



# **IBM ILOG JViews Diagrammer V8.6**

## **Integrating BPMN facilities**

# Copyright

## Copyright notice

© Copyright International Business Machines Corporation 1987, 2009.

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

## Trademarks

IBM, the IBM logo, ibm.com, WebSphere, ILOG, the ILOG design, and CPLEX are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

## Notices

For further copyright information see `<installdir>/license/notices.txt`.

## *Table of contents*

<b>Integrating BPMN facilities.....</b>	<b>5</b>
<b>What is BPMN?.....</b>	<b>6</b>
<b>Using the BPMN Modeler.....</b>	<b>7</b>
Object palettes.....	8
Using the toolbar.....	9
Creating an Interrupt Intermediate event.....	10
Creating a Flow.....	11
Using Artifacts and Associations.....	12
<b>Integrating BPMN support into a graphical application.....</b>	<b>13</b>
Overview.....	14
BPMN interactors.....	15
BPMN palettes.....	16
The BPMN style sheet.....	17
<b>Index.....</b>	<b>19</b>



# *Integrating BPMN facilities*

Explains how to integrate Business Process Modeling Notation (BPMN) facilities into your graphical application.

## **In this section**

### **What is BPMN?**

Defines Business Process Modeling Notation support in IBM® ILOG® JViews Diagrammer, and how it allows you to build BPMN-compliant diagrams.

### **Using the BPMN Modeler**

The BPMN Modeler lets you edit Business Process Diagrams (BPDs) through a multidocument interface. There is a conventional GUI for handling files, editing commands, view settings, and options.

### **Integrating BPMN support into a graphical application**

Explains how to integrate the BPMN support into your own application.

---

## What is BPMN?

The BPMN support in JViews Diagrammer lets you edit and display Business Process Diagrams (BPDs). It consists of:

- ◆ A sample which shows a BPMN Modeler application.

The BPMN Modeler is based on an instance of `IlvDiagrammer` and makes use of the other items of BPMN support. It permits users to build BPMN-compliant business process diagrams. This application is supplied in **<installdir>/jviews-diagrammer86/samples/diagrammer/bpmnmodeler**.

- ◆ A set of special interactors, needed to create some BPMN elements. These interactors are included in the API.
- ◆ A BPMN style sheet, with extensive use of composite graphics to represent BPMN elements.
- ◆ A set of XML elements suitable for the BPMN style sheet, using the BPMN vocabulary for tasks, links, events, and so on.

# *Using the BPMN Modeler*

The BPMN Modeler lets you edit Business Process Diagrams (BPDs) through a multidocument interface. There is a conventional GUI for handling files, editing commands, view settings, and options.

## **In this section**

### **Object palettes**

Describes the BPMN palettes that allow you to create objects.

### **Using the toolbar**

Describes how to use the BPMN toolbar to create one or more objects.

### **Creating an Interrupt Intermediate event**

Shows how to create an interrupt intermediate event in a task or a collapsed subprocess.

### **Creating a Flow**

Shows how to create a Flow that connects two BPMN Flow objects.

### **Using Artifacts and Associations**

Describes how to use Artifact and Association objects.

---

## Object palettes

In addition to the standard commands, the toolbar provides you with palettes that let you create BPMN objects in your diagrams. These BPMN palettes offer the following categories of object:

- ◆ Activities: Tasks and Subprocesses
- ◆ Start Events
- ◆ End Events
- ◆ Intermediate Events
- ◆ Gateways
- ◆ Flows
- ◆ Artifacts and Associations
- ◆ Pools and Lanes



---

## Using the toolbar

### **To create an object:**

1. Select an icon in the toolbar.
2. Click at the location where you want your object to be created.

By default, after you have created an object, the icon is no longer selected.

### **To create more than one object of the same type:**

- ◆ Select the Sticky Modes option in the Options menu before selecting an icon.

Once this option is selected, a toolbar icon remains selected until you select another one.

---

## Creating an Interrupt Intermediate event

**To create an interrupt intermediate event in a task or a collapsed subprocess:**

1. Select the intermediate event icon in the toolbar.
2. Click the task or collapsed subprocess for which you want to create an interrupt intermediate event.

**To create an interrupt intermediate event (or another intermediate event) in an expanded subprocess:**

1. Select the intermediate event icon in the toolbar.
2. Click the bottom border of the subprocess.

If you click elsewhere, you create an intermediate event that is not an interrupt intermediate event.

---

## Creating a Flow

**To create a Flow that connects two BPMN Flow objects:**

1. Choose a flow type in the toolbar.
2. Click a valid source object in the diagram.  
Valid source objects are highlighted when you move the mouse over them.
3. Click a valid target object in the diagram.  
Valid target objects for the chosen flow and source are highlighted when you move the mouse over them.

---

## Using Artifacts and Associations

IBM® ILOG® JViews BPMN support comes with three predefined Artifacts: Annotation, Data Object, and Group.

Unlike Activities, Gateways, and Events, which are represented by regular JViews nodes, Artifacts are ignored by the graph layout operations.

You cannot use a Sequence Flow or a Message Flow (implemented as JViews links) to connect an Artifact to another object. Instead, you must use an Association.

The Association object is implemented as a specialized node, not as a link. It can be used to connect an Artifact to a node or link. An Association is also ignored during layout operations.

**Note:** The Association object is completely different from the Compensation Association Link (displayed as a link), which is designed to connect a Compensation Interrupt Intermediate Event to its Associated Compensation Activity.

# ***Integrating BPMN support into a graphical application***

Explains how to integrate the BPMN support into your own application.

## **In this section**

### **Overview**

Provides a source code example of integrating BPMN support into a graphical application.

### **BPMN interactors**

Shows how to use the BPMN-specific interactors.

### **BPMN palettes**

Shows how to create and use the BPMN palettes.

### **The BPMN style sheet**

Describes the BPMN Style Sheet, which defines the appearance and behavior of BPMN objects.

---

## Overview

As an example, you can look at the source code of the BPMN Modeler sample in `<installdir>/jviews-diagrammer86/samples/diagrammer/bpmnmodeler/src/bpmn/BPMNModeler.java`.

In the BPMN Modeler, a BPMN diagram is viewed and edited in an `IlvDiagrammer` object. An `IlvDiagrammer` object is a Swing component that displays a graph of nodes and links.

The BPMN Modeler is derived from the `IlvBPMNModeler` class. It uses BPMN-specific interactors, palettes, and style sheet.

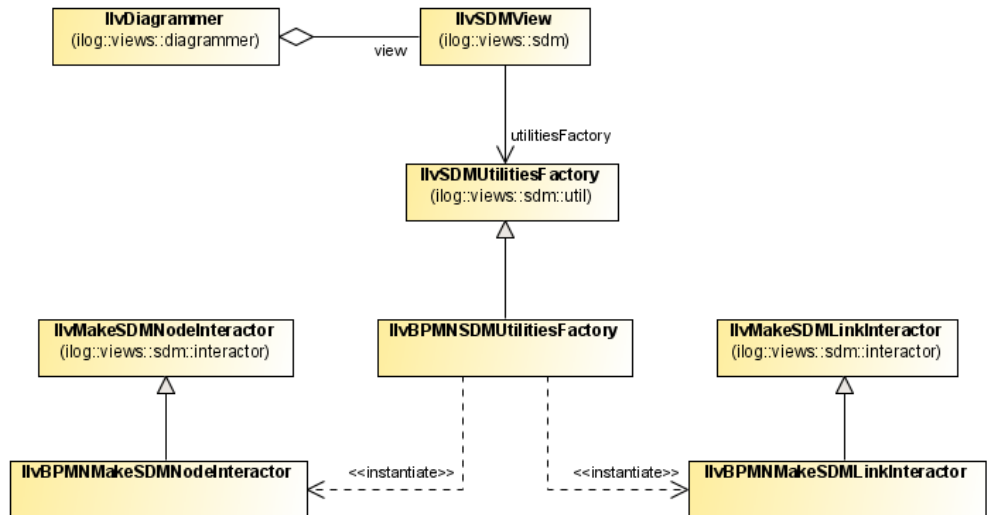
## BPMN interactors

You use the BPMN-specific interactors through the BPMN-specific factory, `IlvBPMNSDMUtilitiesFactory`. This factory is derived from `IlvSDMUtilitiesFactory`. It redefines the methods `createMakeSDMNodeInteractor (ilog.views.sdm.IlvSDEngine, ilog.views.IlvManagerView)` and `createMakeSDMLinkInteractor (ilog.views.sdm.IlvSDEngine, ilog.views.IlvManagerView)`.

The `createMakeSDMNodeInteractor` method returns an `IlvBPMNMakeSDMNodeInteractor` object, derived from `IlvMakeSDMNodeInteractor`. This interactor is used to create all BPMN objects except flows.

The `createMakeSDMLinkInteractor` methods returns `IlvBPMNMakeSDMLinkInteractor` object, derived from `IlvMakeSDMLinkInteractor`. This interactor is used to create BPMN flows.

The following figure illustrates the BPMN interactors class relationships.



*BPMN interactors class relationships*

The following code example shows how to set a BPMN factory for an `IlvDiagrammer` object. The constructor `this` is an instance of an `IlvBPMNModeler` object.

### Setting a BPMN factory

```
IlvDiagrammer diagrammer = new IlvDiagrammer();
diagrammer.getView().setUtilitiesFactory(new IlvBPMNSDMUtilitiesFactory(this));
```

---

## BPMN palettes

The BPMN palettes contain template objects that the user of the modeler can use to create BPMN objects with predefined tags and properties in XML. These palettes are specified by XML files located in

**<installdir>/jviews-diagrammer86/samples/diagrammer/bpmnmodeler/resources/palettes.**

Each XML file contains the description of the template objects for a palette.

To create a BPMN palette as a Swing toolbar from a palette XML file, you use an `IlvDiagrammerPaletteBar` object as shown in the following code example.

### Creating a BPMN palette toolbar

```
new IlvDiagrammerPaletteBar(  
    new URL(getBaseURL(), "resources/palettes/"+filename),  
    getStyleSheetURL(),  
    JToolBar.HORIZONTAL);
```



---

## The BPMN style sheet

The BPMN Style Sheet is available for you to examine at **<installdir>/jviews-diagrammer86/data/bpmn/bpmn.css**. Any files that it refers to are in the same directory. It defines the appearances and behavior of the BPMN objects. In addition to the definition of graphic properties for the objects (such as foreground color, line width), the style sheet defines the structure of the graphic objects. The nodes of the graph are represented by composite graphics, and the composition of these objects is specified in the style sheet.

The following code example shows the style rule that sets the foreground color of the Intermediate Events to blue.

### A style rule that sets a property value

```
#intermediate {  
    foreground : "blue";  
}
```

The BPMN Style Sheet is also in a JAR file; in fact, the BPMN Modeler sample makes use of the style sheet in the JAR file.



# Index

- A**
  - Annotation **12**
- B**
  - BPMN
    - Artifacts **12**
    - Associations **12**
    - Flows **11**
    - interactors **15**
    - interrupt intermediate events **10**
    - palettes **8, 16**
    - Sticky Modes option **9**
    - style sheet **17**
    - toolbar **8**
  - BPMN Modeler **7**
- D**
  - Data Object **12**
- G**
  - Group **12**
- I**
  - interrupt intermediate event **10**
- M**
  - multidocument interface **8**
- S**
  - Sticky Modes option **9**