>
<td>Displays the total number of workstep instances that have used this resource.</td>
</tr>
<tr>
<td>Consumed Count</td>
<td>Displays the total number of the resource units consumed.</td>
</tr>
<tr>
<td>Total Cost</td>
<td>Displays the total cost of the resource, which is the product of cost of the resource and the consumed count.</td>
</tr>
</tbody>
</table>

- **NonConsumables** view: displays the utilization of non-consumable resources in the simulation run for this scenario.

  **Figure 274: NonConsumables View**

  ![](image2.png)

  The **NonConsumables** view displays the following runtime information of non-consumable resources.
Table 78: Non-consumable resource information

<table>
<thead>
<tr>
<th>Heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the name of the resource used.</td>
</tr>
<tr>
<td>Units</td>
<td>Indicates the number of available resource units.</td>
</tr>
<tr>
<td>Cost per Hour</td>
<td>Displays the cost of resource as defined for this scenario.</td>
</tr>
<tr>
<td>Required in Tasks</td>
<td>Displays the number of workstep (or task) instances that require this resource.</td>
</tr>
<tr>
<td>Requirement</td>
<td>Displays the total count of the resource required.</td>
</tr>
<tr>
<td>Used in Tasks</td>
<td>Displays the number of workstep (or task) instances that have used this resource.</td>
</tr>
<tr>
<td>Total Usage Time</td>
<td>Indicates the total duration (in hours, minutes, seconds) of consumption of the resource.</td>
</tr>
<tr>
<td>Total Cost</td>
<td>Displays the total cost of the resource, which is the product of cost of the resource and the total usage time.</td>
</tr>
</tbody>
</table>

After the simulation run is completed, all status indicators are green (for completed) and the Exit Simulation button is activated.

After completion of simulation

You can view the final status of the performers (individual and system) and resources (consumable and non-consumable) used during this process simulation, from the respective views. You can view any violations and recommendations by clicking the respective view, as described in this section.

- **Violations** view: lists violations (if any) in the objectives defined (see Defining objectives on page 358) in the current scenario.

**Figure 275: Violation view**

The table provides the following violation information.
Table 79: Violations

<table>
<thead>
<tr>
<th>Heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Indicates the source whose objective has been violated. Possible sources are the simulation project, process, workstep, performer, resource, and path.</td>
</tr>
<tr>
<td>Name</td>
<td>Displays the objective that was violated.</td>
</tr>
<tr>
<td>Condition</td>
<td>Displays the condition defined in the violated objective.</td>
</tr>
<tr>
<td>Path</td>
<td>Displays the path to the source whose objective has been violated. This is applicable only for defined paths and worksteps (including worksteps of an inline subprocess). The format used is: <code>&lt;processname-&gt;&lt;source&gt;</code>.</td>
</tr>
</tbody>
</table>

- **Recommendations** view: lists bottlenecks and other problems encountered during the simulation run. An example of a bottleneck can be that a particular workstep had a high waiting time, before using a particular resource or performer.

Figure 276: Recommendations view

The table provides the following information.

Table 80: Recommendations

<table>
<thead>
<tr>
<th>Heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Indicates the source where the problem has arisen.</td>
</tr>
<tr>
<td>Problem</td>
<td>Displays the description of the problem.</td>
</tr>
<tr>
<td>Recommendation</td>
<td>Displays the statement of a possible solution to the problem. For example, “Increase resources used by work-step.”</td>
</tr>
</tbody>
</table>

### Generating a simulation report

After running a simulation for a scenario, you can generate a Simulation result (*.SRF) and report (*.HTML or *.XLS) files.

- **Result**: Only one result (*.SRF) file is generated for each simulation run, which could have a single process or multiple processes. You can use the result file later to generate simulation reports. You can also generate comparison reports (see Comparing simulation results on page 377) by using multiple result files. For example, you can generate multiple result (*.SRF) files after simulating with different scenarios.

- **Report**: You can generate a simulation report in HTML or Excel format either directly after a simulation run or later from the result (*.SRF) file. You can select and create reports according to your requirement.
To generate the Simulation report:

1. From the Generate section of the Simulation Control Editor, click the Result icon to open the Simulation Report dialog box.
2. Modify (if required) the name of the default result file.
3. Click OK to save the result file.

You can view the result file in the Project Explorer view in the <project>\results folder.

4. Right-click the result file (*.SRF), then click Simulation Report > Generate Report.

Alternatively, from the Generate section of the Simulation Control Editor, click the Report icon to open the Simulation Reports dialog box. In this case, you do not need to generate the simulation result file, as discussed in Step 1 to Step 3.

5. Depending on your requirement, you can generate any of the listed report types. Click each report type to view its description in the adjoining area.

6. From the Output section, select the format (HTML or Excel) from the Format drop-down list. Modify (if required) the name of the generated file in the File box.

7. Select the report type to be generated, then click OK. Figure 277 on page 372 displays a Complete HTML report with all the generated information.

Figure 277: HTML report on simulation results

The Simulation report (in HTML format) contains the main section in links, which you can click to navigate to the respective topic. Additionally, it displays the summary charts (only in HTML format) displaying the overall cost per process and cost per resource (and performer) used in the simulation project. For more information, see the following Viewing simulation report details on page 373 section.

Note: Simulation reports in Excel format do not display charts.
Viewing simulation report details

The generated Simulation report contains the following main sections:

- **Process simulation result** on page 373
- **Resource usage results** on page 376
- **Violations and recommendations** on page 377
- **Setting calendar** on page 360

**Note:** The simulation report is saved directly under the simulation project folder
`Workspace_Home\<project_name>\reports\`.

## Process simulation result

This section provides the following sub-sections for each of the added processes and subprocesses:

- **Instance Completion Status**: (Can be viewed in Basic, Cost, and Time reports also) Displays the total number of process instances required, created, completed, and terminated.
- **Instance Execution Time**: (Can be viewed in Basic and Time reports also) Displays the duration of the completed process instances, along with the minimum, maximum, and average time.
- **Instance Completion Cost**: (Can be viewed in Cost report also) Displays the total completion cost of all workitems, along with minimum, maximum, and average cost.
- **Workstep Execution Status and Time**: (Can be viewed in Basic and Time reports also) Displays the instance count (created, activated, completed, terminated, timed out) for each of the process worksteps, along with the duration (total, minimum, maximum, average) taken to complete all the workstep instances. You can view this information in Tabular and Chart format.

**Figure 278: Worksteps**

- **Workitems Time and Cost**: (Can be viewed in Cost and Time reports also) Displays the workitem count (created, completed, terminated) for each task, along with the completion cost (total, average, minimum, maximum) of each workitem. Additionally, this section displays the number of workitem instances, which waited in queue for the performer to become available,
along with the total, minimum, maximum, and average waiting time. You can view this information in Tabular and Chart format (which displays the workitem status and total cost).

**Figure 279: Workitems**

![Workitems](image)

- **Workstep Connectors:** (Can be viewed in Basic, Cost, and Time reports also) Displays the instance count (created, completed, terminated) for each of the workstep connector, along with the aggregate delay duration taken to complete or terminate the connector traversals. You can view this information in Tabular format only.

**Figure 280: Connectors**

<table>
<thead>
<tr>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
<th>Type</th>
<th>Distribution</th>
<th>Created</th>
<th>Completed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1or2arrange</td>
<td>ORJoin1</td>
<td>Arrange Interview</td>
<td>DEFAULT</td>
<td>Constant (mean = 00:00:00)</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>2or2archive</td>
<td>ORJoin2</td>
<td>Archive</td>
<td>DEFAULT</td>
<td>Constant (mean = 00:00:00)</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>Accepted</td>
<td>Decision3</td>
<td>Check References</td>
<td>CONDITIONAL</td>
<td>Constant (mean = 00:00:00)</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Another</td>
<td>Decision3</td>
<td>ORJoin1</td>
<td>CONDITIONAL</td>
<td>Constant (mean = 00:00:00)</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Approved</td>
<td>Decision1</td>
<td>ORJoin1</td>
<td>CONDITIONAL</td>
<td>Constant (mean = 00:00:00)</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Connection 23</td>
<td>Archive</td>
<td>NotifyRejection</td>
<td>DEFAULT</td>
<td>Constant (mean = 00:00:00)</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>Proctor</td>
<td>Decison4</td>
<td>Hiring Process</td>
<td>CONDITIONAL</td>
<td>Constant (mean = 00:00:00)</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>No</td>
<td>Decision2</td>
<td>ORJoin2</td>
<td>DEFAULT</td>
<td>Constant (mean = 00:00:00)</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Rejected</td>
<td>Decision3</td>
<td>ORJoin2</td>
<td>DEFAULT</td>
<td>Constant (mean = 00:00:00)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Rejected_1</td>
<td>Decision4</td>
<td>ORJoin2</td>
<td>DEFAULT</td>
<td>Constant (mean = 00:00:00)</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>Rejected_2</td>
<td>Decision1</td>
<td>ORJoin2</td>
<td>DEFAULT</td>
<td>Constant (mean = 00:00:00)</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Yes</td>
<td>Decision2</td>
<td>Interview</td>
<td>CONDITIONAL</td>
<td>Constant (mean = 00:00:00)</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>arrange2dec2</td>
<td>Arrange Interview</td>
<td>Decision2</td>
<td>DEFAULT</td>
<td>Constant (mean = 00:00:00)</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Check2dec4</td>
<td>Check References</td>
<td>Decision4</td>
<td>DEFAULT</td>
<td>Constant (mean = 00:00:00)</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- **Path Analysis Status and Execution Time:** (Can be viewed in Basic, Cost, and Time reports also) Displays the traversal count, the traversal duration (total, minimum, maximum, average),
and traversal cost (total, minimum, maximum, average) for each defined path. You can view this information in Tabular and Chart format.

**Figure 281: Paths**
Resource usage results

This section provides a usage summary for both performers and resources in the following sub-sections. You can view this section in Basic, Cost, and Time reports also.

- **Human Performers**: Displays the cost (hourly, total), duration available for, assigned task count, total usage time, and utilization percentage for each human performer. You can view this information in Tabular and Chart format.

Figure 282: Human performers

- **System Performers**: Displays the number of invocations, cost (hourly, total), duration available for, assigned task count, total usage time, and utilization percentage for each system performer. You can view this information in Tabular and Chart format.

Figure 283: System performers

- **Consumable Resources**: Displays cost, consumed count, and consumer count for each consumable resource. You can view this information in Tabular and Chart format.

Figure 284: Consumable resources

- **Non-consumable Resources**: Displays cost, required in tasks, requirement, used in tasks, and total usage time for each consumable resource. You can view this information in Tabular format.
and Chart format (similar in format to that of Consumable resource).

Violations and recommendations

This section lists all the violations and recommendations generated for the simulation run. You can view this section in the Violations and Recommendations report also.

Calendar

You can view this section in the Time report also. This section contains the following sub-sections:

- **Process Calendar**: Displays the calendar settings for each of the added processes. The calendar information includes calendar type, workdays, and details of the work sessions.

- **Human Performer**: Displays the calendar settings for each human performer (individual and group). The calendar information includes calendar type, as well as workdays and work sessions for simulation calendar type.

Comparing simulation results

You can generate a comparison report (HTML or Excel) from multiple result (*.srf) files, each of which represent the result of a simulation run. You can also generate a real-time comparison report to compare the results of a simulation run with the real-time server data. You can select the type of the comparison report, similar to that for simulation report types (Generating a simulation report on page 371).

**Note**: You need to create at least two results files, before you can generate a comparison report. You can also compare result files generated across simulation projects.

To generate a comparison report:

1. If you are in the **OpenEdge BPM Run Simulation** perspective, then click the **Simulation Configuration** perspective to switch to the **Progress OpenEdge BPM Simulation** perspective.
2. From the **Project Explorer** view, select the result (*.SRF) files (located in <project>\results folder), which you want to compare.
3. Right-click the selected result files, then select **Simulation Report > Generate Comparison Report** to open the **Simulation Reports** dialog box (see Generating a simulation report on page 371).
4. Depending on your requirement, you can generate any of the listed report types. Click each report type to view its description in the adjoining area. Select the report type to be generated.
5. From the **Output** section, select the format (HTML or Excel) from the **Format** drop-down list. Modify (if required) the name of the generated file in the **File** box.

6. Click **OK**. **Figure 285** on page 378 displays a sample Complete Comparison report with all the generated information.

**Figure 285: Simulation Comparison Report**

The comparison report includes a generated Result ID (Result 1, Result 2, and so on, depending on the number of result files being compared). Additional information includes the scenario used, the simulation run start (and stop) date and time, the simulation duration and the total cost for each of the Result IDs.

Additionally, the comparison report includes all the main sections of the corresponding simulation report as described in **Viewing simulation report details** on page 373. Each of the main and sub-sections includes data for all the generated Result IDs in a tabular format.

**Note:** The “Violations and Recommendations” section is included only in the Violation and Recommendation comparison report.

### Generating Real-time Comparison Report

You can also generate a real-time comparison report (HTML or Excel) by comparing the result of a simulation run to the real-time data from a server.

**To generate the real-time comparison report:**

1. After performing a simulation run, click the **Realtime Report** ( ) icon from the **Generate** section of the Simulation Control Editor to open the first page of the **Realtime Comparison** wizard, which is similar to second page of the **Import Scenario Data** wizard (Figure 251 on page 346).
Alternatively, you can right-click a result file (*.SRF), then click **Simulation Report > Generate Realtime Comparison Report**.

2. To establish connection to your server:
   a) Specify the server IP address and port number in the respective boxes.
   b) Enter the user name and password required to connect to the server.
   c) Modify (if required) the start date and end date in the **Start Date** and **End Date** boxes by clicking the respective **ellipsis** button. For details, see **Specifying a date** on page 361.

   Data of all the process instances that have been completed in this date range (between start and end date) are retrieved from the server.

   d) Click **Connect** to verify the connection to the server.

3. Click **Next** to open the second page of the **Realtime Comparison** wizard, which lists the report type that you can generate.

4. Review the available report type and its description.

5. From the **Output** section, select the format (HTML or Excel) from the **Format** drop-down list. Modify (if required) the name of the generated file in the **File** box.

6. Click **Finish** to generate the realtime comparison report, which is similar to the comparison report shown in **Figure 285** on page 378.
Working with User Management tool

Business Process Modeler provides the User Management tool, which allows you to add and manage users, groups, and queues, as well as view and update the organization hierarchy.

All the members created using the User Management tool, are available to newly created process templates in Process Models and can be assigned to Activity worksteps.

The User Management tool includes a Organization Chart Viewer, that enables you to view the specified organization as a hierarchy of users and groups, similar to as shown in Figure 286 on page 381.

Figure 286: Sample organization chart
Types of members

The User Management tool supports three types of members, as described in Table 81 on page 382.

Table 81: Member types

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Users</td>
<td>Represents the basic entity of the user management tool. Typical attributes of a user include user name, real name, and title.</td>
</tr>
<tr>
<td>Groups</td>
<td>Represents a set of users performing related tasks, for instance, the HR group. Attributes include leader name and group members. A group can include another group.</td>
</tr>
<tr>
<td>Queues</td>
<td>Represents a set of users and groups.</td>
</tr>
</tbody>
</table>
Starting User Management tool

You can start the User Management tool by clicking **Tools > Users**.

**Figure 287: User Management tool**

The User Management tool consists of two panes. The left pane displays a tree consisting of the member node types namely **Users**, **Groups**, and **Queues**. You can define members, which then appear under the respective node type (see *Managing nodes* on page 384).

You can also search for members using the Search box in the left pane. Click the down-arrow provided and select either of the following options:

- **Flat search**: This search provides a flat list of all matching members, containing the search text.
- **Tree search**: This search displays the result in a tree structure, indicating the node type and if present, its position in the organization hierarchy.

The right pane (as seen in **Figure 287** on page 383) displays by default, the Organization Chart Viewer initially with blank tree node. After adding users and groups, the viewer is automatically updated to display the organization hierarchy. Additionally, you can expand (and collapse) parts of the organization hierarchy, thus providing views at different detail levels.

The Organization Chart Viewer (as shown in **Figure 286** on page 381) has the following features:

- Each expandable (or collapsible) node displays the group and the group leader (optional). The topmost node displays the organization name and the leader (optional). The title of the added user, if defined, is also displayed.
- You can double-click any node to view the user details.
- You cannot add (or remove) nodes directly from the Chart Viewer.
- Queues are not displayed in the Chart Viewer. The Chart Viewer only displays users and groups (including groups within a group).
To customize the Organization Chart Viewer:

1. In the **Organization Name** box (in the right pane), type the name of your organization.
2. To specify the organization leader, click the **ellipsis** button beside the **Leader** box to open the **Select User** dialog box, which displays a list of defined users (as described in Managing users on page 385).
3. Click the user from the list, then click **OK**. You can also use the **Search** box to search for a user.
4. Click **Apply** to add the organization name and the leader name to the blank topmost node in the Organization Chart Viewer. The organization name is also displayed and represents the root node in the left pane of the User Management tool.

**Note:** Henceforth, you can click the root node at any point of time, to view the updated Organization Chart Viewer.

---

**Printing organization chart**

You can also print the organization chart displayed in the Organization Chart Viewer.

1. Click **File > Print chart** to open the **Print** dialog box, which displays a print preview of the organization chart.

**Figure 288: Printing Chart**

![Print dialog box](image)

2. Modify the print settings, if required, then click **Print**.

---

**Managing nodes**

You can use the User Management tool to add users, groups, and queues as members in the respective member nodes. You can then use the defined members as performers in newly created process templates.
Managing nodes

You can click any member node or member in the left pane to view the details screen in the right pane. Alternatively, you can double-click any node in the Organization Chart Viewer to display the member details.

Managing users

You can add (and manage) individual users from the User list.

1. Click **Users** node in the left pane, displaying the user list in the right pane.

   The user name and the real name is displayed by default. To hide (or show) either of the columns, right-click any column header and mark (or unmark) the appropriate options.

   **Figure 289: User List screen**

   ![User List screen](image)

2. To create a user, click **New** to open the user details screen, where you can enter the following user details.

   **Table 82** on page 385 describes each user information field.

   **Table 82: User information**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>(Required) Enter the system name of the user. The username appears as the performer in your process template. You can also search for a user by entering the username in <strong>Search</strong> box in the User List screen.</td>
</tr>
<tr>
<td>Real Name</td>
<td>Enter the full name of the user (example, David Brown). This name appears in the Organization Chart Viewer for this user. To display the user in the Chart Viewer, you need to add the user to a group.</td>
</tr>
<tr>
<td>Title</td>
<td>Enter the designation (or title) of the user in the organization. If specified, then the title also appears along with the real name in the Organization Chart Viewer.</td>
</tr>
<tr>
<td>E-mail</td>
<td>Enter a valid email address of this user.</td>
</tr>
</tbody>
</table>
3. After entering the user information, click **OK** to add the user to the user list. The user is also added as a member to the **Users** node in the left pane.

To remove a user (or multiple users) from the user list, select the users and click **Remove**. The User Management tool prompts you for confirmation. Click **Yes** to remove the selected users. To modify user details, select the user and click **Modify** to open the user details screen, where you can modify the user information.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phone</td>
<td>Enter the telephone number of this user.</td>
</tr>
<tr>
<td>Calendar</td>
<td>Click the <strong>ellipsis</strong> button beside the <strong>Calendar</strong> box to specify a calendar (if defined). This action includes the business calendar restrictions for this user.</td>
</tr>
</tbody>
</table>
Managing groups

You can add (and manage) groups from the Groups list.

1. Click Groups node in the left pane, displaying the group list in the right pane.

   The group name and label is displayed by default. To hide (or show) either of the columns, right-click any column header and mark (or unmark) the appropriate options.

   Figure 290: Group List screen

   ![Group List screen](image)

2. To create a group, click **New** to open the group details screen, where you can enter the following group details.

   Figure 291: Creating group

   ![Creating group](image)
Table 83 on page 388 describes the group detail fields.

### Table 83: Group information

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Name</td>
<td>(Required) Enter the name of the group. The group name appears as the group performer in your process template. You can also search for a group by entering the group name in <strong>Search</strong> box in the Group List screen (see Figure 290 on page 387).</td>
</tr>
<tr>
<td>Group Label</td>
<td>Enter the group label (example, Human Resources). This label appears as the group label (see Figure 286 on page 381) in the Organization Chart Viewer for all added users.</td>
</tr>
<tr>
<td>Leader</td>
<td>Click the <strong>ellipsis</strong> button beside the <strong>Leader</strong> box to select a group leader (see Figure 291 on page 387). From the <strong>Select User</strong> dialog box, click the user to be selected as the group leader, then click <strong>OK</strong>.</td>
</tr>
<tr>
<td>Calendar</td>
<td>Click the <strong>ellipsis</strong> button beside the <strong>Calendar</strong> box to specify a calendar (if defined). This action includes the business calendar restrictions for the group members.</td>
</tr>
<tr>
<td>Group Members</td>
<td>Use this section to add the group members, as discussed in Adding group members on page 388.</td>
</tr>
</tbody>
</table>

3. Click OK to add the group to the groups list. The group is also added as a member to the Groups node in the left pane.

To remove a group (or multiple groups) from the groups list, select the groups and click **Remove**. The User Management tool prompts you for confirmation. Click **Yes** to remove the selected groups. To modify group details, select the group and click **Modify** to open the group details screen (Figure 291 on page 387), where you can modify the group information.

### Adding group members

You can use the **Group Members** section (see Figure 290 on page 387) to add group members that can include individual users and other groups.

**To add and configure a group member:**

1. Click **Add** to open the **Select Members** dialog box, which lists all the defined users and groups.
2. Click the member to be added, then click **OK** to add the member to the list of members.

   You can also search for a member using the **Search** box. To remove a member from the member list, select the user and click **Remove**.

3. You can also specify a role (optional) for the group member.
   a) Select the added member, then click **Role** to open the **Select Role** dialog box.
   b) Select an existing role from the drop-down list provided or enter a new role.
   c) Click **OK** to define the role for the group member.
Managing queues

You can add (and manage) queues from the queue list.

1. Click **Queues** node in the left pane, displaying the queue list in the right pane.

   The queue name and label is displayed by default. To hide (or show) either of the columns, right-click any column header and mark (or unmark) the appropriate optionns.

   **Figure 292: Queue List screen**

2. To create a queue, click **New** to open the queue details screen, where you can enter the queue details.

   **Table 84** on page 390 describes the queue information fields.
Table 84: Queue information

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Queue Name</td>
<td>(Required) Enter the name of the queue. The queue name appears as the queue performer in your process template. You can also search for a queue by entering the queue name in Search box in the Queue List screen (see Figure 292 on page 389).</td>
</tr>
<tr>
<td>Queue Label</td>
<td>Enter the queue label. Queue labels do not appear in the Organization Chart Viewer.</td>
</tr>
<tr>
<td>Queue Members</td>
<td>You can use this section to add queue members that can include individual users and other groups. Click Add to open the Select Members dialog box, which lists all the defined users and groups. Click the member to be added, then click OK to add the member to the list of members. You can also search for a member using the Search box. Note: You cannot add another queue as the queue member. To remove a member from the member list, select the member and click Remove.</td>
</tr>
</tbody>
</table>

3. Click OK to add the queue to the queue list. The queue is also added as a member to the Queues node in the left pane.

To remove a queue (or multiple queues) from the queue list, select the queues and click Remove. The User Management tool prompts you for confirmation. Click Yes to remove the selected queues. To modify queue details, select the queue and click Modify to open the queue details screens, where you can modify the queue information.

Importing users

You can also import user information from a selected server.

1. Click File > Import to open the Import dialog box.

   ![Figure 293: Importing user information](image)

2. In the Server URL box, modify the URL (if required) of the server from where you want to retrieve user information. You cannot import user information from an LDAP Realm.

3. Click OK.
Working with Business Calendar

Business Process Modeler provides the Business Calendar tool, using which you can define an organization-level calendar and assign them to workstep performers for simulation projects. For details regarding usage of business calendar in the Simulation perspective, see Configuring and running simulation on page 339.
You can start the Business Calendar by clicking **OpenEdge > Tools > Calendars**.

**Figure 294: Calendar**

You can use the Business Calendar tool to add (and manage) calendars and schedules.

### Managing calendars

You can perform the following operations to manage calendars.

- To add a calendar, click the **Add** icon from the **Calendar** tool, to add a calendar (example, “Calendar1”) to the **Calendar** drop-down list.
Note: Each added calendar (Calendar1, Calendar2, and so on) is a copy of the configuration set in the working schedule of the default calendar.

- To rename the added calendar, click the Rename icon to open the Calendar name dialog box.

Figure 295: Renaming a calendar

1. Type the modified name of the calendar.
2. Click OK to rename the calendar.

- To delete a calendar, select the calendar in the Calendar drop-down list, then click the Remove icon. The Business Calendar tool prompts you for confirmation. Click OK to remove the selected calendar.

Note: You cannot rename nor delete the default calendar.

Managing schedules

You can create customized working schedules, which are independent of calendars. You can then apply a schedule to a single day, or a sequence of days, or even to all the days of the year. You can also apply a schedule to multiple calendars.
To define a schedule:

1. Click a single day of your calendar, or drag across consecutive days to open the **Select Schedule** dialog box, using which you can define (or select) a schedule.

   ![Select Schedule dialog box](image)

2. Click the **Add** icon to add a schedule (example, “Schedule1”) to the **Schedule** drop-down list.

   **Note:** Each added schedule (Schedule1, Schedule2, and so on) contains the original settings of the default schedule. Any changes made to the default schedule are not reflected when you add a schedule.

3. Enter a description for the schedule (example, “Recruitment Schedule”) in the **Description** box.

   **Note:** You can rename the schedule by clicking the **Rename** icon, to open the **Schedule name** dialog box, where you can enter the modified name. To delete a schedule, click the **Remove** icon. You cannot rename nor delete the default schedule.

4. From the **Workdays** section, select (or clear) the appropriate checkboxes for the work days in this schedule. For the default schedule, Monday to Friday are marked as working days, with Saturday and Sunday designated as non-working days.

   **Note:** To create a schedule for company holidays, clear all the checkboxes in the Workdays section.

5. The **Workhours** section displays two time segments namely, **AM** and **PM**. For the default schedule, the work-hours are 8-12 (in the AM segment) and 1-5 (in the PM segment). To manage work-hours:
   - Use the slider along the starting and ending time of each segment to change the default work-hours.
   - You can remove a work-hour segment by right-clicking on it, then clicking **Remove**.
• You can also insert additional work-hour segments, by right-clicking a blank portion of the **Workhours** section, then clicking **Insert**.

The work-hours that you set is applied to all the working days selected in the **Workdays** section.

6. You can set different work-hours for single (or multiple) working days in the **Exceptions** tab.

**Figure 297: Select Schedule dialog box – Exceptions tab**

![Select Schedule dialog box – Exceptions tab](image)

a) Select the checkbox for the working day whose work-hours you want to modify.

b) Modify the work-hours, as described in Step 5.

7. Click **OK** to create and apply the schedule to the selected days of your calendar.

The selected days are highlighted by a color, which is unique and randomly selected for each customized schedule. Move your pointer over any highlighted day to view the name of the customized schedule.

**Overwriting a schedule**

You can overwrite a schedule for selected day (or days) with another schedule.

1. Drag across the calendar days, whose schedule is to be overwritten.

2. From the **Select Schedule** dialog box (Figure 296 on page 394), select any schedule from the **Schedule** drop-down list.

3. Click **OK** to apply the schedule to the selected days.
Managing business objects

Business Process Modeler provides the Business Object Manager tool, which enables you to define and manage business objects. For each business object, you can add and manage attributes, where data is stored and accessed.

Business objects are fully interoperable, plug-and-play, distributed components that encapsulate traditional lower-level objects that implement a business process. In Business Process Modeler, business objects act as a collection of dataslots that can behave as a single, reusable unit. Business Objects also supports complex nested structures, including collection of other business objects. Using business objects, you can incrementally adjust business applications to fit your changing organizational needs, use them to rapidly build new applications entirely from reusable components, and dramatically reduce your development and maintenance costs.

For details, see the following topics:

- Using the Business Object Manager
- Using business objects in a link
- Using business objects in applications

Using the Business Object Manager

You can use the Business Object Manager tool to create and maintain business objects in the Process Repository. The stored objects can be persisted internally and externally.
To create and manage a business object:

1. Open a Process Model or Web application in Process Modeler.
2. From the **Tools** menu, click **Business Objects** to open the **Business Object Manager** dialog box, which displays the list of default business objects with category, "demo."

**Figure 298: Business Object Manager dialog box**

![Business Object Manager dialog box](image)

**Note:** To modify an existing business object, select it from the **Business Object Manager** dialog box, then click **Edit** to make the changes in the **Edit Business Object** dialog box (similar to that shown in **Figure 298** on page 398).

3. To create a new business object, click **New** to open the **Create New Business Object** dialog box.

**Figure 299: Create New Business Object dialog box**

![Create New Business Object dialog box](image)

4. Type the name of the business object in the **Name** box and select a category from the **Category** drop-down list. To define a category, click the **ellipsis** button beside the **Category** list, to open the **New Category** dialog box, where you can define a new category.

5. In the **Data Source Reference** box, enter the description of the datasource (if any) being referenced. In the **Description** box, enter a description of the business object. This information is optional and help users identify business objects.
6. To define attributes for the business object, click **Add** to open the **New Attribute** dialog box.

**Figure 300: New Attribute**

![New Attribute dialog box](image)

a) Define the attribute properties by entering appropriate information in the available field components.

Table 85 on page 399 describes each of the business object attributes.

**Table 85: Defining business object attributes**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the attribute. Attribute name must not contain any special characters, and can have up to 64 characters. You need to precede any upper-case character with at least two lower-case characters.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of attribute. Options include CHARACTER, INT64, INTEGER, DECIMAL, LOGICAL, DATETIME-TZ, Java datatypes, and any business object that you have previously defined. <strong>Note:</strong> Business Process Modeler does not support business objects with circular reference. For example, if a business object, &quot;bo1&quot; contains the business object, &quot;bo2&quot; as an attribute, then &quot;bo2&quot; cannot contain &quot;bo1&quot; as an attribute.</td>
</tr>
<tr>
<td>Size</td>
<td>The size of the attribute. You can modify the size only for String type.</td>
</tr>
<tr>
<td>Default value</td>
<td>Enter (or modify) the default value for the attribute, if required. The default values are 0 (for Long), 0.0 (for Double and BigDecimal), false (for Boolean), and current date (for TimeStamp). To change the current date for Timestamp, click the <strong>Open Time Chooser</strong> icon.</td>
</tr>
<tr>
<td>Primary Key</td>
<td>Select this checkbox to make this attribute the primary key. Only one attribute can be assigned as the primary key.</td>
</tr>
<tr>
<td>Read Only</td>
<td>Select this checkbox to make this attribute read-only.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Select this checkbox to make this attribute mandatory (or required).</td>
</tr>
</tbody>
</table>
If you select an existing business object as the attribute type and select the **Collection** check box, then you can retrieve multiple records from the selected business object, which is displayed as a line in a multi-line table (see Figure 300 on page 399). For example, if you create a attribute called Addresses and select a business object for “Employee Records,” you can display the “Employee Records” attribute data. This data is then displayed in Business Process Portal as a table.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collection</td>
<td>If you select an existing business object as the attribute type and select the <strong>Collection</strong> check box, then you can retrieve multiple records from the selected business object, which is displayed as a line in a multi-line table (see Figure 300 on page 399). For example, if you create a attribute called Addresses and select a business object for “Employee Records,” you can display the “Employee Records” attribute data. This data is then displayed in Business Process Portal as a table.</td>
</tr>
</tbody>
</table>

**Note:** You can add a business object with “Collection” property as a Business Object Collection (BO Collection) widget to your form. For details, see [Using business objects in a form](#) on page 405.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Description of the attribute. This is optional.</td>
</tr>
</tbody>
</table>

b) Click **OK** to add the attribute to the **Attributes** list in the **Create New Business Object** dialog box. Figure 301 on page 400 displays a sample business object for storing employee information.

Figure 301: Create New Business Object dialog box with Added Attributes

---

7. After you have added all the required attributes, click **OK**.

The new business object is added to the list in the **Business Object Manager** dialog box.
Importing business objects

You can define business objects by importing files of the following types:

- Business Object XML files, as discussed in Importing a business object from Business Object XML file on page 401.
- XSD files, as discussed in Importing a business object from an XSD file on page 401.
- Java Bean files, as discussed in Importing a business object from a Java Bean file on page 403.

Importing a business object from Business Object XML file

You can import one or more business object from an XML file, which is only available to you (or the ones that others have shared with you).

To import a business object from BO XML:

1. From the Local section of the Business Object Manager dialog box (Figure 298 on page 398), click Import. The Import dialog box appears.

2. Select BusinessObject XML and then click OK. The Open dialog box appears.
3. Select single (or multiple) XML files and click Open to display the list of business objects. By default, all business objects are selected.
4. Click Open. The details of the business object file appears in business object details dialog box.
5. Click OK. The selected business objects are added to the list in the Business Object Manager.

Importing a business object from an XSD file

An XSD file can contain multiple tags each of which can be converted to a business object. You can select the tags to be imported as business objects.

Prerequisite:

1. You must have a JDK version 5.0 or higher installed on your machine for importing business object from an XSD.
2. In the System Environment Variables for PATH variable add the JDK bin directory. For example C:\Program Files\Java\jdk1.6.0_26\bin.

This must be updated because, the XSD functionality uses the xjc utility bundled with a JDK version 5.0 or higher found in the bin directory of the JDK installation.
To import a business object from an XSD file:

1. From the Local section of the Business Object Manager dialog box (Figure 298 on page 398), click Import. The Import dialog box appears.
2. Select XSD and then click OK. The XSD Import dialog box appears.
3. To add an XSD file as the source, click Add. The Add Source dialog box appears.
   a) You can enter the absolute path of the XSD file in the Location box. Alternatively, click the ellipsis button beside the Location box to browse and select the XSD file.
   b) Select the existing Category or create a new category.
   c) Click OK.

Figure 303: XSD Import dialog box

In the XSD Import dialog box, the XSD details appear in the Sources section and the business objects available in the XSD are displayed in the Types section. Click Modify in the Sources section to select another XSD file or category. Click Delete in the Sources section to delete the selected XSD file. To modify any of the listed business objects in the Types section, select the business object and click Modify. You can modify the details including name and category.

4. Click Generate.

The list of business object that you can import appears in Business Objects To Import dialog box.

5. Select one (or more) business object and click OK. The selected business objects are added to the list in the Business Object Manager.
Importing a business object from a Java Bean file

Java Beans encapsulates multiple objects into a single object (or bean).

To import a business object from a Java Bean file:

1. From the Local section of the Business Object Manager dialog box (Figure 298 on page 398), click Import. The Import dialog box appears.
2. Select JavaBean and then click OK. The Java Bean Import dialog box appears.
3. To add a Java Bean file as the source, click Add. The Add Source dialog box appears.

   ![Add Source dialog box](image)

   a) Click the ellipsis button beside the Location box to browse and select the Java file or a folder containing multiple Java files.
   b) Select the existing Category or create a new category based on the requirement.
   c) Select Linked Source checkbox. If you select the Linked Source checkbox, then the information of linking the source is passed on to the BO type as well.

**Note:** The Linked Source feature is available only for importing a Java file and it is unavailable if you have selected a folder which contains multiple Java files. If you generate the Java file in a common resource project, the original Java file is referenced if you have selected the Linked Source checkbox.
Note: The Linked Source, whose value is initially set to true, loses its value if you modify the imported business object.

d) Click **OK**.

**Figure 304: JavaBean Import dialog box**

![JavaBean Import dialog box]

Note: The default (or initial) value of Java files with initialized variables is not imported.

In the **JavaBean Import** dialog box, the JavaBean details appear in the **Sources** section and the business objects available in the Java bean are displayed in the **Types** section. Click **Modify** in the **Sources** section to select another JavaBean file or category. Click **Delete** in the **Sources** section to delete the selected JavaBean file. To modify any of the listed business objects in the **Types** section, select the business object and click **Modify**. You can modify the details including name, category, and linked source attribute.

4. Click **Generate**.

The list of business object that you can import appears in **Business Objects To Import** dialog box.

5. Select one (or more) business object and click **OK**. The selected business objects are added to the list in the Business Object Manager.

**Exporting a business object**

**To export a business object:**

1. From the **Business Object Manager** dialog box (Figure 298 on page 398), select one or more listed business objects, then click **Export**.
2. The **Save** dialog box appears, in which you can export the business objects in XML format.
3. Select the folder where you want to export the selected business objects.
4. Click **Save**.

The selected business objects are saved to the selected folder. For each selected business object, a separate XML file is created, which contains the business object definitions encoded in it.
Using business objects in a link

You can use business objects in links to filter for specific conditions. For information regarding links, see Defining link properties on page 240.

To use a business object in a link:

1. Open the Properties view for a conditional link, then click Advanced option to open the Advanced setting for that link.
2. Drag the business object dataslot (in this example “BOds”) from the Dataslots section to add it to the Condition section.
3. Build the condition for the business object. For business objects, you can use the AutoComplete feature, which lists all the attributes of the business object.
4. To see if the condition is valid, click Validate. A message stating “Validation successful” must appear.

Using business objects in applications

You can use Business Objects in your process templates, as described in the following sections.

Note: Business objects are not supported on worksteps with default HTML presentation format.

Using business objects in a form

You can add business objects with “Collection” property as a Business Object Collection (BO Collection) widget to your form using Form Editor. The BO Collection widget supports:

• Complex rendition capabilities of the added widget in forms.
• Form Preview capability, which allows you to test the runtime rendition of the added widget.
• Configuring the individual attributes of the business object using Properties view.

Note: For information on Form Editor, see Using Form Editor on page 275.
The following figure illustrates the structure of sample business object dataslot, "Invention" containing business objects with Collection attribute, namely "inventors," "patents," and "submitters." Business objects with Collection attribute are depicted with the 🐠 icon.

**Figure 305: Sample business object dataslot structure**

To add a business object to your form:

1. Click the **Data Sources** link in the Form Editor Tasks pane (see Defining the form’s data sources on page 305).
2. Expand the **User Dataslots** folder under the **Dataslots** folder to display the available dataslots.
3. Expand the Business Object dataslot (for example, “Invention1”) and drag the business object with "Collection" attribute (for example, "patents") to your form.

The BO Collection widget is added to your form.
**Note:** You cannot drag individual attributes of a business object with "Collection" attribute to your form.

**Figure 306: BO Collection widget in the Form Editor**

![Diagram of BO Collection widget in the Form Editor]

**Note:** You can also add other attributes of a business object by individually dragging them to your form.

The figure above displays the BO collection widget (in this case, "patents") as a grey box. You can define (or modify) the properties of the BO collection widget using the **Business Object Collection Properties** view.

**Note:** Similar to the other form controls, you can add the BO Collection widget to any form editor layout control including tabbed pane, table, fieldset, and panel.

4. Click **File > Save** to save the changes made to your form.

**Configuring a BO Collection widget**

You can use the **Business Object Collection Properties** view to configure a BO Collection widget in your form.

The left pane of the **Business Object Collection Properties** view displays the tree structure of the top-level collection attribute (in this case, "patents") and its individual attributes. You can perform the following operations in the **Business Object Collection Properties** view:
Table 86: Operations in Business Object Collection Properties

<table>
<thead>
<tr>
<th>To...</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remove an attribute</td>
<td>Clear the checkbox associated with the attribute. The selected attributes are rendered in the same order as controls in your form. The attributes, which are not selected, are not rendered in your form at runtime.</td>
</tr>
<tr>
<td>Move an attribute</td>
<td>You can change the position of an attribute in the tree. Drag the attribute to another location within the tree.</td>
</tr>
<tr>
<td>Define the attribute properties.</td>
<td>You can set the presentation format and other properties of each attribute (including the top-level collection attribute). Select the attribute to view its properties in the right pane of the Business Object Collection Properties view. You can define (or modify) the properties, as described in the following steps.</td>
</tr>
</tbody>
</table>

To define (or modify) the attribute properties:

1. Select the top-level collection attribute (in this case, "patents"). From the Presentation tab, you can set the presentation type to Grid (default) or Combobox.
   - For Grid presentation, all selected attributes of this collection attribute are rendered in a grid format.
   - For Combobox presentation, only one of the primitive attributes of this collection attribute can be selected and rendered in a combo box. For this example, if "patentID" is the selected attribute for "patents" widget, the IDs of all patents are rendered in the combo box. Use the Combobox presentation in your form only when the business object collection is already populated with data.

For the individual attributes, the presentation type options vary according to the attribute type. The table below list the presentation types for each attribute type.

Table 87: Supported Presentation types for each attribute

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Supported presentation types</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHARACTER</td>
<td>• Text field</td>
</tr>
<tr>
<td></td>
<td>• Combo box</td>
</tr>
<tr>
<td>INTEGER, INT64, DECIMAL</td>
<td>• Text field</td>
</tr>
<tr>
<td></td>
<td>• Combo box</td>
</tr>
<tr>
<td>Business object with Collection attribute</td>
<td>For collection business objects nested in the &quot;parent&quot; business object.</td>
</tr>
<tr>
<td></td>
<td>• Grid</td>
</tr>
<tr>
<td></td>
<td>• Combo box</td>
</tr>
<tr>
<td>Attribute Type</td>
<td>Supported presentation types</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>LOGICAL</td>
<td>• Check box</td>
</tr>
<tr>
<td></td>
<td>• Radio button</td>
</tr>
<tr>
<td>DATETIME-TZ</td>
<td>Date Time</td>
</tr>
</tbody>
</table>

2. Depending on the presentation type you select in the **Type** drop-down list, the list of properties in the right pane of the **Business Object Collection Properties** view vary. If required, modify the value of the **Title** property (available for all presentation types), which is the title of the attribute on your form. The properties specific to the presentation types are described below:

- For collection attributes with "Grid" presentation type, you can configure the following properties:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| Create Mode  | The default mode is Allow Create, which allows you to add a row to the grid. To change the mode to any of the following options, click the **ellipsis** button in the **Value** column box:  
  • Custom Create: Adds a **Create** button at runtime. Clicking this button displays a customized dialog box where the user can enter the data for the new record in the grid. Repeat clicking the **Create** button to enter additional records to the grid.  
  • Disable Create: to disable the addition of rows in the grid. |
| Edit Mode    | The default mode is Allow Edit, which allows you to edit the contents of the grid. To make the grid read-only, click the **ellipsis** button in the **Value** column box and select the **Disable Edit** option. |
| Allow Delete | The default value is "False" which means you cannot delete any row in the grid. You can change the value to "True" by clicking the **Value** column box. |
| Number Of Rows | The default value is 5, which is the number of grid rows displayed in a single page. You can edit the value by clicking the **Value** column box. |
For collection attributes with "Combobox" presentation type, you can configure the properties as follows.

1. For collection attribute with "Combobox" presentation type, you can select only one attribute of the collection. Use the **Choices** property to select this attribute. Click the **ellipsis** button in the **Value** column box to open the **Collection Properties** dialog box, which lists the attributes that you can select. Select a attribute and click **OK**.

   ![Collection Properties dialog box](image)

2. Use the **Sub Collection** property to select the collection business object nested in the parent business object (for example, "submitters" nested in "patents"). Depending on the option you select in the **Choices** property field at runtime, the **Sub Collection** property field (in this case, "submitters") is automatically updated.

   **Note:** To create a cascaded set of two combo boxes, set the presentation type of the nested collection business object to "Combobox."

   Click the **ellipsis** button in the **Value** column box to open the **Collection Properties** dialog box, which lists the nested collection business objects that you can select.

   - For Timestamp attribute, you can use the **Date Only** property to display only date (in this case, the value is "True") or date and time (in this case, the value is "False").

3. From the **Validation** tab (not available for business objects with collection attribute), you can use the **Required** property (default value, "False") to specify if the corresponding form control is mandatory or not.

### Using business objects in scripts

You can use scripts to set the value of business object attributes. You can add the required script in the **Before Activation** and **When Completed** tabs of the workstep Properties, as well as an overdue action.

Business Process Modeler provides the `getExprEval()` method, which returns a `ObjectGraphService` object. You can use this object to set the attribute values of the specified business object.
The following code provides an example of this functionality:

```javascript
var empObj_DS = jst.getDataSlotValue("Employee_DS");
var exprEval = jst.getExprEval();
exprEval.setValue(empObj_DS, "firstName", "Sam");
jst.putDataSlot("Employee_DS", empObj_DS);
```

where Employee_DS is the datslot of type, "Business Object" mapped to the "empObj_DS" business object with "firstName" attribute.
Product support contact information

If the product documentation does not provide a solution to your specific issue, or if you need clarification on the issue, then contact our Product Support team. You can contact the team through the Internet, telephone, or postal mail, as per the details provided in Table 88 on page 413.

Table 88: Product Support Contact Information

<table>
<thead>
<tr>
<th>To contact by</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web site</td>
<td><a href="http://progresslink.progress.com/supportlink">http://progresslink.progress.com/supportlink</a></td>
</tr>
<tr>
<td></td>
<td>If you are an existing customer, then you can log in to the above site for product support. If you are a first time user, then you need to create an account first.</td>
</tr>
<tr>
<td>Telephone⁵</td>
<td>1-781-280-4999 for US, Latin America and Canada</td>
</tr>
<tr>
<td></td>
<td>1-781-280-4543 for the Product Support Fax Line</td>
</tr>
<tr>
<td>Postal Address⁵</td>
<td>Progress Software Corporation</td>
</tr>
<tr>
<td></td>
<td>14 Oak Park Drive</td>
</tr>
<tr>
<td></td>
<td>Bedford, MA 01730, USA.</td>
</tr>
</tbody>
</table>

To enable us to quickly answer your questions, please provide the following information:

- Your name, installation site address and the license key for Business Process Server software.
- Your Business Process Server version and build number.

⁵ For support telephone numbers and offices in your region, visit the support web site above. This contact information is for customer support only.
Chapter 22: Product support contact information

- Your operating system, application server and browser, with version and service pack details, if any.
- Your database management system and version, and information on JVM and JDBC used.
See Table of Contents for location of Third party acknowledgements within this documentation.

For details, see the following topics:

- Third-party acknowledgments

Third-party acknowledgments

One or more products in the Progress OpenEdge v11.3 release includes third party components covered by licenses that require that the following documentation notices be provided:

Progress OpenEdge v11.3 may incorporate ANT v1.5.4. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1, applies to all versions of up to ant 1.6.0 included. The Apache Software License, Version 1.1 - Copyright (C) 2000-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: “This product includes software developed by the Apache Software Foundation (http://www.apache.org/).” Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
4. The names “Ant” and “Apache Software Foundation” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
5. Products derived from this software may not be called “Apache”, nor may “Apache” appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.
Progress OpenEdge v11.3 may incorporate Apache Avalon 2002/08/06, Apache Batik v1.1, Apache Jakarta POI v1.1, Apache POI 2003/05/14, Regexp (Pure Java Regular Expression) v1.2, and XMLTask v1.15.1 from The Apache Software Foundation. Such technologies are subject to the following terms and conditions: The Apache Software License, Version 1.1 Copyright (c) 2000 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: “This product includes software developed by the Apache Software Foundation (http://www.apache.org/).” Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
4. The names “Apache” and “Apache Software Foundation” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
5. Products derived from this software may not be called “Apache”, nor may “Apache” appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>. Portions of this software are based upon public domain software originally written at the National Center for Supercomputing Applications, University of Illinois, Urbana-Champaign.
Progress OpenEdge v11.3 may incorporate Apache JAXP v1.3.04 from The Apache Software Foundation. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1 Copyright (c) 1999-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Xalan" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED `AS IS' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999, Lotus Development Corporation., http://www.lotus.com. For more information on the Apache Software Foundation, please see http://www.apache.org/.
Progress OpenEdge v11.3 may incorporate Crimson v1.1.3 from Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1 Copyright (c) 1999-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Xerces" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999, International Business Machines, Inc., http://www.ibm.com. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Progress OpenEdge v11.3 may incorporate jfor v0.7.2 from jfor. Such technology is subject to the following terms and conditions: jfor Apache-Style Software License. Copyright (c) 2002 by the jfor project. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the jfor project (http://www.jfor.org)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The name "jfor" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact info@jfor.org. 5. Products derived from this software may not be called "jfor", nor may "jfor" appear in their name, without prior written permission of info@jfor.org. THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JFOR PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate Apache LogKit v1.2 from The Apache Software Foundation. Such technologies are subject to the following terms and conditions: The Apache Software License, Version 1.1 -- Copyright (C) 1997-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "LogKit", "Jakarta" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.
Progress OpenEdge v11.3 may incorporate Xerces for Java XML Parser v2.6.2. Such technology is subject to the following terms and conditions:
The Apache Software License, Version 1.1
Copyright (c) 1999 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
4. The names "Xerces" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999, International Business Machines, Inc., http://www.ibm.com. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Progress OpenEdge v11.3 may incorporate SOAP v2.3.1 from Apache Foundation. Such technology is subject to the following terms and conditions:
The Apache Software License, Version 1.1
Copyright (c) 1999 The Apache Software Foundation. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
4. The names "SOAP" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.
THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Progress OpenEdge v11.3 may incorporate ANTLR (Another Tool for Language Recognition) v2.7.6. Such technology is subject to the following terms and conditions: ANTLR 3 License [The BSD License] Copyright (c) 2003-2006, Terence Parr All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the author nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate Castor v0.9.3 from castor.org. Such technology is subject to the following terms and conditions: Original Intalio license - Copyright 1999-2004 (C) Intalio Inc., and others. All Rights Reserved. Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The name "Exolab" must not be used to endorse or promote products derived from this Software without prior written permission of Intalio Inc. For written permission, please contact info@exolab.org. 4. Products derived from this Software may not be called "Castor" nor may "Castor" appear in their names without prior written permission of Intalio Inc. Exolab, Castor and Intalio are trademarks of Intalio Inc. 5. Due credit should be given to the ExoLab Project (http://www.exolab.org/). THIS SOFTWARE IS PROVIDED BY INTALIO AND CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL INTALIO OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate Decimal Conversion Code (dtoa.c; g_fmt.c; rnd_prod.s; decstrod.c; decstrtof.c; dmisc.c; gdtoa.c; gdtoa.h; gdtoaimp.h; gethex.c; gmisc.c; hd_init.c; misc.c; smisc.c; strtodg.c; strtord.c; sum.c; ulp.c). Such technologies are subject to the following terms and conditions: dtoa.c License: The author of this software is David M. Gay. Copyright (c) 1991, 2000, 2001 by Lucent Technologies. Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software. THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE. g_fmt.c License: The author of this software is David M. Gay. Copyright (c) 1991, 1996 by Lucent Technologies. Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software. THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.
OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. gethex.c License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. gmsic.c License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. hd_init.c License: The author of this software is David M. Gay. Copyright (C) 2000 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. strtord.c License: The author of this software is David M. Gay. Copyright (C) 1998, 2000 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. sum.c License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. ulp.c License: The author of this software is David M. Gay. Copyright (C) 1998, 1999 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Progress OpenEdge v11.3 may incorporate DOM4J v1.6.1 from MetaStuff, Ltd. Such technology is subject to the following terms and conditions: Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved. Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The name "DOM4J" must not be used to endorse or promote products derived from this Software without prior written permission of MetaStuff, Ltd. For written permission, please contact dom4j-info@metastuff.com. 4. Products derived from this Software may not be called "DOM4J" nor may "DOM4J" appear in their names without prior written permission of MetaStuff, Ltd. DOM4J is a registered trademark of MetaStuff, Ltd. 5. Due credit should be given to the DOM4J Project - http://www.dom4j.org THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate GraphicsMagick v1.3.14 from GraphicsMagick Group. Such technology is subject to the following terms and conditions:

.. This text is in reStucturedText format, so it may look a bit odd. .. See http://docutils.sourceforge.net/rst.html for details.

======================================
GraphicsMagick Copyrights and Licenses
======================================

This file is part of the GraphicsMagick software distributed by the GraphicsMagick Group.

["Please note that the legal community considers 15 or more total lines of code or text (not necessarily contiguous) to be significant for the purposes of copyright. Repeated changes such as renaming a symbol has similar significance to changing one line of code."]

The licenses which components of this software fall under are as follows.

1) In November 2002, the GraphicsMagick Group created GraphicsMagick from ImageMagick Studio's ImageMagick and applied the "MIT" style license: Copyright (C) 2002 - 2012 GraphicsMagick Group
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

2) In October 1999, ImageMagick Studio assumed the responsibility for the development of ImageMagick (forking from the distribution by E. I. du Pont de Nemours and Company) and applied a new license:

Copyright (C) 2002 ImageMagick Studio, a non-profit organization dedicated to making software imaging solutions freely available.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files ("ImageMagick"), to deal in ImageMagick without restriction, including without limitation

the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of ImageMagick, and to permit persons to whom the ImageMagick is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of ImageMagick.

The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall ImageMagick Studio be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with ImageMagick or the use or other dealings in ImageMagick.

Except as contained in this notice, the name of the ImageMagick Studio shall not be used in advertising or otherwise to promote the sale, use or other dealings in ImageMagick without prior written authorization from the ImageMagick Studio.

3) From 1991 to October 1999 (through ImageMagick 4.2.9), ImageMagick was developed and distributed by E. I. du Pont de Nemours and Company: Copyright 1999 E. I. du Pont de Nemours and Company

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files ("ImageMagick"), to deal in ImageMagick without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of ImageMagick, and to permit persons to whom the ImageMagick is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of ImageMagick.

The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement.

In no event shall E. I. du Pont de Nemours and Company be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with ImageMagick or the use or other dealings in ImageMagick.
Except as contained in this notice, the name of the E. I. du Pont de Nemours and Company shall not be used in advertising or otherwise to promote the sale, use or other dealings in ImageMagick without prior written authorization from the E. I. du Pont de Nemours and Company.

4) The GraphicsMagick Base64Decode() and Base64Encode() functions are based on source code obtained from OpenSSH. This source code is distributed under the following license: Copyright (c) 2000 Markus Friedl. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

5) Many of the pattern images in coders/logo.c are derived from XFig, which is distributed under the following license:

| FIG : Facility for Interactive Generation of figures
| Copyright (c) 1985-1988 by Supoj Sutanthavibul
| Parts Copyright (c) 1989-2000 by Brian V. Smith
| Parts Copyright (c) 1991 by Paul King

Any party obtaining a copy of these files is granted, free of charge, a full and unrestricted irrevocable, world-wide, paid up, royalty-free, nonexclusive right and license to deal in this software and documentation files (the "Software"), including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons who receive copies from any such party to do so, with the only requirement being that this copyright notice remain intact.

6) The documentation for the composition operators is copied from the rlecomp manual page, which is authored by Rod Bogart and John W. Peterson. Rlecomp is part of the Utah Raster Toolkit distributed by the University of Michigan and the University of Utah. The copyright for this manual page is as follows: Copyright (c) 1986, University of Utah

This software is copyrighted as noted below. It may be freely copied, modified, and redistributed, provided that the copyright notice is preserved on all copies.

There is no warranty or other guarantee of fitness for this software, it is provided solely "as is". Bug reports or fixes may be sent to the author, who may or may not act on them as he desires.

You may not include this software in a program or other software product without supplying the source, or without informing the end-user that the source is available for no extra charge.

If you modify this software, you should include a notice giving the name of the person performing the modification, the date of modification, and the reason for such modification.

7) The source code comprising swab.c is originally derived from libtiff which has the following license:
Copyright (c) 1988-1997 Sam Leffler

Copyright (c) 1991-1997 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

8) The C++ API known as "Magick++", and which resides in the Magick++ directory, is distributed under the following license: Copyright 1999 - 2003 Bob Friesenhahn <bfriesen@simple.dallas.tx.us>

Permission is hereby granted, free of charge, to any person obtaining a copy of the source files and associated documentation files ("Magick++"), to deal in Magick++ without restriction, including without limitation of the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of Magick++, and to permit persons to whom the Magick++ is furnished to do so, subject to the following conditions:

This copyright notice shall be included in all copies or substantial portions of Magick++. The copyright to Magick++ is retained by its author and shall not be subsumed or replaced by any other copyright.

The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall Bob Friesenhahn be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with Magick++ or the use or other dealings in Magick++.

9) The GraphicsMagick HaldClutImagePixels() function in magick/hclut.c is based on source code from the HaldCLUT package by Eskil Steenberg (http://www.quelsolaar.com/technology/clut.html) which is distributed under the following license: Copyright (c) 2005 Eskil Steenberg. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

10) GraphicsMagick makes use of third-party "delegate" libraries to support certain optional features. These libraries bear their own copyrights and licenses, which may be more or less restrictive than the GraphicsMagick license. For convenience, when GraphicsMagick is bundled with (or compiled with) "delegate" libraries, a copy of the licenses for these libraries is provided in a "licenses" directory.

OVERVIEW and LEGAL ISSUES from jpeg.txt file (from GraphicsMagick):

The Independent JPEG Group's JPEG software

README for release 6b of 27-Mar-1998

This distribution contains the sixth public release of the Independent JPEG Group's free JPEG software. You are welcome to redistribute this software and to use it for any purpose, subject to the conditions under LEGAL ISSUES, below. Serious users of this software (particularly those incorporating it into larger programs) should contact IJG at jpeg-info@uunet.uu.net to be added to our electronic mailing list. Mailing list members are notified of updates and have a chance to participate in technical discussions, etc. This software is the work of Tom Lane, Philip Gladstone, Jim Boucher, Lee Crocker, Julian Minguillon, Luis Ortiz, George Phillips, Davide Rossi, Guido Vollbeding, Ge' Weijers, and other members of the Independent JPEG Group. IJG is not affiliated with the official ISO JPEG standards committee.

This package contains C software to implement JPEG image compression and decompression. JPEG (pronounced "jay-peg") is a standardized compression method for full-color and gray-scale images. JPEG is intended for compressing "real-world" scenes; line drawings, cartoons and other non-realistic images are not its strong suit. JPEG is lossy, meaning that the output image is not exactly identical to the input image. Hence you must not use JPEG if you have to have identical output bits. However, on typical photographic images, very good compression levels can be obtained with no visible change, and remarkably high compression levels are possible if you can tolerate a low-quality image. For more details, see the references, or just experiment with various compression settings. This software implements JPEG baseline, extended-sequential, and progressive compression processes. Provision is made for supporting all variants of these processes, although some uncommon parameter settings aren't implemented yet.

For legal reasons, we are not distributing code for the arithmetic-coding variants of JPEG; see LEGAL ISSUES. We have made no provision for supporting the hierarchical or lossless processes defined in the standard. We provide a set of library routines for reading and writing JPEG image files, plus two sample applications "cjpeg" and "djjpeg", which use the library to perform conversion between JPEG and some other popular image file formats. The library is intended to be reused in other applications.
In order to support file conversion and viewing software, we have included considerable functionality beyond the bare JPEG coding/decoding capability; for example, the color quantization modules are not strictly part of JPEG decoding, but they are essential for output to colormapped file formats or colormapped displays. These extra functions can be compiled out of the library if not required for a particular application. We have also included "jpegtran", a utility for lossless transcoding between different JPEG processes, and "rdjpgcom" and "wrjpgcom", two simple applications for inserting and extracting textual comments in JFIF files.

The emphasis in designing this software has been on achieving portability and flexibility, while also making it fast enough to be useful. In particular, the software is not intended to be read as a tutorial on JPEG. (See the REFERENCES section for introductory material.) Rather, it is intended to be reliable, portable, industrial-strength code. We do not claim to have achieved that goal in every aspect of the software, but we strive for it.

We welcome the use of this software as a component of commercial products.

No royalty is required, but we do ask for an acknowledgement in product documentation, as described under LEGAL ISSUES.

LEGAL ISSUES

In plain English:
1. We don't promise that this software works. (But if you find any bugs, please let us know!)
2. You can use this software for whatever you want. You don't have to pay us.
3. You may not pretend that you wrote this software. If you use it in a program, you must acknowledge somewhere in your documentation that you've used the IJG code.

In legalese:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-1998, Thomas G. Lane.

All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.

(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".

(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

ansi2knr.c is included in this distribution by permission of L. Peter Deutsch, sole proprietor of its copyright holder, Aladdin Enterprises of Menlo Park, CA.
ansi2knr.c is NOT covered by the above copyright and conditions, but instead by the usual
distribution terms of the Free Software Foundation; principally, that you must include source code
if you redistribute it. (See the file ansi2knr.c for full details.) However, since ansi2knr.c is not needed
as part of any program generated from the IJG code, this does not limit you more than the foregoing
paragraphs do.

The Unix configuration script "configure" was produced with GNU Autoconf.

It is copyright by the Free Software Foundation but is freely distributable.

The same holds for its supporting scripts (config.guess, config.sub, ltconfig, ltmain.sh). Another
support script, install-sh, is copyright by M.I.T. but is also freely distributable.

It appears that the arithmetic coding option of the JPEG spec is covered by patents owned by IBM,
AT&T, and Mitsubishi. Hence arithmetic coding cannot legally be used without obtaining one or
more licenses. For this reason, support for arithmetic coding has been removed from the free JPEG
software. (Since arithmetic coding provides only a marginal gain over the unpatented Huffman
mode, it is unlikely that very many implementations will support it.)

So far as we are aware, there are no patent restrictions on the remaining code.

The IJG distribution formerly included code to read and write GIF files.

To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed
altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique
does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by
all standard GIF decoders.

We are required to state that "The Graphics Interchange Format(c) is the Copyright property of
CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

Contents of png.txt file (from GraphicsMagick):

This copy of the libpng notices is provided for your convenience. In case of any discrepancy
between this copy and the notices in the file png.h that is included in the libpng distribution, the
latter shall prevail.

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:

If you modify libpng you may insert additional notices immediately following this sentence. libpng
versions 1.0.7, July 1, 2000, through 1.2.0, September 1, 2001, are Copyright (c) 2000 Glenn
Randers-Pehrson and are distributed according to the same disclaimer and license as libpng-1.0.6
with the following individuals added to the list of Contributing Authors

    Simon-Pierre Cadieux
    Eric S. Raymond
    Gilles Vollant

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement.
There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs.
This library is provided with all faults, and the entire risk of satisfactory quality, performance,
accuracy, and effort is with the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998, 1999
Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as
libpng-0.96, with the following individuals added to the list of Contributing Authors:

    Tom Lane
    Glenn Randers-Pehrson
    Willem van Schaik
libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996, 1997 Andreas Dilger

Distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc. For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage. Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any source or altered source distribution. The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

A "png_get_copyright" function is available, for convenient use in "about" boxes and the like:
printf("%s", png_get_copyright(NULL));

Also, the PNG logo (in PNG format, of course) is supplied in the files "pngbar.png" and "pngbar.jpg (88x31) and "pngnow.png" (98x31).

Libpng is OSI Certified Open Source Software. OSI Certified Open Source is a certification mark of the Open Source Initiative.

Glenn Randers-Pehrson
randeg@alum.rpi.edu

September 1, 2001
Contents of tiff.txt file (from GraphicsMagick):
Copyright (c) 1988-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.
Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics. THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Contents of zlib.txt file (from GraphicsMagick):

zlib 1.1.3 is a general purpose data compression library. All the code is thread safe. The data format used by the zlib library is described by RFCs (Request for Comments) 1950 to 1952 in the files ftp://ds.internic.net/rfc/rfc1950.txt (zlib format), rfc1951.txt (deflate format) and rfc1952.txt (gzip format). These documents are also available in other formats from ftp://ftp.uu.net/graphics/png/documents/zlib/zdoc-index.html

All functions of the compression library are documented in the file zlib.h (volunteer to write man pages welcome, contact jloup@gzip.org). A usage example of the library is given in the file example.c which also tests that the library is working correctly. Another example is given in the file minigzip.c. The compression library itself is composed of all source files except example.c and minigzip.c.

To compile all files and run the test program, follow the instructions given at the top of Makefile. In short "make test; make install" should work for most machines. For Unix: "configure; make test; make install"

For MSDOS, use one of the special makefiles such as Makefile.msc.

For VMS, use Make_vms.com or descrip.mms.

Questions about zlib should be sent to <zlib@quest.jpl.nasa.gov>, or to Gilles Vollant <info@winimage.com> for the Windows DLL version.

The zlib home page is http://www.cdrom.com/pub/infozip/zlib/
The official zlib ftp site is ftp://ftp.cdrom.com/pub/infozip/zlib/

Before reporting a problem, please check those sites to verify that you have the latest version of zlib; otherwise get the latest version and check whether the problem still exists or not.

Mark Nelson <markn@tiny.com> wrote an article about zlib for the Jan. 1997 issue of Dr. Dobb's Journal; a copy of the article is available in http://web2.airmail.net/markn/articles/zlibtool/zlibtool.htm

The changes made in version 1.1.3 are documented in the file ChangeLog.
The main changes since 1.1.2 are:

- fix "an inflate input buffer bug that shows up on rare but persistent occasions" (Mark)
- fix gzread and gztell for concatenated .gz files (Didier Le Botlan)
- fix gzseek(...., SEEK_SET) in write mode
- fix crc check after a gseek (Frank Faubert)
- fix miniunzip when the last entry in a zip file is itself a zip file

(J Lilge)
- add contrib/asm586 and contrib/asm686 (Brian Raiter)

See http://www.muppetlabs.com/~breadbox/software/assembly.html
- add support for Delphi 3 in contrib/delphi (Bob Dellaca)
- add support for C++Builder 3 and Delphi 3 in contrib/delphi2 (Davide Moretti)
- do not exit prematurely in untgz if 0 at start of block (Magnus Holmgren)
- use macro EXTERN instead of extern to support DLL for BeOS (Sander Stoks)
- added a FAQ file

plus many changes for portability.

Unsupported third party contributions are provided in directory "contrib". A Java implementation of zlib is available in the Java Development Kit 1.1 http://www.javasoft.com/products/JDK/1.1/docs/api/Package-java.util.zip.html

See the zlib home page http://www.cdrom.com/pub/infozip/zlib/ for details.

A Perl interface to zlib written by Paul Marquess <pmarquess@bfsec.bt.co.uk> is in the CPAN (Comprehensive Perl Archive Network) sites, such as: ftp://ftp.cis.ufl.edu/pub/perl/CPAN/modules/by-module/Compress/Compress-Zlib*

A Python interface to zlib written by A.M. Kuchling <amk@magnet.com> is available in Python 1.5 and later versions, see http://www.python.org/doc/lib/module-zlib.html

A zlib binding for TCL written by Andreas Kupries <a.kupries@westend.com> is available at http://www.westend.com/~kupries/doc/trf/man/man.html

An experimental package to read and write files in .zip format, written on top of zlib by Gilles Vollant <info@winimage.com>, is available at http://www.winimage.com/zLibDll/unzip.html and also in the contrib/minizip directory of zlib.

Notes for some targets:

- To build a Windows DLL version, include in a DLL project zlib.def, zlib.rc and all .c files except example.c and minigzip.c; compile with -DZLIB_DLL

The zlib DLL support was initially done by Alessandro Iacopetti and is now maintained by Gilles Vollant <info@winimage.com>. Check the zlib DLL home page at http://www.winimage.com/zLibDll

From Visual Basic, you can call the DLL functions which do not take a structure as argument: compress, uncompress and all gz* functions.

See contrib/visual-basic.txt for more information, or get http://www.tcfb.com/dowseware/cmp-z-it.zip

- For 64-bit Irix, deflate.c must be compiled without any optimization. With -O, one libpng test fails. The test works in 32 bit mode (with the -n32 compiler flag). The compiler bug has been reported to SGI.

- zlib doesn't work with gcc 2.6.3 on a DEC 3000/300LX under OSF/1 2.1 it works when compiled with cc.

- on Digital Unix 4.0D (formerly OSF/1) on AlphaServer, the cc option -std1 is necessary to get gzprintf working correctly. This is done by configure.
- zlib doesn't work on HP-UX 9.05 with some versions of /bin/cc. It works with other compilers. Use "make test" to check your compiler.

- gzdopen is not supported on RISCOS, BEOS and by some Mac compilers.

- For Turbo C the small model is supported only with reduced performance to avoid any far allocation; it was tested with -DMAX_WBITS=11 -DMAX_MEM_LEVEL=3

- For PalmOs, see http://www.cs.uit.no/~perm/PASTA/pilot/software.html Per Harald Myrvang <perm@stud.cs.uit.no> Acknowledgments:

The deflate format used by zlib was defined by Phil Katz. The deflate and zlib specifications were written by L. Peter Deutsch. Thanks to all the people who reported problems and suggested various improvements in zlib; they are too numerous to cite here.

Copyright notice:

(C) 1995-1998 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

If you use the zlib library in a product, we would appreciate *not* receiving lengthy legal documents to sign. The sources are provided for free but without warranty of any kind. The library has been entirely written by Jean-loup Gailly and Mark Adler; it does not include third-party code.

If you redistribute modified sources, we would appreciate that you include in the file ChangeLog history information documenting your changes.
Progress OpenEdge v11.3 may incorporate International Classes for Unicode (International Components for Unicode) v2.4 from IBM. Such technology is subject to the following terms and conditions:

**ICU License** - The ICU project is licensed under the X License (see also the x.org original), which is compatible with GPL but non-copyleft. The license allows ICU to be incorporated into a wide variety of software projects using the GPL license. The X license is compatible with the GPL, while also allowing ICU to be incorporated into non-open source products. License ICU License - ICU 1.8.1 and later COPYRIGHT AND PERMISSION NOTICE Copyright (c) 1995-2003 International Business Machines Corporation and others All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

Progress OpenEdge v11.3 may incorporate International Components for Unicode v4.8.0. Such technology is subject to the following terms and conditions:

**ICU License** - ICU 1.8.1 and later COPYRIGHT AND PERMISSION NOTICE Copyright (c) 1995-2011 International Business Machines Corporation and others All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.
Progress OpenEdge v11.3 may incorporate java.net args4j v2.0.12 from Kohsuke Kawaguchi. Such technology is subject to the following terms and conditions: Copyright (c) 2003, Kohsuke Kawaguchi All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate Jaxen v1.1-beta-3 from jaxen.org. Such technology is subject to the following terms and conditions: Project License $Id: LICENSE.txt 1128 2006-02-05 21:49:04Z elharo $ Copyright 2003-2006 The Werken Company. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Jaxen Project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate Jing 20030619 portion of Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: Jing Copying Conditions. Copyright (c) 2001-2003 Thai Open Source Software Center Ltd. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Thai Open Source Software Center Ltd nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Third-party JARs - This distribution includes some additional JAR files, which have their own copying conditions:

saxon.jar Comes from the Saxon 6.5.2 distribution and is covered by these conditions
xercesImpl.jar xml-apis.jar Come from the Xerces-J 2.4.0 distribution and are covered by the Apache Software License
isorelax.jar Comes from ISO RELAX 2003/01/08 distribution and is covered by the following license: Copyright (c) 2001-2002, SourceForge ISO-RELAX Project (ASAMI Tomoharu, Daisuke Okajima, Kohsuke Kawaguchi, and MURATA Makoto) Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Progress OpenEdge v11.3 may incorporate Sun's Jmaki framework v1.0 and Sun's Jmaki framework v1.0.3 from Sun Microsystems, Inc. Such technology is subject to the following terms and conditions: Copyright 1994-2006 Sun Microsystems, Inc. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission. This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN MICROSYSTEMS, INC. ("SUN") AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. You acknowledge that this software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.

Progress OpenEdge v11.3 may incorporate Jmaki framework v1.8.0 from Sun Microsystems, Inc. Such technology is subject to the following terms and conditions: Copyright 1994-2006 Sun Microsystems, Inc. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: * Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer. * Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission. This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN MICROSYSTEMS, INC. ("SUN") AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. You acknowledge that this software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.
Progress OpenEdge v11.3 may incorporate JSTL v1.0 from Sun Microsystems, Inc. Such technologies are subject to the following terms and conditions: Code sample License Copyright 1994-2006 Sun Microsystems, Inc. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission. This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN MICROSYSTEMS, INC. ("SUN") AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. You acknowledge that this software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.

Progress OpenEdge v11.3 may incorporate Looks v2.1.2 from JGoodies. Such technology is subject to the following terms and conditions: The BSD License for the JGoodies Looks - Copyright (c) 2001-2007 JGoodies Karsten Lentzsch. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: o Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of JGoodies Karsten Lentzsch nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate OpenSSL v0.9.8 from The OpenSSL Project. Such technology is subject to the following terms and conditions: OpenSSL License. Copyright (c) 1998-2006 The OpenSSL Project. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)." 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org. 5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project. 6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)." THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. OpenSSL includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com). Original SSLeay License. Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved. This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL. This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, Ihash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com). Copyright remains Eric Young’s, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer, and/or other materials provided with the distribution. 3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)." The word ‘cryptographic’ can be left out if the routines from the library being used are not cryptographic related :-). 4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)." THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. The licence and distribution terms for any publically available version or derivative of this code cannot be changed, i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

Progress OpenEdge v11.3 may incorporate Quartz Enterprise Job Scheduler v1.3.2 from James House. Such technologies are subject to the following terms and conditions: Copyright James House (c) 2001-2003 All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. The licence and distribution terms for any publically available version or derivative of this code cannot be changed, i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

Progress OpenEdge v11.3 may incorporate Quartz Enterprise Job Scheduler v1.3.2 from James House. Such technologies are subject to the following terms and conditions: Copyright James House (c) 2001-2003 All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate SLF4J v1.6.2 and SLF4J Log4J v1.6.1 from sfl4j.org. Such technology is subject to the following terms and conditions: Copyright (c) 2004-2008 QOS.ch All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Progress OpenEdge v11.3 may incorporate STAX API (JSR 173) v3.1.1 from Woodstox Project. Such technology is subject to the following terms and conditions: Copyright (c) 2004-2010, Woodstox Project (http://woodstox.codehaus.org/)All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Woodstox XML Processor nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate Trang 20030619 portion of Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: Copyright (c) 2002, 2003 Thai Open Source Software Center Ltd. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Thai Open Source Software Center Ltd nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate xpp3-1.1.3.4.O portion of Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: Indiana University Extreme! Lab Software License Version 1.1.1 Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the Indiana University Extreme! Lab (http://www.extreme.indiana.edu/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Indiana Univeristy" and "Indiana Univeristy Extreme! Lab" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact http://www.extreme.indiana.edu/.

5. Products derived from this software may not use "Indiana Univeristy" name nor may "Indiana Univeristy" appear in their name, without prior written permission of the Indiana University.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate Xpp3 v1.1.3.4 from Indiana University. Such technology is subject to the following terms and conditions: Indiana University Extreme! Lab Software License Version 1.1.1 Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: “This product includes software developed by the Indiana University Extreme! Lab (http://www.extreme.indiana.edu/).” Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names “Indiana University” and “Indiana University Extreme! Lab” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact http://www.extreme.indiana.edu/. 5. Products derived from this software may not use “Indiana University” name nor may “Indiana University” appear in their name, without prior written permission of the Indiana University. THIS SOFTWARE IS PROVIDED “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate XStream v1.3.1 from XStream. Such technology is subject to the following terms and conditions: Copyright (c) 2003-2006, Joe Walnes Copyright (c) 2006-2007, XStream Committers All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of XStream nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate YAJL v0.4.0 from Lloyd Hilaiel. Such technology is subject to the following terms and conditions: Copyright 2007, Lloyd Hilaiel. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. Neither the name of Lloyd Hilaiel nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate GNU ZLIB v1.1.4 and GNU ZLIB v1.2.3 from Jean-loup Gailly & Mark Alder. Such technology is subject to the following terms and conditions: Copyright notice: (C) 1995-2002 Jean-loup Gailly and Mark Adler This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software. Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions: 1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required. 2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software. 3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Alder
jloup@gzip.org madler@alumni.caltech.edu

Progress OpenEdge v11.3 may incorporate ZLIB.NET Free v1.0.4 from ComponentAce. Such technology is subject to the following terms and conditions: Copyright (c) 2006-2007, ComponentAce http://www.componentace.com All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of ComponentAce nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate LALR Parser Generator in Java v0.10k. Such technology is subject to the following terms and conditions: CUP Parser Generator Copyright Notice, License, and Disclaimer – Copyright 1996-1999 by Scott Hudson, Frank Flannery, C. Scott Ananian - Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

Progress OpenEdge v11.3 includes the RSA Data Security, Inc. MD5 Message-Digest Algorithm. Copyright ©1991-2, RSA Data Security, Inc. Created 1991. All rights reserved. (MD5 Encryption Library v3.0 and MD5 Encryption vMD5C.C) These technologies are subject to the following terms and conditions: RSA Data Security MD5 message-digest algorithm RSA Data Security, Inc. MD5C.C - RSA Data Security, Inc., MD5 message-digest algorithm Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All rights reserved. License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function. License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing the derived work. RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind. These notices must be retained in any copies of any part of this documentation and/or software.
In this tutorial, you learn how to use Process Modeler to create a model of your process. This model can later be expanded and detailed in the Progress Developer Studio for OpenEdge. This tutorial covers the entire procedure from designing a business process to generating instances of the installed application at runtime.

At the end of this tutorial, you will be able to:

• Design a Process Model. This tutorial shows you how to create a simple process called Approval.
• Import the process model into Progress Developer Studio for OpenEdge.
• Publish the process model from Progress Developer Studio for OpenEdge to Business Process Portal and install it as a Business Process Server application.
• Create an application instance at runtime.

For details, see the following topics:

• Designing a process template diagram
• Defining workstep properties
• Saving and installing the process
• Creating an instance and performing a task

Designing a process template diagram

This chapter describes how to design a process template diagram in Business Process Modeler.
Note: Alternatively, you can design a process template without the knowledge of BPMN shapes. For details, see Working with the Overview page on page 79.

**Reviewing the process requirements**

For Approval, the objective is to define a process in which an employee requests a purchase order, and the manager either approves the request, or rejects it. The employee is notified in both cases.

**Defining the process flow**

In Business Process Modeler, any business process can be represented by a process flow, with each workstep representing a unit of work to be executed.

In Approval, the process flow can be illustrated as shown in Figure 308 on page 452.

**Figure 308: Approval - Process Flow**

- **Start**: The requester starts the process by filling out a request for approval form.
- **Request to Manager**: You need to decide how the request is made, and how the employee forwards it to the manager. A possible method is to access or design a Web form, that is sent to the manager as an attachment.
- **Approved?**: Choose a criterion for approving the request, which in this case is the amount of purchase order.
- **Notify Rejection/Notify Approval**: You can decide a suitable mode for notification, such as an e-mail.

Based on these steps, you can design the process template for Approval.

**Creating the process model**

Open the Business Process Modeler and click **File > New > Process Model** to create a new process template. The New Process Model wizard opens, where you can provide the required information. Some important attributes are:

- **Name (Required)**: Provide a unique model name to identify the new process.
- **Label**: This appears as the title of the process in Business Process Portal. If no Label is provided, then the Name appears as the title.
• Group (Optional): If a group is specified, then only the group members can initiate instances of the defined application.

• Manager (Optional): The default user who receives any task if an invalid user is given as a performer during an instance creation.

In the Business Process Modeler, defining the process flow includes deciding the types of worksteps, the order in which they occur, and through which types of connections they communicate. You can use the drag-and-drop features of the Business Process Modeler to add worksteps and connectors. For more information, see Designing a process template diagram on page 49.

Note that a workstep can have a user, an adapter, or a subprocess as the performer.

The flow of the process with worksteps identified is illustrated in Figure 309 on page 453.

**Figure 309: Approval - Process Model**

1. **Request**: The requester starts the process, providing necessary information.
2. **Approve**: The Manager reviews the request and evaluates it for approval.
3. **Decision**: The Decision gateway represents the choice made in the Approve workstep.
4. **NotifyApprove**: This workstep sends an e-mail to the employee that the request is approved. You can use a customized Email adapter for this step.
5. **NotifyReject**: This workstep sends an e-mail to the employee that the request is rejected. You can use a customized Email adapter for this step.
6. **Or-Join**: This workstep connects two flow paths from NotifyApprove and NotifyReject worksteps to the End workstep.
7. **End**: This is the final workstep. No attributes are attached to this workstep.

**Defining the process information flow**

Business Process Modeler supports defining the flow of information in the process with dataslots. To define the process-specific dataslots, click the *Dataslots* tab in the Business Process Modeler interface. For more information on defining dataslots, see Creating user-defined dataslots on page 101. Define the required dataslots for Approval as described in Table 89 on page 454.
### Table 89: Dataslot Definition

<table>
<thead>
<tr>
<th>Dataslot Name</th>
<th>Purpose</th>
<th>Data Type</th>
<th>Data Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Approved</td>
<td>Answers: “Is the request approved by the Manager?” The Decision gateway uses this dataslot to determine the next step in the process flow.</td>
<td>LOGICAL</td>
<td>True</td>
</tr>
<tr>
<td>EmailContentApprove</td>
<td>Answers: “What is the content of the notification e-mail to be sent to the requester, when the manager approves the request?” In the workstep NotifyApprove, the built-in Javascript feature is used to build this dataslot value at runtime.</td>
<td>CHARACTER</td>
<td>Editable</td>
</tr>
<tr>
<td>EmailContentReject</td>
<td>Answers: “What is the content of the notification e-mail to be sent to the requester when the manager rejects the request?” In the workstep NotifyReject, the built-in Javascript feature is used to build this dataslot value at runtime.</td>
<td>CHARACTER</td>
<td>Editable</td>
</tr>
<tr>
<td>EmailHeader</td>
<td>Answers: “What is the common e-mail header for approval as well as rejection?”</td>
<td>CHARACTER</td>
<td>Editable</td>
</tr>
<tr>
<td>EmailSubject</td>
<td>Answers: “What is the subject of the notification e-mail?”</td>
<td>CHARACTER</td>
<td>Editable</td>
</tr>
<tr>
<td>MailTo</td>
<td>Answers: “To whom is the e-mail sent?”</td>
<td>CHARACTER</td>
<td></td>
</tr>
<tr>
<td>Manager</td>
<td>Answers: “To whom should the request go for approval?” Approval task goes to the Task List of the specified manager.</td>
<td>CHARACTER</td>
<td>Editable</td>
</tr>
<tr>
<td>Remark</td>
<td>Answers: “What remarks does the manager enter while making the decision?”</td>
<td>CHARACTER</td>
<td>Editable, Required</td>
</tr>
<tr>
<td>Request</td>
<td>Answers: “What is the text of the request?”</td>
<td>CHARACTER</td>
<td>Editable</td>
</tr>
<tr>
<td>Subject</td>
<td>Answers: “What is the subject of the request?”</td>
<td>CHARACTER</td>
<td>Editable</td>
</tr>
</tbody>
</table>

### Defining process performers

To define performers, click the **Performers** tab in Business Process Modeler. Click **New**, then select a Performer type. Enter data in the **Performer** dialog box to define the required performers for **Approval** as described in **Table 90** on page 455. For more information on defining performers, see **Defining performers** on page 118.
Table 90: Performer Definition

<table>
<thead>
<tr>
<th>Performer</th>
<th>Type</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emailer</td>
<td>Adapter</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Creator</td>
<td>User</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Manager</td>
<td>User</td>
<td>Manager</td>
</tr>
</tbody>
</table>

Tip: When a performer name is underlined and italicized (as in Manager), it indicates that the performer is a dataslot with a dynamic value. In this case, the performer called Manager is derived from the value of the Manager dataslot at runtime.

Defining workstep properties

You can define the properties of each workstep in the Properties view. An Activity workstep's Properties view has six tabs: General, Description, Fields, Collaboration, Alerts and Advanced tab. For more information on setting workstep properties, see Setting workstep properties on page 217.

Step Table 91 on page 455 describes the worksteps and lists their performers in the Approval sample process.

Table 91: Workstep Properties

<table>
<thead>
<tr>
<th>Workstep</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request</td>
<td>This is the Start workstep for Approval.</td>
</tr>
<tr>
<td>Approve</td>
<td>This workstep is specific to the Manager approving the request.</td>
</tr>
<tr>
<td></td>
<td><strong>Performer:</strong> Manager</td>
</tr>
<tr>
<td>Decision</td>
<td>The Decision gateway represents the available paths.</td>
</tr>
<tr>
<td>NotifyApprove</td>
<td>This is an Email adapter workstep. The adapter sends an e-mail to the specified recipient, which in this case is the Requester. The e-mail notifies the requester regarding the manager's decision to approve the request. Note that the performer is an external performer.</td>
</tr>
<tr>
<td></td>
<td><strong>Performer:</strong> common.externalperformers.Emailer: sendMail</td>
</tr>
<tr>
<td>NotifyReject</td>
<td>This is an Email adapter workstep. The adapter sends an e-mail to the specified recipient, which in this case is the Requester. The e-mail notifies the requester regarding the manager's decision to reject the request. Note that the performer is an external performer.</td>
</tr>
<tr>
<td></td>
<td><strong>Performer:</strong> common.externalperformers.Emailer: sendMail</td>
</tr>
<tr>
<td>Or-Join</td>
<td>This connects two different flow paths from NotifyApprove and NotifyReject worksteps to the End workstep.</td>
</tr>
<tr>
<td>End</td>
<td>This is the final workstep. No attributes are attached to this workstep in particular.</td>
</tr>
</tbody>
</table>
Assigning dataslots to worksteps

From the individual workstep's Properties view, open the Fields tab for Start and Activity worksteps (Dataslots tab for other worksteps) to associate dataslots to each workstep. For more information, see the relevant sections in Setting workstep properties on page 217.

<table>
<thead>
<tr>
<th>Workstep</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request</td>
<td><strong>Editable Dataslots</strong>: Manager, Request, Subject</td>
</tr>
<tr>
<td></td>
<td><strong>Presentation Format</strong>: Form</td>
</tr>
<tr>
<td>Approve</td>
<td><strong>Non-editable Dataslots</strong>: Request, Subject</td>
</tr>
<tr>
<td></td>
<td><strong>Editable Dataslots</strong>: Approved, Remark</td>
</tr>
<tr>
<td></td>
<td><strong>Presentation Format</strong>: Form</td>
</tr>
<tr>
<td>NotifyApprove</td>
<td><strong>Input to Adapter</strong>: EmailContentApprove, EmailSubject, MailTo</td>
</tr>
<tr>
<td>NotifyReject</td>
<td><strong>Input to Adapter</strong>: EmailContentReject, EmailSubject, MailTo</td>
</tr>
</tbody>
</table>

After assigning dataslots, you can also define presentation forms for Start and Activity worksteps by right-clicking the workstep and selecting Open, which opens the Form Editor. For more information on defining presentation forms, see Using Form Editor on page 275.

Saving and installing the process

You can verify and export the process to the Process Repository as an SPT file. You can simulate the process in a simulation project before you export it, in order to quickly test a process model for design flaws, estimated costs, and bottlenecks, as described in Configuring and running simulation on page 339.

Note that the Business Process Modeler output cannot be directly installed in Business Process Server's BPM Portal. You must first publish the project using Progress Developer Studio for OpenEdge (see Publishing the process on page 65). Your Business Process Server administrator can then install this process template in Business Process Portal environment, and make it available to you and other users as an installed application.

In this tutorial, you have learned to design the sample process Approval. You can compare it with the process provided with the Progress Developer Studio for OpenEdge installation, located in the Studio_Home\workspace folder.

Installing the published application

Once you have opened the process model in Progress Developer Studio for OpenEdge, you can directly publish the process template Approval.spt to Business Process Portal.

To install the published application in Business Process Portal:

1. Log into Business Process Portal with administrator privileges. If you do not have administrator privileges, then the Administration module is not visible.
2. In the Administration module, go to Applications > BP Server. From the list of published BP Server applications, select Approval and then click Install.
When you install the Approval process template, it becomes available to other users as an installed application.

If a group or permission is defined at the time of process creation, then the application is available only to the members of that group or members with appropriate permissions.

Creating an instance and performing a task

This example displays how you can create an instance of the Approval application and perform a task.

1. Log into Business Process Portal as an Application User (in this example, as ‘ebms’ user).
2. To create an instance of the Approval application, open the Applications tab in the Home module and, from the installed applications, click the Approval link.

The Start workstep page for the Approval application opens, as shown in Figure 311 on page 457. The dataslots previously defined in the Start workstep are displayed as editable text boxes or read-only fields in this page.

3. Provide the required information in the editable fields, and click Create to initiate the instance.
4. Once you create an instance, the task is assigned to the performer of the next workstep. Go to My Instances to see the Instance status (Figure 312 on page 458). In this example, 'ebms' is the performer of the next workstep.

**Figure 312: Approval Application - Process Instance Status**

![Image](image1)

5. To see the tasks assigned to you, open the My Tasks tab (Figure 313 on page 458) in the Home module. As the user 'ebms', you are assigned the ‘Approve’ task of the ‘Approval (125)’ instance in the ‘Approval’ application.

**Figure 313: Approval Application - Tasks Status**

![Image](image2)
6. As the user ‘ebms’, you can click on the ‘Approve’ task hyperlink to open the Task Details page. Enter data in the editable fields, as required.

**Figure 314: Approval Application - Task Details**

7. Click **Complete** to finish the task.
Tutorial: Using the Graphical Event Logic (GEL) Editor

The following tutorial provides step-by-step instructions for using the Graphical Event Logic (GEL) Editor (as described in Using the Graphical Event Logic Editor on page 316) to configure actions on a designed form.

Let us design a form for entering Billing and Mailing addresses for an online product order with the following form elements:

- Billing address includes the street address, city, state, and postal code boxes, all of which are mandatory.

- Mailing address includes the street address, city, state, and postal code boxes, all of which are mandatory. The mailing address can be either the same as the billing address, or you can specify a different mailing address.

- The **Same as billing address** checkbox: If you select this checkbox, the billing address information is copied to the corresponding boxes in the mailing address. Otherwise, you can enter the mailing address details.

- A divider layout control for displaying an inline message for any missing mailing information. On form load, this control is hidden and is displayed only when an error occurs.

The following steps comprise this tutorial:

1. Design the form for entering the Billing and Mailing addresses, as described in **Step 1: Designing the form** on page 462.

2. Add the required events for the form and controls, as described in **Step 2: Adding the events** on page 463.

3. Configure the actions for the added events using the Graphical Event Logic (GEL) Editor, as described in **Step 3: Configuring actions** on page 463.
For details, see the following topics:

- Step 1: Designing the form
- Step 2: Adding the events
- Step 3: Configuring actions

## Step 1: Designing the form

To design the form for entering addresses, you must create a BPM process (or Web application) project and add a workstep (with Form presentation format). Let us name the workstep, “Enter address.”

**To design the form:**

1. Right-click the "Enter address" form workstep, and select the Open Form option.

   The Form Editor appears.
2. Design the form in the Layout tab with the form elements shown in the figure below.

**Figure 315: Form Design**

![Form Design](image)

3. Add the following controls for the above form design:
   - **Billing Address** containing:
     - Address, with text area control (ID: billing_address and Required attribute value as "true")
     - City, with text field control (ID: billing_city and Required attribute value as "true")
• State, with text field control (ID: billing_state and Required attribute value as "true")
• Postal code, with text field control (ID: billing_postalCode and Required attribute value as "true")

• **Mailing Address** containing:
  • Same as billing address checkbox (ID: mailing_same_as_billing)
  • Divider control (ID: div1)
  • Address, with text area control (ID: mailing_address and Required attribute value as "true")
  • City, with text field control (ID: mailing_city and Required attribute value as "true")
  • State, with text field control (ID: mailing_state and Required attribute value as "true")
  • Postal code, with text field control (ID: mailing_postalCode and Required attribute value as "true")

4. Save your form.

### Step 2: Adding the events

After designing the form for entering addresses, let us now add the events, which are triggered when users:

• Load the form.
• Select the **Same as billing address** checkbox.
• Click **Complete** to submit the form.

**To add the following events:**

1. Select the **Same as billing address** checkbox and from the **Events** tab of **Checkbox Properties** view, add the onClick event, namely, "mailing_same_as_billing_onClick."
2. Click the blank portion of your form and from the **Events** tab of **Form Properties** view, add the following events:
   • onSubmit event, namely, "form_onSubmit"
   • onLoad event, namely, "form_onLoad"

We can now configure the actions for the added events in the **Logic** tab.

### Step 3: Configuring actions

Let us now configure actions for the added events using the GEL Editor in the **Logic** tab.
To configure the following actions:

1. For the "mailing_same_as_billing_onClick" event, configure the action (as shown in the following figure) using the IF-THEN-ELSE action construct with the "Empty/Nonempty string" condition (in the IF block), the "Copy widget value" (in the THEN block), and "Popup message" (in the ELSE block) expressions.

   **Figure 316: IF-THEN-ELSE construct**

   ![IF-THEN-ELSE construct](image)

   At runtime, this action checks if the user has entered the billing address in the **Address** box. If yes, the billing address is copied to the **Address** box of the mailing address, which is then disabled. Otherwise, it displays a popup message, "Enter billing address."

2. As all Mailing address boxes must contain the same details as the corresponding Billing address boxes, add an IF-THEN-ELSE construct (with the same condition and expressions shown in **Figure 316 on page 464**) in the "mailing_same_as_billing_onClick" event for each of the remaining address fields, namely, City, State, and Postal Code.

   **Figure 317: Example**
3. For the "form_onLoad" event, add the "Hide widget" expression (as shown in the following figure).

**Figure 318: Form Load event**

On form load, this action hides the "div1" divider layout control on the form.

4. For the "form_onSubmit" event, configure the action (as shown in the following figure) using the IF-THEN action construct with the "Complex condition" (in the IF block) and the "Inline message" and "Stop script execution" (in the THEN block) expressions.

**Figure 319: IF-THEN construct**

On form submission at runtime, this action checks if the user has entered the mailing address details. If any of the boxes are empty, the following actions are executed:

- Displays the hidden "div1" divider layout on the form.
- An inline message, "Mailing information is missing" is displayed in the "div1" divider layout.
- Executes the "Stop script execution" action to stop the submission of the form.

**Caution:** We recommend you to use the "Stop script execution" expression only if required, as it stops the execution of the script (in this case, the script to submit the form).
Third-party acknowledgments

One or more products in the Progress OpenEdge v11.3 release includes third party components covered by licenses that require that the following documentation notices be provided:
Appendix A: Third-party acknowledgments

Progress OpenEdge v11.3 may incorporate ANT v1.5.4. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1, applies to all versions of up to ant 1.6.0 included. The Apache Software License, Version 1.1 - Copyright (C) 2000-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Ant" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.
Progress OpenEdge v11.3 may incorporate Apache Avalon 2002/08/06, Apache Batik v1.1, Apache Jakarta POI v1.1, Apache POI 2003/05/14, Regexp (Pure Java Regular Expression) v1.2, and XMLTask v1.15.1 from The Apache Software Foundation. Such technologies are subject to the following terms and conditions: The Apache Software License, Version 1.1 Copyright (c) 2000 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: “This product includes software developed by the Apache Software Foundation (http://www.apache.org/).” Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
4. The names “Apache” and “Apache Software Foundation” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
5. Products derived from this software may not be called “Apache”, nor may “Apache” appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>. Portions of this software are based upon public domain software originally written at the National Center for Supercomputing Applications, University of Illinois, Urbana-Champaign.
Appendix A: Third-party acknowledgments

Progress OpenEdge v11.3 may incorporate Apache JAXP v1.3.04 from The Apache Software Foundation. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1 Copyright (c) 1999-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Xalan" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999, Lotus Development Corporation, http://www.lotus.com. For more information on the Apache Software Foundation, please see http://www.apache.org/.
Progress OpenEdge v11.3 may incorporate Crimson v1.1.3 from Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1 Copyright (c) 1999-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Xerces" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

Progress OpenEdge v11.3 may incorporate jfor v0.7.2 from jfor. Such technology is subject to the following terms and conditions: jfor Apache-Style Software License. Copyright (c) 2002 by the jfor project. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the jfor project (http://www.jfor.org)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The name "jfor" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact info@jfor.org. 5. Products derived from this software may not be called "jfor", nor may "jfor" appear in their name, without prior written permission of info@jfor.org.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999, International Business Machines, Inc., http://www.ibm.com. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.
Appendix A: Third-party acknowledgments

Progress OpenEdge v11.3 may incorporate Apache LogKit v1.2 from The Apache Software Foundation. Such technologies are subject to the following terms and conditions: The Apache Software License, Version 1.1 -- Copyright (C) 1997-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "LogKit", "Jakarta" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED \"AS IS\" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>. 
Progress OpenEdge v11.3 may incorporate Xerces for Java XML Parser v2.6.2. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1 Copyright (c) 1999 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Xerces" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999, International Business Machines, Inc., http://www.ibm.com. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Progress OpenEdge v11.3 may incorporate SOAP v2.3.1 from Apache Foundation. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1

Copyright (c) 1999 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "SOAP" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.
Appendix A: Third-party acknowledgments

This software is provided "as is" and any expressed or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the Apache Software Foundation or its contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Progress OpenEdge v11.3 may incorporate ANTLR (Another Tool for Language Recognition) v2.7.6. Such technology is subject to the following terms and conditions: ANTLR 3 License [The BSD License] Copyright (c) 2003-2006, Terence Parr All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the author nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate Castor v0.9.3 from castor.org. Such technology is subject to the following terms and conditions: Original Intalio license - Copyright 1999-2004 (C) Intalio Inc., and others. All Rights Reserved. Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The name "ExoLab" must not be used to endorse or promote products derived from this Software without prior written permission of Intalio Inc. For written permission, please contact info@exolab.org. 4. Products derived from this Software may not be called "Castor" nor may "Castor" appear in their names without prior written permission of Intalio Inc. Exolab, Castor and Intalio are trademarks of Intalio Inc. 5. Due credit should be given to the ExoLab Project (http://www.exolab.org/). THIS SOFTWARE IS PROVIDED BY INTALIO AND CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL INTALIO OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate Decimal Conversion Code (dtoa.c; g_fmt.c; rnd_prod.s; decstrod.c; decstrtof.c; dmisc.c; gdtoa.c; gdtoa.h; gdtoaimp.h; gethex.c; gmisc.c; hdi.m.c; misc.c; smisc.c; strtodg.c; strtord.c; sum.c; ulp.c). Such technologies are subject to the following terms and conditions: dtoa.c License: The author of this software is David M. Gay. Copyright (c) 1991, 2000, 2001 by Lucent Technologies. Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software. THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE. g_fmt.c License: The author of this software is David M. Gay. Copyright (c) 1991, 1996 by Lucent Technologies. Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software. THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.
OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. gotoa.h License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. gotoaimp.h License: The author of this software is David M. Gay. Copyright (C) 1998-2000 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. gethex.c License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. gmisc.c License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. hd_init.c License: The author of this software is David M. Gay. Copyright (C) 2000 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
software and its documentation for any purpose and without fee is hereby granted, provided that
the above copyright notice appear in all copies and that both that the copyright notice and this
permission notice and warranty disclaimer appear in supporting documentation, and that the name
of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of
the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES
BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. misc.c
License: The author of this software is David M. Gay. Copyright (C) 1998, 1999 by Lucent
Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software
and its documentation for any purpose and without fee is hereby granted, provided that the above
copyright notice appear in all copies and that both that the copyright notice and this permission
notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent
or any of its entities not be used in advertising or publicity pertaining to distribution of the software
without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD
TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY
SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF
CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. smisc.c License:
The author of this software is David M. Gay. Copyright (C) 1998, 1999 by Lucent Technologies All
Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation
for any purpose and without fee is hereby granted, provided that the above copyright notice appear
in all copies and that both that the copyright notice and this permission notice and warranty
disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities
not be used in advertising or publicity pertaining to distribution of the software without specific,
written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS
SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS.
IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL,
INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING
FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF
CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. strtodg.c License: The author of this
software is David M. Gay. Copyright (C) 1998-2001 by Lucent Technologies All Rights Reserved
Permission to use, copy, modify, and distribute this software and its documentation for any purpose
and without fee is hereby granted, provided that the above copyright notice appear in all copies
and that both that the copyright notice and this permission notice and warranty disclaimer appear
in supporting documentation, and that the name of Lucent or any of its entities not be used in
advertising or publicity pertaining to distribution of the software without specific, written prior
permission.
LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. strtord.c License: The author of this software is David M. Gay. Copyright (C) 1998, 2000 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. sum.c License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. ulp.c License: The author of this software is David M. Gay. Copyright (C) 1998, 1999 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Appendix A: Third-party acknowledgments

Progress OpenEdge v11.3 may incorporate DOM4J v1.6.1 from MetaStuff, Ltd. Such technology is subject to the following terms and conditions: Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved. Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The name "DOM4J" must not be used to endorse or promote products derived from this Software without prior written permission of MetaStuff, Ltd. For written permission, please contact dom4j-info@metastuff.com. 4. Products derived from this Software may not be called "DOM4J" nor may "DOM4J" appear in their names without prior written permission of MetaStuff, Ltd. DOM4J is a registered trademark of MetaStuff, Ltd. 5. Due credit should be given to the DOM4J Project - http://www.dom4j.org THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate GraphicsMagick v1.3.14 from GraphicsMagick Group. Such technology is subject to the following terms and conditions:

."This text is in reStucturedText format, so it may look a bit odd. .. See http://docutils.sourceforge.net/rst.html for details.

=================================================================

GraphicsMagick Copyrights and Licenses

=================================================================

This file is part of the GraphicsMagick software distributed by the GraphicsMagick Group.

[*Please note that the legal community considers 15 or more total lines of code or text (not necessarily contiguous) to be significant for the purposes of copyright. Repeated changes such as renaming a symbol has similar significance to changing one line of code.*]

The licenses which components of this software fall under are as follows.

1) In November 2002, the GraphicsMagick Group created GraphicsMagick from ImageMagick Studio’s ImageMagick and applied the ”MIT” style license: Copyright (C) 2002 - 2012 GraphicsMagick Group
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

2) In October 1999, ImageMagick Studio assumed the responsibility for the development of ImageMagick (forking from the distribution by E. I. du Pont de Nemours and Company) and applied a new license:

Copyright (C) 2002 ImageMagick Studio, a non-profit organization dedicated to making software imaging solutions freely available.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (“ImageMagick”), to deal in ImageMagick without restriction, including without limitation

the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of ImageMagick, and to permit persons to whom the ImageMagick is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of ImageMagick.

The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall ImageMagick Studio be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with ImageMagick or the use or other dealings in ImageMagick.

Except as contained in this notice, the name of the ImageMagick Studio shall not be used in advertising or otherwise to promote the sale, use or other dealings in ImageMagick without prior written authorization from the ImageMagick Studio.

3) From 1991 to October 1999 (through ImageMagick 4.2.9), ImageMagick was developed and distributed by E. I. du Pont de Nemours and Company: Copyright 1999 E. I. du Pont de Nemours and Company

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (“ImageMagick”), to deal in ImageMagick without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of ImageMagick, and to permit persons to whom the ImageMagick is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of ImageMagick.

The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall E. I. du Pont de Nemours and Company be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with ImageMagick or the use or other dealings in ImageMagick.
Except as contained in this notice, the name of the E. I. du Pont de Nemours and Company shall not be used in advertising or otherwise to promote the sale, use or other dealings in ImageMagick without prior written authorization from the E. I. du Pont de Nemours and Company.

4) The GraphicsMagick Base64Decode() and Base64Encode() functions are based on source code obtained from OpenSSH. This source code is distributed under the following license: Copyright (c) 2000 Markus Friedl. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

5) Many of the pattern images in coders/logo.c are derived from XFig, which is distributed under the following license:

| FIG : Facility for Interactive Generation of figures
| Copyright (c) 1985-1988 by Supoj Sutanthavibul
| Parts Copyright (c) 1989-2000 by Brian V. Smith
| Parts Copyright (c) 1991 by Paul King

Any party obtaining a copy of these files is granted, free of charge, a full and unrestricted irrevocable, world-wide, paid up, royalty-free, nonexclusive right and license to deal in this software and documentation files (the "Software"), including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons who receive copies from any such party to do so, with the only requirement being that this copyright notice remain intact.

6) The documentation for the composition operators is copied from the rlecomp manual page, which is authored by Rod Bogart and John W. Peterson. Rlecomp is part of the Utah Raster Toolkit distributed by the University of Michigan and the University of Utah. The copyright for this manual page is as follows: Copyright (c) 1986, University of Utah

This software is copyrighted as noted below. It may be freely copied, modified, and redistributed, provided that the copyright notice is preserved on all copies.

There is no warranty or other guarantee of fitness for this software, it is provided solely "as is". Bug reports or fixes may be sent to the author, who may or may not act on them as he desires.

You may not include this software in a program or other software product without supplying the source, or without informing the end-user that the source is available for no extra charge.

If you modify this software, you should include a notice giving the name of the person performing the modification, the date of modification, and the reason for such modification.

7) The source code comprising swab.c is originally derived from libtiff which has the following license:
Copyright (c) 1988-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

8) The C++ API known as "Magick++", and which resides in the Magick++ directory, is distributed under the following license: Copyright 1999 - 2003 Bob Friesenhahn <bfriesen@simple.dallas.tx.us>

Permission is hereby granted, free of charge, to any person obtaining a copy of the source files and associated documentation files ("Magick++"), to deal in Magick++ without restriction, including without limitation of the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of Magick++, and to permit persons to whom the Magick++ is furnished to do so, subject to the following conditions:

This copyright notice shall be included in all copies or substantial portions of Magick++. The copyright to Magick++ is retained by its author and shall not be subsumed or replaced by any other copyright.

The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall Bob Friesenhahn be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with Magick++ or the use or other dealings in Magick++.

9) The Graphics Magick HaldClutImagePixels() function in magick/hclut.c is based on source code from the HaldCLUT package by Eskil Steenberg (http://www.quelsolaar.com/technology/clut.html) which is distributed under the following license: Copyright (c) 2005 Eskil Steenberg. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
THIS SOFTWARE IS PROVIDED BY THE AUTHOR \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

10) GraphicsMagick makes use of third-party "delegate" libraries to support certain optional features. These libraries bear their own copyrights and licenses, which may be more or less restrictive than the GraphicsMagick license. For convenience, when GraphicsMagick is bundled with (or compiled with) "delegate" libraries, a copy of the licenses for these libraries is provided in a "licenses" directory.

OVERVIEW and LEGAL ISSUES from jpeg.txt file (from GraphicsMagick):

The Independent JPEG Group’s JPEG software

README for release 6b of 27-Mar-1998

This distribution contains the sixth public release of the Independent JPEG Group’s free JPEG software. You are welcome to redistribute this software and to use it for any purpose, subject to the conditions under LEGAL ISSUES, below. Serious users of this software (particularly those incorporating it into larger programs) should contact IJG at jpeg-info@uunet.uu.net to be added to our electronic mailing list. Mailing list members are notified of updates and have a chance to participate in technical discussions, etc. This software is the work of Tom Lane, Philip Gladstone, Jim Boucher, Lee Crocker, Julian Minguillon, Luis Ortiz, George Phillips, Davide Rossi, Guido Vollbeding, Ge’ Weijers, and other members of the Independent JPEG Group. IJG is not affiliated with the official ISO JPEG standards committee. OVERVIEW

This package contains C software to implement JPEG image compression and decompression. JPEG (pronounced "jay-peg") is a standardized compression method for full-color and gray-scale images. JPEG is intended for compressing "real-world" scenes; line drawings, cartoons and other non-realistic images are not its strong suit. JPEG is lossy, meaning that the output image is not exactly identical to the input image. Hence you must not use JPEG if you have to have identical output bits. However, on typical photographic images, very good compression levels can be obtained with no visible change, and remarkably high compression levels are possible if you can tolerate a low-quality image. For more details, see the references, or just experiment with various compression settings. This software implements JPEG baseline, extended-sequential, and progressive compression processes. Provision is made for supporting all variants of these processes, although some uncommon parameter settings aren't implemented yet.

For legal reasons, we are not distributing code for the arithmetic-coding variants of JPEG; see LEGAL ISSUES. We have made no provision for supporting the hierarchical or lossless processes defined in the standard. We provide a set of library routines for reading and writing JPEG image files, plus two sample applications "cjpeg" and "djpeg", which use the library to perform conversion between JPEG and some other popular image file formats. The library is intended to be reused in other applications.
In order to support file conversion and viewing software, we have included considerable functionality beyond the bare JPEG coding/decoding capability; for example, the color quantization modules are not strictly part of JPEG decoding, but they are essential for output to colormapped file formats or colormapped displays. These extra functions can be compiled out of the library if not required for a particular application. We have also included "jpegtran", a utility for lossless transcoding between different JPEG processes, and "rdjpgcom" and "wrjpgcom", two simple applications for inserting and extracting textual comments in JFIF files.

The emphasis in designing this software has been on achieving portability and flexibility, while also making it fast enough to be useful. In particular, the software is not intended to be read as a tutorial on JPEG. (See the REFERENCES section for introductory material.) Rather, it is intended to be reliable, portable, industrial-strength code. We do not claim to have achieved that goal in every aspect of the software, but we strive for it.

We welcome the use of this software as a component of commercial products.

No royalty is required, but we do ask for an acknowledgement in product documentation, as described under LEGAL ISSUES.

LEGAL ISSUES

In plain English:

1. We don't promise that this software works. (But if you find any bugs, please let us know!)
2. You can use this software for whatever you want. You don't have to pay us.
3. You may not pretend that you wrote this software. If you use it in a program, you must acknowledge somewhere in your documentation that you've used the IJG code.

In legalese:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-1998, Thomas G. Lane.

All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.

(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".

(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

ansi2knr.c is included in this distribution by permission of L. Peter Deutsch, sole proprietor of its copyright holder, Aladdin Enterprises of Menlo Park, CA.
ansi2knr.c is NOT covered by the above copyright and conditions, but instead by the usual
distribution terms of the Free Software Foundation; principally, that you must include source code
if you redistribute it. (See the file ansi2knr.c for full details.) However, since ansi2knr.c is not needed
as part of any program generated from the IJG code, this does not limit you more than the foregoing
paragraphs do.

The Unix configuration script "configure" was produced with GNU Autoconf.

It is copyright by the Free Software Foundation but is freely distributable.

The same holds for its supporting scripts (config.guess, config.sub, ltconfig, ltmain.sh). Another
support script, install-sh, is copyright by M.I.T. but is also freely distributable.

It appears that the arithmetic coding option of the JPEG spec is covered by patents owned by IBM,
AT&T, and Mitsubishi. Hence arithmetic coding cannot legally be used without obtaining one or
more licenses. For this reason, support for arithmetic coding has been removed from the free JPEG
software. (Since arithmetic coding provides only a marginal gain over the unpatented Huffman
mode, it is unlikely that very many implementations will support it.)

So far as we are aware, there are no patent restrictions on the remaining code.

The IJG distribution formerly included code to read and write GIF files.

To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed
altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique
does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by
all standard GIF decoders.

We are required to state that "The Graphics Interchange Format(c) is the Copyright property of
CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

Contents of png.txt file (from GraphicsMagick):

This copy of the libpng notices is provided for your convenience. In case of any discrepancy
between this copy and the notices in the file png.h that is included in the libpng distribution, the
latter shall prevail.

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:

If you modify libpng you may insert additional notices immediately following this sentence. libpng
versions 1.0.7, July 1, 2000, through 1.2.0, September 1, 2001, are Copyright (c) 2000 Glenn
Randers-Pehrson and are distributed according to the same disclaimer and license as libpng-1.0.6
with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Gilles Vollant

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement.
There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs.
This library is provided with all faults, and the entire risk of satisfactory quality, performance,
accuracy, and effort is with the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998, 1999
Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as
libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik
libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996, 1997 Andreas Dilger

Distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

  John Bowler
  Kevin Bracey
  Sam Bushell
  Magnus Holmgren
  Greg Roelofs
  Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc. For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

  Andreas Dilger
  Dave Martindale
  Guy Eric Schalnat
  Paul Schmidt
  Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage. Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any source or altered source distribution. The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

A "png_get_copyright" function is available, for convenient use in "about" boxes and the like:

  printf("%s",png_get_copyright(NULL));

Also, the PNG logo (in PNG format, of course) is supplied in the files "pngbar.png" and "pngbar.jpg" (88x31) and "pngnow.png" (98x31).

Libpng is OSI Certified Open Source Software. OSI Certified Open Source is a certification mark of the Open Source Initiative.

  Glenn Randers-Pehrson
  randeg@alum.rpi.edu
  September 1, 2001
Appendix A: Third-party acknowledgments

Contents of tiff.txt file (from GraphicsMagick):

Copyright (c) 1988-1997 Sam Leffler

Copyright (c) 1991-1997 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics. THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Contents of zlib.txt file (from GraphicsMagick):

zlib 1.1.3 is a general purpose data compression library. All the code is thread safe. The data format used by the zlib library is described by RFCs (Request for Comments) 1950 to 1952 in the files ftp://ds.internic.net/rfc/rfc1950.txt (zlib format), rfc1951.txt (deflate format) and rfc1952.txt (gzip format). These documents are also available in other formats from ftp://ftp.uu.net/graphics/png/documents/zlib/zdoc-index.html

All functions of the compression library are documented in the file zlib.h (volunteer to write man pages welcome, contact jloup@gzip.org). A usage example of the library is given in the file example.c which also tests that the library is working correctly. Another example is given in the file minigzip.c. The compression library itself is composed of all source files except example.c and minigzip.c.

To compile all files and run the test program, follow the instructions given at the top of Makefile. In short "make test; make install" should work for most machines. For Unix: "configure; make test; make install"

For MSDOS, use one of the special makefiles such as Makefile.msc.

For VMS, use Make_vms.com or descrip.mms.

Questions about zlib should be sent to <zlib@quest.jpl.nasa.gov>, or to Gilles Vollant <info@winimage.com> for the Windows DLL version.

The zlib home page is http://www.cdrom.com/pub/infozip/zlib/

The official zlib ftp site is ftp://ftp.cdrom.com/pub/infozip/zlib/

Before reporting a problem, please check those sites to verify that you have the latest version of zlib; otherwise get the latest version and check whether the problem still exists or not.

Mark Nelson <markn@tiny.com> wrote an article about zlib for the Jan. 1997 issue of Dr. Dobb's Journal; a copy of the article is available in http://web2.airmail.net/markn/articles/zlibtool/zlibtool.htm

The changes made in version 1.1.3 are documented in the file ChangeLog.

The main changes since 1.1.2 are:

- fix "an inflate input buffer bug that shows up on rare but persistent occasions" (Mark)
- fix gzread and gztell for concatenated .gz files (Didier Le Botlan)
- fix gzseek(..., SEEK_SET) in write mode
- fix crc check after a gseek (Frank Faubert)
- fix miniunzip when the last entry in a zip file is itself a zip file  
  (J Lillge)
- add contrib/asm586 and contrib/asm686 (Brian Raiter)
- add support for Delphi 3 in contrib/delphi (Bob Dellaca)
- add support for C++Builder 3 and Delphi 3 in contrib/delphi2 (Davide Moretti)
- do not exit prematurely in untgz if 0 at start of block (Magnus Holmgren)
- use macro EXTERN instead of extern to support DLL for BeOS (Sander Stoks)
- added a FAQ file
  plus many changes for portability.

Unsupported third party contributions are provided in directory "contrib". A Java implementation of zlib is available in the Java Development Kit 1.1
http://www.javasoft.com/products/JDK/1.1/docs/api/Package-java.util.zip.html

See the zlib home page http://www.cdrom.com/pub/infozip/zlib/ for details.

A Perl interface to zlib written by Paul Marquess <pmarquess@bfsec.bt.co.uk> is in the CPAN (Comprehensive Perl Archive Network) sites, such as:

A Python interface to zlib written by A.M. Kuchling <amk@magnet.com> is available in Python 1.5 and later versions, see http://www.python.org/doc/lib/module-zlib.html

A zlib binding for TCL written by Andreas Kupries >a.kupries@westend.com> is available at

An experimental package to read and write files in .zip format, written on top of zlib by Gilles Vollant <info@winimage.com>, is available at http://www.winimage.com/zLibDLL/unzip.html and also in the contrib/minizip directory of zlib.

Notes for some targets:

- To build a Windows DLL version, include in a DLL project zlib.def, zlib.rc and all .c files except example.c and minigzip.c; compile with -DZLIB_DLL

The zlib DLL support was initially done by Alessandro Iacopetti and is now maintained by Gilles Vollant <info@winimage.com>. Check the zlib DLL home page at
http://www.winimage.com/zLibDLL

From Visual Basic, you can call the DLL functions which do not take a structure as argument: compress, uncompress and all gz* functions.

See contrib/visual-basic.txt for more information, or get
http://www.tcfb.com/dowseware/cmp-z-it.zip

- For 64-bit Irix, deflate.c must be compiled without any optimization. With -O, one libpng test fails. The test works in 32 bit mode (with the -n32 compiler flag). The compiler bug has been reported to SGI.

- zlib doesn't work with gcc 2.6.3 on a DEC 3000/300LX under OSF/1 2.1 it works when compiled with cc.

- on Digital Unix 4.0D (formely OSF/1) on AlphaServer, the cc option -std1 is necessary to get gzprintf working correctly. This is done by configure.
- zlib doesn't work on HP-UX 9.05 with some versions of /bin/cc. It works with other compilers. Use "make test" to check your compiler.

- gzdopen is not supported on RISCOS, BEOS and by some Mac compilers.

- For Turbo C the small model is supported only with reduced performance to avoid any far allocation; it was tested with -DMAX_WBITS=11 -DMAX_MEM_LEVEL=3

- For PalmOs, see http://www.cs.uit.no/~perm/PASTA/pilot/software.html Per Harald Myrvang <perm@stud.cs.uit.no> Acknowledgments:

The deflate format used by zlib was defined by Phil Katz. The deflate and zlib specifications were written by L. Peter Deutsch. Thanks to all the people who reported problems and suggested various improvements in zlib; they are too numerous to cite here.

Copyright notice:
(C) 1995-1998 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

   Jean-loup Gailly Mark Adler
   jloup@gzip.org madler@alumni.caltech.edu

If you use the zlib library in a product, we would appreciate *not* receiving lengthy legal documents to sign. The sources are provided for free but without warranty of any kind. The library has been entirely written by Jean-loup Gailly and Mark Adler; it does not include third-party code.

If you redistribute modified sources, we would appreciate that you include in the file ChangeLog history information documenting your changes.
Progress OpenEdge v11.3 may incorporate International Classes for Unicode (International Components for Unicode) v2.4 from IBM. Such technology is subject to the following terms and conditions: ICU License - The ICU project is licensed under the X License (see also the x.org original), which is compatible with GPL but non-copyleft. The license allows ICU to be incorporated into a wide variety of software projects using the GPL license. The X license is compatible with the GPL, while also allowing ICU to be incorporated into non-open source products. License ICU 1.8.1 and later COPYRIGHT AND PERMISSION NOTICE Copyright (c) 1995-2003 International Business Machines Corporation and others All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

Progress OpenEdge v11.3 may incorporate International Components for Unicode v4.8.0. Such technology is subject to the following terms and conditions: ICU License - ICU 1.8.1 and later COPYRIGHT AND PERMISSION NOTICE Copyright (c) 1995-2011 International Business Machines Corporation and others All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.
Progress OpenEdge v11.3 may incorporate java.net args4j v2.0.12 from Kohsuke Kawaguchi. Such technology is subject to the following terms and conditions: Copyright (c) 2003, Kohsuke Kawaguchi All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate Jaxen v1.1-beta-3 from jaxen.org. Such technology is subject to the following terms and conditions: Project License $Id: LICENSE.txt 1128 2006-02-05 21:49:04Z elharo $ Copyright 2003-2006 The Werken Company. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Jaxen Project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate Jing 20030619 portion of Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: Jing Copying Conditions. Copyright (c) 2001-2003 Thai Open Source Software Center Ltd. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Thai Open Source Software Center Ltd nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Third-party JARs - This distribution includes some additional JAR files, which have their own copying conditions:

saxon.jar Comes from the Saxon 6.5.2 distribution and is covered by these conditions
xml-apis.jar Come from the Xerces-J 2.4.0 distribution and are covered by the Apache Software License
isorelax.jar Comes from ISO RELAX 2003/01/08 distribution and is covered by the following license: Copyright (c) 2001-2002, SourceForge ISO-RELAX Project (ASAMI Tomoharu, Daisuke Okajima, Kohsuke Kawaguchi, and MURATA Makoto) Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Appendix A: Third-party acknowledgments

Progress OpenEdge v11.3 may incorporate Sun's Jmaki framework v1.0 and Sun's Jmaki framework v1.0.3 from Sun Microsystems, Inc. Such technology is subject to the following terms and conditions: Copyright 1994-2006 Sun Microsystems, Inc. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission. This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN MICROSYSTEMS, INC. ("SUN") AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. You acknowledge that this software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.

Progress OpenEdge v11.3 may incorporate Jmaki framework v1.8.0 from Sun Microsystems, Inc. Such technology is subject to the following terms and conditions: Copyright 1994-2006 Sun Microsystems, Inc. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: * Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer. * Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission. This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN MICROSYSTEMS, INC. ("SUN") AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILIT OF SUCH DAMAGES. You acknowledge that this software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.
Progress OpenEdge v11.3 may incorporate JSTL v1.0 from Sun Microsystems, Inc. Such technologies are subject to the following terms and conditions: Code sample License Copyright 1994-2006 Sun Microsystems, Inc. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission. This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN MICROSYSTEMS, INC. ("SUN") AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. You acknowledge that this software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.

Progress OpenEdge v11.3 may incorporate Looks v2.1.2 from JGoodies. Such technology is subject to the following terms and conditions: The BSD License for the JGoodies Looks - Copyright (c) 2001-2007 JGoodies Karsten Lentzsch. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of JGoodies Karsten Lentzsch nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate OpenSSL v0.9.8 from The OpenSSL Project. Such technology is subject to the following terms and conditions: OpenSSL License. Copyright (c) 1998-2006 The OpenSSL Project. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)." 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org. 5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project. 6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)." THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. OpenSSL includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com). Original SSLeay License. Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved. This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL. This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com). Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer, and/or other materials provided with the distribution. 3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)." The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :). 4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)." THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. The licence
and distribution terms for any publicly available version or derivative of this code cannot be
changed, i.e. this code cannot simply be copied and put under another distribution licence [including
the GNU Public Licence.]

Progress OpenEdge v11.3 may incorporate Quartz Enterprise Job Scheduler v1.3.2 from James
House. Such technologies are subject to the following terms and conditions: Copyright James
House (c) 2001-2003 All rights reserved. Redistribution and use in source and binary forms, with
or without modification, are permitted provided that the following conditions are met: 1.
Redistributions of source code must retain the above copyright notice, this list of conditions and
the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the documentation and/or other materials
provided with the distribution. THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND
CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS;
OR BUSINESS INTERRUPTION). HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF
 ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate relaxngDatatype.jar portion of Progress Extensions
for Eclipse v2.2. Such technology is subject to the following terms and conditions: Copyright (c)
2001, Thai Open Source Software Center Ltd, Sun Microsystems. All rights reserved. Redistribution
and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met: Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution. Neither the names of the copyright holders
nor the names of its contributors may be used to endorse or promote products derived from this
software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE
COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN
NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Appendix A: Third-party acknowledgments

Progress OpenEdge v11.3 may incorporate SLF4J v1.6.2 and SLF4J Log4J v1.6.1 from sfl4j.org. Such technology is subject to the following terms and conditions: Copyright (c) 2004-2008 QOS.ch All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Progress OpenEdge v11.3 may incorporate STAX API (JSR 173) v3.1.1 from Woodstox Project. Such technology is subject to the following terms and conditions: Copyright (c) 2004-2010, Woodstox Project (http://woodstox.codehaus.org/) All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Woodstox XML Processor nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate Trang 20030619 portion of Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: Copyright (c) 2002, 2003 Thai Open Source Software Center Ltd. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Thai Open Source Software Center Ltd nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate xpp3-1.1.3.4.O portion of Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: Indiana University Extreme! Lab Software License Version 1.1.1 Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the Indiana University Extreme! Lab (http://www.extreme.indiana.edu/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Indiana University" and "Indiana University Extreme! Lab" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact http://www.extreme.indiana.edu/.

5. Products derived from this software may not use "Indiana University" name nor may "Indiana University" appear in their name, without prior written permission of the Indiana University.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Appendix A: Third-party acknowledgments

Progress OpenEdge v11.3 may incorporate Xpp3 v1.1.3.4 from Indiana University. Such technology is subject to the following terms and conditions: Indiana University Extreme! Lab Software License Version 1.1.1 Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: “This product includes software developed by the Indiana University Extreme! Lab (http://www.extreme.indiana.edu/).” Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Indiana Univeristy" and "Indiana Univeristy Extreme! Lab" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact http://www.extreme.indiana.edu/. 5. Products derived from this software may not use "Indiana Univeristy" name nor may "Indiana Univeristy" appear in their name, without prior written permission of the Indiana University. THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY , OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate XStream v1.3.1 from XStream. Such technology is subject to the following terms and conditions: Copyright (c) 2003-2006, Joe Walnes Copyright (c) 2006-2007, XStream Committers All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of XStream nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate YAJL v0.4.0 from Lloyd Hilaiel. Such technology is subject to the following terms and conditions: Copyright 2007, Lloyd Hilaiel. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. Neither the name of Lloyd Hilaiel nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate GNU ZLIB v1.1.4 and GNU ZLIB v1.2.3 from Jean-loup Gailly & Mark Adler. Such technology is subject to the following terms and conditions: Copyright notice: (C) 1995-2002 Jean-loup Gailly and Mark Adler This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software. Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions: 1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required. 2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software. 3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

Progress OpenEdge v11.3 may incorporate ZLIB.NET Free v1.0.4 from ComponentAce. Such technology is subject to the following terms and conditions: Copyright (c) 2006-2007, ComponentAce http://www.componentace.com All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of ComponentAce nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate LALR Parser Generator in Java v0.10k. Such technology is subject to the following terms and conditions: CUP Parser Generator Copyright Notice, License, and Disclaimer – Copyright 1996-1999 by Scott Hudson, Frank Flannery, C. Scott Ananian - Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

Progress OpenEdge v11.3 includes the RSA Data Security, Inc. MD5 Message-Digest Algorithm. Copyright ©1991-2, RSA Data Security, Inc. Created 1991. All rights reserved. (MD5 Encryption Library v3.0 and MD5 Encryption vMD5C.C) These technologies are subject to the following terms and conditions: RSA Data Security MD5 message-digest algorithm RSA Data Security, Inc. MD5C.C - RSA Data Security, Inc., MD5 message-digest algorithm Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All rights reserved. License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function. License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing the derived work. RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind. These notices must be retained in any copies of any part of this documentation and/or software.