TIBCO iProcess® Modeler

Procedure Management

Software Release 11.4
July 2013

Two-Second Advantage®

Important Information

Contents

Preface

This guide explains how to manage iProcess procedure objects - procedures, sub-procedures, sub-procedure parameter templates, libraries, and versions.

This guide is aimed at business analysts, system designers, IT specialists, and other staff involved in the development and control of iProcess procedures. It assumes that you are familiar with the basics of procedure definition in iProcess. You should have either attended a suitable iProcess training course, or be familiar with the material presented in *TIBCO iProcess Modeler Getting Started* and *TIBCO iProcess Modeler Basic Design*.

Topics

Related Documentation

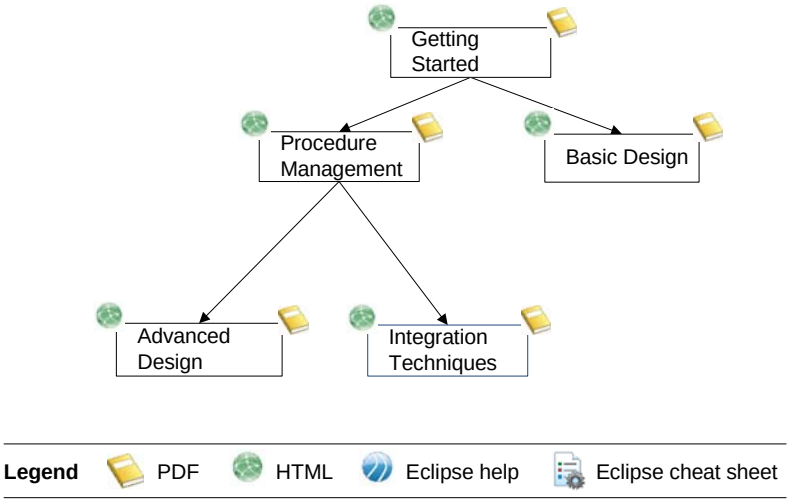
This section lists documentation resources you may find useful.

TIBCO iProcess Modeler Documentation

The following documents form the TIBCO iProcess Modeler and TIBCO iProcess® Workspace (Windows) documentation set, which are supplied with the TIBCO iProcess Workspace (Windows) software:

- *TIBCO iProcess Workspace (Windows) Installation* Read this manual for instructions on site preparation and installation.
- *TIBCO iProcess Workspace (Windows) Release Notes* Read the release notes for a list of new and changed features. This document also contains lists of known issues and closed issues for this release.
- TIBCO iProcess Suite Documentation This documentation set contains all the manuals for TIBCO iProcess Modeler, TIBCO iProcess® Workspace (Windows), and other TIBCO products in TIBCO iProcess® Suite. The manuals for TIBCO iProcess Modeler and TIBCO iProcess Workspace (Windows) are as follows:
  - *TIBCO iProcess Workspace (Windows) User's Guide*
  - *TIBCO iProcess Modeler Getting Started*
  - *TIBCO iProcess Modeler Procedure Management*
  - *TIBCO iProcess Modeler Basic Design*
  - *TIBCO iProcess Modeler Advanced Design*
  - *TIBCO iProcess Modeler Integration Techniques*
  - *TIBCO iProcess Expressions and Functions Reference Guide*
  - *TIBCO iProcess Workspace (Windows) Manager's Guide*

If you are new to iProcess procedure development, you are advised to follow the reading path shown next. The documentation road map shows the relationships between the books and online references in this product's documentation set.



Other TIBCO Product Documentation

You may find it useful to read the documentation for the following TIBCO products:

- TIBCO ActiveMatrix BusinessWorks™
- TIBCO Business Studio™
- TIBCO Enterprise Message Service™
- TIBCO Hawk®
- TIBCO Rendezvous®

Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
SWDIR	TIBCO iProcess Engine installs into a directory. This directory is referenced in documentation as *SWDIR*. The value of *SWDIR* depends on the operating system. For example, • on a Windows server (on the `C:` drive) if *SWDIR* is set to the `C:\swerver\staffw_nod1` directory, then the full path to the `swutil` command is in the `C:\swerver\staffw_nod1\bin\swutil` directory. • on a UNIX or Linux server if *SWDIR* is set to the `/swerver/staffw_nod1` directory, then the full path to the `swutil` command is in the `/swerver/staffw_nod1/bin/swutil` directory or the `$`*SWDIR*`/bin/swutil` directory. **Note**: On a UNIX or Linux system, the environment variable $*SWDIR* should be set to point to the iProcess system directory for the *root* and *swadmin* users.
`code font`	Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example: Use `MyCommand` to start the foo process.
`bold code font`	Bold code font is used in the following ways: • In procedures, to indicate what a user types. For example: Type **`admin`**. • In large code samples, to indicate the parts of the sample that are of particular interest. • In command syntax, to indicate the default parameter for a command. For example, if no parameter is specified, `MyCommand` is enabled: `MyCommand [`**`enable`**` | disable]`

Table 1 General Typographical Conventions (Cont'd)

Convention	Use
italic font	Italic font is used in the following ways: • To indicate a document title. For example: See *TIBCO ActiveMatrix BusinessWorks Concepts*. • To introduce new terms. For example: A portal page may contain several portlets. *Portlets* are mini-applications that run in a portal. • To indicate a variable in a command or code syntax that you must replace. For example: `MyCommand` *PathName*
Key combinations	Key name separated by a plus sign indicate keys pressed simultaneously. For example: Ctrl+C. Key names separated by a comma and space indicate keys pressed one after the other. For example: Esc, Ctrl+Q.
	The note icon indicates information that is of special interest or importance, for example, an additional action required only in certain circumstances.
	The tip icon indicates an idea that could be useful, for example, a way to apply the information provided in the current section to achieve a specific result.
	The warning icon indicates the potential for a damaging situation, for example, data loss or corruption if certain steps are taken or not taken.

Table 2 Syntax Typographical Conventions

Convention	Use
`[ ]`	An optional item in a command or code syntax. For example: `MyCommand [optional_parameter] required_parameter`
`|`	A logical `OR` that separates multiple items of which only one may be chosen. For example, you can select only one of the following parameters: `MyCommand param1 | param2 | param3`

Table 2 Syntax Typographical Conventions (Cont'd)

Convention	Use

`{ }`	A logical group of items in a command. Other syntax notations may appear within each logical group.
	
	For example, the following command requires two parameters, which can be either the pair `param1` and `param2`, or the pair `param3` and `param4`.
	
	`MyCommand {param1 param2} | {param3 param4}`
	
	In the next example, the command requires two parameters. The first parameter can be either `param1` or `param2` and the second can be either `param3` or `param4`:
	
	`MyCommand {param1 | param2} {param3 | param4}`
	
	In the next example, the command can accept either two or three parameters. The first parameter must be `param1`. You can optionally include `param2` as the second parameter. And the last parameter is either `param3` or `param4`.
	
	`MyCommand param1 [param2] {param3 | param4}`

Connecting with TIBCO Resources

How to Join TIBCOmmunity

TIBCOmmunity is an online destination for TIBCO customers, partners, and resident experts. It is a place to share and access the collective experience of the TIBCO community. TIBCOmmunity offers forums, blogs, and access to a variety of resources. To register, go to http://www.tibcommunity.com.

How to Access TIBCO Documentation

You can access TIBCO documentation here:

http://docs.tibco.com

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, contact TIBCO Support as follows:

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:

  http://www.tibco.com/services/support

- If you already have a valid maintenance or support contract, visit this site:

  https://support.tibco.com

  Entry to this site requires a user name and password. If you do not have a user name, you can request one.

Chapter 1 Introduction

This chapter describes the different types of procedure objects you can manage, and the main tools that you can use to manage them.

Topics

Procedure Objects

There are several different types of procedure object that you can manage. These objects are:

- Procedures - which also includes sub-procedures and sub-procedure parameter templates.
- Versions - of procedures.
- Libraries - containers which you can use to organize your procedure objects.
- Shortcuts - references to procedure objects that exist somewhere else in the library structure.

Procedures

Procedures are the basic component of iProcess operation. For the purposes of procedure management, procedures also include:

- **Sub-procedures**. A sub-procedure is simply a procedure that can only be called from another procedure. It can receive case data from its parent procedure and can return case data to it.
- **Sub-procedure parameter templates**. A parameter template is used to define a consistent set of input and output data for multiple sub-procedures.

For more information about how to use:

- sub-procedures, see “Using and Defining Sub-Procedures” in *TIBCO iProcess Modeler Advanced Design*.
- sub-procedure parameter templates, see “Defining Sub-Procedure Parameter Templates” in *TIBCO iProcess Modeler Advanced Design*.

Versions

When you create a procedure you create a **version** of that procedure, which has a unique **version number**. Version control provides a way of keeping track of changes made to a procedure.

Libraries

Libraries are simply containers you put procedures in, in the same way that you put files in folders (or directories). As the number of procedures you have grows, you can expand and develop your library structure to organize your procedures as effectively as possible.

Every iProcess system has a default root procedure library called **Procedure Management**. It provides the access point to all other libraries and procedures.

Shortcuts

A shortcut is a reference to a procedure object that exists somewhere else in your library structure. You can create a shortcut to a procedure, sub-procedure or sub-procedure parameter template. You cannot create a shortcut to a library.

Shortcuts provide a convenient means of accessing frequently-used procedure objects without having to continually navigate through the library structure. When you perform an operation on a shortcut, you are actually performing the operation on the referenced procedure object, with the following exceptions:

- You can move a shortcut.
- You can copy a shortcut (and so have multiple shortcuts to the same procedure object, in different locations).
- When you delete a shortcut, the referenced procedure object is not affected.

Procedure Management Tools

iProcess provides three main tools that you can use to perform procedure management tasks. These tools are:

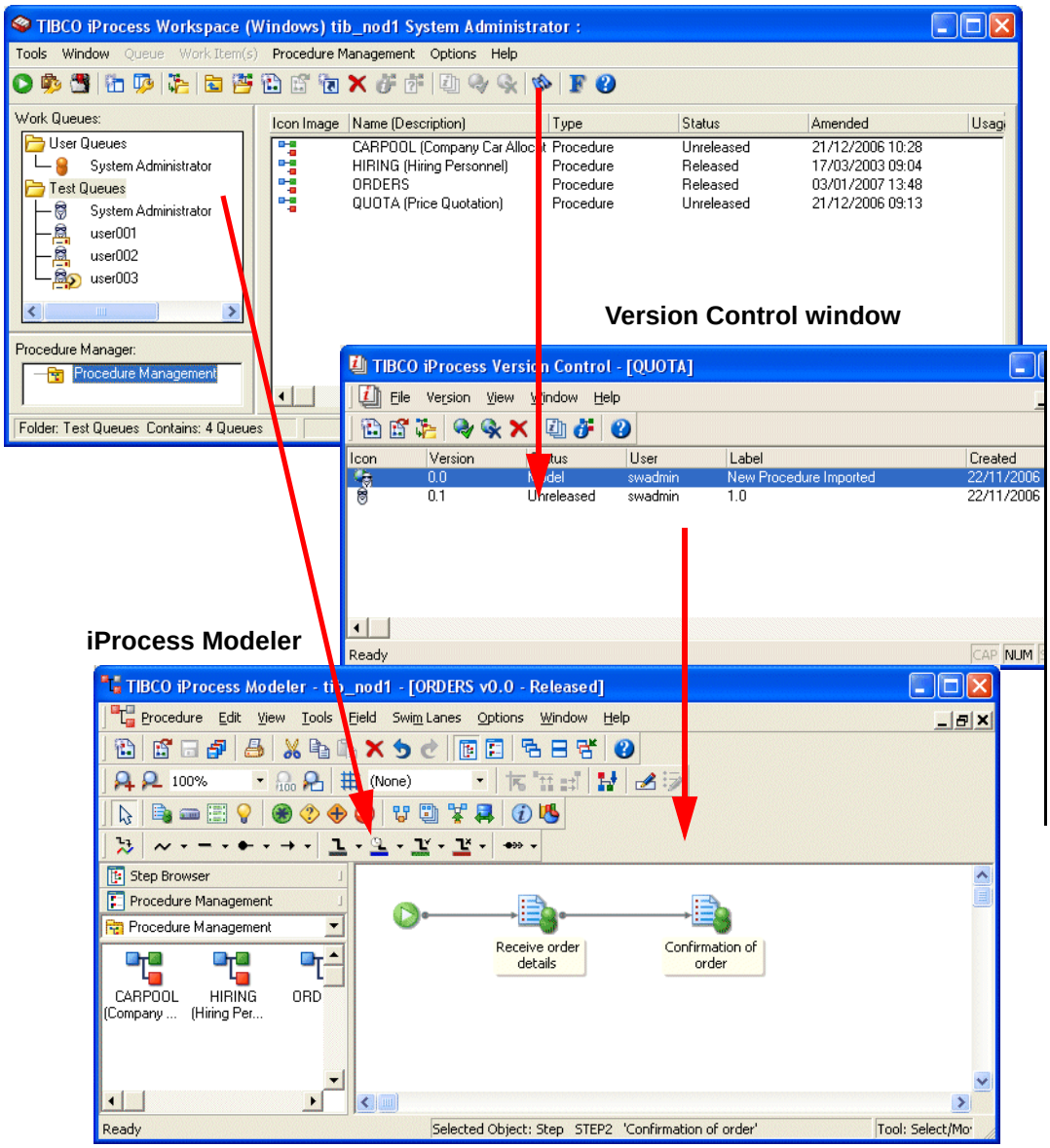
- The Procedure Manager, which allows you to organize and manage your procedures as a hierarchical structure of procedure libraries. You access the Procedure Manager from TIBCO iProcess Workspace (Windows).
- The Version Control Window, which allows you to manage procedure versions. You access the Version Control window from the Procedure Manager.
- The TIBCO iProcess Modeler, which allows you to view and edit the definition of a (specific version of a) procedure. You access TIBCO iProcess Modeler from either the Procedure Manager or from the Version Control window.

The diagram on the following page shows how these tools interact.

You can perform certain tasks from only one tool; others from multiple tools. For example, you can only view a procedure's version history from the Version Control window, but you can release a version of a procedure from either the Version Control window or the Procedure Manager.

The following sections describe each tool in more detail.

Procedure Manager



The Procedure Manager

The Procedure Manager allows you to organize and manage your procedures as a hierarchical structure of procedure libraries, in the same way as, for example, you manage files and directories.

The following sections describe:

Displaying the Procedure Manager

The Procedure Manager is part of TIBCO iProcess Workspace (Windows). It is only available if you are logged in as a user with a MENUNAME attribute of PRODEF or ADMIN.

You can display the Procedure Manager either:

- in place of the Work Queue Manager window. To do this, click **Window > View Procedure Manager**. The Work Queue Manager is replaced by the Procedure Manager.

  If you want to return to the Work Queue Manager at any time, click **Window > View Work Queue List**.

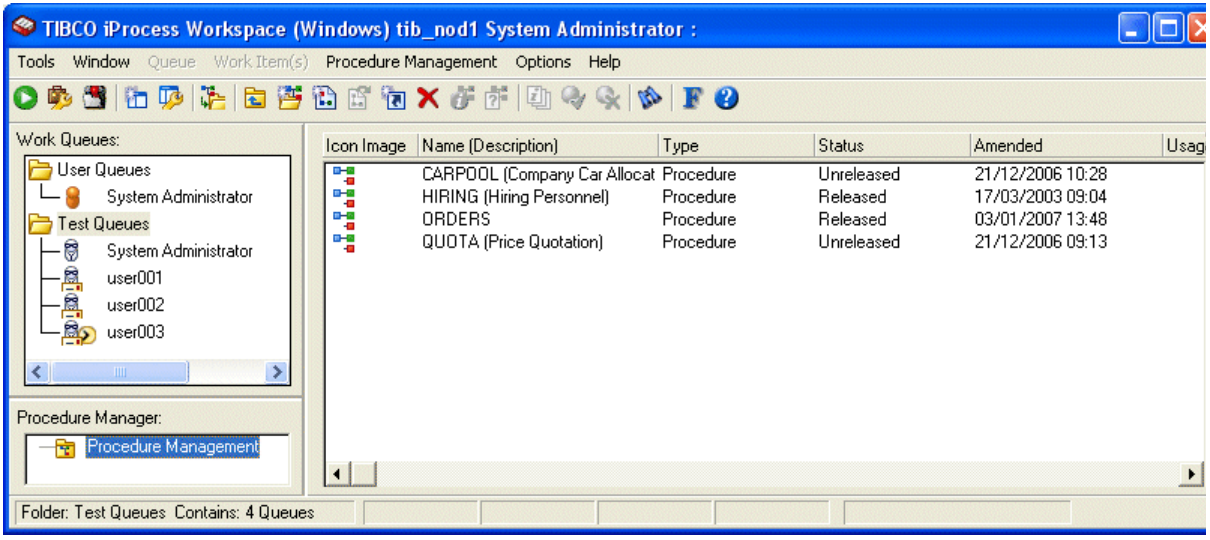
- by splitting the Work Queue Manager window. To do this, click **Window > Split Left Hand Window**. The work queues list is displayed at the top of the left-hand window and the procedure library list is displayed underneath it.

  If you want to return to using a single window, click in either the Work Queue Manager or Procedure Manager, depending on which one you want to use, then click **Window > Split Left Hand Window** again.

The Procedure Manager is divided into two parts:

- The **Procedure Library list**, on the left-hand side of the window, shows the procedure libraries that you have access to.

- The **Procedure Details list**, on the right-hand side of the window, shows the procedure objects in the currently selected library.



To change the size of either part of the window, drag the bar that separates the two sides.

The Procedure Library List

The Procedure Library list, on the left-hand side of the window, shows the procedure libraries that you have access to. All libraries are held under the root **Procedure Management** library.

The following table describes the icons that are displayed in this list.

Icon	Description
	A closed library. Click the library to open it, displaying its contents in the Procedure Details list.
	The currently open library. The contents of this library are displayed in the Procedure Details list.
	A minus (-) sign next to a library indicates that the library contains further libraries, which are currently shown below this library. Click the minus sign to contract the Procedure Library list to hide these libraries.
	A plus (+) sign next to a library indicates that the library contains further libraries, which are not currently shown. Click the plus sign to expand the Procedure Library list to show these libraries.

The Procedure Details List

The Procedure Details list, on the right-hand side of the window, shows the procedure objects in the currently selected library. By default, the following information is shown about each object:

- its **Icon Image**. By default, an icon is shown that indicates the object's type, as shown in the following table. However, you can define a custom icon for an object if you wish - see Icon Image on page 110.

Icon	Description
	Library
	Procedure
	Shortcut to a procedure
	Sub-procedure
	Shortcut to a sub-procedure

Icon	Description
	Sub-procedure parameter template
	Shortcut to a sub-procedure parameter template
	Locked procedure, sub-procedure or sub-procedure parameter template
	Locked shortcut

- its **Name**.
- its **Type**. If you want to change a procedure's type from being a main procedure to a sub-procedure or vice versa, see Changing the Procedure Type of a Version on page 81.
- its current release **Status**, which determines how you can use the object. See Release Status on page 63.
- the date and time the object was last amended.
- its **Usage Instructions** - the web address (URL) of a web page or other document that contains text describing what the object is used for. See Usage Instructions on page 111.

Note that:

- The Procedure Manager displays and works with the default version of a procedure. If you want to perform an operation on a specific version of a procedure you should use the Version Control window instead. See Using Version Control on page 61 for more information.
- Whether or not you can see, open or edit a particular procedure object depends on the object's access permission rules. See Setting Access Controls on page 114.
- An object is normally shown as **locked** when it is open in TIBCO iProcess Modeler.

  An object can also be locked if it was not closed properly from a previous TIBCO iProcess Modeler session - for example, if the system failed while the procedure was open. If this happens the object cannot be accessed again until it is unlocked in the iProcess database. Please contact TIBCO Technical Support for more information about how to do this.
- Objects are listed first by type (libraries first, then procedures, sub-procedures, sub-procedure parameter templates and shortcuts), then alphabetically.

You can change:

- the order in which columns are displayed, by dragging a column heading left or right to a new position, then releasing it.
- the width of a column, by dragging the right hand border of the column left or right until it is the size you want.
- the columns that are displayed. See Customizing the Procedure Details List on page 10 for more information.

Customizing the Procedure Details List

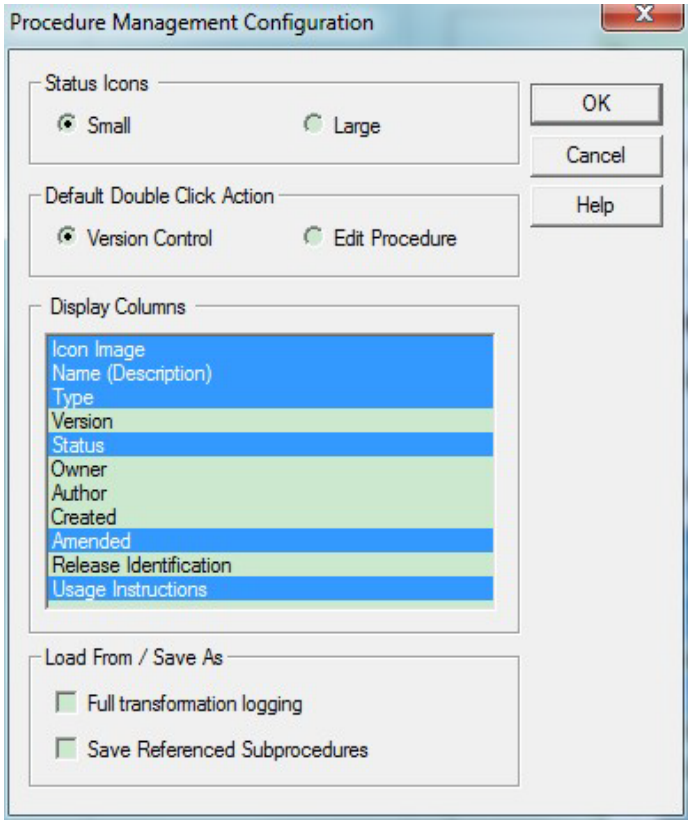
You can tailor the Procedure Details list to:

- display large or small icons.
- open either the Version Control window or iProcess Modeler, when you double-click a single procedure.
- display additional or different information about each procedure object.

To do this:

1. Click anywhere in Procedure Manager, then click  .

The **Procedure Management Configuration** dialog is displayed.



2. In the **Status Icons** area, select either **Large** or **Small**.
3. In the **Default Double Click Action** area, select the action that you want to happen whenever you double-click a single procedure:
   - **Version Control**. Open the Version Control window for the selected procedure.
   - **Edit Procedure**. Open a procedure window in iProcess Modeler for the default version of the selected procedure.
4. In the **Display Columns** area, select each column that you want to display in the Procedure Details list. (Clicking an entry selects it; clicking it again de-selects it.)

5. The following table describes the available columns that you can display.

Column	Description
Icon Image	The bitmap image that is used to graphically represent the object. See Icon Image on page 110.
Name	The object's name. See Setting Properties on page 109.
Type	The object's type. See Setting Properties on page 109.
Version	The version number of the object. See Version Numbers on page 62.
Status	The release status of the object. See Release Status on page 63.
Owner	The user name of the iProcess user who owns the object. See Owner on page 118.
Author	The name (or other identifier) of the iProcess user who created the object. See Extended Properties on page 112.
Created	The date and time that the object was created. See Extended Properties on page 112.
Amended	The date and time that the object was last modified. See Extended Properties on page 112.
Release Identification	Free text that you can use to identify an object. See Extended Properties on page 112.
Usage Instructions	The URL of a web page or other document that contains text describing what the object is used for. See Usage Instructions on page 111.

6. In the Load From/Save As area, check the **Full Transformation Logging** checkbox if you want to produce logs when transforming iProcess procedures into XPDL and vice versa. See "Interpreting the Logs Produced From the Transformation Process" in *TIBCO iProcess Modeler Integration Techniques* for more information.
7. In the Load From/Save As area, check the **Save Referenced Subprocedures** checkbox if you want to save an iProcess procedure and its sub-procedures as

XPDL. For more information about how to save a procedure object as XPDL, see "Saving and Loading iProcess Procedures as XPDL" in *TIBCO iProcess Modeler Integration Techniques*.

Procedure Manager Toolbar Buttons and Menu Items

The following table describes the toolbar buttons that provide procedure management functions, along with their associated menu options (all of which are on the **Procedure Management** menu).

Button	Menu Item (Procedure Management >)	Description
	Update	Update the Procedure Library list and Procedure Details list.
	None	Close the current library and open the parent library.
	New Library	Create a new library in the currently selected library. See Creating a New Library on page 42.
	New Procedure	Create a new procedure in the currently selected library. See Creating a New Procedure From the Procedure Manager on page 40.
	Edit Procedure(s)	Edit the currently selected procedure. See Opening Procedures From the Procedure Manager on page 43.
	New Shortcut	Create a new shortcut to a procedure object. See Creating a New Shortcut on page 42.
None	Load From...	Load XPDL into iProcess Workspace (Windows). See "Translating iProcess Definitions into XPDL" in *TIBCO iProcess Modeler Integration Techniques*.
None	Save As...	Save the currently selected procedure object as XPDL. See "Translating iProcess Definitions into XPDL" in *TIBCO iProcess Modeler Integration Techniques*.

Button	Menu Item (Procedure Management >)	Description
	Delete	Delete the currently selected procedure object. See Deleting Procedures, Libraries and Shortcuts on page 55.
	Properties	View the Properties of the currently selected procedure object. See Working with Properties of Procedures on page 107.
	View Usage Instructions	View the Usage Instructions for the currently selected object. See Usage Instructions on page 111.
	Version Control	Display the version details of the selected procedure(s) in the Version Control window. See The Version Control Window on page 17.
	Release Procedure	Release the latest (Unreleased or Model) version of the selected procedure(s). See Releasing the Unreleased or Model Version of One or More Procedures on page 76.
	Withdraw Procedure	Withdraw the latest (Released or Model) version of the selected procedure(s). See Withdrawing a Specific Version of a Procedure on page 79.
	Find	Search the library structure for a procedure object. See Searching for Libraries, Procedures and Shortcuts on page 52.
n/a	Precedence	Set the precedence option to determine which versions of sub-procedures will be called when test cases are run. See Defining Which Version of a Sub-Procedure is Called on page 50.

Button	Menu Item (Procedure Management >)	Description
	None	Change the display font for the list of procedures shown in the Procedure Manager.
	None	Display the Procedure Manager Help.

The Version Control Window

The Version Control window allows you to view and manage procedure versions.

The following sections describe:

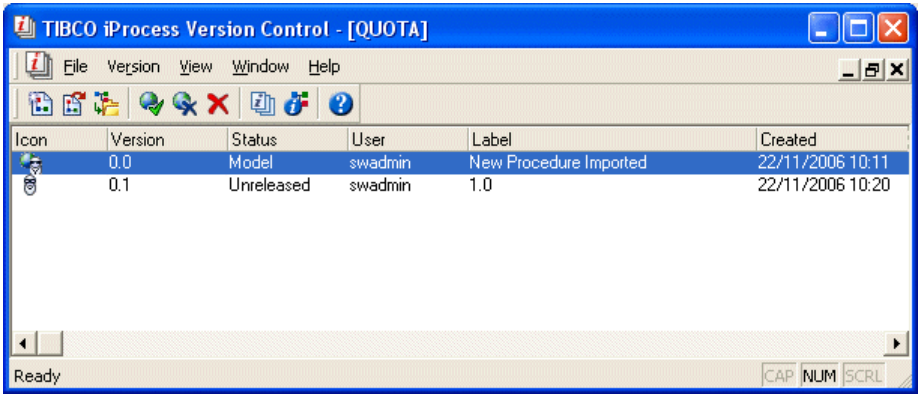
Displaying the Version Control Window

The Version Control window is not displayed by default. To display it, either:

- Select one or more procedures in the Procedure Manager, then click  , or
- Double-click a single procedure in the Procedure Manager.

By default, double-clicking a single procedure displays the Version Control window for the procedure. However, you can change this behavior so that double-clicking instead opens the default version of the procedure in iProcess Modeler. To do this, change the **Default Double Click Action** setting in the **Procedure Management Configuration** dialog.

The Version Control window is displayed, showing the different versions of your selected procedure(s).



If you select further procedures in Procedure Manager and click  , the Version Control window is activated and the version information for each procedure is displayed in a separate version window.

If you want to close a particular version window, click anywhere inside it and click **File > Close**.

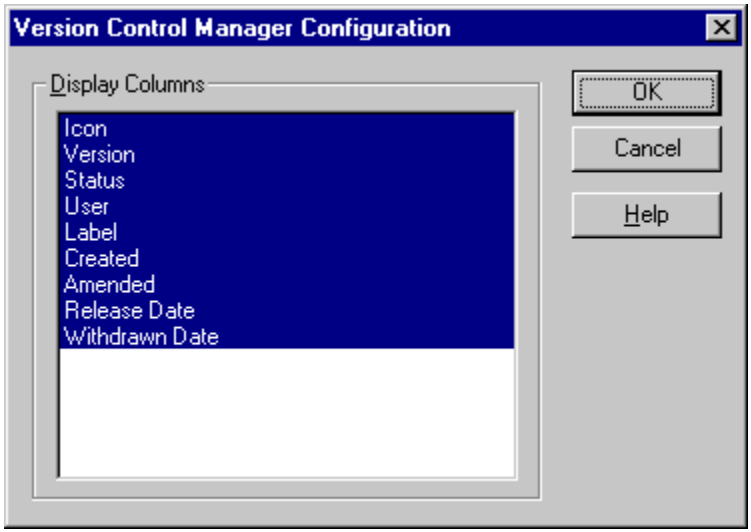
If you want to close the Version Control window itself, click **File > Exit**.

Configuring What Information is Displayed

You can configure the Version Control window to display additional or different information about each version.

To do this:

1. In the Version Control window, click **View > Configuration**. The **Version Control Configuration** dialog is displayed.



2. Select each column that you want to display in the Version Control window. (Clicking an entry selects it; clicking it again de-selects it.)

The following table describes the available columns that you can display.

Column	Description
Icon Image	The image associated with the version's current release **Status**.
Version	The version number of the version. See Version Numbers on page 62.
Status	The release status of the version. See Release Status on page 63.
User	The user name of the iProcess user who last modified the version.
Comment	The comment (if any) associated with the version's current **Status**. See Release Status on page 63.
Created	The date and time when the version was created.
Amended	The date and time when the version was last modified.
Released	The date and time when the version was released.
Withdrawn	The date and time when the version was withdrawn.

Version Control Window Toolbar Buttons and Menu Items

The following table describes the Version Control window toolbar buttons, along with their associated menu options.

Button	Menu Item	Description
	Version > Create Version	Create a new version of a procedure (based on the selected version). See Creating a New Version of a Procedure on page 72.
	Version > View History	Display the version history for the selected version. See Displaying the Audit Trail for a Version on page 84.
	Version > Release	Release the selected version. See Releasing a Specific Version of a Procedure on page 75.
	Version > Withdraw	Withdraw the selected version. See Withdrawing a Specific Version of a Procedure on page 79.
	Version > Delete	Delete the selected version(s). See Deleting a Version of a Procedure on page 82.
	Version > Properties	Display the properties for the selected version. See Setting and Viewing Properties for a Specific Version on page 86.
	View > Refresh	Update the information displayed in the Version Control window.
	Version > Edit Version	Open the selected version in iProcess Modeler. See Opening a Specific Version of a Procedure on page 45.
	Help	Display the Version Control Help.
None	View > Configuration	Configure what information to display about each version. See Configuring What Information is Displayed on page 18.

Button	Menu Item	Description
None	Window > Windows	Display the **Windows** dialog, which allows you to organize, save, or close multiple windows at the same time. *See Working with Multiple Procedure Windows on page 34*.

The TIBCO iProcess Modeler

TIBCO iProcess Modeler allows you to view and edit the definition of a (specific version of a) procedure.

The following sections describe those aspects of iProcess Modeler that relate to procedure management. See:

- Standard Toolbar Buttons and Menu Items on page 22
- Chart Toolbar Buttons and Menu Items on page 24
- The Procedure Browser on page 26
- The Step Browser on page 31
- Working with Multiple Procedure Windows on page 34

For more information about how to use TIBCO iProcess Modeler to design procedures, see *TIBCO iProcess Modeler Basic Design*. (See "Placing Procedure Objects" in that guide for a description of the Object Palette toolbar.)

Standard Toolbar Buttons and Menu Items

The **Standard** toolbar is displayed by default when you open iProcess Modeler.

You can turn the **Standard** toolbar on or off by choosing **View** > **Toolbars** > **Standard Toolbar**.

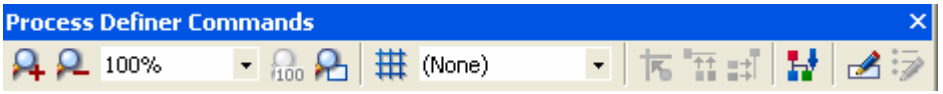
The following table describes the **Standard** toolbar buttons, along with their associated menu options.

Button	Menu Item	Description
	Procedure > New	Create a new procedure window. See Creating a New Procedure in iProcess Modeler on page 41.
	Procedure > Open	Open a procedure. See Opening a Procedure From the Procedure Browser on page 44.
	Procedure > Save	Save the current procedure. See Saving a Procedure on page 46.
	Procedure > Save All	Save all currently open procedures. See Saving a Procedure on page 46.
	Procedure > Print	Print the current procedure. See Printing Procedures on page 57.
	Edit > Cut	Cut the selected object and place it on the clipboard.
	Edit > Copy	Copy the selected objects to the clipboard.
	Edit > Paste	Paste the objects from the clipboard to the cursor location in the procedure.
	Edit > Delete	Delete the currently selected object in the active procedure window.
	Edit > Undo	Undo the previous edit.
	Edit > Redo	Redo the previous edit.
	View > View Step Browser	Show or hide the Step Browser. See The Step Browser on page 31
	View > New Procedure Browser	Display a new Procedure Browser. See The Procedure Browser on page 26.
	Window > Cascade	Cascade all procedure windows.

Button	Menu Item	Description
	Window > Tile	Horizontally tile all procedure windows. **Note: If you want to vertically tile procedure windows you must click Window > Windows. See Working with Multiple Procedure Windows on page 34.**
	Window > Windows	Display the **Windows** dialog, which allows you to organize, save, or close multiple windows at the same time. *See Working with Multiple Procedure Windows on page 34.*
	Help	Display the iProcess Modeler help.

Chart Toolbar Buttons and Menu Items

The **Chart** toolbar is displayed by default when you open iProcess Modeler.



You can turn the **Chart** toolbar on or off by choosing **View** > **Toolbars** > **Chart Toolbar**.

The following table describes the **Chart** toolbar buttons, along with their associated menu options.

Button	Menu Item	Description
	View > Zoom In	Zoom one level of magnification into the active procedure window.
	View > Zoom Out	Zoom one level of magnification out of the active procedure window.
	View > Map	View the entire process definition (the "process map") in the active procedure window. A window outline is shown over the process map. Position the outline over a part of the map and click on it to zoom in to that part of the process definition.
	Field > Definition	Open the **Field Definition** dialog. See "Defining Fields" in *TIBCO iProcess Modeler Basic Design*.
	Edit > Allowed Values	Define a list of values that can be selected for a parameter when the sub-procedure mapping is performed. See "Defining a Sub-Procedure Parameter Template" in *TIBCO iProcess Modeler Advanced Design*. **Note: This option is only available for a sub-procedure parameter template.**
	View > Snap Grid Size	Select the size of the grid.
	View > Normal	Display the default size.
	View > Display Snap Grid	Display the snap-to grid

Button	Menu Item	Description
	Edit > Snap To Grid	Snap the selected object to the center of the nearest grid square.
	Edit > Align Objects	Horizontally align all selected objects with the currently focused object.
	Edit > Align Objects	Vertically align all selected objects with the currently focused object.
	Edit > Layout Top-Down	Change the chart orientation from left-to-right to top-down.

The Procedure Browser

The Procedure Browser is a separate "dockable" window in iProcess Modeler, which allows you to quickly and easily navigate through and display the contents of your procedure libraries. You can use it to:

- *locate and open a procedure*. See Opening a Procedure From the Procedure Browser on page 44.
- *drag and drop a sub-procedure* into a procedure window. The sub-procedure object appears where you release the mouse button, and the **Sub-Procedure Call Definition** dialog appears, with the name of the sub-procedure name already entered as the **Call Reference Name**. See "Adding a Static Sub-Procedure Call" in *TIBCO iProcess Modeler Advanced Design* for more information.

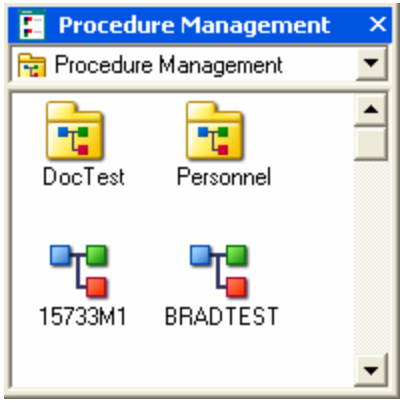
In previous versions of iProcess you could drag and drop sub-procedures from the Procedure Manager into iProcess Modeler. The Procedure Browser makes it easier to do this because, as it is part of iProcess Modeler, you don't have to arrange two separate application windows on the desktop.

Displaying a Procedure Browser

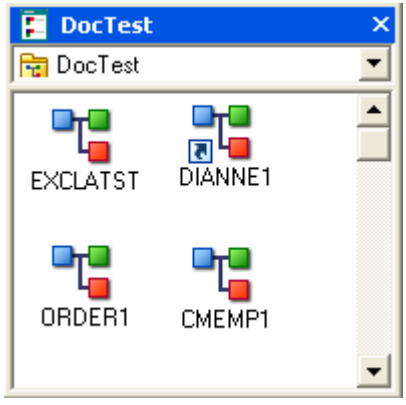
By default, when you open iProcess Modeler the Procedure Browser is not displayed.

To display a new Procedure Browser, click  . The Procedure Browser is displayed in its own window in iProcess Modeler. It contains two areas:

- *a drop-down list of libraries*, which you use to navigate through your library structure.



- *a list area*, which displays the contents of the currently selected library.



Whether or not you can see, open or use a particular procedure object depends on the object's access permission rules. See Setting Access Controls on page 114.

Using Multiple Procedure Browsers

You can display as many Procedure Browsers in iProcess Modeler as you want. You may find it useful to use multiple Procedure Browsers if, for example:

- you use sub-procedures extensively. The Procedure Browser offers a rapid way to place sub-procedures. If you have multiple libraries of sub-procedures, organized (perhaps) by function, using multiple Procedure Browsers gives you rapid access to multiple libraries without having to repeatedly browse your library structure.
- you want rapid access to particular libraries to open procedures.

To open another Procedure Browser, simply click  .

If you want to display several Procedure Browsers (for example, to give you rapid access to multiple libraries), you may find it useful to "dock" them to maximize your available screen space. See Positioning Procedure Browser Windows on page 30 for more information.

When you exit from iProcess Modeler your Procedure Browser windows and their positions are saved in the registry. When you next open iProcess Modeler, these windows are restored in the same state as when you exited.

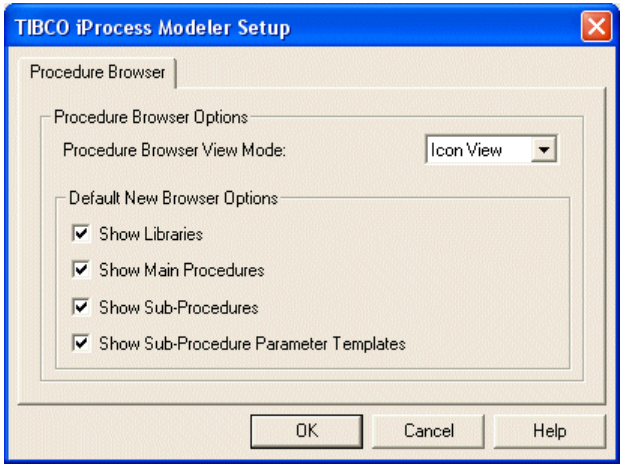
Customizing the Procedure Browser

You can customize two aspects of the Procedure Browser display:

- how the contents of the currently selected library in the list area are displayed - either as large icons or as a list.
- which types of object are displayed - libraries, main procedures, sub-procedures and/or sub-procedure parameter templates.

To do this:

1. In iProcess Modeler, click **Options** > **Setup**. The **TIBCO iProcess Modeler Setup** dialog is displayed.



2. Click the **Procedure Browser** tab.
3. In the **Procedure Browser View Mode** field, select either:
   - **Icon View**, to display the contents of the currently selected library in the list area as large icons.
   - **List View**, to display the contents of the currently selected library in the list area as a list.

Changing this option affects all open Procedure Browser windows as soon as you click **OK** to exit the dialog.

4. In the **Default New Browser Options** area, select or clear the individual check boxes to choose the objects that are displayed in the Procedure Browser.

Changing this option does not affect any existing Procedure Browser windows. It takes effect when you open a new Procedure Browser.

You can also access these options on a per-Procedure Browser basis by right-clicking the Procedure Browser window's title bar. If you do this:

- changing the **Procedure Browser View Mode** field takes effect immediately and affects *all* Procedure Browser windows.
- changing the **Default New Browser Options** takes effect *immediately*, but affects *only* the selected Procedure Browser.

Positioning Procedure Browser Windows

Procedure Browser windows are “dockable”. A dockable window is one that, when you drag it to the edge of a target window, the dockable window’s outline snaps into place along the length of the target window edge.

You can dock a Procedure Browser window to:

- the top (i.e. underneath the menu and toolbars), bottom, left or right-hand side of the iProcess Modeler window.
- the top, bottom, left or right (as appropriate) of another Procedure Browser window that is already docked.
- the top, bottom, left or right (as appropriate) of the Step Browser window, if that is already docked.

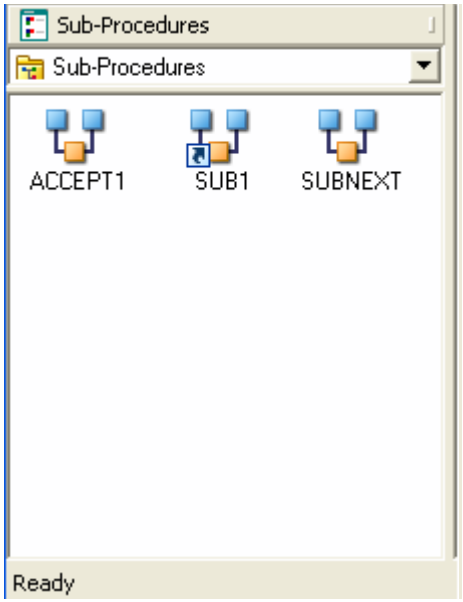
Alternatively, you can use the Procedure Browser as a normal “floating” window that you can position anywhere on your screen.

To dock a Procedure Browser to another window:

1. Drag the Procedure Browser to the edge of the window you want to dock to. An outline of the new Procedure Browser window will appear, showing the new position.
2. If you are happy with the new position, release the mouse button.

If you have trouble repositioning the window, because the outline disappears almost immediately after it appears, drag the mouse very slowly when you are in the vicinity of the target window edge.

You can use docking to "stack" Procedure Browsers (and/or the Step Browser), by dragging a second window over the top edge of an already docked window. You may find this an efficient way to use screen space if, for example, you want to display several Procedure Browsers to give you rapid access to multiple libraries.



When windows are stacked in this way, simply click the title bar of the Procedure Browser you want to look at to display its contents.

The Step Browser

The Step Browser is a separate "dockable" window in iProcess Modeler, which allows you to quickly and easily navigate to any step in your currently open procedures - particularly useful if you have large procedures that span multiple pages with large numbers of steps.

You can use it to:

- *go to a particular step*. Left-clicking on a step activates the procedure window containing that step and jumps the display to that step.
- *display the step menu for a particular step*. Right-clicking on a step displays the step menu for that step, without jumping to it in the procedure window. This means that you can edit the details for steps that are not currently in view or even in the active procedure window.

Displaying the Step Browser

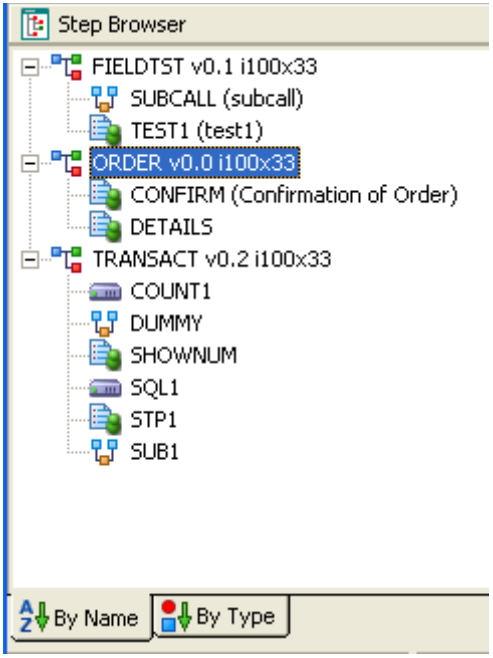
By default, when you open iProcess Modeler the Step Browser is displayed (docked to the left of the procedure window).

You can hide or show the Step Browser at any time by clicking  .

The Step Browser displays a hierarchical view of all steps in all currently open procedures. There are two different views:

- The **By Name** tab shows steps grouped by procedure, then sorted by name. Each step name is preceded by an icon that shows the step type, and followed by the step description in brackets.
- the **By Type** tab shows steps grouped by procedure, then by step type, then sorted by name. Each step name is followed by the step description in brackets.

The following example shows the Step Browser **By Name** view with two procedures, **DOMORD v0.1** and **QUOTA v1.2**, open in the Process Definer.



If you want to display the Step Browser along with several Procedure Browsers (for example, to give you rapid access to multiple libraries), you may find it useful to "dock" them to maximize your available screen space. See Positioning the Step Browser Window on page 33 for more information.

Positioning the Step Browser Window

The Step Browser window is “dockable”. A dockable window is one that, when you drag it to the edge of a target window, the dockable window’s outline snaps into place along the length of the target window edge.

You can dock the Step Browser window to:

- the top (i.e. underneath the menu and toolbars), bottom, left or right-hand side of the iProcess Modeler window.
- the top, bottom, left or right (as appropriate) of a Procedure Browser window that is already docked.

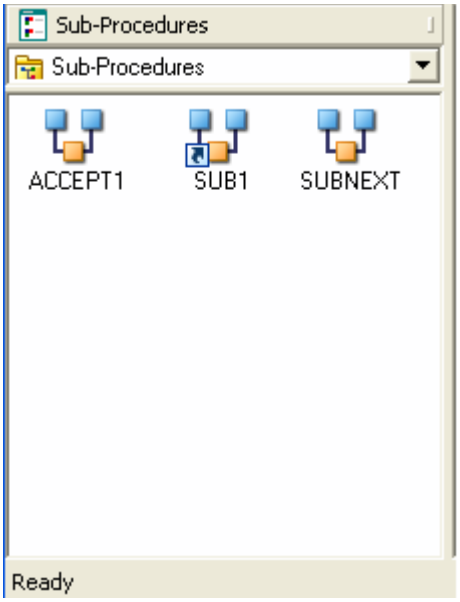
Alternatively, you can use the Step Browser as a normal “floating” window that you can position anywhere on your screen.

To dock the Step Browser to another window:

1. Drag the Step Browser to the edge of the window you want to dock to. An outline of the new Step Browser window will appear, showing the new position.
2. If you are happy with the new position, release the mouse button.

If you have trouble repositioning the window, because the outline disappears almost immediately after it appears, drag the mouse very slowly when you are in the vicinity of the target window edge.

You can use docking to "stack" the Step Browser and multiple Procedure Browsers, by dragging a second window over the top edge of an already docked window. You may find this an efficient way to use screen space if, for example, you want to display several Procedure Browsers to give you rapid access to multiple libraries, along with the Step Browser.



When windows are stacked in this way, simply click the title bar of the Step Browser when you want to display its contents.

Working with Multiple Procedure Windows

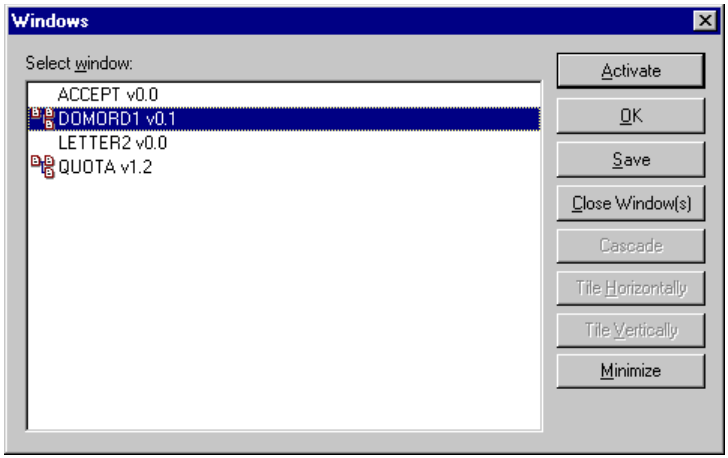
You can display as many procedure windows as you want to in the main iProcess Modeler window. To make it easier to work with large numbers of windows, the iProcess Modeler provides a number of options:

- To horizontally tile your procedure windows, click **Windows > Tile**.
- To cascade your procedure windows, click **Windows > Cascade**.
- Use the Step Browser and Chart toolbars to rapidly find particular steps.
- Use the Windows dialog. The **Windows** dialog provides a number of options that allow you to quickly go to a particular window, and to organize, save, or close multiple windows at the same time.

Using the Windows Dialog

The **Windows** dialog is available in iProcess Modeler or Version Control window. It provides a number of options that allow you to quickly go to a particular window, and to organize, save, or close multiple windows at the same time.

To display the **Windows** dialog, click **Window > Windows**.



The **Windows** dialog lists all the procedure or version windows (as appropriate) you currently have open. You can then:

1. Select one or more windows from the list.
2. Perform any of the operations shown in the following table by clicking the appropriate button.

3. Click **OK** when you have finished with the dialog.

Button	Meaning with: One window selected	Multiple windows selected
Activate	Activate the selected window.	Not available
OK	Close the **Windows** dialog.	
Save	Save the selected procedure(s). **Note: This option is not available in the Version Control window (because there is nothing to save).**	
Close Window(s)	Close the selected window(s). **Note: In iProcess Modeler, you will be prompted to save any procedures you have edited but not already saved.**	
Cascade	Not available	Cascade the selected windows.
Tile Horizontally	Not available	Tile the selected windows horizontally.
Tile Vertically	Not available	Tile the selected windows vertically.
Minimize	Minimize the selected window(s).	

Chapter 2 Managing Procedures, Libraries and Shortcuts

This chapter describes how to manage procedures (including sub-procedures and sub-procedure parameter templates).

Topics

Creating Procedures, Libraries and Shortcuts

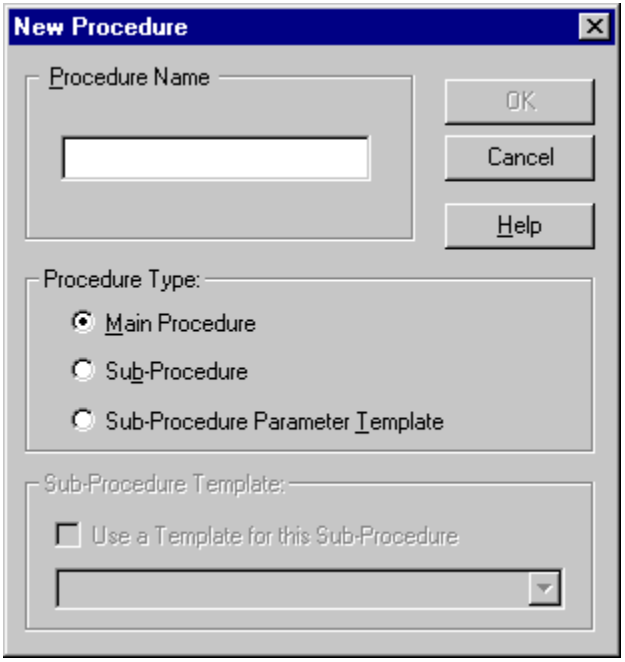
You can create:

- a new procedure - in two ways, either from the Procedure Manager, or in iProcess Modeler.
- a new library - from the Procedure Manager.
- a new shortcut - from the Procedure Manager.

Creating a New Procedure From the Procedure Manager

To create a new procedure from the Procedure Manager:

1. Open the library where you want to create the procedure.
2. Click  . The **New Procedure** dialog is displayed.



3. Enter a **Procedure Name** of up to 8 characters. The name must be unique and can contain any characters except ^ (caret) and, (comma).
4. Select the **Procedure Type** that you want to create - either **Main Procedure**, **Sub-Procedure** or **Sub-Procedure Parameter Template**.

5. If you have chosen **Sub-Procedure**, the **Sub-Procedure Template** section is activated. If you want to base the sub-procedure on a template:
   a. select the **Use a Template for this Sub-Procedure** check box.
   b. select the appropriate template from the drop-down list.

   If you don't want to use a template for this sub-procedure, leave the check box blank.
6. Click **OK**. An entry for the new procedure appears in the library and a new procedure window (or sub-procedure parameter template chart) is displayed in iProcess Modeler
7. Edit the procedure (or template) in iProcess Modeler as required.
8. Click  to save the procedure.

If you do not save the procedure, when you exit iProcess Modeler the entry for it in the Procedure Manager will be deleted.

Creating a New Procedure in iProcess Modeler

You can only create a *main* procedure in iProcess Modeler. If you want to create:

- a sub-procedure, you must first create it as a main procedure, then change its type to a sub-procedure. See Changing the Procedure Type of a Version on page 81.
- a sub-procedure parameter template, you must use the Procedure Manager instead.

To create a new (main) procedure in iProcess Modeler:

1. Click  . A new procedure window is displayed.
2. Edit your procedure as required.
3. Click **Procedure > Save As >** Save As New Procedure to save the procedure.

Creating a New Library

To create a new library:

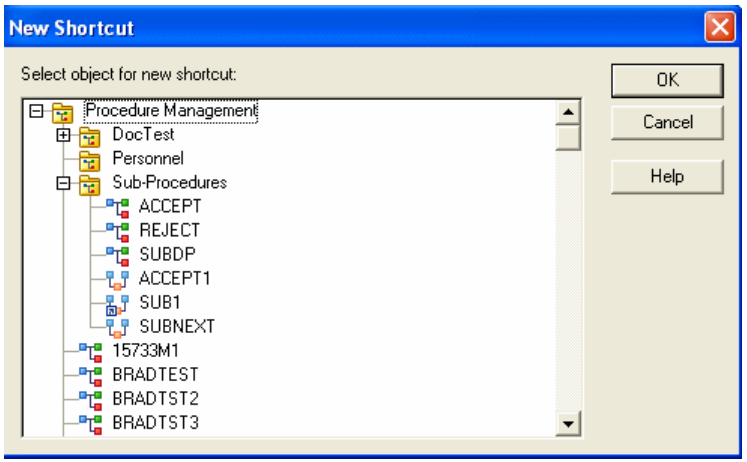
1. In the Procedure Manager, open the library where you want to create the new library.
2. Click  . The **New Procedure Library** dialog is displayed.
3. Enter the name of the new library in the **Library Name** field, of up to 64 characters. You can use any characters except ^ (caret) and , (comma).
4. Click **OK**.

The new library is added under the currently selected library.

Creating a New Shortcut

To create a shortcut to another procedure object:

1. In the Procedure Manager, open the library where you want to create the new shortcut.
2. Click  . The **New Shortcut** dialog is displayed.



3. Select the procedure object that you want to create a new shortcut to.
4. Click **OK**. The shortcut is created with the same name as the procedure object that it references.

Shortcuts are also automatically created if you copy a library that contains procedure objects. Because procedures, sub-procedures and sub-procedure parameter templates must have unique names, they are not copied. Instead, shortcuts to the originals are created in the copied library.

Opening Procedures to View or Edit

Opening a procedure allows you to view or edit it in iProcess Modeler. You can either open the default version or open a specific version of a procedure.

You must have **View** or **Edit** access permission to a procedure to be able to open it. See Setting Access Controls on page 114.

Opening the Default Version of a Procedure

You can open the default version of a procedure from:

- the Procedure Manager
- the iProcess Modeler
- the Procedure Browser

Opening Procedures From the Procedure Manager

You can open the default version of either a single procedure or multiple procedures at once from the Procedure Manager. To do this:

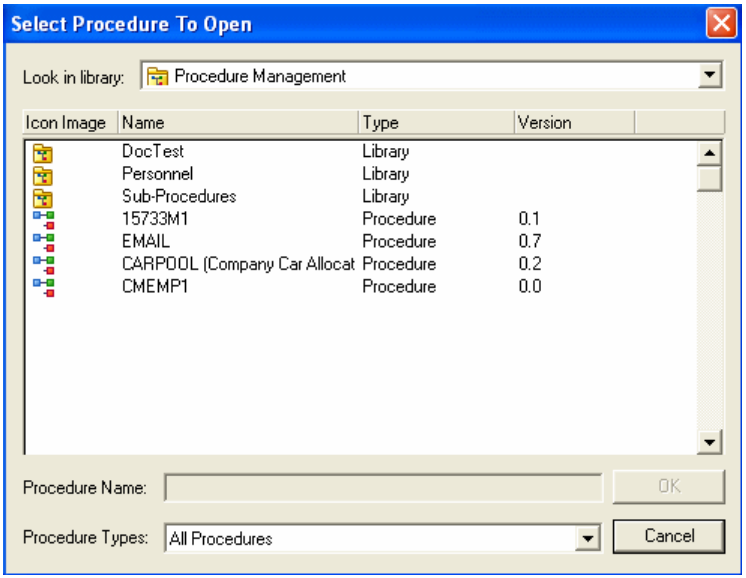
1. Open the library that contains the procedure(s) you want to open. You can only open multiple procedures from the same library.
2. Select the procedure(s) you want to open.
3. Click  . A procedure window is opened in iProcess Modeler for each selected procedure. (The procedure window title bar shows the procedure name and the version.)

Double-clicking a single procedure will either display the Version Control window for the procedure, or open the default version of the procedure in iProcess Modeler. This behavior is controlled by the Default Double Click Action setting in the **Procedure Management Configuration** dialog.

Opening a Procedure in iProcess Modeler

To open the default version of a procedure in iProcess Modeler:

1. Click  . The **Select Procedure To Open** dialog is displayed.



2. Use the **Look in library** drop-down list to navigate to the library containing the procedure you want to open.
3. Select the procedure you want to open from the list.

You can use the **Procedure Types** drop-down list to restrict the display to only show main procedures, sub-procedures or sub-procedure parameter templates.

4. Click **OK**. The procedure definition is displayed in a procedure window.

Opening a Procedure From the Procedure Browser

To open the default version of a procedure from the Procedure Browser:

1. Open the library containing the procedure you want to open.
2. Double-click the procedure.
3. The procedure definition is displayed in a procedure window.

Opening a Specific Version of a Procedure

You can open a specific version of a procedure from the Version Control window.

To open a specific version of a procedure:

1. In the Version Control window, select the version that you want to open.
2. Click  . The version is opened in a new procedure window in iProcess Modeler. If you have opened a version that is:
   - *Incomplete or Unreleased*, it is opened as read-write. You can save any changes you make to the same version.
   - *Released, Model or Withdrawn*, it is opened as read-only. If you make any changes and then try to save the procedure, you will be prompted to save it as a new version or as a new procedure instead.

Saving Procedures

You can save a procedure definition from iProcess Modeler, either:

- as the same procedure (Save).
- as a different procedure (Save As New Procedure).
- as a different version of the same procedure (see Saving a Version of a Procedure as a New Version on page 73).

You can only Save changes to the same version of a procedure if it is Incomplete or Unreleased, and if you have edit permissions to it.

If you make changes to a Released, Model or Withdrawn procedure, or if you do not have edit permissions to the procedure, you must save it either as a different version or as a different procedure.

Saving a Procedure

To save changes to (an Incomplete or Unreleased version of) a procedure:

1. Click anywhere in the procedure window of the procedure that you want to save.
2. Click  .
3. If the **Save Comment** dialog is displayed, enter a comment describing the reason why the procedure has been changed.

The Save Comment dialog may be required, optional or hidden - see the description of the PROC_VER_COMMENT attribute in Configuring Version Control on page 87.

If you are saving:

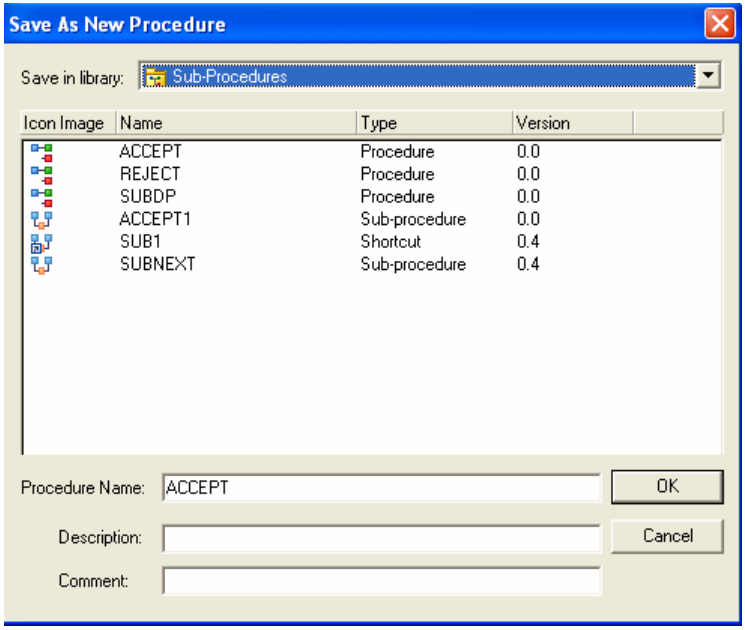
- a *new* procedure, its status is set to Incomplete or Unreleased, as appropriate. Its version number is set to the system default value.
- an *existing* procedure, its status (Incomplete or Unreleased) and version number are unchanged.

If the system is set to increment a procedure's version number whenever changes are made, the minor version number is incremented by 1 instead. See the description of the PROC_VER_INC attribute in Configuring Version Control on page 87.

Saving a Procedure as a Different Procedure

To save a procedure definition with a different procedure name:

1. Click anywhere in the procedure window of the procedure that you want to save.
2. Click **Procedure > Save As > Save As New Procedure**. The **Save As New Procedure** dialog is displayed.



3. Use the **Save in library** drop-down list to navigate to the library where you want to save the procedure.
4. Enter the **Procedure Name** that you want to save the procedure as. The name must be 8 characters or less and can contain any characters except ^ (caret) and , (comma).

You cannot use a procedure name that already exists (in any library).

5. (Optional) Enter a **Description** for the procedure, of up to 24 characters.

6. Enter a **Comment** describing the reason why the procedure is being saved with a new name. This comment will be visible from the audit trail for this version - see Displaying the Audit Trail for a Version on page 84.

The Comment field may be required, optional or hidden - see the description of the PROC_VER_COMMENT attribute in Configuring Version Control on page 87.

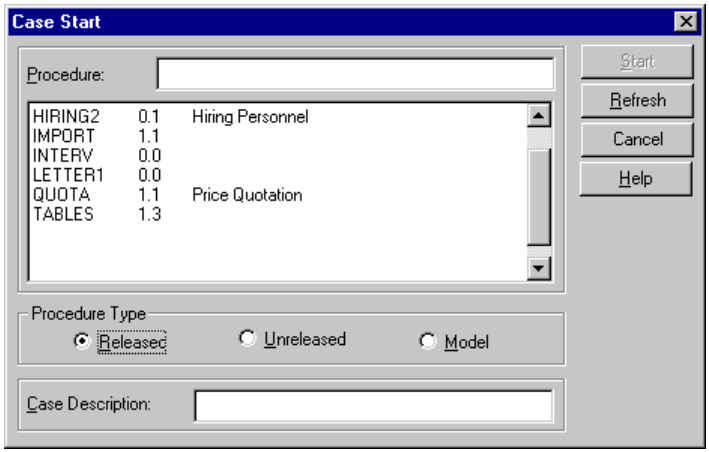
If you are saving:

- a *new* procedure, its status is set to Incomplete or Unreleased, as appropriate. Its version number is set to the system default value.
- an *existing* procedure, its status remains the same. Its version number is set to the system default value.

Testing Procedures

You can start a test case of a procedure from the Procedure Manager (or the Work Queue Manager). To do this:

1. Click  . The **Case Start** dialog is displayed.



2. In the **Procedure Type** section of the dialog, select either **Unreleased** or **Model**. All available procedures of that type are listed (along with the version number).
3. Select the procedure that you want to start a test case of.

If the procedure being tested calls a sub-procedure, the current precedence setting is used to determine which version of the sub-procedure is called.

Note that if an Unreleased version of a procedure calls a Released version of a sub-procedure, then the sub-procedure will be treated as Unreleased and all work items will be sent to test queues.

If a Released version of a main procedure calls an Unreleased version of a sub-procedure, then the sub-procedure work items will be sent to test queues.

4. Enter a **Case Description**. (Depending on the value of the procedure's Case Description at Start property, this may be required or optional, or the Case Description field may not be available.)

5. Click **OK**. The test case is started.

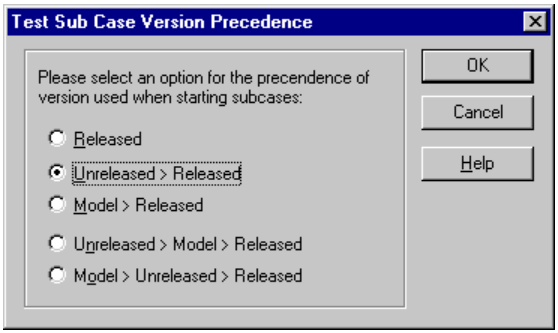
All work items for test cases are sent to appropriate test user queues or test group queues in your **Test Queues** folder in the Work Queue Manager.

Defining Which Version of a Sub-Procedure is Called

If a procedure calls a sub-procedure, the current precedence setting is used to determine which version of the sub-procedure is called.

To define the precedence setting:

1. In the Procedure Manager, click **Procedure Management > Precedence**. The **Test Sub Case Version Precedence** dialog is displayed.

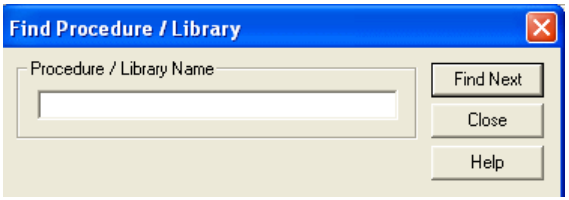


2. Select the order of precedence you want to use.

Searching for Libraries, Procedures and Shortcuts

You can use the **Find** feature to locate a specific library, procedure or shortcut when you don't know where to find it in the library structure. To do this:

1. In the Procedure Manager, click  . The **Find Procedure / Library** dialog is displayed.



2. Enter a text string to search for. The text string can contain the following wild-card characters:

Wildcard	Meaning
*	Zero or more characters of any value.
?	A single character of any value.

A "*" (asterisk) wild-card character is automatically assumed at the start and end of the search-string, and the text search is case insensitive.

3. Click **Find Next**. The Procedure Manager searches the library structure and, if it finds a procedure object whose name or description matches the search string, opens the library containing that object.

The object found is shown in the Procedure Details list but is not selected or highlighted in any way.

4. If this is not the object you are looking for, click **Find Next** again. The Procedure Manager searches for the next object that matches the search string.

   For example, a search for **cust** would find all of the following:

   — a library called **All customer related procedures**.

   — a procedure named **addcust**.

   — a procedure named **Newproc** with a description of **Add a customer record**.

5. When you have found the procedure object you are looking for, click **Close** to close the **Find** dialog.

Moving and Copying Procedures, Libraries and Shortcuts

You can move or copy procedures, libraries and shortcuts that you have access to. To do this:

1. In the Procedure Manager, open the library containing the procedure objects you want to move or copy.
2. Select the object(s) you want to copy or move in the Procedure Details list. If you want to select multiple objects, you can either:
   - hold down the CTRL key and click each object that you want to select.
   - hold down the SHIFT key and click the first and last objects in a range, to select all objects in that range.
3. If you want to:
   - **move** the selected object(s), drag the object(s) to the destination library in the Procedure Library list.
   - **copy** the selected object(s), hold down the CTRL key and drag the object(s) to the destination library in the Procedure Library list.

     You cannot copy a procedure, sub-procedure or sub-procedure parameter template, because these objects must have unique names on the system. Trying to do so will create a shortcut instead.
   - **create a shortcut** to a procedure, sub-procedure or sub-procedure parameter template, hold down the CTRL key and drag the object(s) to the destination library in the Procedure Library list.

     You cannot create a shortcut to a library. Trying to do so will copy the library instead.

This means that if you try and copy a selection that includes libraries and procedures, the library objects will be copied, and the procedures will have shortcuts created to them.

Deleting Procedures, Libraries and Shortcuts

You can delete procedures, libraries and shortcuts using the Procedure Manager.

Deleting a procedure deletes all versions of the procedure. If you only want to delete certain versions of a procedure, use the Version Control window instead - see Deleting a Version of a Procedure on page 82.

You can delete any procedure or library that meets the following criteria. You can delete any shortcut that references a library or procedure that meets these criteria:

- You have Edit access to the procedure or library. (iProcess Administrator users can delete any procedure object.)
- A procedure must not have any outstanding cases associated with it. If a procedure does have outstanding cases, you must purge them before you can delete the procedure. See "Purging Cases" in *TIBCO iProcess Workspace (Windows) Manager's Guide*, or "Purging Cases" in the *TIBCO iProcess swutil and swbatch Reference Guide*, for more information about how to purge cases.

This applies to both live cases and test cases.

- A library must be empty. If a library contains any procedure objects, you must move or delete them before you can delete the library.

To delete one or more procedures or libraries:

1. In the Procedure Manager, select the procedure object(s) you want to delete.
2. Click  . A dialog is displayed asking you to confirm that you want to delete the selected procedure object(s).
3. Click **Yes** to continue.

If you have selected a single object, the object is now deleted.

If you have selected multiple objects, the **Deleting Procedure Manager Objects** dialog is displayed.

The dialog shows what objects are being deleted. If you want to stop the operation at any time, click **Cancel**.

If the delete operation completes successfully, the dialog closes.

If the operation cannot delete an object for any reason, it pauses and displays the details in the **Errors** section of the dialog. You can then either click:

- **Ignore**, to ignore the indicated object and continue deleting other objects.

- **Ignore All**, to ignore the indicated object and any subsequent errors, and continue deleting other objects.
- **Cancel**, to stop the delete operation.

Click **Stop** when the operation has finished to close the dialog.

Printing Procedures

You can print a procedure definition from the iProcess Modeler.

To print the contents of the (currently active) procedure window, click  . The standard Windows **Print Document** dialog is displayed.

To print procedures, a printer with a minimum of 64MB of memory is required.

If you want to change your printer setup options (such as which printer, paper size and orientation to use):

1. Click **Procedure > Print Setup**. The standard Windows **Print Setup** dialog is displayed.
2. Choose the options you want and click **OK**.

There are also a number of print options that you can use to make your printout more informative and easier to read. Click **Options > Printing** for the following options:

- **Show Pagination**. This option prints page numbers on your procedure. There are also dashed lines on-screen to denote where the page breaks are.
- **Print Footers**. When you select this option, iProcess prints the procedure name and page number at the foot of each page.
- **Select Print Font**. This option allows you to change the font used for printing the procedure.
- **Decrease Print Size**. Selecting this option decreases the size of the printed output by 5% each time it is selected.
- **Increase Print Size**. Selecting the Increase Print Size option increases the size of the printed output by 5% each time it is selected.

Controlling Access to Procedures

By default:

- all iProcess users are allowed to start cases of a released procedure.
- any iProcess user with a MENUNAME attribute of MANAGER, PRODEF or ADMIN can administer cases of a procedure (using the **Case Administration** tool in Work Queue Manager).

If you wish, you can restrict the ability to start or administer cases to certain iProcess users and/or groups. You can restrict access to any combination of the following:

- specific named users or groups.
- specific named roles.
- by using a user attribute expression - for example, **sw_user:department="accounts"**.

iProcess Administrator users and the procedure owner always have full access rights irrespective of any entries in the **Procedure Access** dialogs.

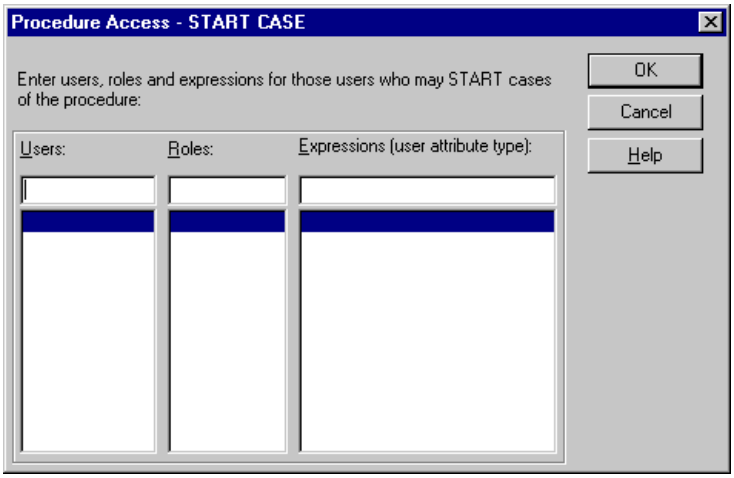
The following sections explain how to do this.

Controlling Who Can Start a Case of a Procedure

To restrict the ability to start cases of a procedure to certain users and/or groups:

1. Open the procedure in iProcess Modeler.

2. Click **Procedure > Access** > **Case Start**. The **Procedure Access - START CASE** dialog is displayed.



3. Add an entry to define who can start a case. To do this:
   a. Click in a blank row in the **Users**, **Roles** or **Expressions** column, as appropriate.
   b. Enter the user or group name, role name or user attribute expression in the input field at the top of the column.
4. Continue adding entries as required. Each column can hold a maximum of 10 entries.

If you want to edit or delete an existing entry, select the entry in the column then edit it as required in the input field.

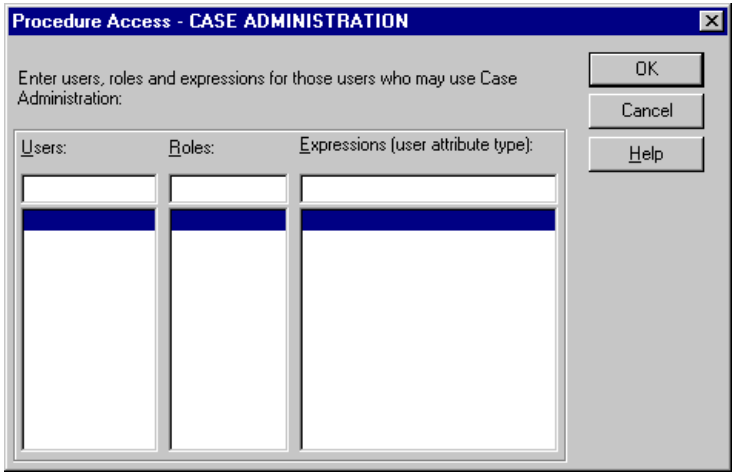
5. Click **OK** to save and exit the dialog.

Only the users specified in the dialog will be able to start cases of this procedure (when it is released).

Controlling Who Can Administer Cases of a Procedure

To restrict the ability to administer cases of a procedure to certain users and/or groups:

1. Open the procedure in iProcess Modeler.
2. Click **Procedure > Access** > **Case Admin**. The **Procedure Access - CASE ADMINISTRATION** dialog is displayed.



3. Add an entry to define who can administer a case. To do this:
   a. Click in a blank row in the **Users**, **Roles** or **Expressions** column, as appropriate.
   b. Enter the user or group name, role name or user attribute expression in the input field at the top of the column.
4. Continue adding entries as required. Each column can hold a maximum of 10 entries.

If you want to edit or delete an existing entry, select the entry in the column then edit it as required in the input field.

5. Click **OK** to save and exit the dialog.

Only the users specified in the dialog will be able to administer cases of this procedure (when it is released).

Chapter 3 Using Version Control

This chapter describes how to use version control to manage procedures.

Topics

About Versions and Version Control

When you create a procedure you create a **version** of that procedure, which has a unique **version number**. Whenever the procedure definition is changed, the version number may be incremented and a new version created.

Version control enhances your procedure management capabilities. You can:

- always see what a procedure looked like at a specific point in time, for auditing purposes (unless the procedure has been deleted).
- develop and test new versions of a procedure while a live version is still in use.
- revert to a previous version of a procedure if you need to.
- test new versions of procedures supplied as XFR files in parallel with the development and use of other versions of procedures on the system.
- allow existing cases to continue to be processed with old versions of the procedures if new procedures are introduced which are incompatible with the old cases.

Version Numbers

Every version of a procedure has a version number in the format:

major_version.minor_version

For example, **1.1**, **1.2**, **1.3**, **2.1** and so on.

Note that:

- Default *major_version* and *minor_version* values are used when a new procedure is saved. These values are defined by the DEF_MAJOR_VERS and DEF_MINOR_VERS process attributes.
- The PROC_VER_INC process attribute defines whether, in iProcess Modeler, a procedure's minor version number will be incremented whenever it is saved, or only when a new version is explicitly created.
- The PROC_VER_NUM_INSTANCES process attribute defines how many instances of a procedure are kept in the iProcess database. Each time you perform an edit on a procedure version a new instance of the procedure version is saved.
- Process attributes are set on iProcess Engine. See "Administering Process Attributes" in *TIBCO iProcess Engine Administrator's Guide* for more information about these attributes and how to set them.

Release Status

A particular version of a procedure always has an associated **release status**, which determines how that version can be used.

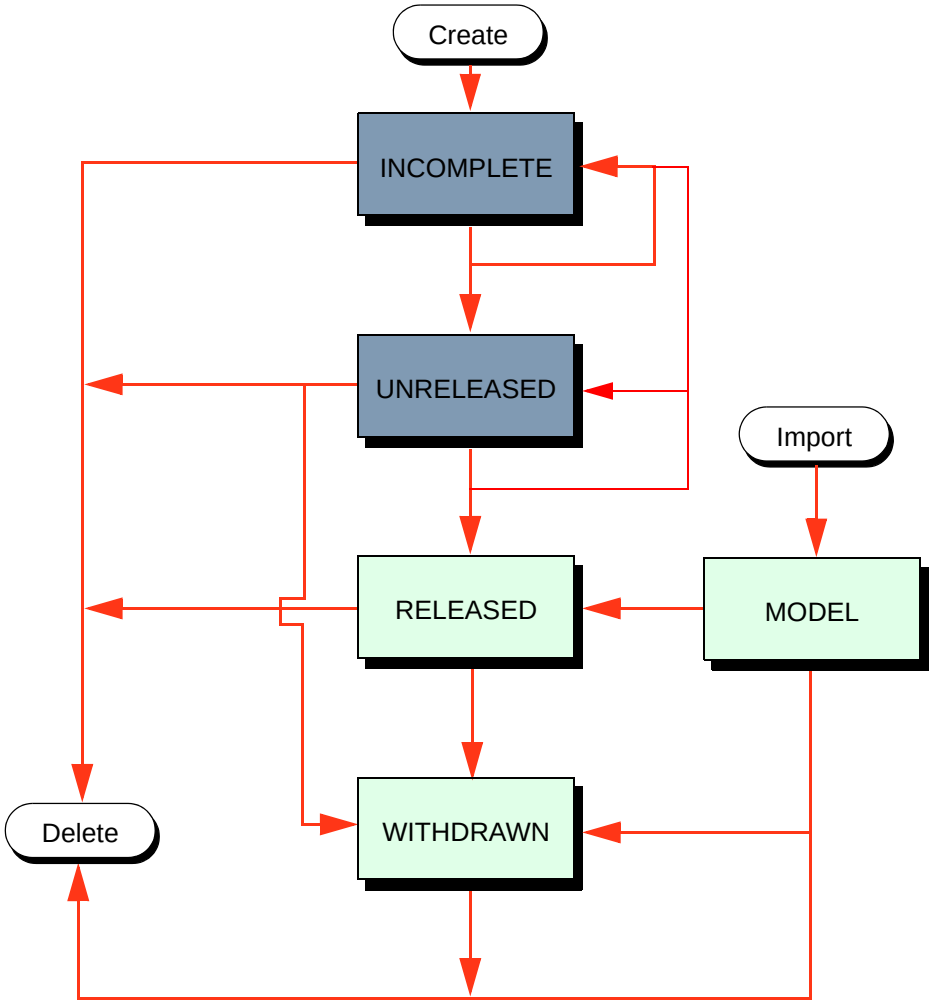
In earlier versions of iProcess, procedures themselves had a release status.

The following table lists the different status types, their meaning, and their intended usage in the life-cycle of a procedure.

Release Status	Meaning	Intended Use
Incomplete	A procedure that cannot be run yet because it has required information missing - for example, a step without an addressee, or a step without a connection from a previous step.	Development
Unreleased	A procedure that is complete and can be run. However, any cases run in the test queue of the user that starts them.	Development/Testing
Released	A procedure that can be used to process live work. Cases can be run and are processed to users' queues.	Operational
Model	A Released procedure that is subsequently imported. However, any cases run in the test queue of the user that starts them. Model status allows a new version to be imported without overwriting the existing Released or Unreleased version, which allows you to test a new version on a target system before adopting it.	Acceptance Testing
Withdrawn	A procedure that is no longer required for development or operational use. No further cases of the procedure can be started. Existing cases will run to completion.	Phase-out

When a procedure is no longer required on the system it can then be deleted. A procedure can only be deleted if it has no live cases associated with it.

The following diagram shows how a version can change status throughout its life cycle.



Edit state; if the procedure definition is edited it can be saved as the same version.

Read-only state; if the procedure definition is edited it must be saved as a new version (or with a new name).

How and When Version Numbers and Release Status Can Change

There can be many different versions of a procedure on the system. iProcess keeps every version of a procedure unless you explicitly delete it. However, at any one time a procedure can only have:

- **one** Incomplete or Unreleased (i.e. development) version.
- **one** Released (i.e. operational) version.
- **one** Model (i.e. test) version.
- **any number** of Withdrawn versions.

The release status (of a particular version) can be changed either directly or indirectly:

- *Directly.* You can perform specific tasks to change a version's release status - for example, releasing, withdrawing or deleting the version.
- *Indirectly.* Performing certain tasks will affect the release status of either:
  - — the version you are working on - for example, when you edit an Incomplete version and complete all the information required for it to run, its status will change to Unreleased when you save it.
  - — another version - for example, an existing Released version is automatically withdrawn when another version is released.

Similarly, the version number (of a particular procedure definition) can be changed either directly or indirectly:

- *Directly.* You can perform specific tasks to change a procedure's version number - for example, creating a new version from an existing version.
- *Indirectly.* Performing certain tasks will affect the version number of the version you are working on. For example, if you edit a Released, Model or Withdrawn procedure you cannot save your changes to the existing version. You must save the procedure using a new version number (or a new name).

The Default Procedure Version

The Procedure Manager displays and works with the default version of a procedure. The default version is the first version that matches the following criteria:

1. the version with an Incomplete or Unreleased status, or
2. the version with a Model or Released status with the latest modification date, or
3. the highest version number with Withdrawn status.

If you want to perform an operation on a specific version of a procedure you should use the Version Control window instead.

How to Use Version Control

The following sections provide some examples of the different ways that you can use iProcess's version control facilities to handle changes to a process.

Scenario 1 - Changing the Procedure Name

This scenario most closely resembles the use of procedure status in pre-Version 10 software.

A company develops and uses simple procedures for internal administration, and has no complex versioning requirements. The process of hiring staff is to be turned into a procedure called HIRING:

1. iProcess Modeler creates a draft of the new HIRING procedure (Version 0.0, INCOMPLETE).
2. Revision and testing of the draft (Version 0.0, UNRELEASED) takes place.
3. When testing has completed, the procedure is released (Version 0.0, RELEASED) for operational use within the company.
4. After a period of time, the hiring process needs to be amended (for example, to use an external agency to handle initial screening of applicants). iProcess Modeler creates a new procedure called HIRING2 (Version 0.0, UNRELEASED) and makes the necessary changes.
5. When testing has completed, the HIRING2 procedure is released (Version 0.0, RELEASED) for operational use. At the same time, the original HIRING procedure must be withdrawn (Version 0.0, WITHDRAWN), and all staff advised to use the new HIRING2 procedure.
6. Finally, all cases of the HIRING procedure are purged and the HIRING procedure is deleted.

Although this scenario is very easy to implement, it suffers from a number of limitations:

- The switchover from HIRING to HIRING2 requires manual handling, and so is susceptible to human error. For example:
  - HIRING must be manually withdrawn.
  - If HIRING still has outstanding cases when HIRING2 is released, it will be necessary to leave both procedures live for a time until all HIRING cases have completed.
  - All staff must be notified (and then remember) to use HIRING2 for new cases.
- It is not suitable for complex implementations of iProcess. For example, where large numbers of procedures are in use and sub-procedures are used extensively. Changing the name of a sub-procedure can then require extensive changes throughout the organization's entire procedure structure.

TIBCO therefore do not recommend the use of this scenario. See scenario 2 instead.

Scenario 2 - Introducing a New Version

As in scenario 1, a company develops and uses simple procedures for internal administration. The process of hiring staff is to be turned into a procedure called HIRING.

Changes to the hiring process are to be handled by creating new versions of the HIRING procedure:

1. iProcess Modeler creates a draft of the new HIRING procedure (Version 0.0, INCOMPLETE).
2. Revision and testing of the draft (Version 0.0, UNRELEASED) takes place.
3. When testing has completed, the procedure is released (Version 0.0, RELEASED) for operational use within the company.
4. After a period of time, the hiring process is amended (for example, to use an external agency to handle initial screening of applicants). iProcess Modeler creates a new version of the procedure (Version 0.1, UNRELEASED) and makes the necessary changes.
5. After revision and testing of the new version, it is released (Version 0.1, RELEASED) for operational use, superseding the original version, which is automatically withdrawn (Version 0.0, WITHDRAWN).

This scenario offers a number of advantages over scenario 1:

- The switchover from Version 0.0 to Version 0.1 is handled automatically, so the possibility for human error is removed:
  - Version 0.0 is automatically withdrawn when Version 0.1 is released.
  - Outstanding cases for Version 0.0 can progress to completion using either Version 0.0 or Version 0.1. (This is an option you can select when you release Version 0.1.)
  - Staff do not need to switch to using a new version. They can simply continue to start cases of HIRING.
- Because no change of procedure name is involved, the introduction of a change in a complex iProcess implementation (where large numbers of procedures are in use and sub-procedures are used extensively) is much easier.

Scenario 3 - Reverting to an Earlier Version

Continuing on from scenario 2, the company has introduced and been using Version 0.1 of the HIRING process. After a period of time, it decides that the use of an external agency to handle initial screening of applicants no longer fits its requirements, and decides to revert to handling the recruitment process in-house again.

Further modifications to the HIRING process are therefore required. However, because of the nature of the changes, the original Version 0.0 procedure offers a better base on which to make the changes than the current Version 0.1:

1. iProcess Modeler creates a new version of the procedure (Version 0.2, UNRELEASED) *based on the original version* (Version 0.0, WITHDRAWN) and makes the necessary changes.
2. After revision and testing the new version is released (Version 0.2, RELEASED) for operational use. The current operational version is automatically withdrawn (Version 0.1, WITHDRAWN).

This scenario illustrates a significant version control benefit - that at any point, you can revert back to any earlier version of a procedure, either for operational use or as a basis for further development. This gives you much greater flexibility in matching your procedure development methodology to your business process development.

Scenario 4 - Controlling Procedure Use in a Multi-site Environment

A company develops procedures on its development system at head office, then distributes them (as .XFR files) to its five regional offices for operational use. Invoicing is handled by an INVOICE procedure, which is currently operational at Version 2.0 (RELEASED). The company now wants to update this procedure and replace it with a new version:

1. The Development department creates a new version of the INVOICE procedure (Version 2.1, UNRELEASED) and makes the necessary changes.
2. After revision and testing of the new version, it is released (Version 2.1, RELEASED), then exported and distributed to the regional offices.
3. Each regional office imports the INVOICE procedure as a Model version (Version 2.1, MODEL). They can then acceptance test the new version in parallel with the existing operational version (Version 2.0, RELEASED). Cases started against the Version 2.1 procedure will be sent to test queues.
4. When acceptance testing has completed, the new version is released (Version 2.1, RELEASED), which automatically withdraws the previous version (Version 2.0, WITHDRAWN).

This scenario illustrates the use of Model versions to control procedure use in a distributed environment. Unlike an Unreleased version, a Model version cannot be edited - it can only be released or withdrawn.

Creating a New Version of a Procedure

You can create a new version of a procedure based on any existing version of a procedure, by doing either of the following:

- Create New Version from the Version Control window.
- Save As New Version from iProcess Modeler.

Note that:

- You can create a new version based on *any* existing version - you do not have to use the latest available version. However, the *new* version number will be assigned automatically by incrementing the *latest* version number.

  For example, if you have versions 0.1, 0.2 and 0.3 of a procedure, you could decide to create a new version based upon version 0.2. The new version would be version 0.4.
- The new version is identical to the version that it is based on, except that it has a new version number, and is automatically created as Unreleased or Incomplete (as appropriate).
- If an Incomplete or Unreleased version already exists, that version will be withdrawn when you create the new version (because you can only have one Unreleased or Incomplete Version of a procedure at a time).
- It is up to you to develop and control your own method of using the version numbering scheme to suit your requirements. For example, you may want to allow reversion to an earlier version as described above. Alternatively, you may want to restrict new versions to be created only from the latest available version.

See How to Use Version Control on page 67 for some examples.

Creating a New Version From the Version Control Window

To create a new version of a procedure:

1. In the Version Control window, select the version that you want to base the new version on.
2. Click  . The **Create New Version for *Procedure* (based on v*n.n*)** dialog is displayed.
3. In the **Version Description** field, enter a description (of up to 128 characters) for the new version. This can be any text that you want, for example,

describing why the version is being created and/or what version it is being based on.

4. If you want to increment the major version number for the new version, select the **Create new major version** check box. If you do not select this check box, the minor version number will be incremented.

   For example, if you are creating a new version based on version 1.3, the new version number will be v2.0 if you select this check box, or 1.4 if you do not.

5. If an Unreleased or Incomplete version of the procedure already exists, the dialog warns you that saving this version will automatically withdraw the existing Unreleased or Incomplete version (because there can only be one Incomplete or Unreleased version in existence at a time).

   If you wish to continue, enter a **Withdraw Description** (of up to 128 characters) describing why the indicated version is being withdrawn.

The new version is saved as an Unreleased or Incomplete version (as appropriate).

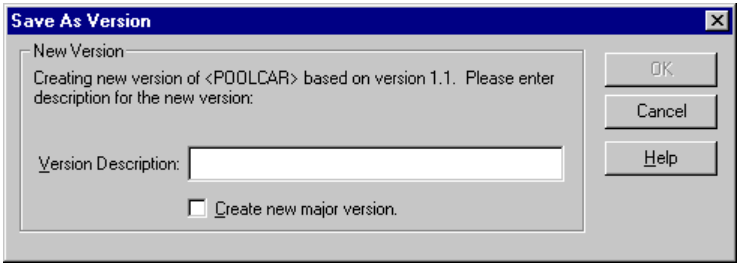
Saving a Version of a Procedure as a New Version

Whenever you edit a procedure, you can if you wish save your changes as a new version.

If you edit a Released, Model or Withdrawn version of a procedure, you *must* save your changes as either a new version or a new procedure. You cannot save changes to the same version.

To save a version of a procedure as a new version:

1. In iProcess Modeler, click **Procedure > Save As > Save As New Version**. The **Save as Version** dialog is displayed.



2. In the **Version Description** field, enter a description (of up to 128 characters) for the new version. This can be any text that you want, for example,

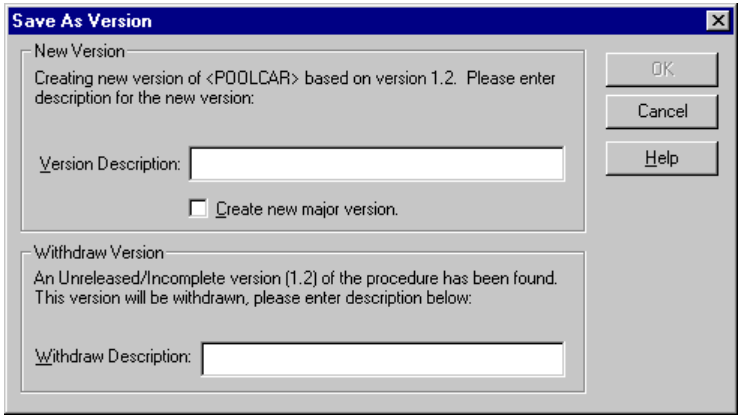
describing why the version is being created and/or what version it is being based on.

3. If you want to increment the major version number for the new version, select the **Create new major version** check box. If you do not select this check box, the minor version number will be incremented.

   For example, if you are creating a new version based on version 1.3, the new version number will be v2.0 if you select this check box, or 1.4 if you do not.

4. If an Unreleased or Incomplete version of the procedure already exists, the dialog warns you that saving this version will automatically withdraw the existing Unreleased or Incomplete version (because there can only be one Incomplete or Unreleased version in existence at a time).

   If you wish to continue, enter a **Withdraw Description** (of up to 128 characters) describing why the indicated version is being withdrawn.



The new version is saved as an Unreleased or Incomplete version (as appropriate).

Releasing a Version of a Procedure

You can release an Unreleased or Model version of a procedure, using either:

- the Version Control window, to release a specific version of a single procedure.

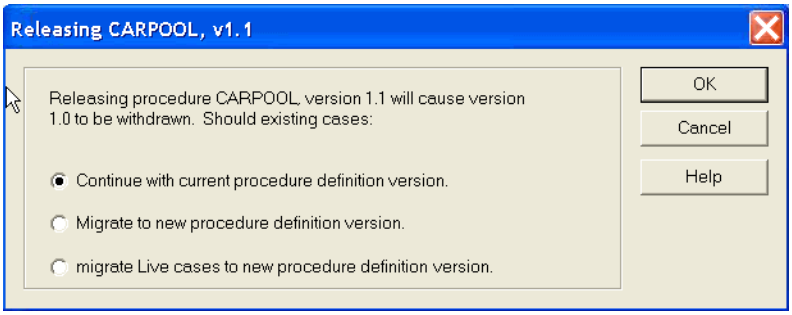
You must use this method if you want to replace an existing Released version with an earlier version.

- the Procedure Manager, to release the Unreleased or Model version of either a single procedure, or of multiple procedures and/or libraries.

Releasing a Specific Version of a Procedure

To release a specific version of a procedure:

1. In the Version Control window, select the version you want to release. You must select a Model or Unreleased version.
2. Click  . The **Release *Procedure*** dialog is displayed.
3. Enter a **Release Comment** describing why the procedure is being released.
4. If the dialog indicates that an existing version will also be withdrawn, you can enter a **Withdraw Comment** as well.
5. Click **OK**.
6. If the **Releasing** dialog is displayed, click the appropriate radio button to specify whether existing cases of the procedure should:
   - — Continue to use the version of the procedure that is being withdrawn.
   - — Migrate to using the version of the procedure that is being released.
   - — Only migrate live cases migrate to the version of the procedure that is being released.



Releasing the Unreleased or Model Version of One or More Procedures

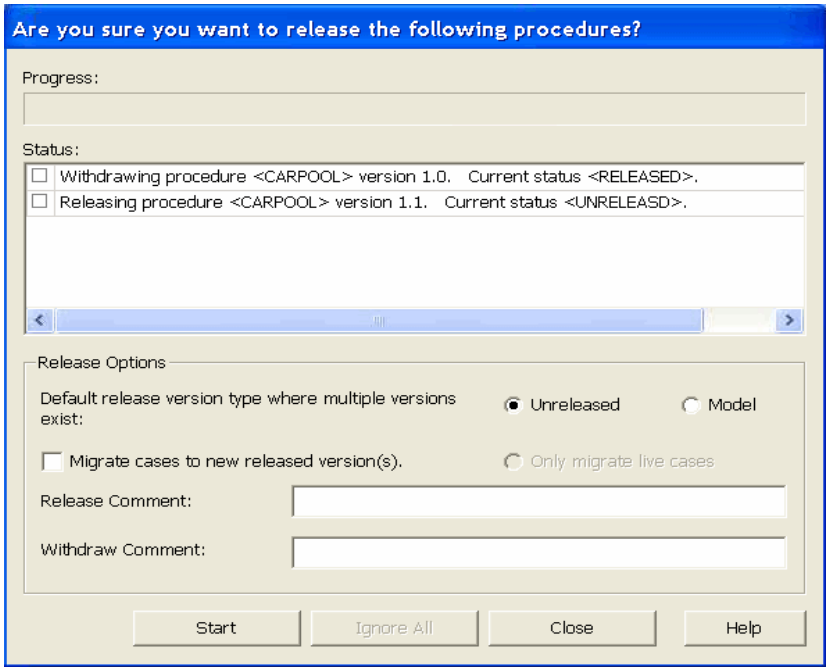
In Procedure Manager, you can select one or more procedures and/or libraries, and release the Unreleased or Model version of each one.

When you select a library, all procedures within that library, and within any child libraries, will be released.

To release the latest versions of one or more procedures:

1. In the Procedure Manager, select the procedures and/or libraries you want to release.
2. Click  . A dialog is displayed showing the procedures and versions that will be released and withdrawn if you continue.

   If a selected procedure cannot be released for any reason (for example, because it is not a Model or Unreleased version), the message "skipping procedure *name*" is displayed in the list, along with the reason.



3. In the Release Options section of the dialog, specify the options you want to use for the selected procedures, as follows:

   — By default, if an Unreleased version and a Model version of a procedure exists, the Unreleased version will be released. If you want to release Model versions instead, select the **Model** radio button. The list is updated to show the Model versions of those procedures instead. (If a procedure only has an Unreleased version, it is not affected by changing this setting - the Unreleased version is still listed.)

   — By default, if a Released version of a procedure exists, that version will be withdrawn, and any existing cases of that procedure version will continue to use the version that is being withdrawn. If you want to migrate those cases to use the version that is being released instead, check the **Migrate cases to new released version(s)** checkbox. If you want to migrate only live cases to use the version that is being released, check the **Migrate cases to**

new released version(s) checkbox, and then select the **Only Migrate Live Cases** radio button.

— Enter a **Release Comment** describing why the procedures are being released. This comment will be applied to all the listed procedures that are being released.

— Enter a **Withdraw Comment** describing why the procedures are being withdrawn. This comment will be applied to all the listed procedures that are being withdrawn.

These options will be applied to **all** the listed procedures. If you want to specify different options for different procedures, you must release those procedures from the Version Control window instead.

4. Click **Start**. The progress meter shows the progress of the operation, and a tick appears next to each procedure as it is released or withdrawn.
5. Click **Close** to close the dialog when all the procedures have been released or withdrawn.

Withdrawing a Version of a Procedure

You can withdraw:

- a specific version (Unreleased, Released or Model) of a procedure, using the Version Control window.
- the latest complete version (Released or Model) of either a single procedure, or of multiple procedures and/or libraries, using the Procedure Manager.

Withdrawing a Specific Version of a Procedure

To withdraw a specific version of a procedure:

1. In the Version Control window, select the version you want to withdraw. You must select a Released, Model or Unreleased version.
2. Click  . The **Withdraw *Procedure*** dialog is displayed.
3. Enter a **Withdraw Comment** describing why the procedure is being withdrawn.

If you are withdrawing the Released version the dialog informs you that this action will leave no current procedure released.

4. Click **OK**.

Withdrawing the Latest Version of One or More Procedures

In Procedure Manager, you can select one or more procedures and/or libraries, and withdraw the latest complete (Released or Model) version of each one.

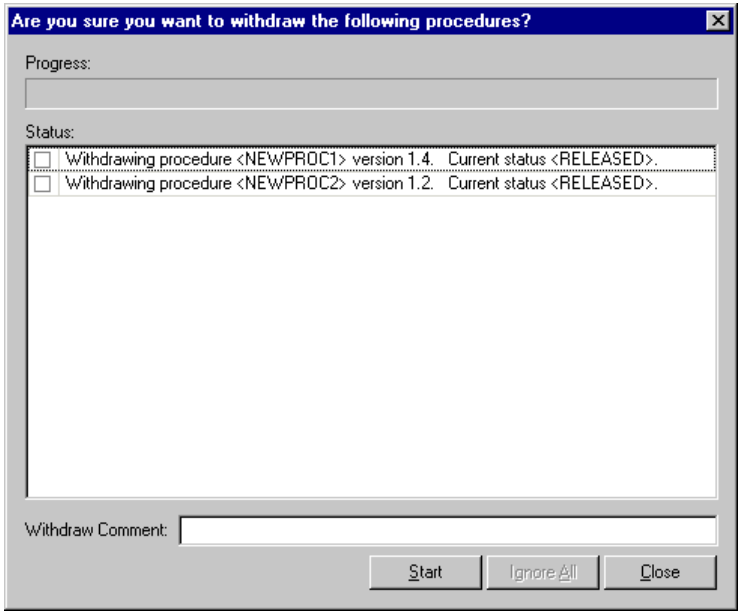
When you select a library, all procedures within that library, and within any child libraries, will be withdrawn.

To withdraw the latest versions of one or more procedures:

1. In the Procedure Manager, select the procedures and/or libraries you want to withdraw.
2. Click  . A dialog is displayed showing the procedures and versions that will be withdrawn if you continue.

   The latest Released or Model version of a procedure will be withdrawn.

If a selected procedure does not have a Released or Model version, the message "skipping procedure *name*" is displayed in the list, along with the reason.



3. Enter a **Withdraw Comment** describing why the procedures are being withdrawn. This comment will be applied to all the listed procedures that are being withdrawn.
4. Click **Start**. The progress meter shows the progress of the operation, and a tick appears next to each procedure as it is withdrawn.
5. Click **Close** to close the dialog when all the procedures have been withdrawn.

Changing the Procedure Type of a Version

You can change a version's procedure type from main procedure to sub-procedure, or vice-versa.

Before changing a version's procedure type, you should note the following:

- Other versions of the procedure are not affected by the change. For example, if HIRING is a main procedure and you change Version 1.4 to a sub-procedure, Versions 1.1, 1.2 and 1.3 will still be main procedures.
- If you are changing a sub-procedure to a main procedure, make sure there are no other procedures (or sub-procedures) that still call the sub-procedure, as they will no longer work properly.
- If you are changing a main procedure to a sub-procedure you will not be able to start cases of it directly - for example, from the **Case Start** dialog, or from the *SWDIR***bin****swutil** utility.

  Also, additional key words, or system values, are automatically created that can be used as fields in the sub-procedure's forms. See "Using Delimiters and Key Words" in *TIBCO iProcess Modeler Basic Design* for more information.

Do not change a version's procedure type unless you are sure that doing so will not cause other procedures or existing cases on your system to go wrong.

To change a main procedure to a sub-procedure:

1. In iProcess Modeler, click **Procedure > Convert Procedure > To Sub-Procedure**. A dialog is displayed asking if you wish to continue.
2. Click **Yes**. The version is now a sub-procedure.

To change a sub-procedure to a main procedure:

1. In iProcess Modeler, click **Procedure > Convert Procedure > To Main Procedure**. A dialog is displayed asking if you wish to continue.
2. Click **Yes**. The version is now a main procedure.

Deleting a Version of a Procedure

You can delete a specific version of a procedure provided that it is not a Released or Withdrawn version that still has outstanding cases associated with it.

Deleting a Single Version of a Procedure

To delete a single version of a procedure:

1. In the Version Control window, select the version that you want to delete.
2. Click  .

The version is deleted.

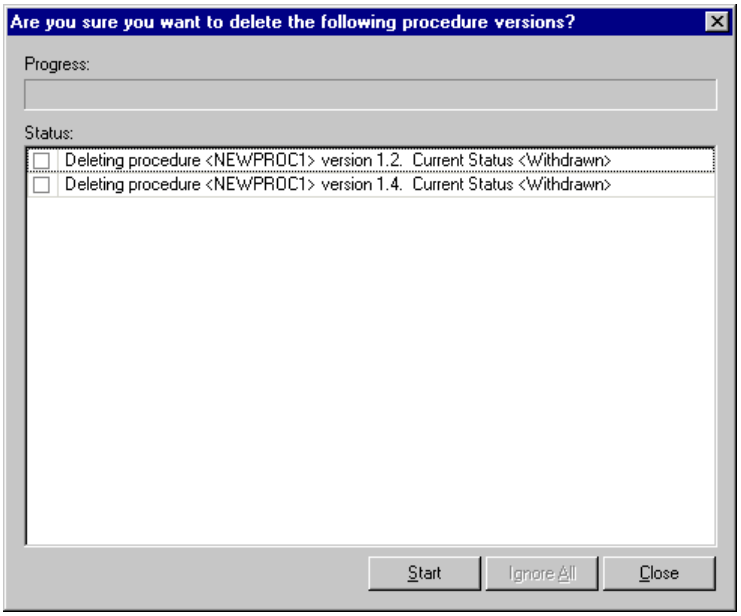
If you try to delete a Released or Withdrawn version that still has outstanding cases, the message **Failed to delete the selected version** is displayed. You will need to manually purge all cases of the procedure that are associated with that version before you can delete it.

For more information about how to purge cases, see either "Purging Cases" in *TIBCO iProcess Workspace (Windows) Manager's Guide*, or "Purging Cases" in *TIBCO iProcess swutil and swbatch Reference Guide*.

Deleting Multiple Versions of a Procedure

To delete multiple versions of a procedure:

1. In the Version Control window, select the versions that you want to delete.
2. Click  . A dialog is displayed showing the versions that will be deleted if you continue.



If a selected version is a Released or Withdrawn version with outstanding cases, the message "skipping version *number*" is displayed in the list, along with the reason. You will need to manually purge all cases associated with that version before you can delete it.

For more information about how to purge cases, see either "Purging Cases" in *TIBCO iProcess Workspace (Windows) Manager's Guide*, or "Purging Cases" in *TIBCO iProcess swutil and swbatch Reference Guide*.

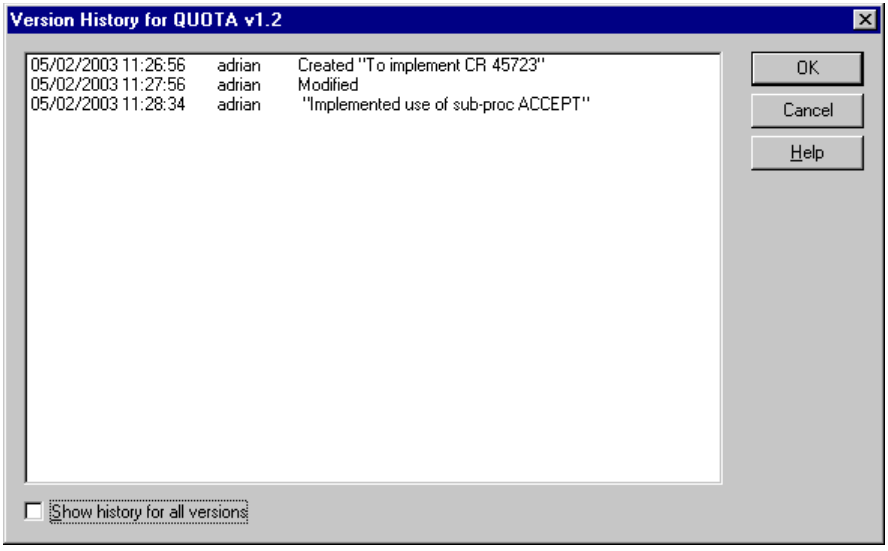
3. Click **Start**. The progress meter shows the progress of the operation, and a tick appears next to each version as it is deleted.
4. Click **Close** to close the dialog when all the versions have been deleted.

If you delete **all** versions of a procedure, the procedure is deleted and the Version Control window for the procedure is closed.

Displaying the Audit Trail for a Version

To display the audit trail of changes made to a specific version or procedure:

1. In the Version Control window, select the version you want to view the audit trail for. If you want to view the audit trail for the entire procedure, select any version.
2. Click  . The **Version History** dialog for the selected version is displayed, showing the changes that have been made to this version.



The following changes are logged:

— when the version was created or imported.

— when the version was last updated (without any change in status).

— when the status of the version was last changed.

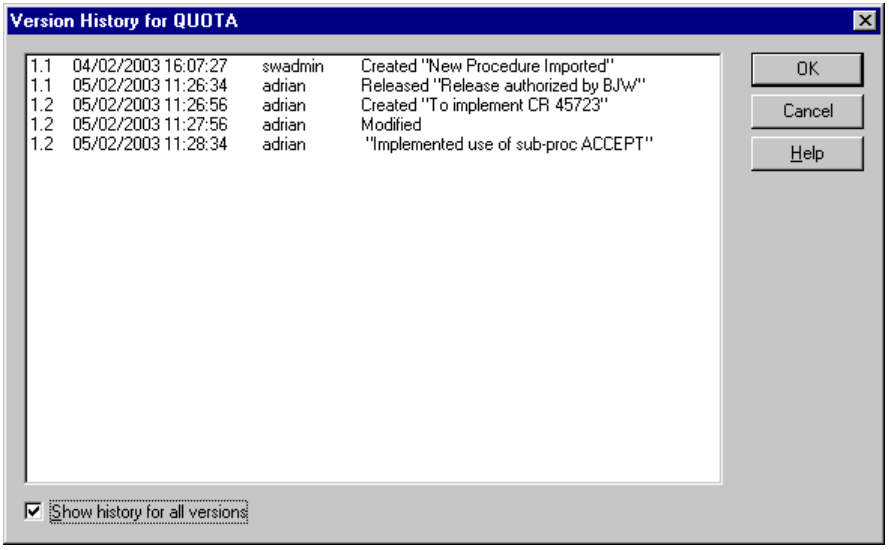
The following information is shown for each change:

— the date and time that the change was made.

— the name of the user who made the change.

— the description of the change and the comment (if any) made by the user about the change.

3. If you want to expand the version history to show the whole procedure, select the **Show history for all versions** check box. The report expands to show the changes made to all versions of the procedure. An additional column shows the version that each change relates to.



Setting and Viewing Properties for a Specific Version

A procedure also has properties that apply to ALL versions of the procedure. You can access those properties from the Procedure Manager - see Working with Properties of Procedures on page 107.

To view and/or set the properties for a specific version of a procedure:

1. In the Version Control w[illegible]ow, select the version you want to view/set properties for and click  . The **Procedure Version Properties** dialog is displayed. This shows the following information about the version:

Field	Description
Procedure	The procedure name
Version	The version number
Status	The release status
Version Label	Descriptive comment

2. If you want to change the descriptive comment for this version, edit the **Version Label** field and click **Change Label**.
3. Click **OK**.

The new comment is shown in the **Label** column.

All previously entered comments are shown in the version's audit trail. See Displaying the Audit Trail for a Version on page 83.

Configuring Version Control

The following table describes the process attributes that can be set on iProcess Engine to define how procedure versions are handled.

See "Administering Process Attributes" in the *TIBCO iProcess Engine Administrator's Guide* for more information about these attributes and how to set them.

Attribute	Description
DEF_MAJOR_VERS	Defines the default major version number that iProcess Modeler will use when a new procedure is saved.
DEF_MINOR_VERS	Defines the default minor version number that iProcess Modeler will use when a new procedure is saved.
PROC_VER_COMMENT	Defines whether or not, in iProcess Modeler, a user has to enter a comment whenever they save a procedure. The comment field can be hidden, optional or required. **Note: You may want the comment field to be hidden if it becomes intrusive when you are working on a large procedure and doing frequent saves. If the comment field is hidden, you can still add a comment to the version when required from the Version Control window - see Setting and Viewing Properties for a Specific Version on page 85.**
PROC_VER_INC	Defines whether, in iProcess Modeler, a procedure's minor version number will be incremented: • whenever it is saved. • only when a new version is explicitly created.

Chapter 4 Viewing Your Procedure in a Browser

This chapter describes how to save your procedure as HTML and view it in a browser. This makes it easier for you to view your procedure as a whole and to browse through it using standard navigation techniques.

Topics

Overview

Saving your procedure as HTML enables you to view your procedure in a web browser.

TIBCO have certified TIBCO iProcess Workspace (Windows) against Internet Explorer 6.0. This means that Internet Explorer 6.0 has been successfully tested by TIBCO for compatibility with this version of TIBCO iProcess Workspace (Windows).

Viewing the procedure in a browser means that the procedure is displayed in sections that are expandible and collapsible. This means you can navigate between various procedure entities, for example, procedures, steps and fields. There is also an option to display the procedure in a printer-friendly form.

The main benefits of viewing a procedure in a browser are:

- You can view the procedure in an industry standard format. This means that the procedure can be viewed in a standard application (Microsoft Explorer). The benefit of this is that the procedure can be easily distributed, for example, on an internet or intranet site, see Viewing Your Procedure as HTML, page 91.
- You can view more than one procedure, sub-procedure or sub-procedure parameter template in a single document, see Saving Your Procedure as HTML, page 91.
- You can use hyperlinks and bookmarks to jump between various procedure entities, see Navigating Around Your Procedure, page 93.
- You can view the procedure in a printer-friendly form, see Viewing a Printer-Friendly Version of Your Procedure, page 101.
- You can use CSS stylesheets to modify how the procedure is displayed in the browser., see Customizing Your HTML Document on page 105.

You cannot edit your procedure from the browser window. To edit your procedure you must use TIBCO iProcess Modeler.

Viewing Your Procedure as HTML

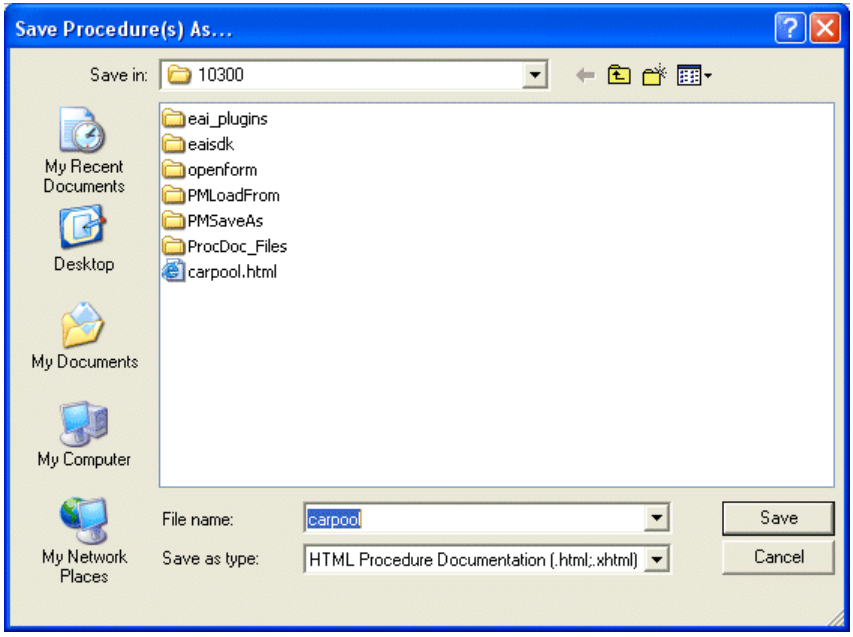
This section describes how to view an iProcess procedure as HTML. To view your procedure as HTML, you need to:

- save your procedure as HTML
- open the HTML document in a browser window.

Saving Your Procedure as HTML

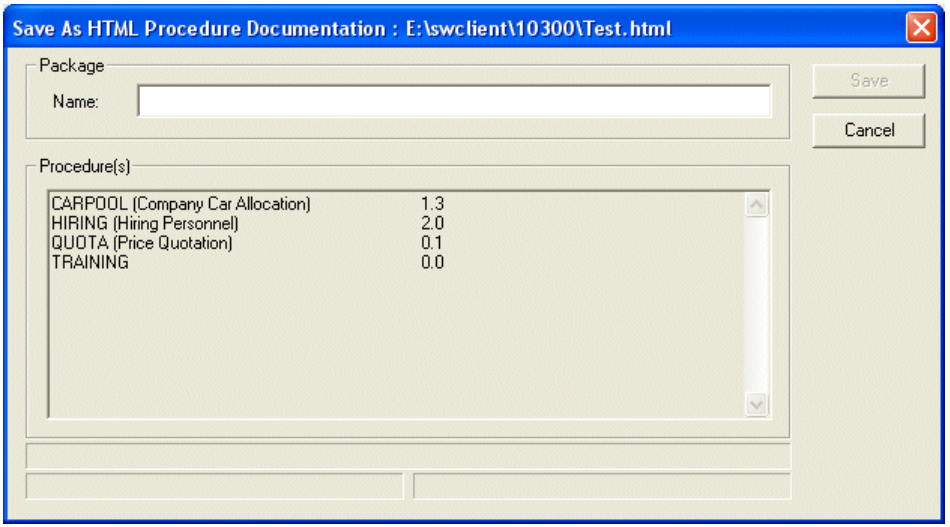
To save a procedure as HTML, do the following:

1. From the **Procedure Manager**, browse to the location of the procedure(s) that you want to save as HTML.
2. Select the procedure(s) that you want to save as HTML.
3. Click **Procedure Management** > **Save as ...**. The **Save Procedure(s) As** dialog is displayed.



4. Navigate to the directory where you want to save the HTML file.
5. In the **Save as type:** box, select **HTML Procedure Documentation (.html; .xhtml)**.

6. In the **Filename:** box, type in the filename for the HTML file that you are creating.
7. Click **Save**. The **Save As HTML Procedure Documentation:** *pathname* dialog is displayed where *pathname* is the path to the HTML file you have selected.



8. In the **Name** box, enter the description for the procedure package that you want to be displayed in the browser window and click **Save**.

   A progress dialog displays the progress of the **Save** instruction.

Viewing Your Procedure in a Browser

Once you have saved your procedure as HTML, open the HTML document in a browser window.

Navigating Around Your Procedure

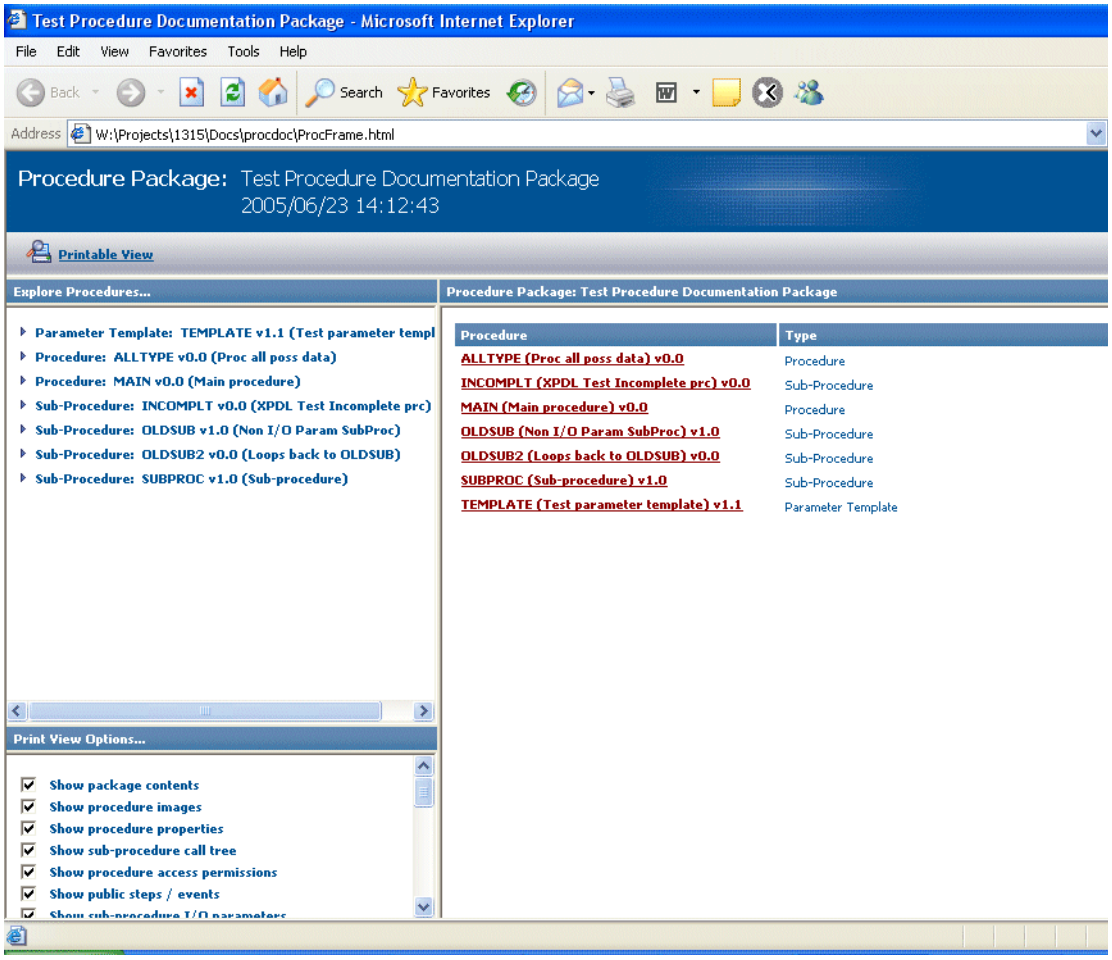
This section describes how to navigate around a procedure and select objects to view.

See:

- About the Browser Window
- Selecting a Procedure to View
- Selecting an Object to View
- Using the History to View Objects

About the Browser Window

When you open your HTML document, the browser window is divided into three panes:

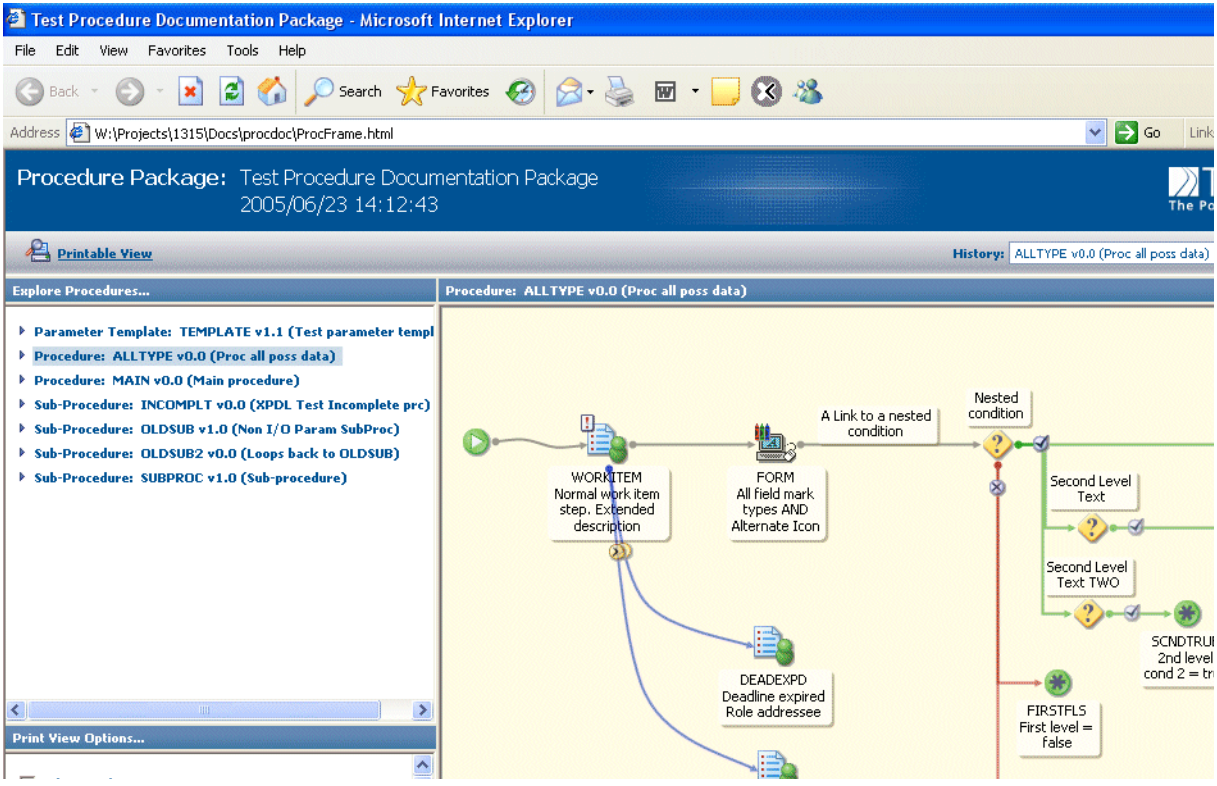


- the **Explore Procedures...** pane displays the procedures that you can view.

- the ***Objecttype*: *objectname*** pane displays information about the currently selected object. The type of information displayed depends on the object you have selected where:
  - — *Objecttype* is the type of object, for example, procedure, sub-procedure, sub-procedure parameter template, step or field.
  - — *objectname* is the name of the currently selected procedure object.
- the **Print View Options...** pane enables you to select which procedure properties you want to display when viewing a printer-friendly version of your procedure. See Viewing a Printer-Friendly Version of Your Procedure, page 101 for more information.

Selecting a Procedure to View

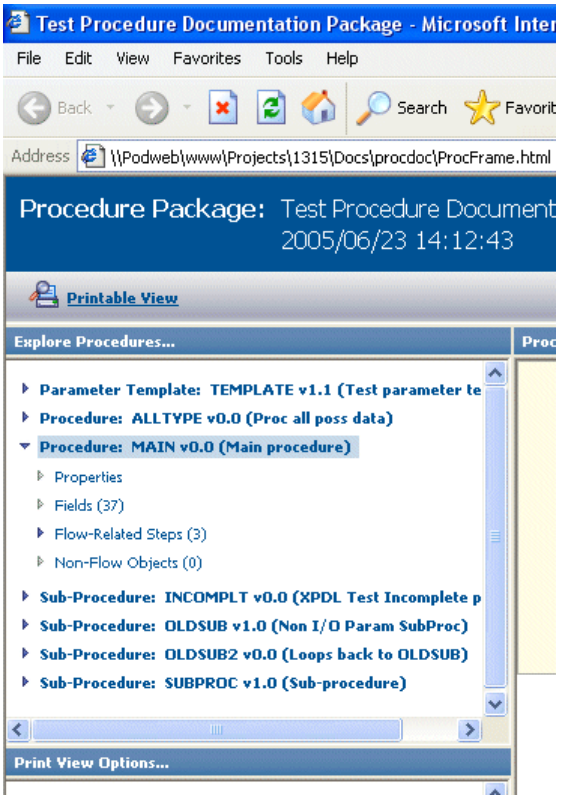
To select a procedure to view, from the **Explore Procedures...** pane, click on the procedure you want to view. The procedure definition is displayed in the right hand side of the window.



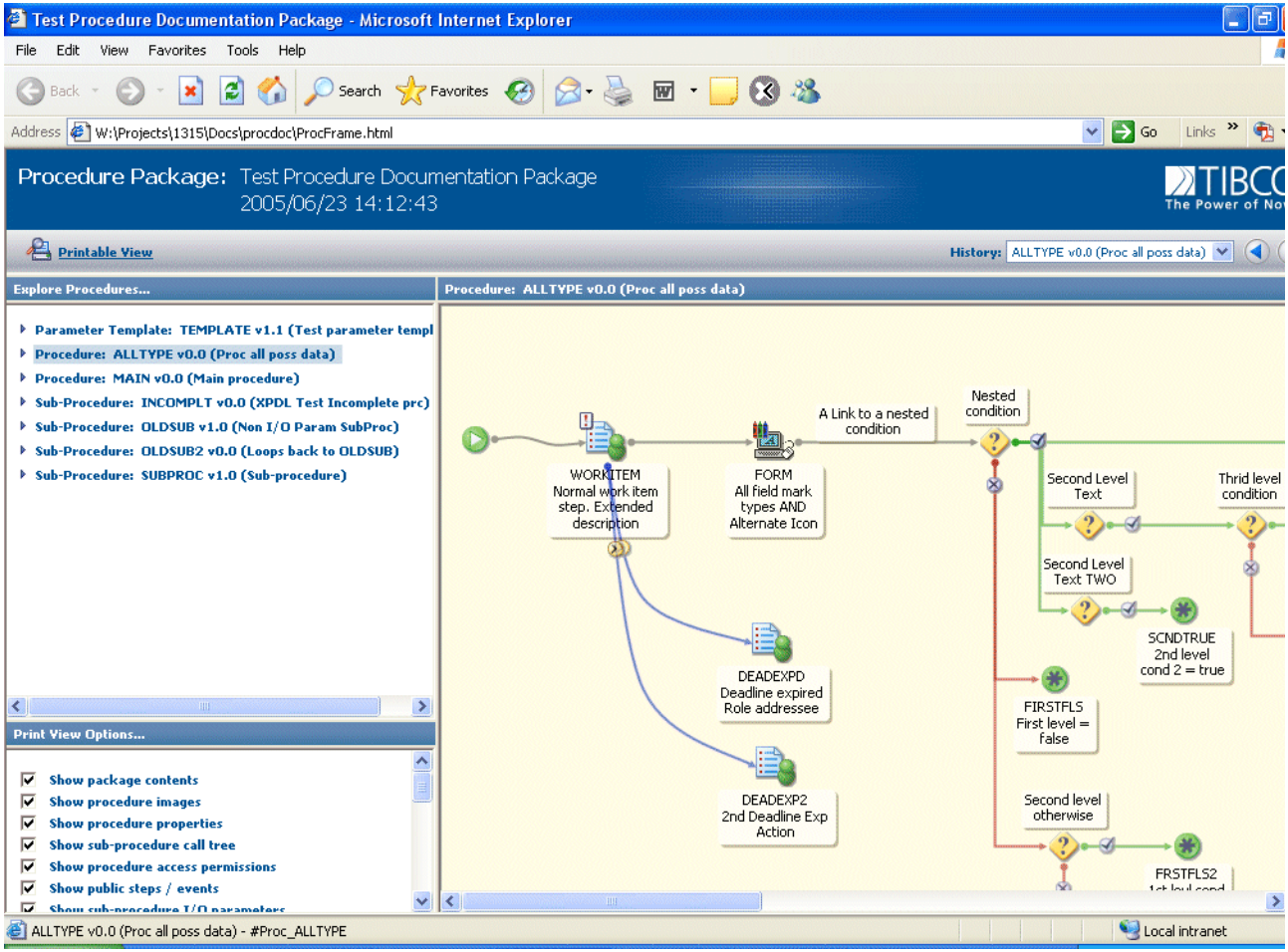
Selecting an Object to View

There are two ways you can select an object to view:

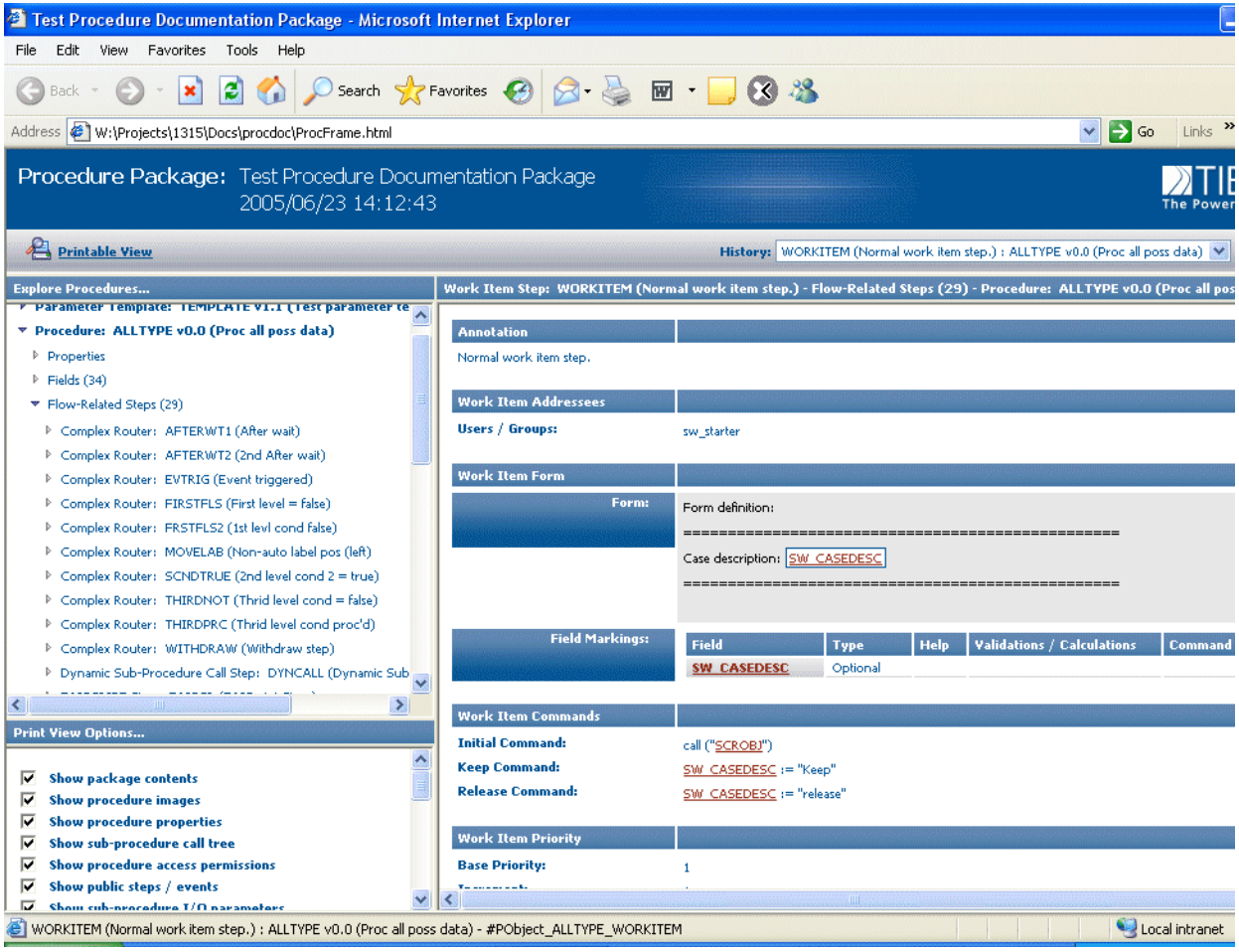
- From the **Explore Procedures...** pane, click ▸ beside the procedure whose objects you want to view. The procedure expands to display the list of objects that you can view for this procedure. Drill down to the object you want to view. Once you have selected the object, you want to view, the details of the object are displayed in the right-hand pane.



- The ***objecttype: objectname*** pane displays the details of the object. The details that are displayed are different depending on the object you chose to view. For example:
  - If you chose to view a procedure or sub-procedure, a visual representation of the procedure is displayed. You can click on the objects in the visual representation to view these objects from this window.



— If you chose to view an object other than a procedure or sub-procedure, for example, a flow-related step, the details of the object are displayed in a tabular form. You can click on hyperlinks to view other objects from this window.

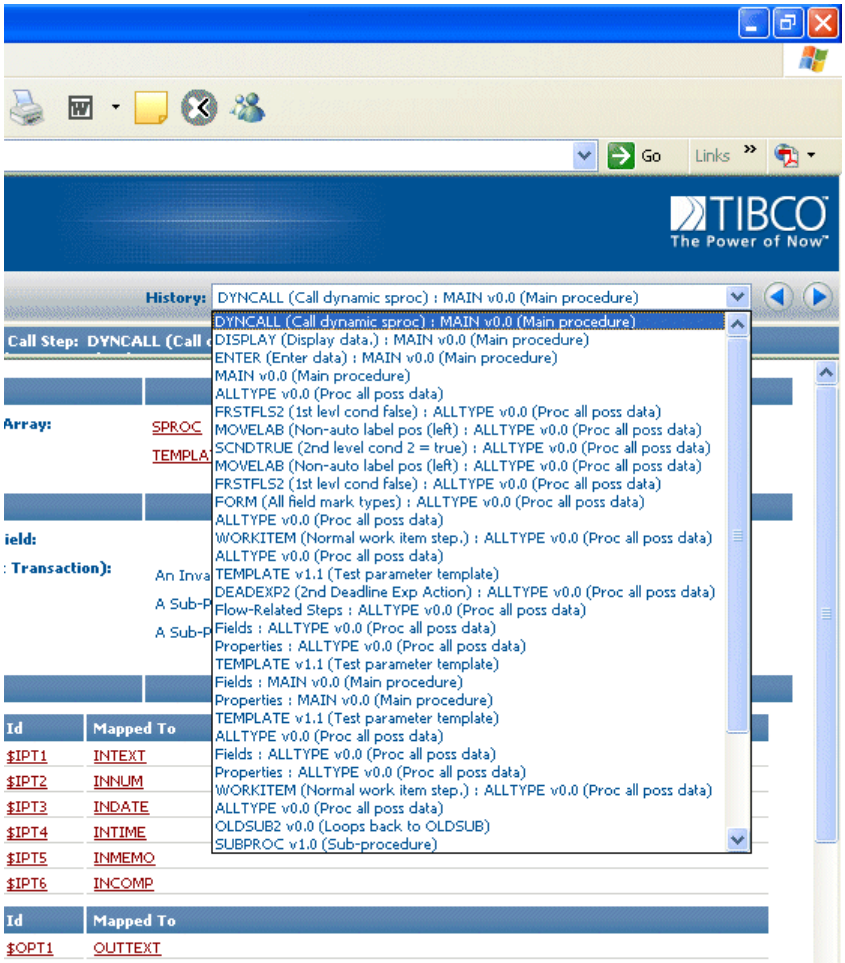


Using the History to View Objects

When viewing a procedure as HTML, the objects you have visited are not added to the browser's main history.

When viewing a procedure in the browser, you can use the **History** controls to move back and forth between the procedure objects that you have been viewing. There are two ways you can do this:

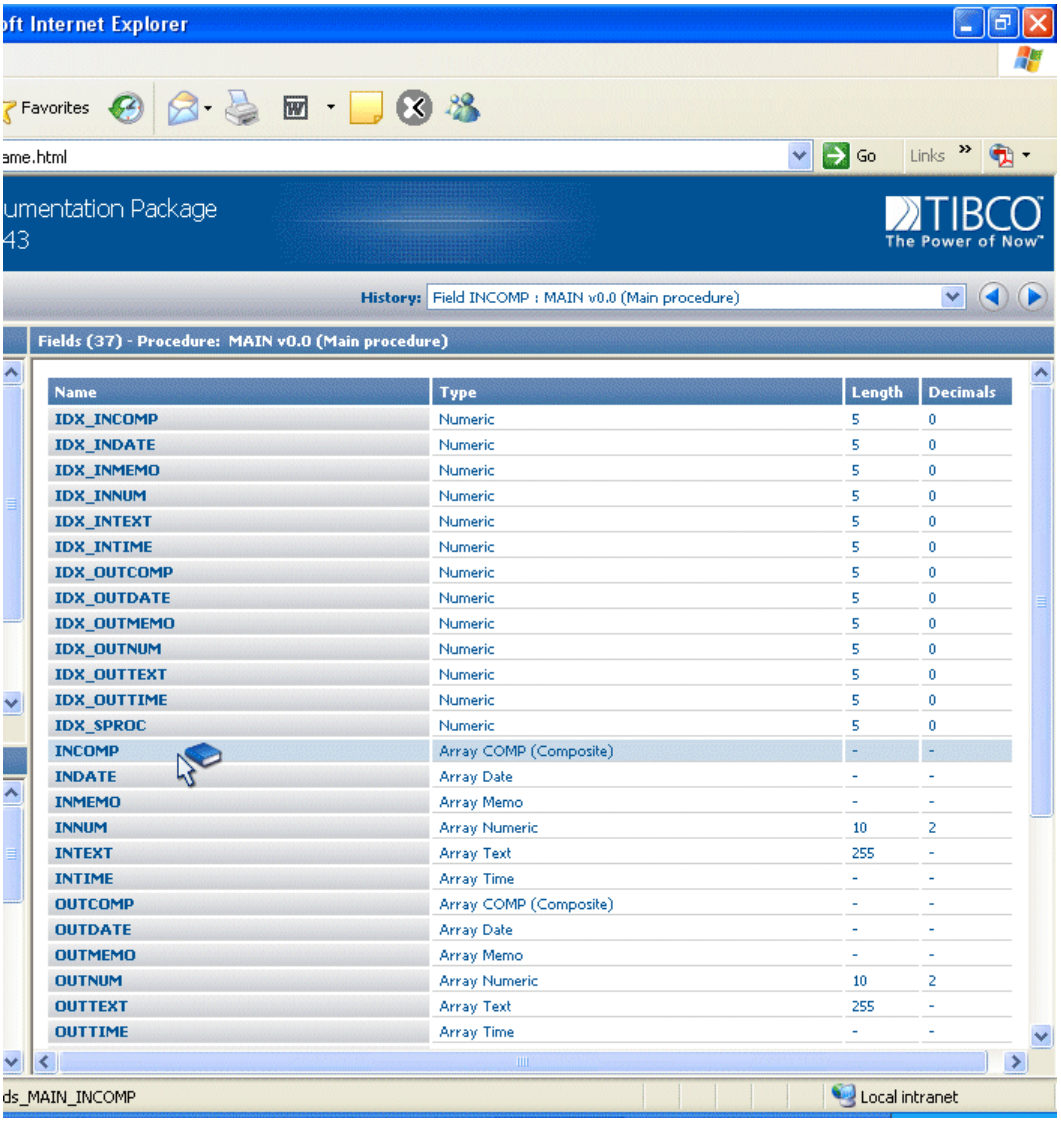
- Using the and buttons.
- Select an object from the **History** drop-down list.



Adding Objects to the History Using Bookmarks

You can add objects, for example fields, to the **History** by bookmarking them. This makes it easier to move between different objects without having to drill down through all the levels of the procedure to get back to your original object.

When you drill down to an object that can be bookmarked, the cursor automatically changes to a book mark. Click on the object to add it to the **History**.



Viewing a Printer-Friendly Version of Your Procedure

To view a printer-friendly version of your procedure, do the following:

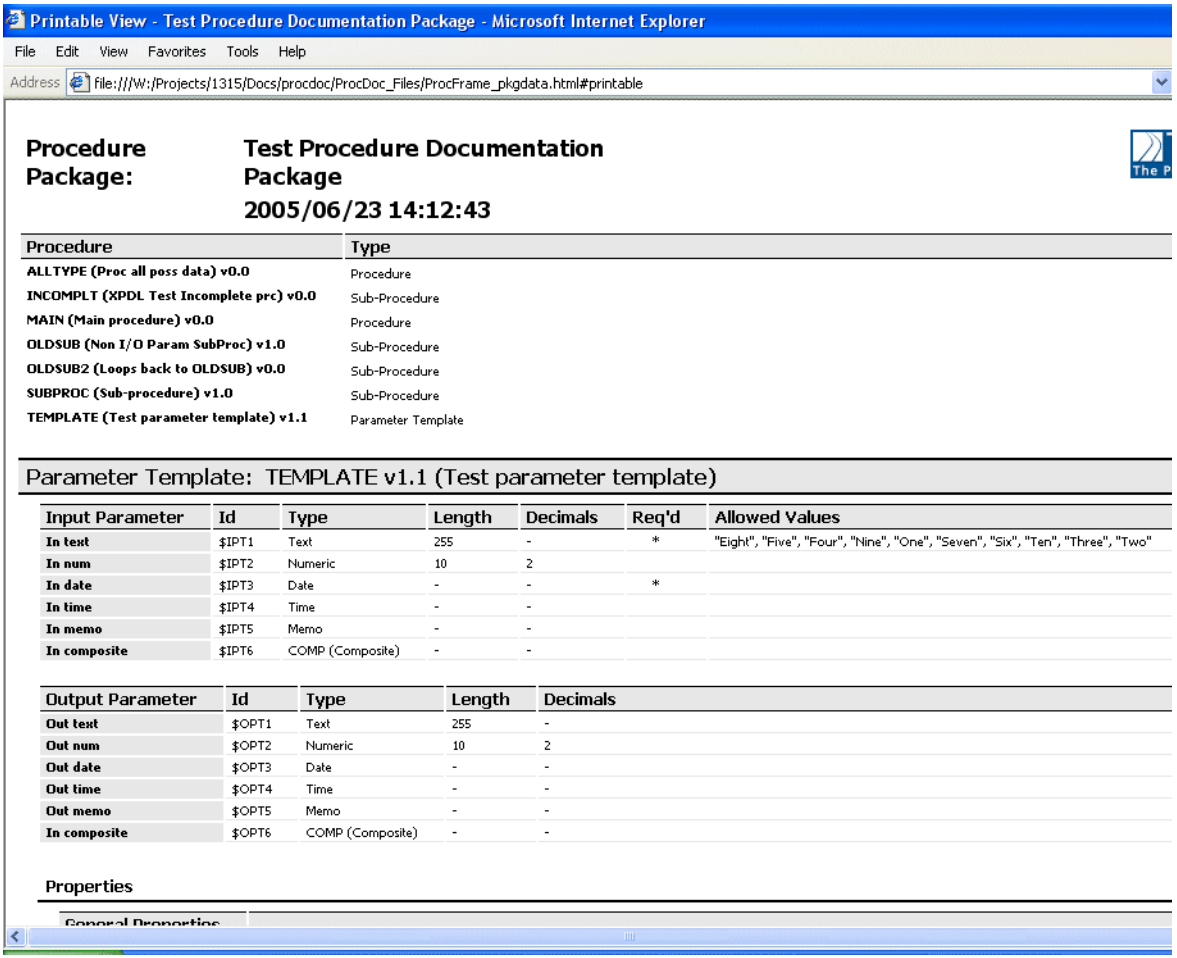
1. From the **Print View Options...** pane, select one, some or all of the following check boxes to configure how your procedure(s) is displayed in the printer-friendly version:.

Option	Description
Show package contents	Lists the procedures contained within the package and their type, for example, procedure or sub-procedure.
Show procedure images	Displays a visual representation of the procedure(s).
Show procedure properties	Displays the procedure's general properties, for example, whether working days are set, if auto-purge is set and whether or not blank addressees are ignored.
Show sub-procedure call tree	Lists the tree of sub-procedures when sub-procedures are called from other sub-procedures.
Show procedure access permissions	Lists the access permissions for the procedure(s), for example, who has permission to start a case.
Show public steps/events	Lists all the public and event steps.
Show sub-procedure I/O parameters	Lists all the sub-procedure input/output parameters for sub-procedures and sub-procedure parameter templates.
Show fields	List all the fields defined in a procedure and their properties.

Option	Description
Show non-flow related objects **Note: If you select this option alone, the non-flow related objects are just listed in the printer-friendly version of the procedure. No information about the properties of the objects is displayed.** **You must select the Show non-flow related objects option along with one of the options listed below to display the options below in the printer-friendly version. If the Show non-flow related steps option is not selected, none of the non-flow related steps or any information about them is displayed.**	Lists scripts and EIS reports which are objects that are not linked into the flow of a procedure. If you select this property, you can also choose to: • Show EIS report data file properties. This lists the EIS data file properties, for example, the owner, whether column headers are used and how the columns are separated. • Show EIS report access permissions. This lists who has access to the EIS report.

Option	Description
Show flow related steps **Note: If you select this option alone, the flow related objects are just listed in the printer-friendly version of the procedure. No information about the properties of the objects is displayed.** **You must select the Show flow related steps option along with one of the sub-options to display the options in the printer-friendly version. If the Show flow related steps option is not selected, none of the flow related steps or any information about them is displayed.**	Lists all objects that are linked together in the procedure(s). If you select this property, you can also choose to display the following options for the flow related steps: • Show objects/steps summary list. Lists a summary of all objects in the procedure. • Show step general properties. Lists the general properties of a step, for example, if the step is a graft step, the name of the array it calls is displayed. • Show object annotations. Lists the contents of any annotations. • Show work item addressees. Lists the addressees for a work item. • Show work item forms. Displays any forms that are defined for work items. • Show work item commands. Lists any commands that are defined for work items, for example, on step initialization or release. • Show work item priority. Lists the work item priority. • Show step deadlines. Lists any deadlines defined for the step. • Show step actions. Lists the actions for the step, for example, which step is processed next. • Show predicted durations. Lists any predicted durations defined for the step. • Show dynamic/graft step error handling. Lists any error handling defined for dynamic/graft steps. • Show call-step I/O parameters. Lists the input/output parameters for sub-procedure call steps. • Show EAI Type-specific data. Lists data for specific EAI types, for example, the call out data for the EAI step. • Show step 'other properties'. Lists other step properties, for example, if the step is deleted on withdraw or if case suspend is ignored.

2. Click **Printable View**. The procedure is displayed in a tabular format that makes it easier to print. For example:



Printable View - Test Procedure Documentation Package - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address file:///W:/Projects/1315/Docs/procdoc/ProcDoc_Files/ProcFrame_pkgdata.html#printable

Procedure Package: **Test Procedure Documentation Package**

2005/06/23 14:12:43

Procedure	Type
ALLTYPE (Proc all poss data) v0.0	Procedure
INCOMPLT (XPDL Test Incomplete prc) v0.0	Sub-Procedure
MAIN (Main procedure) v0.0	Procedure
OLDSUB (Non I/O Param SubProc) v1.0	Sub-Procedure
OLDSUB2 (Loops back to OLDSUB) v0.0	Sub-Procedure
SUBPROC (Sub-procedure) v1.0	Sub-Procedure
TEMPLATE (Test parameter template) v1.1	Parameter Template

Parameter Template: TEMPLATE v1.1 (Test parameter template)

Input Parameter	Id	Type	Length	Decimals	Req'd	Allowed Values
In text	$IPT1	Text	255	-	*	"Eight", "Five", "Four", "Nine", "One", "Seven", "Six", "Ten", "Three", "Two"
In num	$IPT2	Numeric	10	2		
In date	$IPT3	Date	-	-	*	
In time	$IPT4	Time	-	-		
In memo	$IPT5	Memo	-	-		
In composite	$IPT6	COMP (Composite)	-	-		

Output Parameter	Id	Type	Length	Decimals
Out text	$OPT1	Text	255	-
Out num	$OPT2	Numeric	10	2
Out date	$OPT3	Date	-	-
Out time	$OPT4	Time	-	-
Out memo	$OPT5	Memo	-	-
In composite	$OPT6	COMP (Composite)	-	-

Properties

If you amend the **Print View Options**, the amendments are automatically updated in the printer-friendly version of the procedure.

Customizing Your HTML Document

You can customize how the HTML document is displayed in your browser.

The following table describes the CSS stylesheets that specify the visual appearance of the elements within the HTML document. These are located in *swclient***ProcDoc_Files**.

CSS Stylesheet	Description
pd_HdrStyles.css	Style classes for the **Title** frame.
pd_ExpStyles.css	Style classes for the **Explorer** and **Options** frame
pd_ViewStyles.css	Style classes for the **View** frame
pd_PrintStyles.css	Style classes for the **print view document** frame

For more information about the CSS style classes, see the comments within these stylesheets.

For more information on CSS stylesheets, see www.w3schools.com/css.

Chapter 5 Working with Properties of Procedures

This chapter describes how to set and view procedures of properties, access controls, status, events, and deadlines for different types of procedure objects.

Topics

Displaying the Properties Dialog for a Procedure Object

In Procedure Manager, you can display the properties for either a single object or for multiple objects.

To display the **Properties** dialog:

1. Open the library containing the object(s) you want to see the properties for.
2. Select the object(s). You cannot select multiple objects if they are in different libraries.
3. Click  .

The **Properties** dialog is displayed. This dialog has five tabs - Properties, Access Control, Status, Events, and Deadline - which allow you to view and set different types of information about the selected object(s).

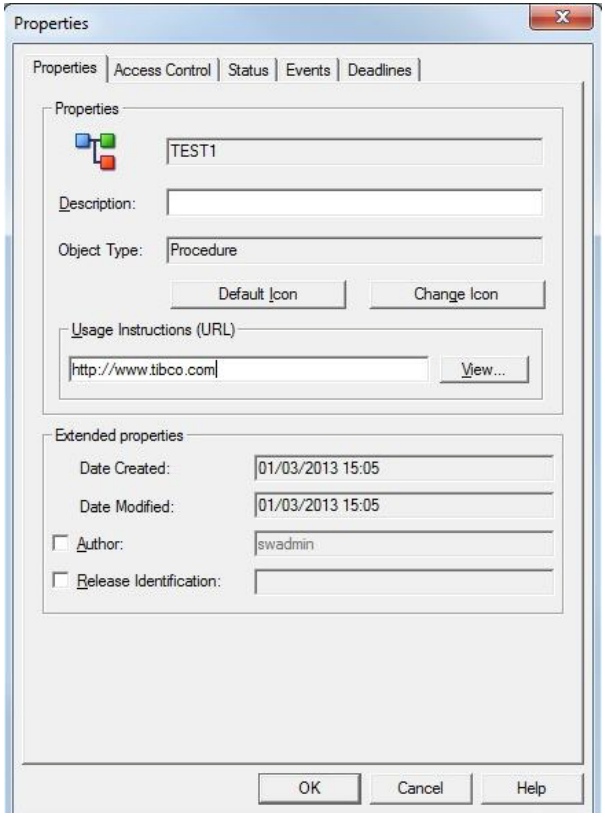
Note that:

- if you have selected a *single object*, the dialog shows the current properties for that object.
- if you have selected *multiple objects*, the property fields are blanked. Any property values that you set will apply to all the objects you selected. However, certain properties cannot be set if you have multiple objects selected - for example, icon image. You will not be able to enter a value for those properties in this case.
- if you have selected multiple objects or a library, the **Status** tab is not shown.

Setting Properties

The following properties can be set from the **Properties** tab of the **Properties** dialog:

- Description
- Icon Image
- Usage Instructions
- Extended Properties



The object's name and type are also displayed on this tab, but you cannot change them. If you want to change a procedure's type from being a main procedure to a sub-procedure or vice versa, see Changing the Procedure Type of a Version on page 81.

Description

Defines the textual description of an object that is displayed in the **Name (Description)** column in the Procedure Manager.

When You Can Define a Description

You can define a description for any procedure object except a library.

You must have a *single* procedure object selected. You cannot define a description for multiple objects in the same operation.

How to Define a Description

To define the description for a procedure object:

1. Display the Properties dialog for the object.
2. On the **Properties** tab, in the **Description** field, enter a text description for the object, up to a maximum of 24 characters.

Icon Image

Defines the bitmap image that is used to graphically represent an object in the **Icon** column in the Procedure Manager, and in the Procedure Browser.

When You Can Define an Icon Image

You can define an icon image for any procedure object.

You must have a *single* procedure object selected. You cannot define an icon image for multiple objects in the same operation.

How to Define an Icon Image

To define the icon image for a procedure object:

1. Display the Properties dialog for the object. The image currently used as the procedure's icon is shown at the beginning of the **Properties** section on the **Properties** tab.

2. If you want to use:
   - — *the default iProcess image*, click **Default Icon**. The default iProcess image is displayed at the beginning of the **Properties** section.
   - — *a different image*, click C**hange Icon**. A standard Windows **Open** dialog is displayed. Locate the **.BMP** file you want to use as the new image and click **OK**. The new image is displayed at the beginning of the **Properties** section.

The bitmap selected will be truncated to 64*64 pixels in size.

Usage Instructions

Specifies the internet address (URL) of a web page or other document that contains text describing what the object is used for.

The URL is shown in the **Usage Instructions** column of the Procedure Manager. To view the usage instructions for an object:

1. Select the object.
2. Click  .

When You Can Define Usage Instructions

You can define usage instructions for any procedure object.

You must have a *single* procedure object selected. You cannot define usage instructions for multiple objects in the same operation.

How to Define Usage Instructions

To define usage instructions for a procedure object:

1. Display the Properties dialog for the object.
2. On the **Properties** tab, in the **Usage Instructions (URL)** field, enter a text string that specifies the internet address of your usage instructions.

Note that:

- The text string used can be a maximum of 1000 characters in length, and can contain forward slashes (/) or backward slashes (\) as path delimiters.

- You can set a **base path** for all your objects by setting the URL for the **Procedure Management** library. All procedure objects will then inherit this base path and add it to the front of their own address.

Setting a base path in this way allows you to move a collection of usage instructions around, without having to edit the location for each procedure object.

- The usage instructions themselves are not stored with the procedure, so you must make sure that the URL is still valid on the target system when the procedure is distributed.
- You can click **View** to test/view your URL.

The following examples are valid entries for the Usage Instructions (URL) field.

- the URL of a web page:

  `http://www.tibco.com/proclibs/HireProc.htm`

- a text file on a network drive

  `m:\docs\HireProc.txt`

- a text file where the file's location is not given for the object itself, but inherited from the **Procedure Management** library. Specify:

  `HireProc.txt`

  in the field for the object, and

  `m:\docs\`

  in the field for the **Procedure Management** library.

Extended Properties

Comprises four separate but related fields:

- Author - the name (or other identifier) of the iProcess user who created the object.
- Date Created - the date and time that the object was created.
- Date Modified - the date and time that the object was last modified.
- Release Identification - free text that you can use to identify an object.

When You Can Define Extended Properties

You can define extended properties for any procedure object. You can have either a single procedure object or multiple objects selected.

You can only define the Author and Release Identification properties. You cannot define or modify the Date Created or Date Modified properties - these are automatically set by iProcess.

How to Define the Author

To define the Author for a procedure object:

1. Display the Properties dialog for the object.
2. On the **Properties** tab, in the **Extended properties** area:
   a. Select the check box to the left of the **Author** field. You cannot edit the entry unless this tick box is checked.
   b. In the **Author** field, enter the name (or other identifier) of the user who created the object, up to a maximum of 64 characters.

When a procedure is saved for the first time, the **Author** property is automatically set to the User Description attribute of the iProcess user who owns the procedure.

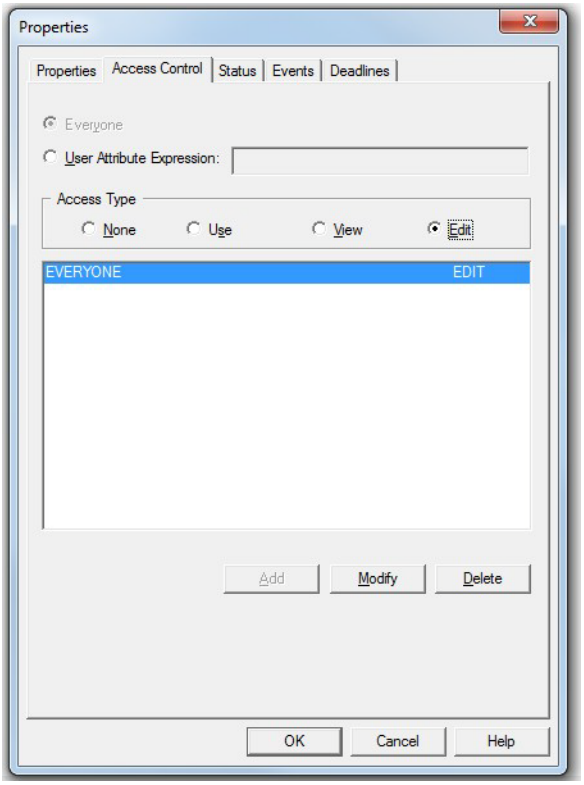
How to Define the Release Identification

To define the Release Identification for a procedure object:

1. Display the Properties dialog for the object.
2. On the **Properties** tab, in the **Extended properties** area:
   a. Select the check box to the left of the **Release Identification** field. You cannot edit the entry unless this tick box is checked.
   b. In the **Release Identification** field, enter a suitable description for the object, up to a maximum of 64 characters.

Setting Access Controls

You can control who has access to a procedure object from the **Access Control** tab of the Properties dialog.



The **Access Control** tab lists the users who have access to this object, and what type of access they have.

The following table defines the meaning of the different **Access Type**s.

Access Type	Description
None	The object is not visible. You have no access to it.
Use	The object is visible but not available. You cannot view its properties or open it in iProcess Modeler. If the object is a sub-procedure, you can define calls to it from iProcess Modeler (or by dragging it from the Procedure Browser).
View	The object is visible. You can view its properties and open it in iProcess Modeler, but you cannot save changes. If the object is a sub-procedure, you can define calls to it from iProcess Modeler (or by dragging it from the Procedure Browser).
Edit	The object is visible. You can edit its properties and edit it in iProcess Modeler.

You can add new entries to the list, and modify or delete existing entries to set up the access controls as you wish.

Note that:

- When a procedure object is created, it inherits the access permissions of its parent library. The default entry for the **Procedure Management** library is **Edit** access for **EVERYONE** (i.e. for all iProcess users).
- iProcess Administrator users always has **Edit** access to all procedure objects, irrespective of its **Access Control** settings. (This is not explicitly shown in the **Access Control** list.)
- Any changes to the access controls for a procedure object take effect as soon as a user updates their Procedure Manager cache (by clicking ).
- If a user has multiple access criteria set (because multiple expressions are defined) then the user has the highest permissions that it meets.

Adding an Entry to the Access List

To add a new entry to the list:

1. Define the users who will have this access. Either:
   - click **Everyone**. There can only be a single entry for **Everyone**, so the button is not available if an entry already exists. (If you have multiple objects selected, you will not see the entries that exist for individual objects.)
   - enter a **User Attribute Expression**. This can be any valid iProcess expression that resolves to a user or group name - for example, **SW_USER:grp_accounts="yes"**.
2. In the **Access Type** section, select either **None**, **Use**, **View** or **Edit**.
3. Click **Add**. The entry is added to the list.

Modifying an Entry in the Access List

To modify an existing entry:

1. Select the entry in the list.
2. Edit the user definition or access type as required.
3. Click **Modify**. The entry is updated in the list.

Deleting an Entry From the Access List

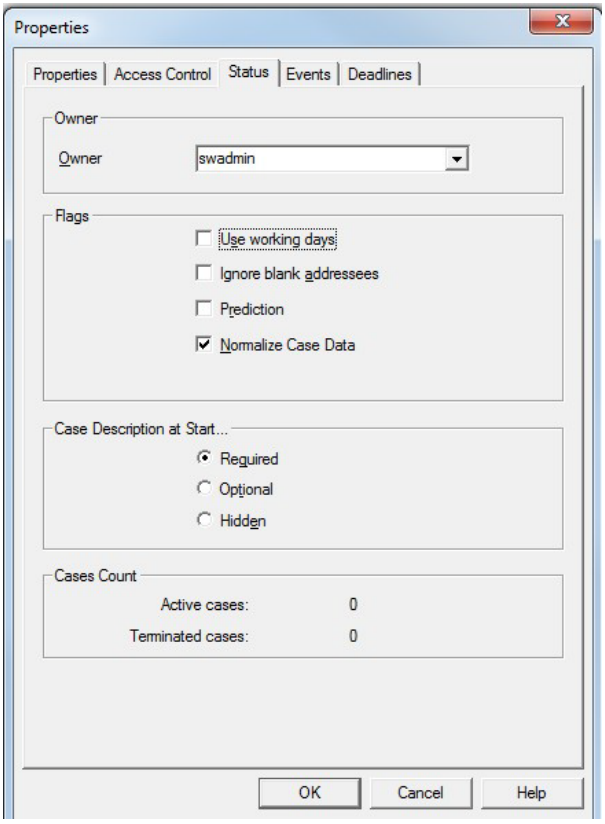
To delete an existing entry:

1. Select the entry in the list.
2. Click **Delete**. The entry is removed from the list.

Setting and Viewing Status Information

The following properties can be set from the **Status** tab of the **Properties** dialog.

- Owner
- Use Working Days Flag
- Ignore Blank Addressees Flag
- Prediction Flag
- Normalize Case Data Flag
- Case Description at Start
- Cases Count



Owner

Defines the iProcess user who owns a procedure. The owner:

- can always start cases of a procedure object.
- always has Edit permissions to the object by default (unless other permissions have been explicitly set, or inherited from the parent object.)

When You Can Define the Owner

You can define the owner for a procedure, sub-procedure or sub-procedure template.

You must have a *single* procedure object selected. You cannot define the owner for multiple objects in the same operation.

How to Define the Owner

To define the **Owner** for a procedure object:

1. Display the Properties dialog for the procedure.
2. On the **Status** tab, select a user name from the drop-down list in the **Owner** field.

The **Status** tab is not displayed if you have a library and/or multiple procedure objects selected.

Use Working Days Flag

Specifies whether to use iProcess Engine's working week or a full 7-day week, when performing date calculations such as checking for deadline expiry.

By default, the flag is set so the working week is used.

You can configure the days that are included as working days in the SWDIR**etc****staffpms** file. For more information see "Specifying the Working Week" in *TIBCO iProcess Engine Administrator's Guide*.

When You Can Define the Use Working Days Flag

You can define the **Use working days** flag for a procedure or sub-procedure.

You must have a *single* procedure object selected. You cannot define this flag for multiple objects in the same operation.

How to Define the Use Working Days Flag

To define the **Use working days** flag for a procedure object:

1. Display the Properties dialog for the procedure.
2. On the **Status** tab, either:
   - select the **Use working days** check box to use iProcess Engine's working week when performing date calculations.
   - *clear* the **Use working days** check box to use a full 7-day week when performing date calculations.

Changing this flag will affect all new work items. Existing work items in queues will not be affected.

For example, consider a default iProcess installation that uses a working week of Monday to Friday. If a step is sent to a queue on a Wednesday and has a deadline that is due to expire in three days, the deadline will expire:

- on the following Monday, if the **Use Working Days** flag is selected.
- on the Saturday, if the **Use Working Days** flag is not selected.

Ignore Blank Addressees Flag

Specifies whether or not a blank addressee for a step should be treated as an error condition.

By default the flag is not set, so a step with a blank addressee is treated as an error.

For more information about using fields to define variable addressees (dynamic routing), see "Routing the Business Process" in *TIBCO iProcess Modeler Basic Design*.

When You Can Define the Ignore Blank Addressees Flag

You can define the **Ignore Blank Addressees** flag for a procedure or sub-procedure.

You must have a *single* procedure object selected. You cannot define this flag for multiple objects in the same operation.

How to Define the Ignore Blank Addressees Flag

To define the **Ignore Blank Addressees** flag for a procedure object:

1. Display the Properties dialog for the object.

2. On the **Status** tab, either:
   - select the **Ignore Blank Addressees** check box. If a step is processed that uses a field as an addressee, and the field is blank, this is treated as intended behavior. No work item is delivered.

     This behavior may be appropriate in some situations. For example, if you have a review step that uses case data to generate the list of reviewers as addressees. In some cases no review will be required, so the addressee list is blank. As this is intended behavior, you do not want to treat this as an error.
   - clear the **Ignore Blank Addressees** check box. If a step is processed that uses a field as an addressee, and the field is blank, this is treated as an error condition and the work item is sent to the iProcess Administrator user's **Undelivered Work Items** queue.

Prediction Flag

Specifies whether or not the use of case prediction is enabled for this procedure.

By default, the flag is not set so case prediction is disabled.

For more information about case prediction, see "Using Case Prediction to Forecast Outstanding Work Items" in *TIBCO iProcess Modeler Advanced Design*.

When You Can Define the Prediction Flag

You can define the **Prediction** flag for a procedure or sub-procedure.

You must have a *single* procedure object selected. You cannot define this flag for multiple objects in the same operation.

How to Define the Prediction Flag

To define the **Prediction** flag for a procedure object:

1. Display the Properties dialog for the object.
2. On the **Status** tab, either:
   - select the **Prediction** check box. Case prediction is enabled for the procedure.
   - clear the **Prediction** check box. Case prediction is disabled for the procedure.

Normalize Case Data Flag

Specifies whether or not case data normalization is enabled for this procedure. By default this check box is either:

- enabled if the global process attribute NORMALISE_CASE_DATA is set to 1 (case data normalization is enabled system-wide), or
- greyed out if the global process attribute NORMALISE_CASE_DATA is set to 0 (case data normalization is disabled system-wide).

If the procedure already has cases, the check box is greyed out and cannot be changed.

For more information about case data normalization, see Chapter 11, "Administering Case Data Normalization" in the *TIBCO iProcess Engine Administrator's Guide.*

When You Can Define the Normalize Case Data Flag

You can define the **Case Data Normalization** flag for a procedure or sub-procedure if:

- the global process attribute NORMALISE_CASE_DATA is set to 1 (case data normalization is enabled system-wide, and
- the procedure does not have any cases.

You must have a *single* procedure object selected. You cannot define this flag for multiple objects in the same operation.

How to Define the Case Data Normalization Flag

To define the **Case Data Normalization** flag for a procedure object:

1. Display the Properties dialog for the object.
2. On the **Status** tab, either:
   - — select the **Case Data Normalization** check box. Case data normalization is enabled for the procedure.
   - — clear the **Case Data Normalization** check box. Case data normalization is disabled for the procedure.

Procedure Import/Export Implications

When you export a procedure, the setting of the procedure's **Case Data Normalization** flag is saved. When you import a procedure:

- if overwriting an existing procedure that has cases, the imported procedure preserves the state of the overwritten procedure's **Case Data Normalization** flag.
- if creating a new procedure, or overwriting a procedure which has no case data, the imported procedure's **Case Data Normalization** flag setting is preserved.

Case Description at Start

Specifies whether users can, cannot or must provide a case description when they start a case of a procedure.

By default, a case description is required.

When You Can Define the Case Description at Start Setting

You can define the **Case Description at Start** setting for a procedure or sub-procedure.

You must have a *single* procedure object selected. You cannot define this setting for multiple objects in the same operation.

How to Define the Case Description at Start Setting

To define the **Case Description at Start** setting for a procedure object:

1. Display the Properties dialog for the object.
2. On the **Status** tab, in the **Case Description at Start** section, select one of the following radio buttons:
   - **Required**. The user must enter a case description in the **Case Start** dialog when they start a case of this procedure.
   - **Optional**. The user can, but does not have to, enter a case description in the **Case Start** dialog when they start a case of this procedure.
   - **Hidden**. The user cannot enter a case description in the **Case Start** dialog when they start a case of this procedure. (The **Case Description** field is not displayed in the **Case Start** dialog.)

     You can use the **Hidden** option if you want to calculate the case description from within the process. For example, you might want to make the case description equal to the customer's postcode (zip code) and name to facilitate searching for the case in response to customer queries.

Cases Count

Shows the number of live (outstanding) and dead (completed or otherwise terminated) cases associated with this procedure.

You cannot delete a procedure that has any live cases associated with it.

When You Can View the Cases Count

You can view the **Cases Count** setting for a procedure or sub-procedure.

You must have a *single* procedure object selected. You cannot view this count for multiple objects in the same operation.

You cannot set or alter this value; it is displayed for information only.

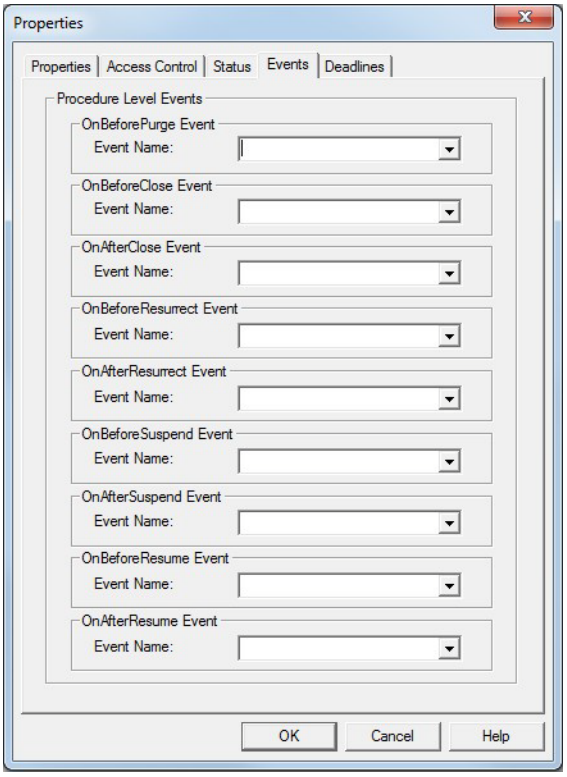
How to View the Cases Count

To view the **Cases Count** for a procedure object:

1. Display the Properties dialog for the object.
2. Click the **Status** tab. The number of **live** and **dead** cases associated with this procedure is shown in the **Cases Count** section.

Setting Procedure Events

You can define an event, which is triggered before or after performing the purge, close, resurrect, suspend, or resume action, from the Events tab of the Properties dialog. The events are triggered to update external data or to run external applications when performing the purge, close, resurrect, suspend, or resume action.



In the Event tab, it contains the following procedure level events:

You can use the `CancelProcEvent` expression to cancel the events that are defined as the OnBefore*Event* procedure level events. There is no effect on the close, resurrect, suspend, and resume actions by calling the `CancelProcEvent` expression from the OnAfter*Event* events, since the actions has already been performed. See Cancelling Procedure Events on page 127 for more information.

- **OnBeforePurge** defines an event, which is triggered when the case is about to purge but before the case is actually purged. In the event, you can define steps to clean up external data that are related to the case in the external systems.
- **OnBeforeClose** defines an event, which is triggered when the case is about to close but before the case is actually closed. In the event, you can define steps to update data in external systems.

  The event is only called when the case is closed abnormally, for example, close a case by running the `swutil CLOSE` command or the SSOLite close command.
- **OnAfterClose** defines an event, which is triggered after closing cases. In the event, you can define steps to update data in external systems. The event is called in one of the following conditions:
  - An external close of the case, for example, close a case by running the `swutil CLOSE` command or the SSOLite close command.
  - A natural close of the case when the case comes to a completion, for example, no outstanding steps in the case.

If the `CancelProcEvent` expression is called when performing the OnBeforeClose event, the close transaction of the case will be aborted. In such cases, the OnAfterClose event if defined in the case is not triggered.

- **OnBeforeResurrect** defines an event, which is triggered when the case is about to resurrect but before the case is actually resurrected. In the event, you can define steps to update iProcess data to external systems, and receive data from external systems to iProcess.
- **OnAfterResurrect** defines an event, which is triggered after resurrecting cases. In the event, you can define steps to update iProcess data to external systems, or receive data from external systems to iProcess.
- **OnBeforeSuspend** defines an event, which is triggered when the case is about to suspend but before the case is actually suspended. In the event, you can define steps to update external data or to run external applications.
- **OnAfterSuspend** defines an event, which is triggered after suspending cases. In the event, you can define steps to update data in external systems.
- **OnBeforeResume** defines an event, which is triggered when the case is about to resume but before the case is actually resumed. In the event, you can define steps to update data in external systems.

- **OnAfterResume** defines an event, which is triggered after resuming cases. In the event, you can define steps to update data in external systems.

When the main case is purged, closed, suspended, or resumed (except the resurrect action), the action related procedure events defined in the sub-procedures are triggered as well. For example, after closing a case, its sub-cases are also closed. So the event defined as the OnAfterClose procedure event in the main procedure and sub-procedures are all triggered.

How to Define a Procedure Level Event

To define procedure events, perform the following steps:

1. In TIBCO iProcess Modeler, create an event in a procedure. Save the procedure

The event must be created in the procedure that you want to define a procedure level event; the Event object in the procedure should have no links on the left hand side (the events that start up an entire branch of a case).

2. Click the procedure in the Procedure Details list on the right-hand side of the TIBCO iProcess Workspace (Windows) window, and press **Ctrl+O** to open the Properties dialog.
3. In the Events tab, define a procedure level event. In the procedure event panel, for example, in the OnBeforePurge Event panel, select the event name defined in the first step from the Event Name drop-down list.

You can use any types of procedure objects to define an event, but TIBCO recommends that you use the EAI Step object as the right hand side of the Event object. If the event is defined as the OnAfterSuspend or OnBeforeResume procedure event, the normal Step object in the event can not be performed after triggering the event. For example, if you set a normal Step object to the right hand side of an Event object, and define the event as the OnAfterSuspend event. When the case is suspended, the OnAfterSuspend event is triggered, the event step cannot be processed since the case is suspending.

The event will be triggered before purging the case. See Example of a Procedure with Procedure Events on page 128 for more information.

Adding, Updating, and Deleting Procedure Events

To set procedure events, you may need to add, update, or delete procedure events:

- Adding a Procedure Event
- Updating a Procedure Event
- Deleting a Procedure Event

Adding a Procedure Event

To add a procedure event, perform the following steps:

1. In the procedure event panel, select an event name you want to trigger in the Event Name drop-down list. You can also type the event name in the Event Name drop-down list. If an event name you entered is not defined in the procedure, an error dialog is displayed to remind you to set a valid event name when clicking the OK button.
2. After setting all the events for the procedure, click the **OK** button to save the setting.

Updating a Procedure Event

1. In the procedure event panel, reselect an event name in the Event Name drop-down list. You can also type an event name in the Event Name drop-down list. If the event name you entered is not defined in the procedure, an error dialog is displayed to remind you to set a valid event name when clicking the OK button.
2. After setting all the events for the procedure, click the **OK** button to save the changes.

Deleting a Procedure Event

1. In the procedure event panel, which you want to delete the event, delete the value in the Event Name drop-down list.
2. After setting all the events for the procedure, click the **OK** button to save the changes.

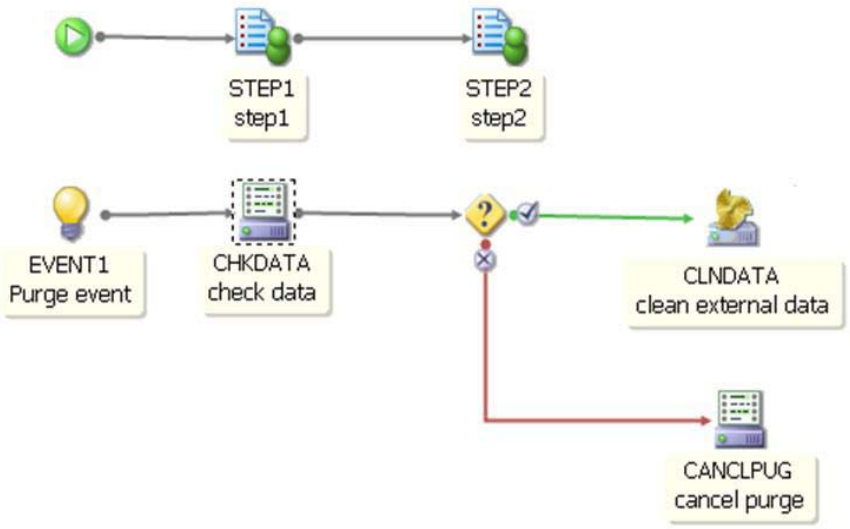
Cancelling Procedure Events

You can use the `CancelProcEvent` expression to cancel the purge, close, resurrect, suspend, or resume action that is set in the OnBefore*Event* procedure level events. If this expression is called when performing the OnBefore*Event* event, the corresponding OnAfter*Event* will not be triggered. See *TIBCO iProcess Expressions and Functions Reference Guide* for more information.

Example of a Procedure with Procedure Events

This section describes a holiday booking procedure with procedure events. In this booking process, the requirements of customers are initially input in an external holiday booking system, and a case of the booking procedure is started automatically to process necessary fulfilment activities. When you purge the case, a condition is evaluated to determine whether to purge the customer data in the external holiday booking system or to cancel the purge action.

The procedure is defined as follows.



The first branch defines a case of the booking procedure to process customer requirements. The second branch defines an event for purging data in the external holiday booking system. The event performs the following steps:

1. Before performing the purge case action, the `EVENT1` event step is triggered. The `EVENT1` is defined as the `OnBeforePurge` procedure event, so it is triggered before purging the case.

   To trigger the `EVENT1` event before purging a case, you need to define a `OnBeforePurge` event, and select the `EVENT1` event name from the Event Name drop-down list of the OnBeforePurge Event panel in the Events tab of

the Properties dialog, as shown in the following figure. See Adding a Procedure Event on page 127 for more information.

2. After the `EVENT1` step is released, the `CHKDATA` script step is processed to check the data and then a condition is evaluated to determine whether to purge the data in the external booking system or not.
   - If the condition is evaluated to `True`, then the `CLNDATA` step is processed to purge the customer data in the external holiday booking system.
   - If the condition is evaluated to `False`, then the `CancelProcEvent` expression in the `CANCLPUG` script step is called to cancel the purge action. For more information about the `CancelProcEvent` expression, see Cancelling Procedure Events on page 127.

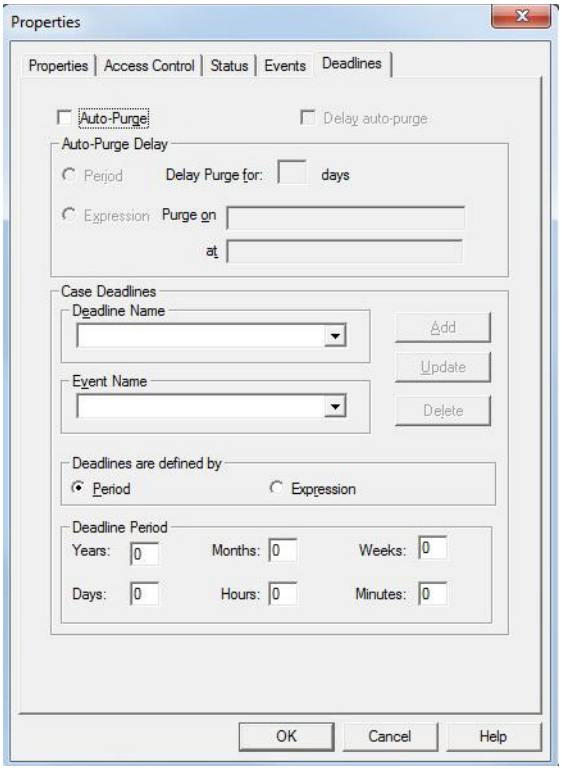
Procedure Events in Database

The `proc_event` table stores the definition of procedure events. The `case_event` table stores information of cases that are interrupted by triggered events when processing the purge, close, resurrect, suspend, or resume operation. The case information is recorded in this table only when the `BG` process is handling the delayed release EAI steps, which are defined in the triggered events. After finishing the event, the case resumes execution and fetches the temporary case data from this table.

For more information about the tables, see the appropriate *TIBCO iProcess Engine (Database) Administrator's Guide*.

Setting Auto-Purge and Deadlines

The procedure properties of auto-purge and deadlines can be set in the Deadlines tab of the Properties dialog, as shown in the following figure.



The following sections describe how to configure the properties of auto-purge and deadline:

- Case Auto-purge Setting
- Case Deadline Setting

Case Auto-purge Setting

The auto-purge function automatically purges the cases when the cases are closed. You can also delay the auto-purge function by setting a delay date. When a case is purged, all records of the case are deleted, including the audit trail. This option should therefore be used with care. By default, the auto-purge function is off.

Description of Auto-purge Fields

You can set the fields of the auto-purge property to:

- Purge cases automatically.
- Delay the auto-purge operation.

Field	Description
Auto-Purge	Checking this field enables you to purge cases automatically when cases complete.
Delay Auto-purge	You can delay the auto-purge operation by checking this field. This field is only available when the Auto-Purge checkbox is checked.
If the Delay Auto-purge checkbox is checked, the following fields are available. You can set periods or expressions after which the auto-delay operation is processed.	
Period	Selecting this field enables you to set a period for the duration of the auto-purge delay.
Delay Purge For	Specify the number of days to delay the auto-purge operation. This field is only available when the Period radio button is selected.
Expression	Selecting this field enables you to set date and time expressions as the duration of the auto-purge delay.

Field	Description
Purge	This field is only available when the Expression radio button is selected. The Purge field contains the following sub-fields: • **On** Specifies an expiration date after which the auto-delay operation is processed. The expiration date can be specified either as a date expression or as a date type field. • **At** Specifies an expiration time after which the auto-delay operation is processed. The expiration time can be specified either as a time expression or as a time type field. For more information about date and time expressions, see *TIBCO iProcess Expressions and Functions Reference Guide*. For more information about the date type and time type fields, see "Creating Fields and Forms" in *TIBCO iProcess Modeler Basic Design*. **Note**: If the On field is set as the date type field and the At field is set as the time type field: • If the value of the date type field and the value of the time type field are set in your predefined form, the expiration for the auto-purge delay will be calculated based on these values. • If the value of the date type field is set and the value of the time type field is not set in your predefined form, the value of the time type field will be automatically set to `23:59`. The expiration for the auto-purge delay will be calculated based on these values. • If the value of the date type field is not set in your predefined form, the expiration for the auto-purge delay will be `90` days from the current date. The default value of the date type field is `90` days. You can specify the default value of the date type field by using the `AUTO_PURGE_DELAY` process attribute. See "Administering Process Attributes" in *TIBCO iProcess Engine Administrator's Guide* for more information.

Activate Auto-purge or Delay Auto-purge Functions

This section describes how to activate the auto-purge function and how to delay the auto-purge function.

Activate the Auto-purge Function

To activate the auto-purge function, perform the following steps:

1. Check the **Auto-purge** checkbox in the Deadlines tab.

2. Click the **OK** button to save the configuration.

Activate the Delay Auto-purge Function

To delay the auto-purge function, perform the following steps:

1. Check the **Auto-purge** checkbox in the Deadline tab and then check the **Delay Auto-purge** checkbox.
2. Specify the duration of the delay. Select the **Period** radio button or the **Expression** radio button in the Auto-Purge Delay panel.
   - If the Period radio button is selected, specify the number of days for the delay. When the case is completed the delay period is calculated.
   - If the Expression radio button is selected, specify a date expression in the Purge On field and specify a time expression in the At field. When the case is completed the delay period is evaluated.
3. Click the **OK** button to save the configuration.

Case Auto-purge in Database

The information of the auto-purge delay are recorded in the `proc_index` table.

For more information about the `proc_index` table, see the appropriate *TIBCO iProcess Engine (Database) Administrator's Guide*.

Case Deadline Setting

A case deadline can be set on a case to trigger an event after a specified time period. If a case is not completed and its deadline expires, the event is triggered.

After migrating cases of a procedure to a new version of a procedure definition or after migrating live cases of a procedure to a new version of a procedure definition, the deadlines of cases are not migrated to the new version of the procedure definition.

After a case is closed, all the deadlines of the case are removed. If the case is reopened, you can reset the deadlines by running the `CreateCaseDeadline` function.

Description of Deadline Fields

You can set case deadlines in the following fields.

Field		Description
Deadline Name		The name of a case deadline. You can define or select a case deadline in this field. **Note**: Deadline naming limitations: • The name must be up to eight characters in length. • The first character is limited to uppercase and lowercase letters (A-Z). • The name may contain letters, numbers, and underscore characters. • The name is not case sensitive.
Event Name		The name of an event. When creating a new deadline, you can select an existing event from this field to connect to a deadline. When the deadline expires, the event you specified in this field is triggered. **Note**: This field only displays the events defined in the procedure and the events have no links to the left hand side (the events that start up an entire branch of a case).
Add		Click this button to add a definition of a case deadline.
Update		Click this button to update a definition of a case deadline.
Delete		Click this button to delete a definition of a case deadline.
Deadlines Are Defined By	Period	Selecting this field enables you to set a period as a case deadline.
	Expression	Selecting this field enables you to set the date and time expressions as a case deadline.

Field	Description
If the Period radio button is selected, the following fields are available.	
Deadline Period	The Deadline Period field contains the following sub-fields. The deadline is calculated at the time the case is started. • **Years** Specifies the number of years when the case deadline expires. • **Months** Specifies the number of months when the deadline expires. • **Weeks** Specifies the number of weeks when the deadline expires. • **Days** Specifies the number of days when the deadline expired. • **Hours** Specifies the number of hours when the deadline expires. • **Minutes** Specifies the number of minutes when the deadline expires.
If the Expression radio button is selected, the following fields are available.	
Deadline Expression	The Deadline Expression field contains the date and time expressions. The expressions are evaluated at the time the case is started. • **Date Expression** Specifies a deadline date after which the event that is set in the Event Name field will be triggered. The deadline date can be specified either as a date expression or as a date type field. • **Time Expression** Specifies a deadline time after which the event that is set in the Event Name field will be triggered. The deadline time can be specified either as a time expression or as a time type field.

Field	Description
Deadline Expression	For more information about how to use the date and time expressions, see "Date and Time Functions" in *TIBCO iProcess Expressions and Functions Reference Guide*. For more information about the date type and time type fields, see "Creating Fields and Forms" in *TIBCO iProcess Modeler Basic Design*. **Note**: If you use the date type field to define the date expression and the time type field to define the time expression: • If the value of the date type field and the value of the time type field are not given in your predefined form, then no deadline is set to the case. • Otherwise: — If the value of the date type field is given in your predefined form, the deadline date is calculated based on this value. If the value of the date type field is not given, the current date will be set as a default value of the deadline date. — If the value of the time type field is given in your predefined form, the deadline time is calculated based on this value. If the value of the time type field is not given, the current time will be set as a default value of the deadline time. For example: — If the value of the date type field is not given for the Date Expression field and the Time Expression field is not set, the current date and time are set as default values of the case deadline. — If the value of the time type field is not given to the Time Expression field and the Date Expression field is not set, the current date and time are set as default values of case deadline.

Creating, Updating, and Deleting Case Deadlines

To set the case deadline property, you may need to create, update, or delete a case deadline:

- Creating a Case Deadline
- Updating a Case Deadline
- Deleting a Case Deadline

Creating a Case Deadline

To define a case deadline, perform the following steps:

You can only create a maximum of nine case deadlines for a procedure in the Deadlines tab of the Properties dialog. In addition, a live case can have 32 or less case deadlines. If the number of case deadlines in a live case exceeds the maximum number, 32, you cannot to create new case deadlines by using the `CreateCaseDeadline` script function. If an attempt is made to do so, then error messages are generated in the `sw_warn` log file.

1. In the Deadline Name field, enter a name for the new case deadline. You must follow the deadline naming rules, see Deadline Name.
2. In the Event Name drop-down list, select the event item you want to link to the deadline.
3. Specify the deadline. Check the **Period** checkbox or the **Expression** checkbox in the Deadlines Are Defined By panel.
   - If the Period checkbox is checked, specify the number of years, months, weeks, days, hours, and minutes for the deadline in the Deadline Period panel.
   - If the Expression checkbox is checked, specify a date expression and a time expression for the deadline in the Deadline Expression panel.
4. Click the **Add** button next to the Deadline Name field to save the definition of the case deadline and then click the **OK** button.

You can also use the `CreateCaseDeadline` script function to create case deadlines in the EAI steps. See "Procedure Functions" in *TIBCO iProcess Expressions and Functions Reference Guide* for more information.

Updating a Case Deadline

To update a case deadline, perform the following steps:

1. In the Deadline Name field, select the deadline name you want to update from the drop-down list.
2. Edit the definition of the deadline based on your request.
3. Click the **Update** button next to the Deadline Name field to save the changes of the case deadline and then click the **OK** button.

You can also use the `UpdateCaseDeadline` script function to update existing case deadlines. See "Procedure Functions" in *TIBCO iProcess Expressions and Functions Reference Guide* for more information.

Deleting a Case Deadline

To delete a case deadline, perform the following steps:

1. In the Deadline Name field, select the deadline name you want to delete from the drop-down list.
2. Click the **Delete** button next to the Deadline Name field to delete the definition of the case deadline and then click the **OK** button.

You can also use the `DeleteCaseDeadline` script function to delete existing case deadlines. See "Procedure Functions" in *TIBCO iProcess Expressions and Functions Reference Guide* for more information.

Dynamically Calculating Case Deadlines

You can reset a case deadline by triggering an event with flag settings in the trigger event command. The event recalculates all the deadlines of a case.

Perform the following steps to recalculate the deadlines of a case:

1. Update the value of the case deadline. Update one or more field values (the date type field and the time type field) used in setting the deadline expressions.

You can recalculate the case deadline only when the date type field and the time type field are used to specify the deadline expressions. You cannot recalculate the case deadline if using a period to specify the deadline.

2. Set a flag and trigger an event to recalculate case deadlines. You can use one of the following methods to process this step:

   — Run the `SWDIR\bin\swutil EVENT` or `SWDIR\util\swbatch EVENT` command. In the command, set the `-r`, `-R`, `-pR`, or `-pr` option as the flag for recalculating deadlines.

   See *TIBCO iProcess swutil and swbatch Reference Guide* for more information.

   — Run the `TRIGGEREVENT` function. In the function, set the value of the `options` variable to `8`, `4`, `6`, `or 10` as the flag for recalculating deadlines.

   See *TIBCO iProcess Expressions and Functions Reference Guide* for a detailed description of this function.

   — iProcess Objects. See the relevant *iProcess Objects Programmer's Guide* and help system for more information.

   — SSO. See *TIBCO iProcess Server Objects (Java) Programmer's Guide* for more information.

   — SSOLite. See the appropriate *TIBCO iProcess Engine (Database) Administrator's Guide* for more information.

Case Deadline in Database

The definitions of the case deadlines are recorded in the `proc_deadline` table. When a case is started, the information of the case deadline is saved in the `case_deadline_event` table and the name of the outstanding step is saved in the `outstanding_addr` table.

If a live case is closed or purged, the records that are related to the case deadline will be deleted from the `case_deadline_event` and `outstanding_addr` tables.

For more information about the tables, see the appropriate *TIBCO iProcess Engine (Database) Administrator's Guide*.

Index

A

C

D

E

I

L

M

N

O

P

R

S

T

U

V

W