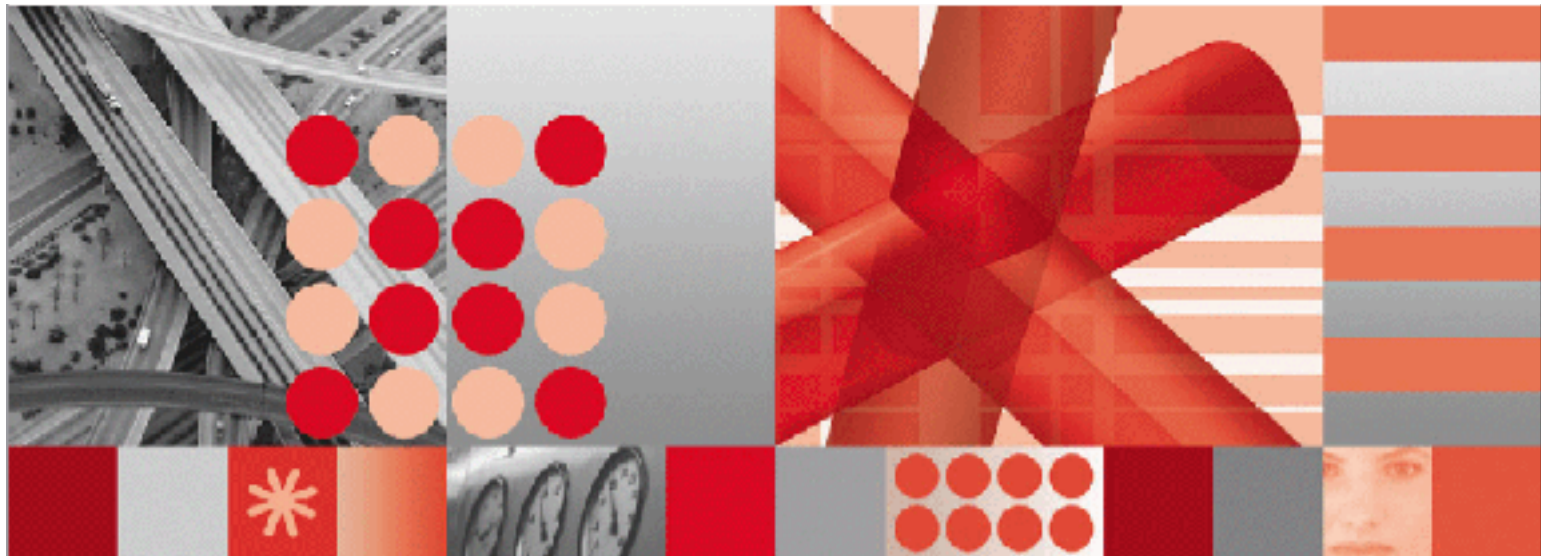


**Tivoli.**



IBM Maximo Asset Management 7.1  
IBM Tivoli Asset Management for IT 7.1  
IBM Tivoli Change and Configuration Management Database 7.1.1  
IBM Tivoli Service Request Manager 7.1



## Application Developer Guide

**Note**

Before using this information and the product it supports, read the information in "Notices" on page 139.

This edition applies to version 7, release 1, modification 0 of IBM Maximo Asset Management, IBM Tivoli Asset Management for IT, and IBM Tivoli Service Request Manager, and version 7, release 1, modification 1 of IBM Tivoli Change and Configuration Management Database and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright International Business Machines Corporation 2008. All rights reserved.

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

# Contents

<b>About This Publication.</b> . . . . .	<b>iii</b>
<b>Chapter 1: What Is the Application Designer?</b> . . . . .	<b>1</b>
The User Interface Architecture. . . . .	1
Controls Available in the Application Designer . . . . .	2
Presentation XML . . . . .	3
User Interface Control Layout . . . . .	5
<b>Chapter 2: Before Using the Application Designer.</b> . . . . .	<b>7</b>
Setting Up an Environment . . . . .	7
Usage Rules and Recommendations. . . . .	7
<b>Chapter 3: Common Application Developer Tasks</b> . . . . .	<b>9</b>
How to Add a Value List to a Field. . . . .	9
Creating Application Restrictions on Applications . . . . .	11
Duplicating an Application . . . . .	12
Creating an Application . . . . .	13
Laying Out a New Tab . . . . .	14
Deleting a User-created Application. . . . .	15
Controlling the Size of Fields. . . . .	15
Optimizing the Placement of Input Fields for Data Entry. . . . .	16
Adding Buttons and Selecting Action Menu Items . . . . .	16
Defining Signature Options for an Application . . . . .	17
Assigning and Overriding the Default Label Value for a Textbox Control . . . . .	18
Assigning a Value to a Default Label Field . . . . .	18
Overriding a Value in the Default Label Field. . . . .	19
Defining Default Values for Fields . . . . .	19
Defining a Default Filter for a Table . . . . .	20
Modifying Application Dialogs. . . . .	21
Moving Controls Across Tabs . . . . .	21
Moving Controls Across Applications . . . . .	21
Moving User Interface Configurations Across Servers . . . . .	22
Exporting Multiple Applications. . . . .	23
Linking to an External URL . . . . .	23
Linking to an Application from a Text Field . . . . .	23
Adding Messages to the Messages Table. . . . .	24
Modifying Contents of the Go To Menu . . . . .	24
<b>Chapter 4: Using Linked Objects and Crossover Domains.</b> . . . . .	<b>25</b>
Database Relationship and Crossover Domain Examples. . . . .	25
Linked Objects and Crossover Domain Scenario . . . . .	26
Using Linked Objects to Display Information in an Application . . . . .	27
Using Crossover Domains to Display Information in an Application . . . . .	34
<b>Chapter 5: Conditional User Interface</b> . . . . .	<b>41</b>
Configuring Conditional Control Properties. . . . .	41
Showing or Hiding Controls . . . . .	42
Showing or Hiding a Control Unconditionally . . . . .	42
Showing or Hiding a Control in a Condition for a Role or Group . . . . .	43

Showing or Hiding a Control in a Condition for All Users .....	44
<b>Appendix A: List of Signature Options.....</b>	<b>47</b>
Common Signature Options .....	47
<b>Appendix B: Controls Reference Information .....</b>	<b>51</b>
Appbar .....	51
Attachments .....	52
Blank Line .....	52
Breadcrumbs.....	53
Button Group .....	56
Checkbox.....	57
Combobox.....	60
Data Source.....	63
Default Value .....	66
Help Grid .....	68
HyperLink.....	69
Image.....	72
Include.....	74
Listbox .....	75
Menubar .....	78
Multiline Textbox.....	78
Multipart Textbox .....	82
Parameter Value.....	87
Parameter Values.....	89
Pushbutton .....	91
Radio Button.....	93
Radio Button Group .....	94
Record Image .....	97
Section .....	99
Section Column .....	104
Section Header .....	105
Section Row .....	107
Static Text .....	108
Tab .....	109
Tab Group.....	112
Table.....	112
Table Column.....	119
Textbox .....	126
Tree .....	131
Tree Attribute.....	134
Tree Node .....	135
<b>Notices .....</b>	<b>139</b>
<b>Index.....</b>	<b>143</b>

# About This Publication

This guide provides a starting point for users who want to use Application Designer to create new applications or modify existing applications.

## Intended Audience

This guide is intended to help any user involved with configuring applications. All users should read the online help content for Application Designer before performing any of the steps outlined in this guide.

## Intended Audience

# What Is the Application Designer?

# 1

The Application Designer is a tool that lets you configure the user interface for applications to meet your specific business needs.

The easy-to-use Application Designer interface enables you to change the appearance of applications without editing a line of code. The drag and drop functionality built into the Application Designer greatly simplifies the process of using this tool.

The Application Designer lets you perform the following common modifications to the user interface of a selected application:

- ▼ Moving fields and sections
- ▼ Adding new fields, tabs, and tables with columns and rows

You can perform these more advanced functions with the Application Designer:

- ▼ Creating or duplicating applications
- ▼ Defining signature options
- ▼ Editing the Toolbar or the Select Action menu.

You can create an application or duplicate an existing application with the Application Designer. But you might need to perform the following additional activities to fully enable a new application:

- ▼ Create database tables and columns
- ▼ Define user access to the application
- ▼ Develop Java code
- ▼ Develop workflow processes

## The User Interface Architecture

A description of the user interface (UI) and general architecture of the Application Designer is covered in the following topics:

- ▼ Controls available in the Application Designer
- ▼ Presentation XML (Extensible Markup Language)
- ▼ Modifying the presentation XML
- ▼ Exporting and importing presentation XML files
- ▼ Editing presentation XML files
- ▼ User interface control layout

## Controls Available in the Application Designer

Applications use "controls," numerous pre-fabricated elements like tables and textboxes, to build the user interface. The individual controls are part of the code and cannot be modified. Some types of controls, like sections, act as containers for other types of controls, such as tab groups, textboxes, or static text. The behavior of each control depends on its binding to a MBO attribute. Attributes store values from the database that are referenced in a control.

The following table lists the controls available in the Application Designer.

Control Name	Description	See...
AppBar	A container that displays all the applications, belonging to the current module, except for the currently displayed application	page 51
Attachments	An attachments icon that allows you to attach different types of information to a record, including images, spreadsheets, and Web pages	page 52
Blank Line	Inserts white space in a section	page 52
Breadcrumb	Navigation that tracks the steps you take in an application and facilitates your ability to retrace steps quickly	page 53
Button Group	A container for adding buttons	page 56
Checkbox	An HTML check box	page 57
Combobox	A combo box with drop-down functionality	page 60
Data Source	A miscellaneous type of control mostly used in dialogs	page 63
Default Value	Forces a default value for a control that resides within a container	page 66
Help Grid	Help text displayed in windows	page 68
HyperLink	A hyperlink that supports both text and graphics	page 69
Image	Allows you to insert an image	page 72
Include	Obtains dialog definitions and common application functions like header and footer from the library.xml file.	page 74
Listbox	An open, drop-down menu that displays dynamic content	page 75
Menubar	A menu bar that contains signature options for the Search, Query, and Bookmark menus	page 78
Multiline Textbox	A text input box with multiple lines	page 78
Multipart Textbox	Two text boxes next to each other; one for displaying text and the other a description	page 82
Parameter Value	A container for the parameter value control to be used in sections and table windows	page 87



Control Name	Description	See...
Parameter Values	This control contains the parameter values control which is a container for the parameter value control	page 89
Pushbutton	Individual buttons that belong to the button group	page 91
Radio Button	An individual radio button that belongs to the radio button group	page 93
Radio Button Group	A container for radio buttons	page 94
Record Image	Associates and displays an image with a record	page 97
Section	A container for adding controls such as section columns, section rows, fields, buttons, combo boxes, and check boxes	page 99
Section Column	A column in a section	page 104
Section Header	A container to label a section or a table window, with the capability to display dynamic values in its label	page 105
Section Row	A row in a section	page 107
Static Text	Informational text displayed in an application	page 108
Tab	An individual tab in a tab group	page 109
Tabgroup	A container for adding tabs	page 112
Table	A container for a table window	page 112
Table Column	A container for a column in a table window	page 119
Textbox	A text input box with a single line	page 126
Tree	A container for a hierarchical structure	page 131
Tree Attribute	A value to describe the tree node control within the tree control	page 133
Tree Node	A container for defining the child elements within the tree control	page 134

## Presentation XML

The system maintains information about the controls used to compose a user interface in XML format, and stores this information in the database. The system assigns a unique ID to each control used in an application, and the Application Designer accesses these controls by referencing their unique IDs.

When you run MAXINST to create the database, the install program places the presentation XML content in the MAXPRESENTATION table.

The presentation XML for every application has a tag for each control it uses to build the user interface. Each tag has a series of attributes, and their values

determine how the control behaves when you view the application. The location of the tags in the presentation XML for every application relative to other tags determines the order and sequence of the user interface elements that you see in a screen.

When you access an application, the system retrieves the presentation XML for the application from the database and stores it in the memory of the application server. For each control referenced in the XML, the user interface framework retrieves code for that control. Based on the values of the attributes passed to it in the XML, the system builds the HTML description of the element that the control represents in memory. The application server holds all tags in memory while the system incrementally builds the complete HTML description for a screen. Once the build is complete, the application server passes the HTML from the application server to the client (the user's Web browser). The XML remains in the server cache and waits for the next time a user accesses the application.

### Modifying the Presentation XML

When you display an application in the Application Designer, the system loads its presentation XML in memory. When you edit the application, the Application Designer edits the XML in memory to reflect the changes you make.

When you save an application, the application server saves the XML in memory to the database. Additionally, when you next use that application, the server loads the XML from the database and renders the application in a browser window.

### Exporting and Importing Presentation XML Files

Use the Application Designer to export XML presentation information to a physical file that you can save. This feature enables you to transport application screens from one environment to another, such as from development to production.

Exporting an XML presentation to a file also lets you edit the XML directly with a text editor rather than with the Application Designer. For example, a title modification that applies to all screens is easier to do with the search and replace functionality of a text editor rather than editing each application window using the Application Designer as the user interface editor.

**NOTE** If you manually edit an XML file, you must use Import File Definitions to load that file to the database for changes to take effect.

Limit your XML edits to simple changes, such as modifying labels, to ensure that changes migrate with subsequent application upgrades.

### Editing Presentation XML Files

Sometimes, editing the XML file of a presentation is more efficient than using the Application Designer to make changes. In some cases, the only way to make a change to a presentation is to edit the XML code.

Most applications use a single page, which can contain several tabs. Some applications, such as View Requisitions, use hidden tabs. Each hidden tab contains its own page to display the user interface. These pages can also contain tabs that are not hidden.

For applications that contain hidden tabs, the Application Designer displays only the first tab of the set of hidden tabs. To edit the other hidden tabs in the

application, you must export the presentation XML for the application, edit the file, and import the changed file.

To edit presentation XML files:

- 1 In the Application Designer, open the application for which you want to edit the XML code.
- 2 Click **Export Application Definition**.
- 3 Save the XML to a local directory.
- 4 Edit that file with a text editor and save it.
- 5 In the Application Designer, click **Import Application Definition**.
- 6 Specify the location of the modified XML code file.
- 7 Click **OK**.

## User Interface Control Layout

The positioning of controls in the user interface is completely relative. The user interface is neither pixel based nor is it grid based. There are no set locations where controls can live on a screen. The exact placement location of a control is relative to other controls on the screen. The relative aspect of the spacing also provides for dynamic sizing of the width of a section and its columns. For example, a section that contains fields of various widths will automatically size itself relative to the widest field in that section.



# Before Using the Application Designer

# 2

This section of the document provides pointers about setting up a client machine and the general system environment before you use the Application Designer.

## Setting Up an Environment

Before you use the Application Designer, please consider the following recommendations:

- ▼ To optimize the amount of information you can view with your monitor without scrolling, set your resolution to 1280x1024.
- ▼ A higher resolution makes it easier to keep windows, like the Control Palette and Control Properties, outside of the configurable window area of the application.
- ▼ Employ three separate environments — development, staging, and production— for designing, testing, and deploying changes made with the Application Designer.

For example, the development environment can be on your own dedicated workstation where you can make configuration changes using the Application Designer. The staging environment can be a server machine where you merge and test individual configurations created in the development environment. After staging, migrate the changes to your production environment so that they are available to all users.

If you employ a multi-stage model, it insulates your production environment from configuration changes made in development and staging.

## Usage Rules and Recommendations

Familiarize yourself with the following rules and recommendations when you use the Application Designer:

- ▼ It is recommended that only one user edit an application on a particular server.
- ▼ Create the data source and associated objects for a new application in the Database Configuration application before you create screens with the Application Designer.
- ▼ You can only delete user-created or duplicated applications. You cannot delete applications that are shipped with the product.

## Usage Rules and Recommendations

- ▼ When you drag a control from the palette onto the workspace, the Application Designer positions the control to the top and left of the target control. Target controls are highlighted with a yellow background color.
- ▼ You can drag most controls into container type controls, such as a section or table.
- ▼ The package key you use when you perform the installation determines what applications you can edit with the Application Designer.

# Common Application Developer Tasks

# 3

There are many common tasks that an application developer will be required to perform. Such tasks include creating and deleting applications, changing the appearance and controls of an application, defining signature options, and so on. Procedures for some of the more common application developer tasks are presented in the following sections.

## How to Add a Value List to a Field

The system uses domains to limit the choice of values that you can enter in a field or to restrict the values that can display in a list. To add a domain to a field, you need to perform the following tasks:

- ▼ Use the Domains application to create a new domain.
- ▼ Use the Database Configuration application to create a new attribute and to associate the new domain with it.
- ▼ Configure the database.
- ▼ Use the Application Designer to add a textbox with a lookup list to the user interface and to bind the textbox to the attribute you created in the Database Configuration application.

This example shows how to use the Application Designer to add a textbox control to the Assets application with a lookup that lets you identify your company's assets as being in a state of: ACTIVE, PENDING, or INACTIVE.

- 1** In the Domains application, click **Add New Domain > Add New ALN Domain**.
- 2** Enter `Assetstatus` in the Domain field, `Asset Status` in the Description field, `Upper` in the Data Type field, and `8` for the Length.
- 3** In the ALN Domain window, click **New Row**, and enter `Active` in the Value and Description fields.
- 4** Repeat Step 4, and enter values of `Ending` and `Inactive`.
- 5** Click **OK**, and then click **Save**.
- 6** In Database Configuration, type `Asset` in the Object field, and press **Enter**.
- 7** Click **Asset** to open the object record.

- 8** Click the **Attributes** tab, and then click **New Row**.
- 9** Enter `State` in the **Attribute** field, and then enter the following values:
  - ▼ `Upper` in the **Type** field
  - ▼ `8` in the **Length** field
  - ▼ `Activity State` in the **Title** field
  - ▼ `Inactive` in the **Default Value** field
  - ▼ `ASSETSTATUS` in the **Domain** field
- 10** Click **Save**.
- 11** To save the changes to your database:
  - a** Sign out of the system.
  - b** Shut down the application server (`Ctrl + C`), and then wait a minute.
  - c** Open a command shell (**Start > Run > cmd**), and navigate to the `<root>\tools\maximo` directory.
  - d** Run `configdb.bat`.
  - e** Refer to the *Maximo Asset Management 7.1 Database System Administrator's Guide* for additional details on configuring the database.
  - f** Restart the application server and login to the system.
- 12** In the **Application Designer**, type `Asset` in the **Application** field, and press **Enter**.
- 13** Click **Asset** to open the **Assets** application in the **Application Designer** workspace.
- 14** Click the **Asset** tab.
- 15** Click **Control Palette**.
- 16** Click the **Section** label that holds the **Asset** and **Status** fields. The section background turns green.
- 17** Drag and drop the **textbox** control on top of the **Asset** field.

Notice that the input box displays "Invalid Binding" because the control is not bound to a data field.
- 18** Select the **textbox** control and click **Control Properties**.
- 19** Type `State` in the **Attribute** field, and press **Tab**.

Note that the text box field becomes bound and its label changes to **Activity State** (which is the description you entered when you first created the attribute).



**20** Type `ValueList` in the Lookup field, and press **Tab**.

Note that the Application Designer adds a lookup icon next to the field so that Activity State will now show a list of values (e.g., ACTIVE, PENDING, and INACTIVE) during lookup.

**21** Click **Save**.

**22** Open the Assets application and open an asset in the Asset tab to view your changes.

The preceding example is a generic way to create simple value lists. You can view more application-specific value list presentations in `lookups.xml` in the `<root>\resource\presentations\system` directory. You can add more or modify the existing value lists by editing the contents of this file with a text editor.

If you edit `lookups.xml` or any other presentation-related XML file, you will need use **Import Application Definition** to import the presentation file into the database. The **Import Application Definition** icon is available from both the Applications and Workspace tabs of the Application Designer.

## Creating Application Restrictions on Applications

You can use the Application Designer to define restrictions on what records appear in an application. You can enter a query in the presentation XML of an application to limit what data a user can see in that application. You can also limit the records that an application shows when it uses the same data source that other applications use, but not all of the data is applicable to all of the applications. The following procedure is presented as an example.

To define restrictions on a power application:

- 1** In the Application Designer, open the Assets application in the Workspace tab.
- 2** Select the List Table control.
- 3** Click **Control Properties** to open the Table Properties window.
- 4** In the Application Restrictions field, type

```
"status != 'DECOMMISSIONED'"
```

the Application Designer interprets the query as

```
select * from asset where status != 'DECOMMISSIONED' .
```

- 5** Click **Save**.

When you open the Assets application, you will see all the records that do not have a status of "DECOMMISSIONED".

To define an application restriction on a single-page application, you must export the presentation XML for that application, modify the XML in a text editor and

## Duplicating an Application

import the file to the Application Designer. The following procedure is an example.

To change a presentation by editing the presentation XML:

- 1 In the Application Designer, open the Asset application.
- 2 Click **Export Application Definition**. A new window shows the XML code for that application.
- 3 Save the XML code to a local file.
- 4 Edit that file with a text editor and save it.

Change the `apprestrictions` property on the presentation tag in the XML to the following text:

```
<presentation id="myapp" mboname="ASSET" apprestrictions="STATUS  
!= 'DECOMMISSIONED'" version="7.1.0.0">
```

- 5 In the Application Designer, click **Import Application Definition**.
- 6 Specify the location of the modified XML code file.
- 7 Click **OK**.

See “Editing Presentation XML Files” on page 4 for more information about how to change the presentation XML.

## Duplicating an Application

You can use the Application Designer to duplicate an existing application. If you duplicate an application, you create a copy of the application presentation and the application definition.

To duplicate an existing application:

- 1 In the Application Designer, open the application you want to duplicate in the Workspace tab.
- 2 Open the Duplicate Application window by choosing **Select Action > Duplicate Application Definition**.
- 3 Enter a unique name for the duplicated application in the Application field and a text description in the Description field.

The name you enter in the Description field is what will appear in the navigation bar of the application.

- 4 Click **OK** to open the new application in the Workspace tab.
- 5 Click **Save**.

The new application appears in the same module as the original application. To view the new application, complete the following steps:

- 1 Sign out of the system (This action refreshes the list of applications in the **Go To** menu.)
- 2 Sign in to the system
- 3 Navigate to the new application

## Creating an Application

The Application Designer lets you create three different types of new applications:

- ▼ Power applications
- ▼ Single page applications
- ▼ Self service applications

The system uses a different template for each application type to create the basic user interface structure of the application.

- ▼ A power application is a multi-tab application. Use the power application template to create a power application that has the following characteristics:
  - Multi-tabs, including the standard List and Workspace tabs
  - **Select Action** menu
  - Toolbar buttons and other user interface components contained in a regular application
- ▼ A single page application uses a variation of the template for the power application. There are no tabs, but regular toolbar options are available.
- ▼ A self service application uses a more restrictive template with fewer user interface options. The template is designed for creating a record and not browsing data, so there is no default list table window and no ordinary toolbar.

To create a new application:

- 1 In the Application Designer toolbar, click **New Application Definition**.
- 2 Provide a name and description for the application. The name must be unique within the current database.
- 3 In the Main Object field, enter the name of the primary database table to which you want the application to refer.

Note that the main object you enter here should already exist in the database. Use the Database Configuration application to define new objects.

- 4 In the Module Name field, specify the name of the module that you want this application to belong to, for example assets, inventory, or work orders.
- 5 Every application must reside within a module. You can click **Select Value** to display a list of modules.

- 6 Click one of the buttons at the bottom of the window to select the type of application that you want to create.

---

Standard power application	An application with multiple tabs, a <b>Select Action</b> menu, toolbar buttons, and other components.
Self-service application	An application that is used to create a record. A self-service application does not include the default List Table window or a toolbar.
Single-page application	A non-tabbed application that can include a toolbar and a <b>Select Action</b> menu.

---

(If you want to create a new application to view your audit tables, create it as a single page application.)

The new application opens in the Workspace tab.

Optional: Use the **Control Palette** to add controls such as tabs, buttons, and tables to the new application. You can use **Control Properties** to change the properties of the controls.

- 7 Click **Save Application Definition** to save your new application to the database.

The system sets the default security to Read. All users who have access to the application in their security profile can read records but not insert or delete. You can define additional signature options for a new application using **Add/Modify Signature Options**. Refer to Appendix A for a complete list of signature options.

You can use the the **Control Palette** and **Control Properties** to add or modify controls (such as tabs, buttons, and table windows) in the new application.

To view your new application in the **Go To** menu, sign out and then sign back into the system.

## Laying Out a New Tab

When you create a new application or add a new tab to an existing application, the Application Designer displays an empty screen. This is the starting point to begin laying out the structure and location for all the elements that will make up this new screen.

To begin the layout of a new screen, add a section control. This control designates a space where you can add additional controls. You can break the layout of the page vertically by adding additional section controls.

You can break a section into horizontal segments or columns by adding section column controls to the section control. A section, by default, can not render itself and only acts a container for other controls. You can complete the Section Properties to make a section display a bar and a minimize/maximize button.

Once you create the basic layout of a window, you can begin to add additional controls that display data, such as the textbox control.

## Deleting a User-created Application

You can only delete an application that you created or duplicated in the Application Designer. You must use a database editor (for example, SQL Plus) to delete an application.

**NOTE** You must shut down the application server before you delete an application. If you delete an application, you do not remove any custom tables, fields, or domains that you created to support the application.

To delete an application, open a database editor and complete the following steps:

- 1 Login as an administrator.
- 2 Type the following commands, and replace the variable `<APPLICATION NAME>` with the name of the application you want to delete.

Enter the application name in all uppercase. For example, if the name of the application you want to delete is Test, replace `APPLICATION NAME` with `TEST` in all the commands.

```
delete from maxapps where app='<APPLICATION NAME>';
delete from maxpresentation where app='<APPLICATION NAME>';
delete from sigoption where app='<APPLICATION NAME>';
delete from applicationauth where app='<APPLICATION NAME>';
delete from maxlabels where app='<APPLICATION NAME>';
delete from maxmenu where moduleapp='<APPLICATION NAME>' and menutype
!='Module';
delete from maxmenu where moduleapp='<Module Name>' and
elementtype='APP' and keyvalue='<APPLICATION NAME>';
delete from appdoctype where app= '<APPLICATION NAME>;
```

- 3 Logout of the database editor.

If you execute the above statements, the application will be deleted from the database.

## Controlling the Size of Fields

The system restricts how much of a field appears in the user interface so that fields of varying lengths do not show jagged edges when stacked on top of each other.

The settings are kept in the file:

```
<root>\applications\maximo\maximouiweb\webmodule
\webclient\common\fieldsizegroups.jsp
```

## Optimizing the Placement of Input Fields for Data Entry

For string fields (ALN, UPPER, LOWER), the defined sizes are:

- ▼ If the database size is 1 or 2 characters, then the field length displayed on a screen is 2.
- ▼ If the database size is 3 through 30 characters, then the field length displayed on a screen is 10.
- ▼ If the database size is greater than 30 characters, then the field length displayed on a screen is 40.

**NOTE** There is no specific unit of measure for field lengths displayed on screens.

The code for ALN, LOWER, UPPER field types appears below:

```
fieldtypegroup = new Hashtable();
fieldtypegroup.put(new Integer("2"), new Integer("2"));
fieldtypegroup.put(new Integer("30"), new Integer("10"));
fieldtypegroup.put(new Integer("31"), new Integer("40"));
fieldsizegroups.put(new Integer(FIELD_TYPE_ALN).toString(),
fieldtypegroup);
fieldsizegroups.put(new Integer(FIELD_TYPE_LOWER).toString(),
fieldtypegroup);
fieldsizegroups.put(new Integer(FIELD_TYPE_UPPER).toString(),
fieldtypegroup);
```

**NOTE** If you make any changes to the fieldsizegroups.jsp file, you must rebuild the maximo.ear file

## Optimizing the Placement of Input Fields for Data Entry

When you design the user interface for a predominantly data entry application, it is crucial to order input fields to optimize the use of the Tab key to navigate between fields. Tabbing order goes from top to bottom and from left to right. You can delete fields not needed for data entry on an input screen, or move those fields to another screen.

If you wish to change the position for a large number of data input fields in an application, consider duplicating an existing application and then changing the order of the input fields.

## Adding Buttons and Selecting Action Menu Items

You can add new toolbar buttons and select action menu items to an application by choosing one of the following select actions in the Application Designer:

- ▼ Add/Modify Toolbar Menu
- ▼ Add/Modify Select Action Menu

When adding a new button, the name you enter in the Key Value field will display in the toolbar when you mouse over the icon that represents the action associated with the button, such as Insert or Delete. Additionally, you can specify the following factors:

- ▼ The name of the image file that will represent the button in the toolbar
- ▼ The position of the button in the toolbar
- ▼ Which tabs will display the new toolbar button

The steps to add a new **Select Action** menu item are very similar to adding a new button, and both share the same window for entering attributes.

## Defining Signature Options for an Application

The Application Designer enables you to add or modify signature options for a new or existing application. For more information on signature options, which specify privileges for using applications, menu options, and toolbar items, see Appendix A, "List of Signature Options," on page A-47.

In the system, you can link any user interface control to a signature option and define restrictions on which user interface controls appear for a group of users.

A Signature Option is a record in the SIGOPTION table that the system uses to configure security. All applications have a set of SIGOPTIONS (such as READ, INSERT, SAVE, DELETE, CHANGESTATUS) that determine standard behavior. You can add SIGOPTIONS to applications to control additional behavior, such as new actions in the menu or special behavior for controls.

An administrator grants SIGOPTIONS to Security Groups to provide application access. Every application has a READ SIGOPTION which is granted to provide basic application access; if a group does not have READ for an application, the group cannot use anything in that application.

**TIP** You must create the association of a control with a signature option and the configuration of conditional properties in the Application Designer application.

When a control is bound to a signature option or an attribute, you can configure the behavior of the control in the following ways:

- ▼ Set a data restriction in the Security Groups application
- ▼ Grant the option on the Application tab
- ▼ Configure conditional properties within the Application Designer

To define signature options:

- 1 In the Application Designer, open an application in the Workspace tab.
- 2 Choose **Select Action > Add/Modify Signature Options** to open the Signature Options window.
- 3 Click **New Row**.
- 4 Enter a valid signature value, for example `Insert` or `Save`, in the Option field.

## Assigning and Overriding the Default Label Value for a Textbox Control

- 5 Enter a description for the signature option in the Description field.
- 6 You can enter additional signature options in the Option Also Grants field. The system will automatically grant these signature options along with the primary signature option.

For example, if you add the Insert option to the **Select Action** menu, you may also want to automatically grant CANCEL and DELETE.

- 7 In the Option Also Revokes field, enter any options that will be automatically revoked as a result of adding the primary option.

Use commas to separate multiple entries in the Option Also Grants and Option Also Revokes fields.

- 8 Click OK.

## Assigning and Overriding the Default Label Value for a Textbox Control

All controls that have a text field component, such as Textbox or Multiline Textbox, have two label fields: Default Label and Label. Both of these fields, which appear in **Control Properties**, are blank when you first drag a textbox control into an application. The field name that displays in the user interface is the name for the control. For example, if you drag a Multiline Textbox into an application, its name becomes "multilinetextbox...".

A textbox control acquires a Default Label when you link the control to an object attribute. This Default Label displays in the user interface and is read only. However, you can override the Default Label value for a control by entering a different value in the Label field. The value in the Label field overrides the Default Label name inherited from the object attribute. This feature enables textbox controls linked to the same object attribute to have different labels in different applications. The system stores the label values for textbox fields in the MAXLABELS table.

## Assigning a Value to a Default Label Field

Complete the following steps to assign a value to the Default Label for a textbox control.

- 1 In the Application Designer, open an application in the Workspace tab, for example, the Assets application.
- 2 Click the Asset tab, and select a section control.
- 3 Click **Control Palette**.
- 4 Drag the Textbox control into the highlighted section.
- 5 Select the Textbox control, and click **Control Properties**.



- 6 In the Textbox Properties window, click **Lookup** for the Attribute field and select ORGID.

Notice that the field label in the workspace and the Default Label field in the window change to Organization, which is the value in the Title field for the ORGID attribute.

**NOTE** You define the value of the Title field in the Database Configuration application. All textbox controls that bind to the ORGID attribute, for example, inherit the value of `Organization` in the Default Label field.

- 7 Click **Save**.

## Overriding a Value in the Default Label Field

Complete the following steps to override the Default Label value for a textbox control.

- 1 In the Application Designer, open an application in the Workspace tab, for example, the Assets application.
- 2 Click the Asset tab and select a textbox control.
- 3 Click **Control Properties**. The Textbox Properties window opens.
- 4 In the Textbox Properties window, enter a new name for the textbox control in the Label field. If the Default Label has a value of `Organization`, for example, you may want to enter a different, but similar value in the Label field, such as `Org ID`.

Notice that the field label in the workspace for the textbox control changes to "Org ID."

- 5 Click **Save**.

## Defining Default Values for Fields

You can specify a default value for a field by entering that value in the Default Value field of the attribute to which the field is bound. You specify a default value in the Details section for the attribute in the Database Configuration application. For example, if you want the `glaccount` attribute in the `WORKORDER` table to default to a specific account number, then enter that number in the Default Value field. The GL Account Number you entered will appear as the default value in all fields that are bound to the `glaccount` attribute.

Alternatively, you can define a field default value that is local to an application by using the Default Value control.

To define a field default value using the Default Value control:

- 1 In the Application Designer, open an application in the Workspace tab.
- 2 Open the **Control Palette**.

## Defining a Default Filter for a Table

- 3 Drag the Default Value control into the section that contains the field for which you want to assign a default value.

If you do not see the Default Value control in the section, click **Select Action > Toggle Show All Controls**.

- 4 Open **Control Properties** for the control, and perform the following actions:
  - a Enter the name of the attribute bound to the field to which you are assigning the default value.
  - b Enter the desired default value in the Value field.
  - c Select Insert for the Default Type

- 5 Click **Save**.

## Defining a Default Filter for a Table

You can use the Application Designer to define a default query for a table window using the Default Value control. When you first open the table, the system uses the query value or values you define to filter the records displayed from a table. For example, if you query the attribute `ITEM_DESCRIPTION` and specify a value of `MOTOR`, then the table only displays item records that have `MOTOR` somewhere in the description. You can also use `%` as a wildcard and `=` for exact match. If you set the value to `=MOTOR`, the system displays item records with only `MOTOR` in the item description.

To define a default filter for a table:

- 1 In the Application Designer, open an application in the Workspace tab.
- 2 Drag the Default Value control into the section that contains the table for which you want to define a default filter.

If you do not see the Default Value control in the section, click **Select Action > Toggle Show All Controls**.

- 3 Open **Control Properties** for the control, and enter the following information:
  - ▼ The name of the attribute of the column to which you are assigning the default query
  - ▼ The desired value in the Value field (use a comma as a delimiter if you filter based on multiple values)
  - ▼ Select Query for Default Type
- 4 Click **Save**.

# Modifying Application Dialogs

You can use the Application Designer to edit application-specific windows that the system stores within the XML presentation of the application.

To modify application dialogs:

- 1 In the Application Designer, open an application in the Workspace tab.
- 2 Click **Edit Dialogs** to open a list of windows specific to the application.
- 3 You can edit the window content in the same way that you would edit any other user interface element in the workspace.
- 4 The user interface also provides general purpose windows that are used by more than one application. To edit these windows, such as the Change Status window, you must use a text editor to update the library.xml file located in the following directory:

```
<root>\resources\presentation\system
```

- 5 If you edit library.xml or any other presentation type XML file, you must import the updated file into the database by using **Import Application Definition** in the Application Designer. The import functionality is available on both the Applications and Workspace tabs.

## Moving Controls Across Tabs

Use **Copy** or **Cut and Paste** to move a control or a section grouping of controls from one tab to another within the same application. Select the individual control or a section control that contains other controls, and click **Copy** or **Cut**. Click the desired tab, and then select the target control (usually a section). Click **Paste**.

## Moving Controls Across Applications

You can edit the XML presentation code and use **Import Application Definition** and **Export Application Definition** of the Application Designer to move an individual control or a section container of controls across applications.

To move controls across applications:

- 1 In the Application Designer, open the source application in the Workspace tab.
- 2 Click **Export Application Definition**.
- 3 Save the XML definition to a file.
- 4 Open the destination application in the Workspace tab.

- 5 Repeat Steps 3 and 4.
- 6 Open the saved source and destination XML files in a text editor.
- 7 In the XML of the source file, navigate to the section ID or the individual control ID that you want to copy. Select and copy the line or lines of code.  
  
Use the unique ID of a control that is located in the Control Properties window of the Application Designer to simplify the search in the XML presentation files.
- 8 Open the destination file and search for the location where you want to copy the code from the source file. For example, type `Ctrl+V` or choose **Edit > Paste** in a Windows environment.
- 9 Save both XML files.
- 10 In the Application Designer, open the destination application in the Workspace tab.
- 11 Click **Import Application Definition**, and enter the destination XML file name with your edit.
- 12 Click **OK**.

To view your change, click the tab in the destination application that contains the XML code.

**NOTE** When you copy a control like a Multipart Textbox to a different application, the control must link to an attribute that exists in the main object table for both applications to view data from the database. If the attribute relationship does not exist, the Multipart Textbox control that was copied to the source destination application does not contain any property values.

## Moving User Interface Configurations Across Servers

You can use **Export Application Definition** and **Import Application Definition** in the Application Designer to move your user interface configurations to other servers. These features enable you to easily move applications between multiple environments, for example testing and production, while different people in your organization configure individual applications.

If you or someone in your organization wants to update the user interface configuration for an application, it is good practice first to export the existing user interface XML file or files to a backup directory. If you want to restore this instance of the user interface in the future, use **Import Application Definition**. When you import an application, you are overwriting the application in your database and not in the presentations directory.

You can locate presentation XML files in the `<root>\resources\presentations` directory. These files are for illustration and reference. Do not modify them. However, you can copy one of these files, save it with a new name, and then make

modifications. You can then import the modified file into the system using the Application Designer.

## Exporting Multiple Applications

You can copy multiple applications at the same time to another server by exporting the XML presentations to a single file.

To export multiple applications:

- 1 In the Application Designer, go to the Applications tab and click **Select Records**.
- 2 Check each application whose XML presentation you want to export.
- 3 Click **Export Application Definition**.
- 4 A new browser window opens a single XML file with the presentation code for each application you exported.

Note that **Import Application Definition** reads the beginning and an end point for each application embedded in the XML file and loads the code appropriately. You can find the start point for an application by searching for the "presentation id" text string.

- 5 Save the XML file to a directory.

## Linking to an External URL

You can create a link to an external Web site from an application using the HyperLink control. Drop the HyperLink control into a section or table container, and then enter a valid URL for the destination site in the URL field of the Properties window for the HyperLink control .

When you click the URL link within the application, the system opens the destination site in a new browser window.

## Linking to an Application from a Text Field

Many fields in applications already contain GOTO links to other applications, but you can create your own link by defining the GOTO Application and Menu Type fields of certain controls, such as Multipart Textbox, Textbox, and Table Column.

To link to an application from a text field:

- 1 In the Application Designer, open an application in the Workspace tab.
- 2 Select a textbox control and click **Control Properties**.
- 3 Enter values in the GOTO Application and Menu Type fields.

## Adding Messages to the Messages Table

You can locate the Menu Type ID values in menus.xml, which is located in <root>\resources\presentations\system.

For example, if you select the Asset field in Work Order Tracking, the Properties window displays a value of `Asset` in the GOTO Application field and a value of `ASSETMAIN` in the Menu Type field.

## Adding Messages to the Messages Table

You can add messages to the MAXMESSAGES table to address customized content. The new messages are displayed when the application encounters errors. These messages are displayed in logs to aid in diagnosing and debugging issues with applications. The new messages are also available for translation into various supported languages.

To add messages to the Messages table:

- 1 In the the Application Designer, open an application in the Workspace tab.
- 2 Select **Messages** from the **Select Action** menu.
- 3 In the Messages window, click **New Row**.
- 4 In the Message Group field, type the identifier for the message group that you created to contain this message.
- 5 In the Message Key field, enter a unique message identifier for the message. The message key must be unique in the message group.
- 6 In the Message ID field, select a message ID from the Select Value list.
- 7 In the Value field, type the new message.
- 8 Complete all additional relevant sections and click **OK**.

## Modifying Contents of the Go To Menu

The Application Designer does not currently allow you to customize the **Go To** menu for an application. However, when you create an application, you can specify in which module you would like the new application to appear. When you duplicate applications, the new application is available in the same module as the original application.

# Using Linked Objects and Crossover Domains

# 4

Linked objects and crossover domains represent different configuration mechanisms that you can employ to move data from one field in an application to another field in a different application. The term “source” refers to an originating field or application. The term “destination” refers to a target field or application.

Although similar in purpose, there are distinct differences as to how these configuration mechanisms transfer data across applications. Linked objects, for example, can pass data from a source application to a target application using database relationships. In the destination application, the data value can be read only or read/write depending on how you configure the input field. If you configure the destination input field to be read/write, then the system automatically updates the source application and parent object if you update a displayed value.

Crossover domains, alternatively, pass data from one application to a different application using domains. The system passes a copy of the data value from a source application to a destination application using the crossover domain as a trigger mechanism. You cannot edit the data value in the destination application. The copied data value is essentially static.

## Database Relationship and Crossover Domain Examples

Database relationships that link objects and crossover domains let you export information from a defined source object to a different destination object using List and Validation Where Clauses to filter values. You can, for example, use a database relationship or crossover domain to accomplish the following:

- ▼ Enter an asset number in the Items application and have the system retrieve the serial number of the asset from the ASSET object and copy it into a field.
- ▼ Enter an asset number for a new work order and have the system retrieve the parent asset number from the ASSET object and insert it into a field.
- ▼ Enter the user name for a person creating a material receipt record and have the system retrieve the name of the user’s supervisor from the PERSON object and insert it into a field.
- ▼ Enter a user name in an application and have the system retrieve the user’s phone number and address from the PERSON object and insert it into two different fields.

- ▼ Enter an asset number for a new work order and have the system retrieve "comments" about the item from the ASSET object and copy them to a textbox field in the work order.

You can configure an existing field or you can create an entirely new field or fields in a destination application using the Application Designer to display information that derives from linked objects or a crossover domain. If you decide to add fields that do not have a corresponding database attribute, you must add the new attributes to the source and destination objects using Database Configuration.

## Linked Objects and Crossover Domain Scenario

Because linking objects and creating crossover domains have similar objectives, you must first decide what it is you want to accomplish from a business practice before you choose one method over the other. For example, you decide that you want to add a field to the Asset application that holds important comments about an asset. A comment field is not a description but a remark similar to "John Doe is monitoring this motor. Please contact him before you perform any maintenance." Further, you want the comments field to display in the work order window whenever you add an asset to a work order.

You have two options:

- 1 Make "Comments" a read-only field in Work Order Tracking. It is a "reference only" field that users cannot update in the work order screen. Similarly, make "Comments" a read/write field in Work Order Tracking. Users can update the field in the work order screen, which will automatically update the comments attribute in the ASSET object.
  - You can use linked objects to accomplish Option 1.
- 2 Copy the comments field value from the ASSET object to an editable field on the work order. Any changes you make to the comments field in the work order will not update the comments attribute. Similarly, any subsequent changes to the comments attribute in the ASSET object are not passed to the comments field in the work order.
  - You can use a crossover domain to accomplish Option 2.

The following table summarizes the actions you must perform to complete the two different scenario options.

Option 1 (linked objects)	Option 2 (crossover domain)
1) Add assetcomment attribute to the ASSET object in Database Configuration	1) Add assetcomment attribute to the ASSET object and woassetcomment attribute to the WORKORDER object in Database Configuration
2) Configure your database	2) Configure your database



Option 1 (linked objects)	Option 2 (crossover domain)
3) Add new relationship to the WORKORDER object in Database Configuration (asset2wolink)	3) Create new crossover domain in Domains (asset2wo)
4) Add textbox control for assetcomment attribute to Assets in the Application Designer. Set Label to Comments.	4) Add crossover domain to workorder.assetnum (the "trigger field") in the WORKORDER object in Database Configuration
5) Add new asset in Assets and enter "John Doe is monitoring this motor" in the Comments field	5) Configure your database
6) Add textbox control for <database relationship name>.assetcomment attribute to Work Order Tracking in the Application Designer	6) Add textbox control for assetcomment attribute to Assets in the Application Designer. Set Label to Comments.
7) Configure Input Mode for <database relationship name>.assetcomment to READONLY  (OR)	7) Add new asset in Assets and enter "John Doe is monitoring this motor" in the Comments field
8) Configure Input Mode for <database relationship name>.assetcomment to DEFAULT (Read/Write)	8) Add textbox control for woassetcomment attribute to Work Order Tracking in the Application Designer.
	9) Input Mode, regardless of setting, is always Read Only in this scenario.

Perform the steps in the following scenarios to configure an application to display comments about an asset in a work order:

- ▼ Using Linked Objects to Display Information in an Application
- ▼ Using Crossover Domains to Display Information in an Application on page 34

## Using Linked Objects to Display Information in an Application

The user interface for an application uses controls that bind to database attributes that present data from the parent or children objects of an application in a browser window. The Work Order Tracking application, for example, uses WORKORDER as its parent object and ASSET as a child object.

Because most applications need to present data that resides in other objects, use the Database Configuration application to define "relationships" between objects. The relationships you create between objects enable data from related tables to be available to the parent object and subsequently to your application.

The Work Order Tracking application, for example, needs to display data stored in the ASSET object. The following example describes the steps you must perform

## Linked Objects and Crossover Domain Scenario

to capture user comments about a particular asset when adding that asset to a work order. The procedures in this section describe the following steps you must perform:

- 1 Add the assetcomment attribute to the ASSET object.
- 2 Configure the database.
- 3 Create a database relationship between ASSET and WORKORDER objects using the Database Configuration application.
- 4 Write down the relationship name, for example, ASSET2WOLINK, for future reference.
- 5 Use the Application Designer to add a textbox control for the assetcomment attribute to the Assets application.
- 6 Add an asset in the Assets application with comment information.
- 7 Use the Application Designer to add a textbox control for the <database relationship>.assetcomment attribute to the Work Order Tracking application (where <database relationship> is the name of the relationship created in Step 3).

## Adding Assetcomment Attribute to the ASSET Object

You must add an attribute (if one does not already exist) to the Asset object as the first step in capturing user comment information about an asset.

To add the assetcomment attribute to the Asset object:

- 1 In the Database Configuration application, in the search filter on the List tab, enter `asset` in the Object field.
- 2 Click **Asset** to open the detail record for the Asset table in the Object tab.
- 3 Click the Attributes tab.
- 4 Click **New Row** to display the Details section to add a new attribute.

### Details Section for Adding an Attribute to the ASSET Object

The screenshot shows the 'Details' section for adding an attribute to the ASSET object. The form is divided into two main sections: 'Details' and 'Advanced'.

**Details Section:**

- Attribute \***: [Empty text box]
- Description \***: [Empty text box]
- Type \***: ALN
- Length \***: 50
- Scale**: 0
- Required?**:
- Title \***: [Empty text box]
- Class**: [Empty text box]
- Domain**: [Empty text box]
- Default Value**: [Empty text box]
- Alias**: [Empty text box]
- Status**: Add

**Advanced Section:**

- Entity**: ASSET
- Column**: [Empty text box]
- Same as Object**: [Empty text box]
- Same as Attribute**: [Empty text box]
- Autonumber**: [Empty text box]
- Search Type \***: NONE
- Persistent?**:
- Must Be?**:
- Positive?**:
- User Defined?**:
- Can Autonumber?**:
- Long Description Owner?**:
- Sequence**: [Empty text box]
- Audit Enabled?**:
- Multilanguage Supported?**:
- Multilanguage in Use?**:
- E-signature Enabled?**:
- Primary Column**: [Empty text box]
- Attribute #**: 97
- Next Sequence Number**: [Empty text box]

5 Enter or accept the defaults for the following required fields:

Field Label	Input Value	Description
Attribute	Assetcomment	Attribute name
Description	User comments regarding the asset	Text description of attribute
Type	ALN	The default
Length	100	Field length
Title	Comment	Alias name to substitute in messages and windowlabels
Search Type	Wildcard	Search on this attribute using wild card characters like * and %

6 Click **Save**.

The Status of the new attribute changes to **Add**. The status of the Asset object changes to **To Be Changed**.

**NOTE** Saving a change in the Database Configuration application stores the update in a configuration table. It does not implement the change in the database. You must run the ConfigDB batch file to update the database.

## Running ConfigDB to Configure the Database

Before you continue with linked objects and the process of making asset comments available on work orders, you must run ConfigDB to add the assetcomment attribute to your database.

As a safety measure, back up your existing database before you run ConfigDB. When you run the database configuration program, it automatically configures and restores data in one step. Also, close all open applications before running ConfigDB, including the following:

- ▼ Application Server (Use **Ctrl+C** from a command line or Windows Services Administration if running as a service)
- ▼ Actuate Server and all third-party tools that might access the database, such as SQL\*Plus or Microsoft Excel

To configure your database:

1 In the Database Configuration application, from the **Select Action** menu, select **Apply Configuration Changes**.

The system displays a message stating that you must apply configuration changes to the database using ConfigDB from the command line while the application server is down.

2 Click **OK**, then sign out of the system and any other open applications.

Wait 60 seconds to ensure that any user sessions terminate properly.

## Linked Objects and Crossover Domain Scenario

- 3 Open a Command Prompt window. From a Windows operating system, choose **Start > Run**.
- 4 Type `cmd` in the Run window, and press **Enter**.
- 5 Navigate to the following directory:

```
<root>\tools\maximo
```

- 6 Type `ConfigDB` to configure the database and restore backup tables. When the configuration process completes, you will see the message:

```
RestoreFromBackup completed [Date/Time stamp]
```

- 7 Type `exit` at the command prompt, and press **Enter** to close the window.

To review the log file for the database configuration, navigate to the following directory:

```
<root>\tools\maximo\log
```

Open the "ConfigDB..." log file with the most recent date/time stamp. You will see a statement similar to the following:

```
alter table asset add assetcomment varchar2 (100) null
```

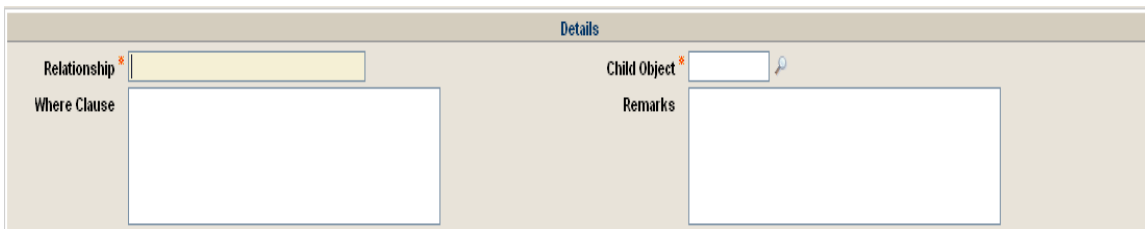
## Linking the WORKORDER and ASSET Objects

To capture comments associated with an asset on a work order, you must create a relationship or link between the WORKORDER and ASSET objects. The link enables Work Order Tracking to display and/or update information stored in the ASSET object.

To create a relationship between the WORKORDER and ASSET objects:

- 1 In Database Configuration, type `workorder` in the Object field, and press **Enter**.
- 2 Click **Workorder** to open the Work Order window .
- 3 Click the Relationships tab, and then click **New Row** to open the Details window .

### WORKORDER Relationships Details Screen



The screenshot shows a web-based form titled "Details" for configuring a relationship between objects. It features four main input areas: a "Relationship" field (a small text box), a "Child Object" field (a small text box with a search icon), a "Where Clause" field (a large empty text area), and a "Remarks" field (a large empty text area). The form is set against a light beige background.

Complete the information on this window to add a database relationship that binds the Workorder and Asset objects using one or more Where Clauses. The Where Clause or Clauses state the conditions under which the two objects can join. In this example, the Where Clauses ensure that Work Order Tracking

returns a unique record value from the Asset object when you enter an asset number in a work order.

**4** Type the following information:

Field Label	Input Value	Description
Relationship	ASSET2WOLINK	The name of the database relationship
Child Object	ASSET	The name of the child object. WORKORDER is the parent object.
Where Clause	assetnum=:assetnum and siteid=:siteid	SQL Where Clause condition or conditions. The system does not validate your input in the Where Clause field. Verify that your syntax is correct.
Remarks	<text>	A description of the relationship. For example:  This relationship binds WORKORDER and ASSET objects when you enter an asset number in a work order.

The following table describes the logic of the Where Clause conditions for this example:

Where Clause Syntax	Description
:assetnum	Represents asset number value entered in work order screen
assetnum=	Look in <b>Asset</b> object for the asset number that matches the value entered in the work order screen
:siteid	The Site value entered in work order screen
siteid=	Look in <b>Asset</b> object for the Site value that matches the current value in the work order screen

The Where Clause above joins the Workorder and Asset objects and locates a unique record.

**5** Click **Save**.

To complete the process of capturing asset comment information on work orders, continue to the next procedure that describes how to add a textbox control to Work Order Tracking in the Application Designer.

**Adding a Textbox for Assetcomment in Assets**

Complete this procedure to use the Application Designer to add a textbox control to the Assets application. The textbox lets you input or edit a comment value for an asset. For example, you might enter the comment "John Doe is monitoring this motor..."

Before you start, verify that you have added assetcomment attribute to the Asset object

To add a textbox control to the Assets application:

## Linked Objects and Crossover Domain Scenario

- 1 In the Application Designer, type `asset` in the Application field, and press **Enter**.
- 2 Click **Asset** to open Assets in the Workspace tab.
- 3 Select the Asset tab.
- 4 Click the section that holds the Asset field.
- 5 Click **Control Palette**.
- 6 Drag the Textbox control into the Asset section.
- 7 With the Textbox control selected, click **Control Properties** to open the Textbox Properties window.
- 8 Enter the following information to bind the Textbox control to the `assetcomment` attribute:

Field	Input Value	Description
Default Label	Defaults to Comment	When you specify <code>assetcomment</code> in the Attribute field, the field defaults to the default label value of the attribute (if available). You can assign your own label value for this attribute that is specific to the application in the Label field. For example, you might enter Asset Comments.
Label	Comments	The value you enter here, for example, Comments, overrides the Default Label setting within the application. The value you enter for Label is specific to the application and not stored in an object.
Attribute	<code>assetcomment</code>	The name of the asset comment attribute in the <b>Asset</b> object.
Input Mode	DEFAULT	The Default option configures the Input Mode for the Comment field to be Read/Write or Edit mode.

- 9 Click **Save**.

## Adding a Textbox for Assetcomment in Work Order Tracking

Complete this procedure to use the Application Designer to add a textbox control in Work Order Tracking. The textbox displays the `assetcomment` attribute value when you enter an asset number in a work order.

Before you start, verify that you have completed the following tasks:

- ▼ Added `assetcomment` attribute to the Asset object
- ▼ Configured your database
- ▼ Added a database relationship (ASSET2WOLINK) to the Workorder object

- ▼ Added textbox control for assetcomment to Assets in the Application Designer (Must subsequently enter a comment for a new or existing asset, for example, "John Doe is monitoring this motor...")

You can configure the Input Mode property of a textbox control in the Work Order Tracking application to a number of different settings. For this example, you can set the Input Mode to Default, which is Read/Write. Alternatively, you can set the value to Read Only. Setting the Comments field to Read/Write updates the assetcomment attribute whenever you edit the attribute value in Work Order Tracking. The Read Only setting displays, but does not let you update, the comment value in the work order application.

To use the Application Designer to add a textbox control to the Work Order Tracking application:

- 1 In the Application Designer, type WOTRACK in the Application field and press **Enter**.
- 2 Click **Wotrack** to open Work Order Tracking in the Workspace tab.
- 3 Select the Work Order tab.
- 4 Click the section that contains the Asset field.

***Asset Section of Work Order Tracking in the Application Designer***

- 5 Click **Control Palette**.
- 6 Drag the Textbox control on top of the Parent WO field to position the control between the Asset and Parent WO fields.
- 7 With the Textbox control selected, click **Control Properties** to open the Textbox Properties window.
- 8 Enter the following information to bind the Textbox control to the assetcomment attribute:

Field	Input Value	Description
Default Label	Defaults to Comment	When you specify assetcomment in the Attribute field, this field defaults to the default label value of the attribute (if available). You can assign your own label value for this attribute that is specific to the application in the Label field. For example, you might enter <code>Asset Comments</code> .
Label	Comments	The value you enter here, for example, <code>Comments</code> , overrides the Default Label setting within the application. The value you enter for Label is specific to the application and not stored in an object.
Attribute	asset2wolink.assetcomment	Type the name of the asset comment attribute in the Asset object. You must preface the assetcomment attribute with the name of the database relationship that binds the Asset and Workorder objects and a dot. For example, <code>asset2wolink.assetcomment</code> .

Field	Input Value	Description
Input Mode	DEFAULT or READONLY	The Default option configures the Input Mode for the Comment field to be Read/Write or Edit mode. You can also configure the Comment field to be Read Only

**9** Click Save.

At this point, the system populates the Comments field whenever you add an asset record to a work order.

## Using Crossover Domains to Display Information in an Application

Using a crossover domain to display comments about an asset in the Work Order Tracking application is similar to using linked objects except that you use a crossover domain value to establish a link between the Workorder and Asset objects and not a database relationship. Additionally, the asset comment that displays in a work order is a copy of the woassetcomment attribute at the time you add an asset number to the work order. When using a crossover domain to copy data across applications, you cannot edit the asset comment value in the destination application.

Because most applications need to present data that resides in other objects, you need to use Database Configuration or Domains to define links between objects. You define the parent and child objects, the trigger fields, and the conditions (SQL Where Clauses) that bind two objects. The relationships you create between objects enable data from related tables to be available to the main object and subsequently to your application.

The Work Order Tracking application, for example, needs to display data stored in the Asset object. The following scenario describes the process of capturing user comments about a particular asset when adding that asset to a work order using a crossover domain. Creating and applying a crossover domain includes the following steps:

- 1** Add the assetcomment attribute to the Asset object and the woassetcomment to the Workorder object.
- 2** Configure the database.
- 3** Add the ASSET2WO crossover domain using the Domains application.
- 4** Add the ASSET2WO crossover domain to the WORKORDER woassetcomment attribute using the Database Configuration application.
- 5** Configure the database.
- 6** Use the Application Designer to add a textbox control for woassetcomment to the Work Order Tracking application.
- 7** Enter an asset number on a work order.



## Creating and Applying a Crossover Domain

A crossover is a mechanism that the system uses to copy data from a child object to a parent object based on a value you enter in the user interface of an attribute. You might build a crossover, for example, to accomplish the following:

- ▼ Copy the supervisor’s name for a user who creates a material receipt to the newly created record.
- ▼ Copy the phone number and address of a person after you enter their name in an application.
- ▼ Copy the parent asset number (asset.assetnum) or asset comments (asset.assetcomment) to a work order after you enter the asset number in a new work order record.

Creating and applying a crossover is a multi-step process. The following set of procedures describe how to employ a crossover to copy asset comments to a new work order record.

The Workorder object manages work order records and can store information about a work order from related objects like Asset. A typical work order record stores asset numbers, and when employing the crossover domain used in this example, asset comments.

The crossover enables the system to retrieve and save comments about an asset in the **Workorder** object when you add an asset number to a work order.

## Adding Woassetcomment Attribute to the WORKORDER Object

The first step in capturing user comment information about an asset is to add an attribute (if one does not already exist) to the Asset object that will store comment information. See "Adding assetcomment attribute to the Asset object" on page 28 for a detailed procedure.

To add the woassetcomment attribute to the Workorder object:

- 1 In the Database Configuration application, in the search filter on the List tab, type `workorder` in the Object field, and press **Enter**.
- 2 Click **Workorder** to open the detail record for the Work Order table in the Object tab.
- 3 Select the Attributes tab.
- 4 Click **New Row** to display the Details section for adding a new attribute.
- 5 Enter or accept the defaults for the following required fields:

Field	Input Value	Description
Attribute	Woassetcomment	Attribute name
Description	User comments regarding the asset	Text description of attribute
Type	ALN	The default

Field	Input Value	Description
Length	100	Field length
Title	Comment	Alias name to substitute in messages and window labels
Search Type	Wildcard	Search on this attribute using wild card characters like * and %

6 Click **Save**.

The Status of the new attribute changes to Add.

7 Configure your database.

See "Running ConfigDB to configure the database" on page 29 for detailed instructions.

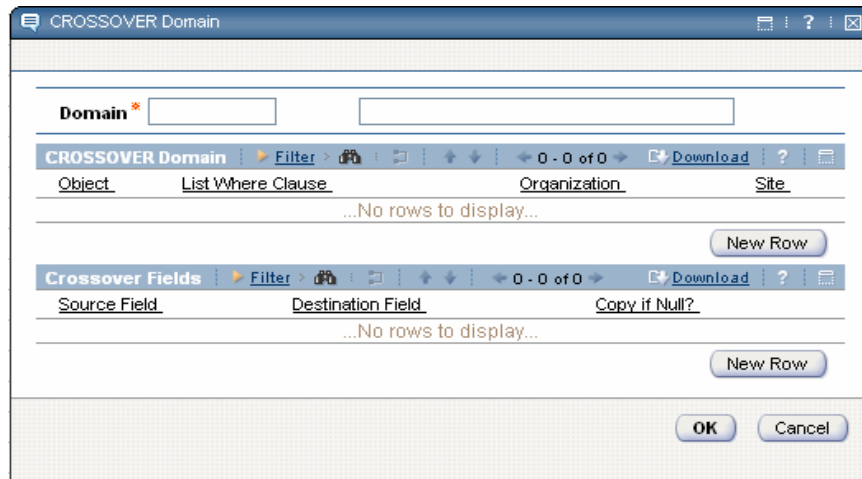
**NOTE** Saving a change in the Database Configuration application stores the update in a cfg table. It does not implement the change in the database. You must run the ConfigDB batch file to update the database.

### Adding a New Crossover Domain

To add a crossover domain named ASSET2WO:

1 In the Domains application, click **Add New Domain > Add New CROSSOVER Domain** to open the Crossover Domain window.

#### CROSSOVER Domain Screen



2 Enter `ASSET2WO` as the Domain name and `Asset to Work Order Crossover Domain` as the full description.

3 Click **New Row** in the Crossover Domain section to open the detail section.

4 Enter `Asset` in the Object field. This step defines `Asset` as the source object of the crossover.

- 5 Enter `assetnum=:assetnum` and `siteid=:siteid` in the Validation Where Clause field.

**NOTE** See Linking the Workorder and Asset Objects procedure on page 30 for more information on the Validation Where Clauses.

- 6 Click **New Row** in the Crossover Fields section to open the detail section.
- 7 Use the lookup to select `assetcomment` from the Asset object in the Source Field, then type `woassetcomment` in the Destination Field.
- 8 Click **OK**.

## Applying the Crossover Domain

Now that you have created the ASSET2WO crossover domain, you must apply it to a field that triggers the crossover conditions. In this example, you apply the crossover to the Asset Number (`assetnum`) field. Subsequently, whenever you enter an asset number in Work Order Tracking, the system executes the crossover domain and copies the value in `assetcomment` to `woassetcomment` for the new work order record.

- 1 In the Database Configuration application, type `WORKORDER` in the Object field, and click **Enter**.
- 2 Click **Workorder** to open the Work Order window in the Object tab.
- 3 Click the Attributes tab to open the Attributes Details window.
- 4 Click **Filter**, and enter `assetnum` in the Attribute field.
- 5 Press **Enter** to open the Assetnum details section, and enter `asset2wo` in the Domain field.
- 6 Click **Save**.

**NOTE** To complete the process of adding and applying the ASSET2WO crossover, you must configure your database. Refer to "Running ConfigDB to update the database" on page 29 for instructions.

## Adding a Textbox for Woassetcomment in Work Order Tracking

Complete this procedure to use the Application Designer to add a textbox control in the Work Order Tracking application. The textbox displays the **woassetcomment** attribute value when you enter an asset number in a work order.

Before you start, verify that you have completed the following tasks:

- ▼ Added `assetcomment` attribute to the Asset object and `woassetcomment` to the Workorder object.
- ▼ Added textbox control for `assetcomment` to Assets in the Application Designer. You must subsequently enter a comment for a new or existing asset, for example, "John Doe is monitoring this motor...".
- ▼ Added ASSET2WO crossover domain in the Domains application.

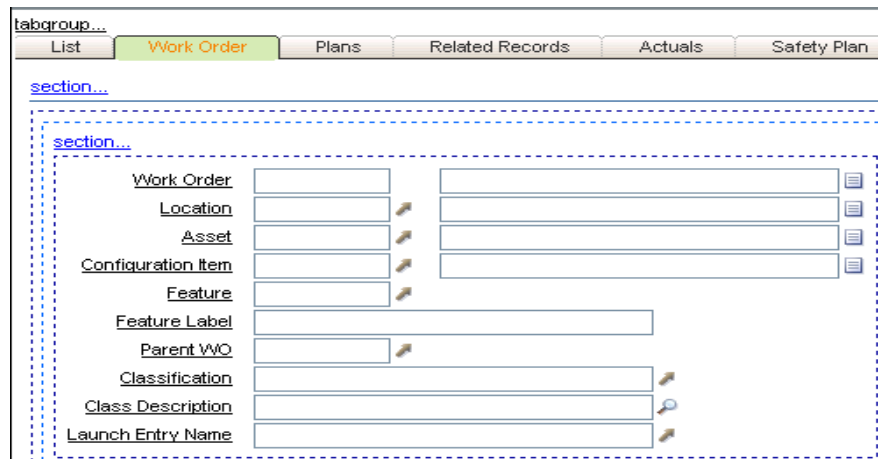
## Linked Objects and Crossover Domain Scenario

- ▼ Added ASSET2WO to the Workorder woassetcomment attribute in the Database Configuration application.
- ▼ Configure your database.

To add a textbox for woassetcomment in the Work Order Tracking application:

- 1 In the Application Designer, type `wotrack` in the Application field, and press **Enter**.
- 2 Click **Wotrack** to open Work Order Tracking in the Workspace tab.
- 3 Select the Work Order tab.
- 4 Click the section that holds the Asset field.

### Asset Section of Work Order Tracking in the Application Designer



- 5 Click **Control Palette**.
- 6 Drag the Textbox control on top of the Parent WO field to position the control between the Asset and Parent WO fields.
- 7 With the Textbox control selected, click **Control Properties** to open the Textbox Properties window.
- 8 Enter the following information to bind the Textbox control to the woassetcomment attribute:

Field	Input Value	Description
Default Label	Defaults to Comment	When you specify woassetcomment in the Attribute field, this field defaults to the default label value for the attribute (if available).
Label	Comments	The value you enter here, for example, Comments, overrides the Default Label setting within the application. The value you enter for Label is specific to the application and not stored in an object.

Field	Input Value	Description
Attribute	woassetcomment	The name of the asset comment attribute in the <b>Workorder</b> object.
Input Mode	READONLY	Configures the Input Mode for the Comments field. In this scenario, regardless of the Input Mode setting, you cannot edit the Comments value.

**9** Click Save.

In summary, by completing the crossover domain procedures you accomplished the business objective of retrieving read-only asset comment information whenever a user enters an asset number in a work order. The crossover domain enables the following actions:

- ▼ Uses the value for ASSETNUM and SITEID and runs it through the Where Clause to find a matching record for the ASSETNUM+SITEID combination in the Asset object.
- ▼ Once located, the crossover takes the value of the assetcomment attribute (the source) of the matching record in the Asset object and copies it to the woassetcomment attribute (the destination) in the Workorder object.
- ▼ The system copies the woassetcomment value to the textbox control you added to the work order record whenever you enter an asset number. The asset comment information copied to the work order is static. This value does not change in the work order record even if someone subsequently modifies the asset comment value in the Asset object.



# Conditional User Interface

# 5

You can configure the system to meet different user requirements. You use the conditional user interface to provide access to data based on user groups and the evaluation of one or more conditions.

You can configure the properties of a control (such as label, color, and value list) to be conditional for specified groups of users or when one or more conditions are true or false.

You can use the features of the conditional user interface independently or combine them to provide a dynamic user interface.

**CAUTION: EXTENSIVE USE OF THE CONDITIONAL USER INTERFACE CAN IMPACT SYSTEM PERFORMANCE.**

The common tasks of the conditional user interface are described in the following sections:

- Configuring conditional control properties
- Showing or hiding a control unconditionally
- Showing or hiding a control in a condition for a role or group
- Showing or hiding a control in a condition for everyone

## Configuring Conditional Control Properties

You can change control properties from their default behavior by setting a condition for a security group. You can configure control properties in the Application Designer after you define both the security groups and conditions.

To configure a conditional property:

- 1** In the Conditional Expression Manager application, create one or more conditions to be evaluated to grant access to the control when it is true. The condition can be an expression or a class.
- 2** Ensure that the required groups exist in the Security Groups application.
- 3** In the Application Designer, open the application and control that you want to configure.
- 4** Open the Control Properties window for the control.
- 5** Click **Configure Conditional Properties**.

- 6 In the Configure Conditional Properties window, specify a signature option for the control.
- 7 In the Security Groups section, specify the group that you want to apply the condition to. To apply the condition globally, select the EVERYONE group.
- 8 In the Conditions for a Security Group section, specify a condition. The conditions that you specify apply only to the group that you specified in the Security Groups section.

If you want to change the property for a group, define and select a condition that is true, such as '1=1'.

- 9 Optional: In the Property Values when Condition is True and the Property Values when Condition is False sections, type the property names and values that you want to change.

Type the property name of the control exactly as it appears in the Control Properties window.

- 10 In the Security Groups application, grant the appropriate signature option to the security group that you specified in step 7.

See “How to Define Signature Options for an Application” on page 17 for more information on how to create a Signature Option.

## Showing or Hiding Controls

User interface controls are created and maintained in the Application Designer. Any control, such as a field, section, tab, or table window, can be associated with a signature option. Any control can be shown or hidden from groups of users.

## Showing or Hiding a Control Unconditionally

You can grant access to a control for users in groups that are granted a particular signature option. Users who are not granted the signature option do not have access to the control.

Multiple controls can be associated with one Signature Option. Access to the controls is granted or revoked as a group.

To show or hide a control unconditionally:

- 1 In the Application Designer, specify your application and control.
- 2 Click **Control Properties**.
- 3 In the Properties window, click **Select Value** next to the Signature Option field and select a signature option to link the control to. Close the Properties window.



- 4 Click **Save Application Definition** on the main page of the Application Designer application to save your changes.
- 5 In the Security Groups application, select or create a group, and then select the Applications tab.
- 6 Select the **Grant Access** check box in the Read Access to Actions row to grant read access to the application

### The Grant Access Checkbox

Options for Actions	
Filter >	1 - 4 of 4
Description	Grant Access ?
Delete Action	<input type="checkbox"/>
New Action	<input type="checkbox"/>
Read Access to Actions	<input type="checkbox"/>
Save Action	<input type="checkbox"/>

- 7 To grant the Signature Option to the security groups that you selected, select the check box of that particular signature option in the Security Groups application.

The control is shown in the interface for only the users in the security groups that this signature option is granted to.

## Showing or Hiding a Control in a Condition for a Role or Group

When a control is associated with a signature option, the condition can be granted unconditionally or it can be granted in a condition. As is the case with showing or hiding the controls unconditionally, they are granted as a group. When a condition evaluates to true, all of the controls associated with the signature option are granted.

To show or hide a control in a condition for a role or group:

- 1 In the Application Designer, specify the application and control.
- 2 Click **Control Properties** to open the Properties window for the control.
- 3 Click **Select Value** next to the Signature Option field to associate the control with an existing signature option.
- 4 Close the window then click **Save Application Definition** on the main page of the Application Designer application to save your changes.
- 5 Go to the **Conditional Expression Manager** application to create a condition record to be evaluated to grant access to the control when it is true. It can be an expression or a class.
- 6 In the Security Groups application, select or create a group and go to the Applications tab.

- 7 Check the **Grant Access** check box in the Read Access to Actions row to grant read access to the application.
- 8 To grant the signature option to the security groups that you specified, select the check box of that particular signature option in security groups.
- 9 Add the condition that you created in step 5 to the signature option by entering the ID in the field next to the **Grant** check box.

When the condition is true the group has access to the control.

## Showing or Hiding a Control in a Condition for All Users

If a group contains all users, you can define specialized behavior that applies to all users. When users are granted read access to the application in another group, the specialized behavior is applied.

To show or hide a control in a condition for all users:

- 1 In the Application Designer, choose your application and control.
- 2 In the Control Properties window for the control, specify a signature option (either existing or new) in the Signature Option field to associate the control with a signature option.
- 3 Close the window then click **Save Application Definition** on the main page to save your changes. .
- 4 In the Conditional Expression Manager application, create a condition record to be evaluated in order to grant access to the control when the condition is true. The condition can be an expression or a class.
- 5 In the Security Groups application, select the EVERYONE security group on the List tab.

This group name is configurable but EVERYONE is the default.

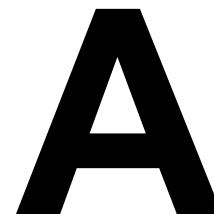
- 6 Go to the Applications tab for the EVERYONE security group and *find* the application that you want to apply the condition to.
- 7 To grant the Signature Option to the EVERYONE group, select the check box associated for that Signature Option in the Security Groups application for the EVERYONE group.
- 8 Add the condition that you created in step 5 to the Signature Option by entering the condition number in the Condition field.
- 9 On the List tab, select or create another group.
- 10 On the Applications tab, grant the group read access to the application.
- 11 Add a user in both groups.

You now have one group of users with read access and the EVERYONE group that has conditional access to the control. When the two groups are combined, users have access to the control when the condition is true.

Users who do not have read access to the application in any of their groups do not have access to the control because they do not have access to the application.



# List of Signature Options



This section lists the signature options available to new and existing applications. For more information on signature options, see "Defining Signature Options for an Application," on page 17.

## Common Signature Options

Control Name	Description	Example
STATUS	Change Status	
INSERT	New	
SAVE	Save	
CLOSE	Close	
CANCEL	Cancel	
CLEAR	Clear Changes	
PREVIOUS	Previous	
NEXT	Next	
ASSOCFOLD	Associate Folders	
MANAGEFOLD	Manage Folders	
MANAGELIB	Manage Library	
DUPLICATE	Duplicate Work Order	
DELETE	Delete	
BOOKMARK	Add to Bookmarks	
RUNREPORTS	Run Reports	
SEARCHBOOK	Bookmarks	
SEARCHMORE	More Search Fields	
SEARCHSQRY	Save Current Query	
SEARCHTIPS	View Search Tips	

## Common Signature Options

Control Name	Description	Example
SEARCHVMQR	View/Manage Queries	
SEARCHWHER	Where Clause	
ACTIVATE	Activate	Activate Process
ADDMOD	Add/Modify	ADDMODACTN – Add/Modify Actions
APPLYIAS	Apply Item Assembly Structure	
APPLYPRAD	Apply Price Adjustment	
APPLYRTE	Apply Route	
APPLYSLA	Apply SLAs	
APPR/APPROVE	Approve	
ASGNEWPRNT	Assign to New Parent	
ASSIGNWF	View Workflow Assignments	
ASSOCCOMM	Associate Commodities	
ASSOCSLA	Associate Slays	
ASTLOCCOMM	Associate Services	
AUTHSITES	Authorize Sites	
COMMCODE	Add/Modify Commodity Codes	
COMP	Complete	Complete Work Order
CONVERSION	Add/Modify Conversions	CREATEWO – Create Work Order
CREATE	Create	
CURBALADJ	Current Balance	
DRAFT	Change Status to Draft	
DRILLDOWN	Open Drilldown	
EDITHIST	Edit History	Edit History Work Order
EXPIRE	Change Status to Expire	
FIELDDEFS	Set Field Defaults	
GENERATEWO	Generate Work Orders	
HELPWF	Workflow Help	
HISTORYWF	View Workflow History	
INACTIVATE	Inactivate	Inactivate a person
INPRG	Change Status to in Progress	
INVTRANS	View Inventory Transactions	
MASSMOVE	Move/Modify Assets	

Control Name	Description	Example
MEASUNIT	Add/Modify Units of Measure	
METERREAD	Enter Meter Readings	
METHIST	Manage Meter Reading History	
METHISTUPD	Update Meter History	
METREADNEW	Enter Most Recent Meter Reading	
METREADOLD	Enter Meter History	
METRESET	Reset/Replace Meters	
MODDELWL	Modify/Delete Work Log	
MODIFYSLAS	Select/Deselect SLAs	
MOVEASSET	Move/Modify Assets	
MOVEVAR	Move Variants	
NEWWRKPKG	Create Work Package	
NOPORTLET	Hide Portlet	
OWNER	Select Owner	
OWNERSHIP	Take Ownership	
PNDREV	Change Status to Pending Revision	
QUEUED	Problem in queue	
READ	Read access	Read access to Work Order Tracking
RECONCILE	Reconcile Balances	
REMOVEWP	Remove Work Plan	
REORDER	Reorder Items	
REORISS	Reorder Direct Issue Itms/Svcs	
RESOLVED	Resolve problem	
REVCONT	Revise Contract	
REVHIST	View Revision History	
ROUTEWF	Route Workflow	
SELASPARTS	Select Asset Spare Parts	
SHOWPLAN	Entire Plan	
SIMILARTKT	Show Similar Tickets	
STARTTIMER	Start Timer	
STOPTIMER	Stop Timer	
STOPWF	Stop Workflow	
SUSPND	Change Status to Suspend	

## Common Signature Options

<b>Control Name</b>	<b>Description</b>	<b>Example</b>
TEMPLATE	Apply Template	Apply Problem Template
TRANSFER	Transfer Current	Transfer Current Item
UNAPPROVE/UNDOAPPR	Unapprove/Undo Approval	Undo Approval of Work Order
USERCUST	Associate Users and Custodians	
VALIDATE	Validate	Validate Process
VIEWCONT	View Contracts	
VIEWCOSTS	View Costs	
VIEWHIST	View History	
WMATL	Set status to Waiting for Material	Set Work Order status to Waiting for Material
WO_SLAS	View SLAs	
WSTART	Change Status to Waiting to Start	
ZEROYTD	Zero Year to Date Quantities	



# Controls Reference Information

# B

The controls listed in this section are available in the control palette of the Application Designer. Each control description contains an introduction, rules, and properties description section. The controls appear in alphabetical order.

## AppBar



The AppBar, short for application bar, is a menu bar that is always open at the top of a page. The AppBar automatically lists all applications in a module except the currently active application. An application bar appears in self service applications like that do not use tabs.

### Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Client area

**NOTE** To view the client area in the Application Designer, select **Toggle Show All Controls** from the **Select Action** menu.

### Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	The name you assign to the AppBar control. This value does not display in the user interface.

# Attachments



You can activate the drop down menus and windows associated with Attachments by dropping the control into any Section container in an application. However, you cannot access the functions used to build the Attachments control through the interface of the Application Designer.

The Attachments control lets you change the label for Attachments that appear in various applications. For example, you can view Attachments in Assets > Asset. A paper clip icon appears next to the Attachments label.

## Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Section

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	<p>The name you assign to the Attachments control. The label value identifies the control to the user. If you do not enter a label, the title defaults to "attachments" in the user interface.</p> <p>The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.</p>

# Blank Line



A Blank Line inserts vertical white space in a section. You might, for example, use this control to add white space between text boxes or rows in a table. Blank lines provide spacing between controls to improve readability or to visually separate different types of controls within a section.

A Blank Line does not have configurable properties and simply functions as graphical element without intelligence. A Blank Line does not have a label. Blank lines, like Sections Columns and Section Rows, are transparent in the user interface.

## Rules

This is not a container control.

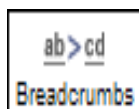
You can place this control inside the following container control:

- ▼ Section

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.

# Breadcrumbs



You can use the Breadcrumb control to enhance navigational functionality in an application. Breadcrumb navigation tracks the steps you take in an application and facilitates your ability to retrace steps quickly. You can use a Breadcrumb control by itself or in conjunction with another control, such as a Tree.

A Breadcrumb complements a hierarchical tree structure by showing a horizontal representation of where you are in a menu structure or service catalog.

For example:

[Work Order Tracking](#) > [Assets](#) > [Locations](#)

Work Order Tracking is the highest level of the hierarchy. Assets is the second and Locations is the third level of the hierarchy. You can configure the greater than character (>) or another character, such as colon (:), as a delimiter for the breadcrumb. Each phrase or ID between delimiters represents one step in a hierarchy. Clicking on a breadcrumb in an application can effect the appearance of other window elements that are tied to the hierarchical MBO.

Applications such as Classifications can use a Breadcrumb control in combination with a Tree control.

To synchronize Tree and Breadcrumb controls, copy the Data Source ID property value from the Tree control to the Breadcrumb control.

The following table describes how a Tree and Breadcrumb control behave in the user interface when combined. The standard behavior of a Tree control described on page 131 is always applicable.

Clicking...	What happens...
Any Node Label	New breadcrumb trail replaces any existing breadcrumb trail. Also: <ul style="list-style-type: none"> <li>▼ Highlights node label in tree</li> <li>▼ Displays node label in breadcrumb</li> <li>▼ All levels in tree hierarchy above the highlighted node label appear in the breadcrumb (in descending order from left to right)</li> </ul>
Plus sign in tree	No effect on breadcrumb or node label highlighting.
Minus sign in tree	Highlights the right-most breadcrumb, which is the same as the node label next to the minus sign.
Dot sign in tree	The corresponding node label highlights in tree and displays in the breadcrumb. All levels in tree hierarchy above the highlighted node label appear in the breadcrumb (in descending order from left to right)
Any node label in the breadcrumb	Equivalent to clicking node label in the tree. The tree contracts to the node label level and any breadcrumb links to the right of the clicked node label disappear.
A delimiter in the breadcrumb	No effect.

To summarize, you can expect the following user interface behaviors when navigating a tree or breadcrumb:

- ▼ Any time you highlight a node label in a tree or breadcrumb, the window displays information related to that label.
- ▼ Clicking a plus sign in a tree expands the node, but does not affect window content, label highlighting, or the breadcrumb.
- ▼ Clicking a minus sign in a tree does not affect window content, but does highlight the node label associated with the minus sign and the breadcrumb reflects the new hierarchy ending with the highlighted node.
- ▼ The highlighted node label in a tree always reflects the right-most element in the breadcrumb. The breadcrumb elements to the left reflect the intervening levels in the tree hierarchy.

## Rules

This is not a container control.

You can place this control inside any container control.

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Bean Class	Reserved for future use.
Business Object	<p>The main MBO reference for the application. For example, Work Order Tracking uses WOTRACK and Assets uses ASSET.</p> <p>Every application works with a MBO and MBOSet. The set of work orders displayed on the List tab in the Work Order Tracking application, for example, is a MBOSet. Each individual work order is a MBO. An presentation tag of an application defines the main MBO for the application. MBOs are responsible for the following types of functions:</p> <ul style="list-style-type: none"> <li>▼ Field validation</li> <li>▼ Updating the database</li> <li>▼ Business processing logic.</li> </ul> <p>By default, the presentation tag for an application defines the main MBO to use when saving or displaying data through the user interface. All controls in the application share the same MBO reference via the Parent Data Source ID property. Unless overridden, all controls added to an application bind to the Parent Data Source ID. If you wish an individual control to display and process information from a different MBO, you can enter a different Parent Data Source ID.</p>
CSS Class Name	Enter the class name for a cascading style sheet. The value you enter can change the layout and appearance of the Breadcrumb control.
Data Source ID	If you want this control to use a different Data Source ID than the default of MAINRECORD, then enter that value here.
Label	<p>The name you assign to the Attachments control. The label value identifies the control to the user. If you do not enter a label, the title defaults to "attachments" in the user interface.</p> <p>The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.</p>
Separator	Reserved for future use.

Attribute Name	Description
Relationship	<p>You enter a relationship value in this field. You can search for values in the Relationships tab of the Database Configuration application. If you enter a relationship, it is automatically inherited by any controls that point to this data source.</p> <p>Relationships enable a control to display information from different business objects. For example, if you entered SPAREPART, the table will be able to display information based on the defined relationship. When you enter a relationship in the Table control, it automatically prefixes any child column attributes added to the table. So, for SPAREPART, you only have to enter the column attribute name and not the full relationship of SPAREPART &lt;attribute&gt;.</p>
Breadcrumbs Source ID	Enter a Data Source ID for the control. If you use the Breadcrumb with a Tree control, enter the Data Source ID for the Tree.
Source Method	The piece of code functionality that calls the menu bar, its signature security options, and layout. The method that calls the standard search menu bar is "getAppSearchOptions."
Align	Reserved for future use.
Signature Option	Reserved for future use.

## Button Group



Button Group is a container control that holds pushbuttons. The Button Group defines the layout for the buttons it holds, including alignment within the container and the title for the group.

### Rules

You can place the following control in this container control:

- ▼ Pushbutton

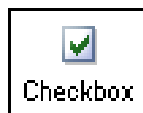
You can place this control inside the following container control:

- ▼ Section
- ▼ Table

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	<p>The name you assign to the Button Group control. The label value identifies the control to the user. If you do not enter a label, the Button Group title is transparent in the user interface. Most Button Groups do not have a label in the system, because the control functions primarily as a container that holds a layout for its children pushbuttons.</p> <p>The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.</p>
Text Alignment	Select an alignment value to position the Button Group label within the control. The options are Left, Center, or Right. The text label always displays to the left of any pushbuttons you add to the group.
Button Alignment	Reserved for future use.
Show Container?	Select this property to display a shaded background of horizontal lines for a Button Group container. Note that you cannot view background shading for a Button Group control in a Table container.

## Checkbox



A Checkbox is an input type of control that typically enables a user to select one or more items from a list or to indicate a yes/no response to a question. A selected box, for example, indicates a Yes response to a question. Also, a Checkbox can display information that is inherited from the database, which you may be able to edit depending on the Input Mode property setting.

Checkbox is similar to a text type of control, such as Textbox and Multiline Textbox, because it has a text field component or label that is visible in the user interface. The windows for these types of controls usually have a Default Label property that enables the control to inherit a read-only label from the attribute you configure for the control. The Default Label value comes from the database object. Alternatively, you can override an inherited Default Label by entering a different value in the Label property. This value is local to the presentation file of the application. Labels you enter at the presentation level are not stored in the MAXATTRIBUTE or MAXLABELS table.

Checkboxes appear in most applications and, depending on the property settings, can accept user input or display information from the database.

## Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Section

**NOTE** The label for the Checkbox control displays in the application with a question mark as "GL Account?"

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Null (aka Default Label)	This label defaults to the text value associated with the database attribute you bound to the Checkbox. This is a read only field.
Label	You can enter a text value in this field that overrides the Default Label value. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Attribute	You can associate a Checkbox with a database attribute by using the Select Value window. Not all attributes you select will have a database default label value.
Value	Reserved for future use.
Input Mode	Select one of the options listed below for the Input Mode: <ul style="list-style-type: none"> <li>▼ Default</li> <li>▼ Password</li> <li>▼ Password Readonly</li> <li>▼ Password Required</li> <li>▼ Query</li> <li>▼ Readonly</li> <li>▼ Required</li> </ul>



Attribute Name	Description
	<p>The Default option sets the Input Mode for all controls to "edit." If you do not select an Input Mode for a control, the system automatically sets the mode to read/write so that you can both enter or display information in the field. If you enter information in a read/write text field, it posts to the database column specified by the Attribute property.</p> <p>Input Mode Option Descriptions:</p> <p><b>Default</b> – sets control to "edit" so you can display or input values and mode is read/write</p> <p><b>Password</b> – sets control to display asterisks (blanked out) for displayed or input values and mode is read/write</p> <p><b>Password Readonly</b> – sets control to display asterisks and mode is read only</p> <p><b>Password Required</b> – sets control to display asterisks and the control is required</p> <p><b>Query</b> – sets control to display a typed value and mode is read only</p> <p><b>Readonly</b> – sets control to display grayed-out text and mode is read only</p> <p><b>Required</b> – sets control to display a typed value and record cannot be saved unless there is a value in the control</p>
Data Change Event	<p>You can configure an input control to perform either of the following actions based on any data change to the control:</p> <ul style="list-style-type: none"> <li>▼ REFRESHTABLE</li> <li>▼ RESETCHILDREN</li> </ul> <p>Choosing REFRESHTABLE forces the table that is the data source for the input control to refresh its data. Choosing RESETCHILDREN forces any children of the data bean of the control to delete and reacquire their MBOSetRemote references.</p>
Data Source ID	<p>If you want this control to use a different Data Source ID than the default of MAINRECORD, then enter that value here.</p> <p>See the Data Source ID description for the Combobox control for additional information on data sources.</p>

# Combobox



A Combobox is an input and text type of control that enables a user to select one or more items from a drop-down list. Like the Checkbox and Textbox controls, the Combobox also has a text field component or label that is visible in the user interface.

The windows for these types of controls usually have a Default Label property that enables the control to inherit a read-only label from the attribute you configure for the control. The Default Label value comes from the database object. Alternatively, you can override an inherited Default Label by entering a different value in the Label property. This value is local to the presentation file of the application. Labels you enter at the presentation level are not stored in the MAXATTRIBUTE or MAXLABELS table.

Comboboxes appear less frequently than Checkboxes but are still widespread throughout most applications. Depending on property settings, Comboboxes typically let you make a single selection from a drop-down list of values displayed from the database.

## Rules

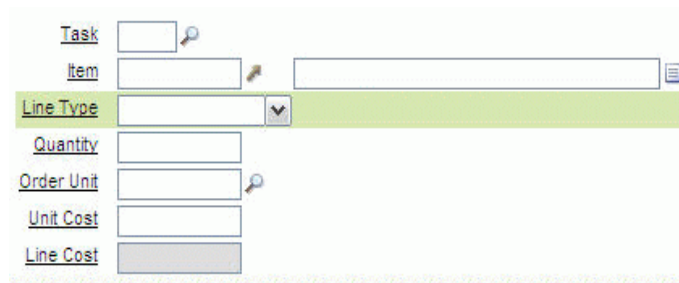
This is not a container control.

You can place this control inside the following container control:

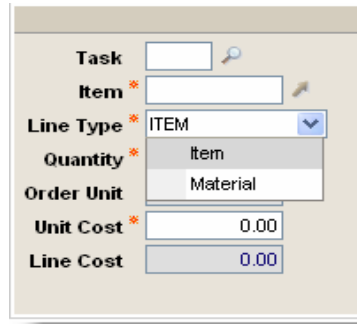
- ▼ Section

## Example

- 1 Combobox control in Details section of the Materials subtab in Work Order Tracking > Plans. The control label is Line Type. The application is the Application Designer.



2 Combobox as it appears in the Materials subtab of Work Order Tracking application.



Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control. You cannot edit this value.
Default Label	This label defaults to the text value associated with the database attribute you bound to the Combobox. This is a read only field.
Label	You can enter a text value in this field that overrides the Default Label value. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Width	The Width field enables you to enter a numeric value to increase or decrease the default size of the Combobox in the user interface. For example, a Combobox has an approximate width of 120 pixels. To enlarge the width of the box, you must enter a value greater than 120 in the Width field.
Select Attribute	The values displayed in this field (if available) are bound to the data source and attribute you select for the control. A Select Attribute value displays in the textbox portion for the Combobox and is a synonym or abbreviation of the Display Attribute. For example, if you create a Combobox called Status, you might default or define values like WAPPR for Waiting for Approval and APPR for Approved.
Display Attribute	The values displayed in this field (if available) are bound to the data source and attribute you select for the control. Display Attribute values display in the pull-down menu for the Combobox. For example, if you create a Combobox called Status, you might default or define values like Waiting for Approval and Approved.

Attribute Name	Description
Attribute	<p>You can associate a Combobox with a database attribute by using the Select Value window. To create the example for this control, you can enter "=wpmaterial" in the Object field to display all the attributes in the WPMATERIAL table. Select "LINETYPE" to inherit the "Line Type" label for the Combobox. Not all attributes you select will have a database default label value.</p>
Input Mode	<p>Select one of the options listed below for the Input Mode:</p> <ul style="list-style-type: none"> <li>▼ Default</li> <li>▼ Password</li> <li>▼ Password Readonly</li> <li>▼ Password Required</li> <li>▼ Query</li> <li>▼ Readonly</li> <li>▼ Required</li> </ul> <p>The Default option sets the Input Mode for all controls to "edit." If you do not select an Input Mode for a control, the system automatically sets the mode to read/write so that you can both enter or display information in the field. If you enter information in a read/write text field, it posts to the database column specified by the Attribute property.</p> <p>Input Mode Option Descriptions:</p> <p><b>Default</b> – sets control to "edit" so you can display or input values and mode is read/write</p> <p><b>Password</b> – sets control to display asterisks (blanked out) for displayed or input values and mode is read/write</p> <p><b>Password Readonly</b> – sets control to display asterisks and mode is read only</p> <p><b>Password Required</b> – sets control to display asterisks and the control is required</p> <p><b>Query</b> – sets control to display a typed value and mode is read only</p> <p><b>Readonly</b> – sets control to display grayed-out text and mode is read only</p> <p><b>Required</b> – sets control to display a typed value and record cannot be saved unless there is a value in the control</p>

Attribute Name	Description
Data Change Event	<p>You can configure an input control to perform either of the following actions based on any data change to the control:</p> <ul style="list-style-type: none"> <li>▼ REFRESHTABLE</li> <li>▼ RESETCHILDREN</li> </ul> <p>Choosing REFRESHTABLE forces the table that is the data source for the input control to refresh its data. Choosing RESETCHILDREN forces any children of the control's data bean to delete and reacquire their MBOSetRemote references.</p>

## Data Source



A Data Source is a miscellaneous type of control that is mostly used in dialogs. Unlike most controls, it is transparent and does not render anything visible in the user interface like a Textbox or a Table. A Data Source control functions as a hook to a business object that other controls can reference and use as its source for table data. If you reference an inserted Data Source control, it enables you to override the default data source for the presentation or the current Parent Data Source. The example for this control shows how the Belongs To field in the Apply Item Assembly Structure uses the "parentitem" Data Source ID to identify a parent asset number.

### Rules

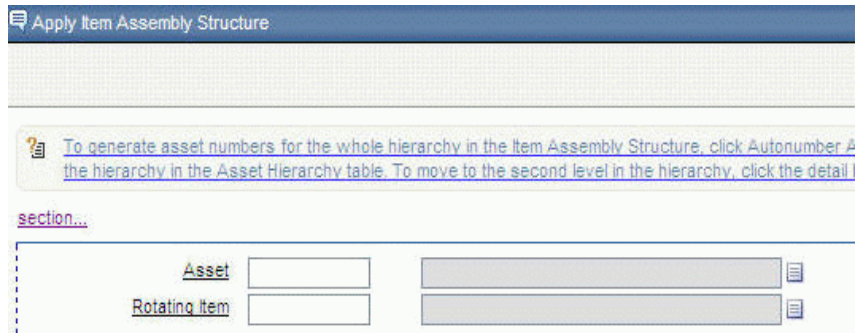
This is not a container control.

You can place this control inside the following container controls:

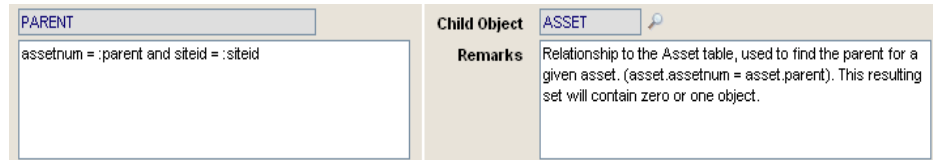
- ▼ Section
- ▼ Tab
- ▼ Table
- ▼ Client area

## Example

- 1 Data Source control as it appears in the Application Designer > Asset > Apply Item Assembly Structure window.



- 2 Parent relationship defined in Database Configuration for Asset.



## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Bean Class	Reserved for future use.

Attribute Name	Description
Business Object	<p>The main MBO reference for the application. For example, Work Order Tracking uses WOTRACK and Assets uses ASSET.</p> <p>Every application works with a MBO and MBOSet. The set of work orders displayed on the List tab in the Work Order Tracking application, for example, is a MBOSet. Each individual work order is a MBO. The presentation tag of an application defines the main MBO for the application. MBOs are responsible for the following types of functions:</p> <ul style="list-style-type: none"> <li>▼ Field validation</li> <li>▼ Updating the database</li> <li>▼ Business processing logic.</li> </ul> <p>By default, the presentation tag for an application defines the main MBO to use when saving or displaying data through the user interface. All controls in the application share the same MBO reference via the Parent Data Source ID property. Unless overridden, all controls added to an application bind to the Parent Data Source ID. If you wish an individual control to display and process information from a different MBO, you can enter a different Parent Data Source ID.</p>
Parent Data Source ID	<p>If you wish this control to inherit data from a source other than MAINRECORD, enter that value in this field. You might, for example, want to define a new parent data source in an application like Work Order Tracking where a table needs to pull data from an object other than WORKORDER. The Parent Data Source ID lets you build relationships with a different data source using Where Clauses.</p>
Relationship	<p>You enter a relationship value in this field. You can search for values in the Database Configuration &gt; Relationships tab. If you enter a relationship, it is automatically inherited by any controls that point to this data source.</p> <p>Relationships enable a control to display information from different business objects. For example, if you entered SPAREPART, the table will be able to display information based on the defined relationship. When you enter a relationship in the Table control, it automatically prefixes any child column attributes added to the table. So, for SPAREPART, you only have to enter the column attribute name and not the full relationship of SPAREPART &lt;attribute&gt;.</p>
Order By	Reserved for future use.
Where Clause	Where Clause

Attribute Name	Description
Listeners	Enter the ID(s) of one or more controls whose display values will be refreshed based on changes to this data source. Control ID(s) can reside anywhere inside the application. You must separate multiple values with a comma.

## Default Value



A Default Value is a miscellaneous type of control that lets you define the insert or query default value for any control. Unlike most controls, it is transparent and does not render anything visible in the user interface like a Textbox or a Table. The following XML from the Inventory presentation shows a query type defaultvalue ID of "rotateasset\_rotasset\_default" in Inventory > Inventory > Rotating Assets. The system configures this control to pull all the rotating asset records from the Asset table for the storeroom location you specify in the query search.

See the Example section for illustrations of how the system configures the Default Value Properties window for this control and how the Associated Assets search table appears in the Application Designer.

```
<table id="rotateasset_rotasset" relationship="ASSET"
label="Associated Assets">

<defaultvalue id="rotateasset_rotasset_default"
dataattribute="location" fromdatasrc="MAINRECORD"
fromattribute="location" defaulttype="query" />

- <tbody id="rotateasset_rotasset_tablebody" filterable="true"
displayrowsperpage="14">

  <tablecol id="rotateasset_rotasset_tablebody_1"
dataattribute="assetnum" />

  <tablecol id="rotateasset_rotasset_tablebody_2"
dataattribute="location" />

  <tablecol id="rotateasset_rotasset_tablebody_3"
dataattribute="location.description" sortable="false"

  />
  ...
```

### Rules

This is not a container control.

You can place this control inside the following container controls:

- ▼ Section
- ▼ Table
- ▼ Presentation



**NOTE** To view the presentation tag in the Application Designer, select **Toggle Show All Controls** from the **Select Action** menu.

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Attribute	<p>Select an attribute from a data source to update or query. You can set the update/query value using one of the following methods:</p> <ol style="list-style-type: none"> <li>1 Define a static value using the Value property.</li> <li>2 Read a value from the database by defining the From Attribute and From Data Source ID properties.</li> </ol> <p><b>NOTE</b> You can use the Select Value window to select an attribute from the MAINRECORD or any other table. If you enter "=" before the object name, the window displays only attributes for the table you specify. You can also enter wildcard characters like "%" before or after a text string in the Attribute or Object field to filter your search.</p>
Value	The alphanumeric value you type in this field becomes a static default value for an insert or query action using the specified Attribute property. In the Inventory application, for example, you might set attribute to LOCATION, Value to CENTRAL, and Default Type to QUERY. As a result, whenever you query inventory records the Storeroom field automatically defaults to CENTRAL. Also, you might set attribute to LASTCOST, Value to 32.95, and Default to INSERT. As a result, when you insert an inventory record the Last Cost field always defaults to 32.95.
From Data Source ID	Enter the Data Source ID for the From Attribute, if configured.
From Attribute	If you do not configure a static default value with the Value property, then you can enter a From Attribute that updates or queries the database with its value.
Default Type	Select a Default Type of either Insert or Query.

# Help Grid



A Help Grid is HTML text that you can insert on top of a page or in a window. The text can function as a label or provide instructions on how to enter data. A Help Grid can have a label and inline HTML text.

Help grids typically appear in windows within the user interface. To view windows in the Application Designer, open an application and click **Edit Dialogs** in the toolbar.

## Rules

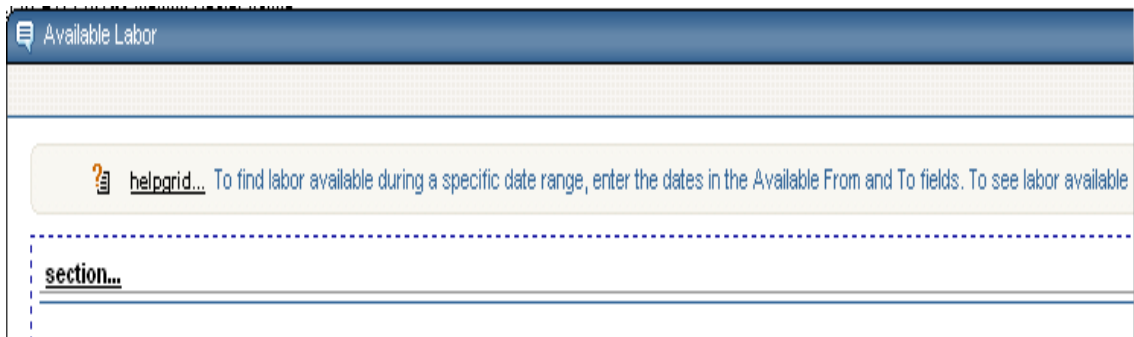
This is not a container control.

You can place this control inside the following container controls:

- ▼ Section
- ▼ Section Column
- ▼ Tab
- ▼ Dialogs

## Example

- 1 Open the Available Labor window. A portion of the Help Grid text appears below.



## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	The name you assign to the Help Grid control for a specific application. The label you enter is not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.

Attribute Name	Description
HTML Content	Informational or instructional help text displayed in the Help Grid control.

## HyperLink



The Hyperlink control supports text and graphics. This control lets you configure a label as a hyperlink. When you click on the label, it can automatically display a menu, a window, or it can take you to an entirely different place within or outside the application. You can, for example, configure a hyperlink to take you to a URL. The Start Center, Go To, and Sign Out links in the navigation bar are examples of hyperlinks.

### Rules

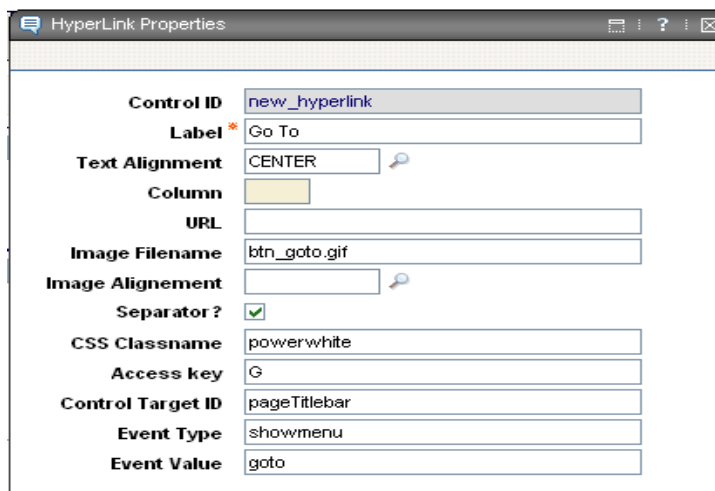
This is not a container control.

You can place this control inside the following container control:

- ▼ Section

### Example

- HyperLink Properties window configured to display the **Go To** menu.



- Presentation text for navigation bar hyperlink IDs in the library.xml system file.

```
< Bulletins
```

```
<hyperlink id="bboardlink" label="Bulletins:"
mboname="BULLETINBOARD" mxevent="showbboard"
classname="powerwhite" image="btn_bboard.gif" accesskey="B" />
```

Go To

```
<hyperlink id="gotolink" label="Go To" mxevent="showmenu"
targetid="pageTitlebar" eventvalue="goto" classname="powerwhite"
image="btn_goto.gif" accesskey="G" />
```

Reports

```
<hyperlink id="reportslink" label="Reports" mxevent="showmenu"
targetid="pageTitlebar" eventvalue="reportsmenu"
classname="powerwhite" image="btn_reporting.gif" accesskey="R" />
```

Start Center

```
<hyperlink id="sslink" label="Start Center" mxevent="startcenter"
classname="powerwhite" image="btn_startcenter.gif"
separator="false" accesskey="C" />
```

Profile

```
<hyperlink id="profilelink" label="Profile" mxevent="showmenu"
targetid="pageTitlebar" eventvalue="profile"
classname="powerwhite" image="btn_profile.gif" accesskey="P" />
```

Sign Out

```
<hyperlink id="signoutlink" label="Sign Out" mxevent="signout"
classname="powerwhite" image="btn_signout.gif" accesskey="S" />
```

Help

```
<hyperlink id="helplink" label="Help" mxevent="showmenu"
Targetid="pageTitlebar" eventvalue="help" classname="powerwhite"
image="btn_help.gif" accesskey="H" />
```

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	The name you assign to the Hyperlink control. This value displays in the user interface. You click the label to initiate the link.  The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Text Alignment	Select Left, Center, or Right to align the hyperlink label within a section.

Attribute Name	Description
Column	<p>Enter a numeric value in this field that will position the hyperlink at the beginning, middle, or end of a column depending on the Text Alignment setting. By default, each Section control has seven columns that are transparent to the user. For example, if you enter "3" in this field and "Center" for Text Alignment, then the system centers the hyperlink above or below column three.</p> <p>If the hyperlink you want to display is wider than its column, you cannot use the Text Alignment property. The system automatically extends the column width to contain the hyperlink.</p>
URL	<p>If you want the hyperlink to take you to a URL, enter the value here.</p>
Image Filename	<p>To add a graphic next to the hyperlink, enter a file name in this field. This field defaults to the path of the images directory so that you do not have to enter the full path for a graphic file.</p> <p>You can specify any type of graphic that a browser is capable of displaying, such as .gif, .jpeg, or .bmps. Some graphic files may require a plug-in.</p> <p><b>NOTE</b> You can place any graphic file into the system library of images. However, if you insert and apply a graphic file without a .gif extension, then you need to rebuild/deploy the .EAR file.</p>
Image Alignment	<p>If you associate an image with the Hyperlink control, you can align the image to the right or left of the control label by selecting Right or Left.</p>
Separator?	<p>Reserved for future use.</p>
CSS Classname	<p>Enter the class name for a cascading style sheet. The value you enter can change the layout and appearance of the hyperlink control. The "powerwhite" value entered in the example for this control changes the hyperlink font color to white.</p> <p>You can search for CSS classnames in the following directory:</p> <pre data-bbox="842 1665 1305 1724">&lt;root&gt;\applications\maximo\ maximouiweb\webmodule\webclient</pre>
Access Key	<p>Enter a letter from the hyperlink label to use as part of a keyboard shortcut. For example, the access key for the Go To hyperlink is G. Press ALT + G &lt;enter&gt; to drop down the main menu. The system underlines the access key in the user interface.</p>

Attribute Name	Description
Control Target ID	Enter a Control ID, if different from the current button control, to which the system sends the Event property (if defined) for processing. The Control Target ID for the Go To hyperlink example is "pageTitlebar.
Event Type	<p>You can link a hyperlink to a user interface operation or "event." When you click the hyperlink, it triggers the event you specify in this field. The Event Type for the Go To hyperlink is "showmenu." The "showmenu" event displays the <b>Go To</b> menu.</p> <p>To view all of the events defined in the system, you must export the system .xml files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;root&gt;\resources\presentation\system directory. When completed, you can search the XML files for event values using the text string "MXEVENT."</p> <p>The Application Designer processes events in the following order:</p> <ol style="list-style-type: none"> <li>1 If you specify a Target ID property, it sends the event to the Control ID specified in the property for processing.</li> <li>2 If you do not specify a Target ID property, it sends the event to the JSP main page of the application for processing. It calls the method (i.e., a function defined in a class) identified by the MXEVENT property to process the event.</li> <li>3 If 1 and 2 do not work, it tries to load the window associated with the specified event.</li> </ol>
Event Value	Defines the event called when you select a hyperlink. The event value for the <b>Go To</b> menu is "goto."

## Image



Images appear in many places throughout the user interface as visual markers that identify an application you can open or action you can perform. Image icons in the toolbar represent common functions like Save, Cut, Copy, and Paste. The settings in the Images Properties window enable you to edit the height, width, and alignment for a specified image file.

The image icons you see in the user interface reside in the following directory:

```
<root>\applications\maximo\maximouiweb\webmodule\webclient\images
```

You can access the graphic files in the images directory using the Image control in the Application Designer.

## Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Section

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Filename	<p>To add a new graphic to the user interface, enter a file name in this field. If you want to edit an existing graphic, its file name appears here. This field defaults to the path of the images directory so that you do not have to enter the full path for a graphic file.</p> <p>You can specify any type of graphic that a browser is capable of displaying, such as .gif, .jpeg, or .bmps. Some graphic files may require a plug-in.</p> <p><b>NOTE</b> You can place any graphic file into the system library of images. However, if you insert a graphic file without a .gif extension into the system, then you need to rebuild/ deploy the .EAR file.</p>
Height	Enter a numeric value in this field that will increase or decrease the height of the image.
Width	<p>Enter a numeric value in this field that will increase or decrease the width of the image.</p> <p>If you change the default width or height of a graphic, it can result in image distortion.</p>
Align Image	Choose Left, Center, or Right to align an image with the left, center, or right margin of a section.
Column	This field subdivides the section in which the image appears into four equal segments or columns that are transparent in the user interface. You can change the position of the image by moving the graphic into one of the columns.

# Include



The Include control is one of the basic controls that the system uses together with presentation, page, and client area to create an XML presentation file. The presentation file represents the graphical layout for an application. The presentation control is a container that can include only one page control. The page control differentiates the main page, which uses the entire client area of the browser, from the dialogs of the application. Dialogs display as "popups" in the presentation. A presentation can define its own dialogs or use the Include control to obtain dialog definitions from the library.xml file. The presentation file also uses the Include control to acquire common application functions like header and footer from library.xml.

The client area control differentiates the body of a page or window from its header and footer. The client area must contain one or more Sections, a Table, a Tabgroup or both, that act as containers for all the other controls that comprise the presentation.

The presentation for applications like Incidents, for example, use the Include control to obtain table definitions for the Work and Communication Logs from the library.xml system file. The Include control enables you to import pre-defined layouts from library.xml into different presentations.

The following example is an extract from the Work Order Tracking application presentation that shows how the include control imports the standard header and footer definitions from library.xml.

```
<presentation id= "wotrack" ... version="6.0.0">
    <page id= "mainrec">
        <include id="pageHeader"/>
        <clientarea id= "clientarea">
            ...
            ...
        </clientarea>
        <include id="pageFooter"/>
    </page>
</presentation>
```

## Rules

This is not a container control.

You can place this control inside the following container controls:

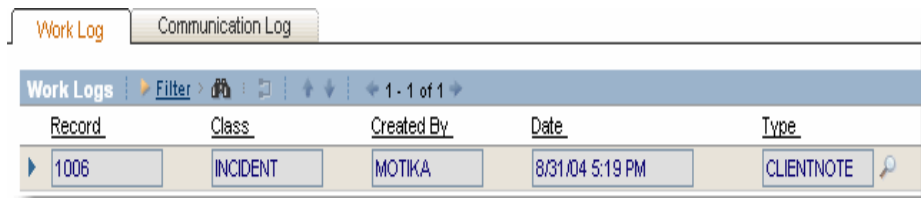
- ▼ Page
- ▼ Tab
- ▼ Section



**NOTE** To view the page tag in the Application Designer, select **Toggle Show All Controls** from the **Select Action** menu.

## Example

1 Work Log table format inherited from "comlog" in library.xml.



## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Control ID to Clone	Enter a Control ID from library.xml. This property enables you to include the same library element multiple times in a presentation.

# Listbox



A Listbox is a text type of control that displays a list of selected items in a box that is always open. The Listbox has a text field component or label that is visible in the user interface.

Listbox controls have a Default Label property that enables the control to inherit a read-only label from the attribute you configure for the control. The Default Label value comes from the database object. Alternatively, you can override an inherited Default Label by entering a different value in the Label property. This value is local to the presentation file of an application. Labels you enter at the presentation level are not stored in the MAXATTRIBUTE or MAXLABELS table.

Listboxes display information from the paired Business Object Name and attribute values you specify in the properties window. Additionally, you can filter what information displays and how it is sorted using the Where Clause and Order By property values.

## Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Section

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Default Label	This label defaults to the text value associated with the database attribute you bound to the Listbox. This is a read only field.
Label	You can enter a text value in this field that overrides the Default Label value. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Height	Enter a numeric value in this field that will increase or decrease the height of the Listbox.
Width	Enter a numeric value in this field that will increase or decrease the width of the Listbox.
Bean Class	Reserved for future use.
Attribute	You can associate a Listbox with a database attribute by using the Select Value window. If the attribute you select has a default label value, it will display in the Default Label field.
Input Mode	Select one of the options listed below for the Input Mode: <ul style="list-style-type: none"> <li>▼ Default</li> <li>▼ Password</li> <li>▼ Password Readonly</li> <li>▼ Password Required</li> <li>▼ Query</li> <li>▼ Readonly</li> <li>▼ Required</li> </ul>

Attribute Name	Description
	<p>The Default option sets the Input Mode for all controls to "edit." If you do not select an Input Mode for a control, the system automatically sets the mode to read/write so that you can both enter or display information in the field. If you enter information in a read/write text field, it posts to the database column specified by the Attribute property.</p> <p>Input Mode Option Descriptions:</p> <p><b>Default</b> – sets control to "edit" so you can display or input values and mode is read/write</p> <p><b>Password</b> – sets control to display asterisks (blacked out) for displayed or input values and mode is read/write</p> <p><b>Password Readonly</b> – sets control to display asterisks and mode is read only</p> <p><b>Password Required</b> – sets control to display asterisks and the control is required</p> <p><b>Query</b> – sets control to display a typed value and mode is read only</p> <p><b>Readonly</b> – sets control to display grayed-out text and mode is read only</p> <p><b>Required</b> – sets control to display a typed value and record cannot be saved unless there is a value in the control</p>
Business Object Name	Reserved for future use.
Parent Data Source ID	Reserved for future use.
Order By	Reserved for future use.
Where Clause	Reserved for future use.
Auto-Refresh	Select this box if you want the information in the Listbox to automatically refresh whenever there is a change to the MBO.
Key Attribute	Reserved for future use.
Key Value	Reserved for future use.

# Menubar



The Menubar is similar in appearance to the Appbar. The bar that appears at the top of the List tab in most applications is an example of a menu bar. This menu bar can contain signature security options like Advanced Search, Save Query, and Bookmark. The options in the menu bar usually contain drop-down menus with additional option choices. Advanced Search, for example, has the More Search Fields, Where Clause, and View Search Tips options.

## Rules

This is not a container control.

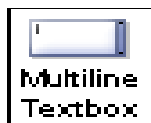
You can place this control inside the following container control:

- ▼ Tab

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	The name you assign to the Menubar control. This value does not display in the user interface.  The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Source Method	The piece of code functionality that calls the menu bar, its signature security options, and layout. The method that calls the standard search menu bar is "getAppSearchOptions."
Event Value	Defines the event called when you select an option on the menu bar. For example, the Event Value for the search menu bar is "search."

# Multiline Textbox



A Multiline Textbox is a control that displays information or accepts user input in a large box that holds several lines of text. The textbox has a text field component or label that is visible in the user interface. Textboxes are often linked to select value lookups and detailed menus. A select value lookup lets you select and then display a single value from a list of records. A textbox linked to a detailed menu can contain diverse options, such as a select value lookup, jumps to other applications, or views of different records. The

Menu Type property you configure for a Textbox controls the format and content of the detailed menu.

Textbox controls have a Default Label property that enables the control to inherit a read-only label from the attribute you configure for the control. The Default Label value comes from the database object. Alternatively, you can override an inherited Default Label by entering a different value in the Label property. This value is local to the presentation file of the application. Labels you enter at the presentation level are not stored in the MAXATTRIBUTE or MAXLABELS table.

Textboxes display table information from the Data Source ID and Attribute values you specify in the properties window.

## Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Section

## Properties

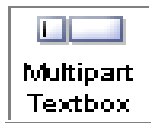
Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Default Label	This label defaults to the text value associated with the database attribute you bound to the textbox. This is a read only field.
Label	You can enter a text value in this field that overrides the Default Label value. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Hide Label?	By default, the label value always appears in the user interface. Select this box to hide the label.
Columns	The number of columns in the Multiline Textbox. Most Multiline Textboxes in the system, including the preceding example, have a column value of "35." To increase or decrease the width of the box, enter a value greater than or less than "35."
Rows	Enter an integer for the number of rows you want in the Multiline Textbox.
Attribute	You can associate a Textbox with a database attribute by using the Select Value window. If the attribute you select has a default label value, it will display in the Default Label field. The Attribute for the preceding example is SELECTSTMT from the KPIMAIN table.

Attribute Name	Description
Data Source ID	<p>The Data Source ID for this control. It points to the main MBO or object that the control uses for entering and displaying data. Each application links to a main object that represents the Parent Data Source for all child controls in the application. Individual or groups of controls within an application can inherit attributes from objects that are not the primary data source. In the preceding example, the table associated with MAINRECORED is KPIMAIN.</p> <p>See the Data Source ID description for the Combobox control on page 60 for additional information on data sources.</p>
Lookup	<p>You can enter a value in the Lookup property that links the Textbox to a pre-formatted lookup table. The lookup table pulls records from the main table. Users can populate the Textbox by selecting a record from the table. The value, called a lookup ID, defines the XML presentation content and format of a select value table. WORKORDERISTASK, for example, is the Lookup property for the Parent WO textbox in Work Order Tracking. Lookup IDs reside in a system XML file called lookups.xml.</p> <p>You cannot edit system XML files from the Application Designer. However, you can view the content of these types of files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;root&gt;\resources\presentation\system directory.</p>
Menu Type	<p>You can enter a value in the Menu Type property that links the textbox to a pre-formatted drop-down menu. The value, called a menu ID, defines the XML presentation content and format of a drop-down menu. WORKORDER, for example, is the Menu Type for the Parent WO textbox in Work Order Tracking. Menu IDs reside in a system XML file called menus.xml.</p> <p>You cannot edit menus.xml from the Application Designer. However, you can view the content of this file or other system .xml files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;root&gt;\resources\presentation\system directory.</p>

Attribute Name	Description
Input Mode	<p>Select one of the options listed below for the Input Mode:</p> <ul style="list-style-type: none"> <li>▼ Default</li> <li>▼ Password</li> <li>▼ Password Readonly</li> <li>▼ Password Required</li> <li>▼ Query</li> <li>▼ Readonly</li> <li>▼ Required</li> </ul> <p>The Default option sets the Input Mode for all controls to "edit." If you do not select an Input Mode for a control, the system automatically sets the mode to read/write so that you can both enter or display information in the field. If you enter information in a read/write text field, it posts to the database column specified by the Attribute property.</p> <p>Input Mode Option Descriptions:</p> <p><b>Default</b> – sets control to "edit" so you can display or input values and mode is read/write</p> <p><b>Password</b> – sets control to display asterisks (blacked out) for displayed or input values and mode is read/write</p> <p><b>Password Readonly</b> – sets control to display asterisks and mode is read only</p> <p><b>Password Required</b> – sets control to display asterisks and the control is required</p> <p><b>Query</b> – sets control to display a typed value and mode is read only</p> <p><b>Readonly</b> – sets control to display grayed-out text and mode is read only</p> <p><b>Required</b> – sets control to display a typed value and record cannot be saved unless there is a value in the control</p>

Attribute Name	Description
Data Change Event	<p>You can configure an input control to perform either of the following actions based on any data change to the control:</p> <ul style="list-style-type: none"> <li>▼ REFRESHTABLE</li> <li>▼ RESETCHILDREN</li> </ul> <p>Choosing REFRESHTABLE forces the table that is the data source for the input control to refresh its data. Choosing RESETCHILDREN forces any children of the control's data bean to delete and reacquire their MBOSetRemote references.</p>
Move to Data Source ID	<p>The value you enter lets you move to a record stored in a different object. You can select any Data Source ID from the presentation. Refer to the Data Source ID property description.</p> <p>By default, the system sends the Move To event to the ResultsBean, which contains a set of records displayed in the List tab. The Move To event affects the Move to Data Source ID object.</p>

## Multipart Textbox



A Multipart Textbox is a two-part control that contains side-by-side textboxes that can display information from different sources. You can input information, populate a box by selecting a value from a list, or simply display an attribute description from a table. Typically, a Multipart Textbox consists of a text label, a textbox pull-down menu, and an expanded description for an attribute. The source of the information in each textbox can come from separate tables. The layout of the control places information from different sources in the same area of the screen.

Textboxes have labels that are usually visible in the user interface. Labels describe the type of information displayed in the box. Textboxes are often linked to select value lookups and detailed menus. A select value lookup lets you select and then display a single value from a list of records. A Textbox linked to a detailed menu can contain diverse options, such as a Select Value window, jumps to other applications, or views of different records. The Menu Type property you configure for a textbox controls the format and content of the detailed menu.

Textbox controls have a Default Label property that enables the control to inherit a read-only label from the attribute you configure for the control. The Default Label value comes from the database object. Alternatively, you can override an inherited Default Label by entering a different value in the Label property. This value is local to the presentation file of an application. Labels you enter at the presentation level are not stored in the MAXATTRIBUTE or MAXLABELS table.

The Multipart Textbox enables you to display a second attribute from the same or a different database table on the same line. Most applications typically use the second part of a Multipart Textbox to display a full description for an asset,



location, or work order number displayed in the first textbox. The example for this control shows that Asset number 11230 is an Emergency Generator.

Textboxes display table information from the Data Source ID and attribute values you specify in the properties window.

## Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Section

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Default Label	This label defaults to the text value associated with the database attribute you bound to the textbox. This is a read only field.
Label	You can enter a text value in this field that overrides the Default Label value. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Attribute	You can associate a textbox with a database attribute by using the Select Value window. If the attribute you select has a default label value, it will display in the Default Label field. The example for this control displays a record number of 11230 with a field label of Asset.
Menu Type	<p>You can enter a value in the Menu Type property that links the textbox to a pre-formatted drop-down menu. The value, called a menu ID, defines the XML presentation content and format of a drop-down menu. This example uses the pre-formatted menu for assets called ASSETMAIN. Menu IDs reside in a system XML file called menus.xml.</p> <p>You cannot edit this file from the Application Designer. However, you can view the content of this file or other system .xml files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;root&gt;\resources\presentation\system directory.</p>

Attribute Name	Description
Lookup	<p>You can enter a value in the Lookup property that links the textbox to a pre-formatted lookup table. The lookup table pulls records from the main table. Users can populate the textbox by selecting a record from the table. The value, called a lookup ID, defines the XML presentation content and format of a select value table. ASSET, for example, is the Lookup property for the Multipart Textbox example. Lookup IDs reside in a system XML file called lookups.xml.</p> <p>You cannot edit system XML files from the Application Designer. However, you can view the content of these types of files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;root&gt;\resources\presentation\system directory.</p>

Attribute Name	Description
Input Mode	<p>Select one of the options listed below for the Input Mode:</p> <ul style="list-style-type: none"> <li>▼ Default</li> <li>▼ Password</li> <li>▼ Password Readonly</li> <li>▼ Password Required</li> <li>▼ Query</li> <li>▼ Readonly</li> <li>▼ Required</li> </ul> <p>The Default option sets the Input Mode for all controls to "edit." If you do not select an Input Mode for a control, the system automatically sets the mode to read/write so that you can both enter or display information in the field. If you enter information in a read/write text field, it posts to the database column specified by the Attribute property.</p> <p>Input Mode Option Descriptions:</p> <p><b>Default</b> – sets control to "edit" so you can display or input values and mode is read/write</p> <p><b>Password</b> – sets control to display asterisks (blanked out) for displayed or input values and mode is read/write</p> <p><b>Password Readonly</b> – sets control to display asterisks and mode is read only</p> <p><b>Password Required</b> – sets control to display asterisks and the control is required</p> <p><b>Query</b> – sets control to display a typed value and mode is read only</p> <p><b>Readonly</b> – sets control to display grayed-out text and mode is read only</p> <p><b>Required</b> – sets control to display a typed value and record cannot be saved unless there is a value in the control</p>

Attribute Name	Description
Change Event	<p>You can configure an input control to perform either of the following actions based on any data change to the control:</p> <ul style="list-style-type: none"> <li>▼ REFRESHTABLE</li> <li>▼ RESETPCHILDREN</li> </ul> <p>Choosing REFRESHTABLE forces the table that is the data source for the input control to refresh its data. Choosing RESETPCHILDREN forces any children of the control's data bean to delete and reacquire their MBOSetRemote references.</p>
Turn Smart Fill Off?	<p>By default, the system enables the "Smart Fill" feature. To disable the smart fill feature, clear the check box. This feature enables you to enter a partial value in a text box field. The system attempts to match the partial text string entered to a valid value or values in the database. If the system finds one matching value, it automatically populates the field so that you can continue entering information. If the system finds more than one matching value, it displays a Select Value window with a list of matching values. If you clear the smart fill feature, The system does not validate the data you enter in the textbox.</p>
GOTO Application	<p>You can configure a pull-down menu associated with a textbox to have options that will take the user to different applications. Use the Select Value window for this property to choose application names that will appear in the pull-down menu. In the example for this control, you would see "Go To Assets" as an option in the pull-down menu because the GOTO Application property lists "ASSET" as a value.</p>
Move to Data Source ID	<p>The value you enter lets you move to a record stored in a different object. You can select any Data Source ID from the presentation. Refer to the Data Source ID definition below.</p> <p>By default, the system sends the Move To event to the ResultsBean, which contains a set of records displayed in the List tab. The Move To event affects the Move to Data Source ID object.</p>
Attribute for Part 2	<p>You can associate the second part of a Multipart Textbox with a database attribute by using the Select Value window. Typically, you configure the second part of a multipart control to display the full description information for the record selected in the first part. The example for this control displays Emergency Generator, which is the ASSET.DESCRPTION for the asset number selected.</p>

Attribute Name	Description
Lookup for Part 2	You can enter a value in the Lookup for Part 2 property that links the second part of the Multipart Textbox to a pre-formatted lookup table. See the previous Lookup property description for additional information.
Input Mode for Part 2	See the previous Input Mode property description for information about this field. In this example and in most cases where the second part of a Multipart Textbox contains a full description, the property value will be READONLY.
Long Description Input Mode	If Attribute for Part 2 is a long description, you can select this box to set the description to read only. If you leave this field cleared (the default), you can edit the long description.
Data Source ID	<p>The Data Source ID for this control. It points to the main MBO or object that the control uses for entering and displaying data. Each application links to a main object that represents the Parent Data Source for all child controls in the application. Individual or groups of controls within an application can inherit attributes from objects that are not the primary data source. In the case of a Multipart Textbox, you must configure the two different attributes that comprise the control to pull data from the same object.</p> <p>See the Data Source ID description for the Combobox control for additional information on data sources.</p>

## Parameter Value



The properties for the Parameter Value control define the dynamic values that display in a Section Header. For example, the task table in Work Order Tracking > Plans tab has a variable value similar to "(0)," where the variable values displayed to the user reflects the attribute configuration setting (i.e., WONUM) in the Parameter Value control.

When you move a section, all of its children, including Section Header and the Parameter Value and Parameter Values controls, automatically move with it.

A Parameter Value control works with the following:

- ▼ Parameter Values
- ▼ Section Header

A Parameter Value is transparent in the user interface. Section headers along with the Parameter Value and Parameter Values controls enable you to display dynamic values in the label for a Table or Section. In the Section Header example,

the settings you configure in the Section Header and Parameter Values set xxx in the Table Header to the current WONUM (i.e., 1000).

## Rules

You can place the following controls in this container control:

- ▼ Parameter Values

You can place this control inside the following container controls:

- ▼ Section
- ▼ Table

You can associate a Parameter Value with Section Headers inserted into Table or section controls. To create a Section Header with dynamic values, complete the following steps:

- 1 Insert a Section Header control into a Section or Table control.
- 2 Drag the Parameter Values control into the Section Header container.
- 3 A Parameter Value control automatically appears with the Parameter Values control.

## Example

See Section Header example on page 105.

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Parameter Position	This property points to the Parameter Values Property field you want to replace. See the Parameter Values Property field for more information.
Attribute	Select a data attribute from a data source to display as the Label property in the Table or Section Header.  <b>NOTE</b> You can use the Select Value window to select an attribute from the MAINRECORD or any other table in the system. If you enter "=" before the Object name, the window displays only attributes for the table that you specify. You can also enter wildcard characters like "%" before or after a text string in the Attribute or Object field to filter your search.

Attribute Name	Description
Data Source ID	<p>If you want this control to use a different Data Source ID than the default of MAINRECORD, then enter that value here.</p> <p>See the Data Source ID description for the Combobox control for additional information on data sources.</p>

## Parameter Values



Parameter Values are container controls that attach to a Section Header and can reside in a Table or Section. This control can hold dynamic values that represent replaceable values in the Section Header label. The Plans tab in the Work Order application, for example, uses the Parameter and Parameter Values controls to display children and tasks of work orders.

The property attribute in the Parameter Values control (e.g., the default is "label") contains the dynamic values defined in the Parameter Value control. For example, the task table in Work Order Tracking > Plans tab has a variable value similar to "(0)," where the variable values displayed to the user is a result of the configuration settings in the Parameter Value and Values controls. The property attribute for the Parameter Values control, for example, has a default value of "label."

When you move a section, all of its children, including Section Header and the Parameter Value and Parameter Values controls, automatically move with it.

A Parameter Values control works with the following:

- ▼ Parameter Value
- ▼ Section Header

Parameter Values are transparent in the user interface. Section Headers along with the Parameter Value and Parameter Values controls enable you to display dynamic values in the label for a Table or Section. In the Section Header example, the settings you configure in the Section Header and Parameter Values set xxx in the table header to the current WONUM (i.e., 1000).

### Rules

You can place the following controls in this container control:

- ▼ Parameter Value

You can place this control inside the following container controls:

- ▼ Section
- ▼ Table

You can associate Parameter Values with Section Headers inserted into Table or Section controls. To create a Section Header with dynamic values, complete the following steps:

- 1 Insert a Section Header control into a Section or Table control.
- 2 Drag the Parameter Values control into the Section Header container.
- 3 A Parameter Value control automatically appears with the Parameter Values control.

### Example

See the Section Header example on page 105.

### Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Property	<p>The Property field has a default value of "Label," which is a variable that displays dynamic values inherited from the Parameter Value control in the Table or Section control.</p> <p>For example, the sample XML from Work Order Tracking &gt; Plans shown below illustrates how Parameter Values and Parameter Value work together to populate the "Children of Work Order (0)" heading with work order numbers.</p> <pre>&lt; table id="plans_children_table" orderby="wosequence"  label="Children of Work Order {0}" collapsible="true"collapsed="true" relationship="SHOWCHILDREN" beanclass="psdi.webclient.beans.workorder. SelectWORecordsBean"&gt;  &lt;sectionheader id="plans_children_table_sectionheader"&gt;  &lt;paramvalues id="plans_children_table_1_sec_params" property="label"&gt;  &lt;paramvalue id="plans_children_table_1_sec_param_1" position="0" datasrc="MAINRECORD" dataattribute="wonum" /&gt;</pre>



Attribute Name	Description
	<p>The XML shows how (0) is dynamically updated to display the current work order according to the value of the Position attribute. The first position value is 0, the second is 1, and the third is 2, and so on. If the work order in the first position was 1003, then the header for this table would display "Children of Work Order 1003.</p>
	<p>The key fields in the XML are:</p>
	<p>Orderby = wosequence</p>
	<p>SQL statement that sorts work orders to display in numeric sequence</p>
	<p>Relationship = SHOWCHILDREN</p>
	<p>Defines database relationship for showing children work orders</p>
	<p>Property = label</p>
	<p>The Parameter Values Property field that is dynamically updated and passed to the Table Header for display</p>
	<p>Position = 0</p>
	<p>Select the work order in the first position</p>
	<p>Datasource ID = MAINRECORD</p>
	<p>Points to the default data source for the Work Order Tracking application (WORKORDER)</p>
	<p>Dataattribute = wonum</p>
	<p>Defines the data value (e.g., work order number) to display in the Table Header</p>

## Pushbutton



A Pushbutton control is a rectangular box with round corners that looks like a button. You push a button to initiate an event like invoking a window with a list of records or displaying an empty row for entering a new record.

### Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Button Group

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	<p>The name you assign to the button control. The label value identifies the control to the user. If you do not enter a label, the button is blank in the user interface.</p> <p>The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.</p>
Image Filename	<p>Enter the name of a button-related graphic file, such as btn_next.gif, to insert the image inside of the button. For example, if you have a button called Next, you may want to insert an arrow icon using btn_next.gif.</p> <p>You can specify icons used in the system that reside in the following directory:</p> <pre>&lt;root&gt;\applications\maximo\maximouiweb\webmodule\webclient\images</pre>
Default Button?	Select this field to make a button the active button in a Button Group. The system highlights the label text of the active button in the user interface. You may want to make the most frequently used button, for example, New Row, the default button.
Menu Type	<p>You can enter a value in the Menu Type property that links the button to a pre-formatted drop-down menu. The value, called a menu ID, defines the XML presentation content and format of a drop-down menu.</p> <p>Menu IDs reside in a system XML file called menus.xml. You cannot edit this file from the Application Designer. However, you can view the content of this file or other system .xml files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the</p> <pre>&lt;root&gt;\resources\presentation\system directory.</pre>

Attribute Name	Description
Event	<p>You can link a button to a user interface operation or "event." When you press the button, it triggers the event you specify in this field.</p> <p>To view all of the events defined in the system, you must export the system .xml files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;root&gt;\resources\presentation\system directory. When completed, you can search the XML files for event values using the text string "MXEVENT."</p> <p>The Application Designer processes events in the following order:</p> <ol style="list-style-type: none"> <li>1 If you specify a Target ID property, it sends the event to the Control ID specified in the property for processing.</li> <li>2 If you do not specify a Target ID property, it sends the event to the JSP main page of the application for processing. It calls the method (i.e., a function defined in a class) identified by the MXEVENT property to process the event.</li> <li>3 If 1 and 2 do not work, it tries to load the window associated with the specified event.</li> </ol> <p>For example, in Work Order Tracking &gt; Plans there is a "Children of Work Order xxxx" table that has a button called <b>Select Assets</b>. When you press <b>Select Assets</b>, a Select Assets window opens that displays a list of asset records. The window displays because the system processes the "selectallassets" event specified for this button using the logic described above.</p>
Value	Reserved for future use.
Target ID	Enter a Control ID, if different from the current button control, to which the system sends the Event property (if defined) for processing.

## Radio Button



A Radio Button is a control that allows users to choose one of a predefined set of options. Radio Buttons appear in groups of two or more. Radio buttons typically appear as little circles that are either empty (for unselected) or contain a dot (selected).

When you select a Radio Button, the previously selected button in the same group is deselected.

## Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Radio Button Group

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Default Label	This label defaults to the text value associated with the database attribute you bound to the Radio Button Group. This is a read only field.
Label	<p>The name you assign to the Radio Button Group control. This value overrides an entry in the Default Label field. The label value identifies the control to the user. If you do not enter a label or inherit a default label, the Radio Button Group title is transparent in the user interface.</p> <p>The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.</p>
Button Value	Enter a data attribute for an individual button control. If you do not automatically populate a Radio Button Group with a predefined list of values linked to a data attribute, then you can enter individual values for each button that you manually add to a group.

## Radio Button Group



A Radio Button Group is a container control that holds Radio Buttons. The Radio Button Group defines the layout for the buttons it holds, including a group title, text alignment for button labels, and whether or not to show a border. You can bind a Radio Button Group to a database attribute that automatically populates the group with buttons, label values, and so on. Binding to a database attribute saves time because you do not have to define the individual Radio Button controls for the group. When you make a selection in the Radio Button Group, the value automatically updates the attribute in the database. If you link a Radio Button Group to a database attribute, then you can edit the group in the Database Configuration and Domains applications. The system binds most Radio Button Groups to an attribute.

## Rules

You can place the following control in this container control:

- ▼ Radio Button

You can place this control inside the following container control:

- ▼ Section

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Default Label	This label defaults to the text value associated with the database attribute you bound to the Radio Button Group. This is a read only field.
Label	The name you assign to the Radio Button Group control. This value overrides an entry in the Default Label field. The label value identifies the control to the user. If you do not enter a label or inherit a default label, the Radio Button Group title is transparent in the user interface. You can enter a text value in this field that overrides the Default Label value. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Hide Label?	Select this field to hide the Label or Default Label value in the user interface.
Button Orientation	Enter a value of Vertical or Horizontal to align the radio buttons in a group either vertically or horizontally.
Text Alignment	Select a value of Left, Center, or Right to position button labels relative to the radio button icon. The text label for the button group always displays to the left of any radio buttons you add to the group.
Show Border?	Select this field to draw a rectangular border around the Radio Button Group.
Attribute	Select an attribute from a data source to display its label value as the Default Label property for the Radio Button Group. Not all attributes you select will have a database default label value. Additionally, when you bind a Radio Button Group to an attribute, you can inherit a list of Radio Buttons without the need to define the buttons individually with the Radio Button control.

Attribute Name	Description
Input Mode	<p>Select one of the options listed below for the Input Mode:</p> <ul style="list-style-type: none"> <li>▼ Default</li> <li>▼ Password</li> <li>▼ Password Readonly</li> <li>▼ Password Required</li> <li>▼ Query</li> <li>▼ Readonly</li> <li>▼ Required</li> </ul> <p>The Default option sets the Input Mode for all controls to edit. If you do not select an Input Mode for a control, the system automatically sets the mode to read/write so that you can both enter or display information in the field. If you enter information in a read/write text field, it posts to the database column specified by the Attribute property.</p> <p>Input Mode Option Descriptions:</p> <p><b>Default</b> – sets control to "edit" so you can display or input values and mode is read/write</p> <p><b>Password</b> – sets control to display asterisks (blanked out) for displayed or input values and mode is read/write</p> <p><b>Password Readonly</b> – sets control to display asterisks and mode is read only</p> <p><b>Password Required</b> – sets control to display asterisks and the control is required</p> <p><b>Query</b> – sets control to display a typed value and mode is read only</p> <p><b>Readonly</b> – sets control to display grayed-out text and mode is read only</p> <p><b>Required</b> – sets control to display a typed value and record cannot be saved unless there is a value in the control</p>

Attribute Name	Description
Change Event	<p>You can configure an input control to perform either of the following actions based on any data change to the control:</p> <ul style="list-style-type: none"> <li>▼ REFRESHTABLE</li> <li>▼ RESETPCHILDREN</li> </ul> <p>Choosing REFRESHTABLE forces the table that is the data source for the input control to refresh its data. Choosing RESETPCHILDREN forces any children of the control's data bean to delete and reacquire their MBOSetRemote references.</p>
Value Attribute	<p>Optionally, if you do not automatically populate a Radio Button Group with a list of buttons, you can enter a data value in this field that automatically updates the database when you select a button. For example, if you have a button with a Description Attribute of "Waiting for Approval," you might define the Value Attribute as "WAPPR." The system writes "WAPPR" to the database and displays "Waiting for Approval" in the user interface.</p>
Description Attribute	<p>Optionally, you can lookup or enter a data attribute that displays a text label for the Value Attribute in the user interface. See Value Attribute.</p>
Data Source ID	<p>If you want this control to use a different Data Source ID than the default of MAINRECORD, then enter that value here.</p> <p>See the Data Source ID description for the Combobox control for additional information on data sources.</p>

## Record Image



You can use the Record Image control to associate and display an image with a record. This control lets you embed an expandable thumbnail image. Adding an image to a record provides a graphical reference. The image provides a visual way to identify products and services quickly and efficiently. You can click on an image to view its full size. Also, you can drag the mouse over the image to display a required text description. The mouse over or ALT Text displayed is the record description. The default label for all images is "Click to enlarge," which you can modify with the Application Designer.

You can add one image or thumbnail per record. Each record, however, can contain multiple image containers with the same image. You might, for example, want the same image to appear in more than one application tab. Adding a Record Image to an application inserts a placeholder but not the actual image. You insert an image using **Add/Modify Image** from the **Select Action** menu of the application. If needed, you can use an autodb script to add the Select Action menu item. You can add images that have a .jpg or .gif extension.

Additionally, you can view images associated with records even if the image is not part of the application with the Record Image control. Use **View Image** in the **Details** menu to open an image in a pop-up window. To add **View Image** to a **Details** menu, edit the menu.xml file. When you are finished, use **Import Application Definition** in the Application Designer to import the menu.xml file.

## Rules

This is not a container control.

You can place this control inside any container control.

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
CSS Class Name	Enter the class name for a cascading style sheet. The value you enter can change the layout and appearance of the Record Image control.
Height	Enter a numeric value in this field that will increase or decrease the height of this control.
Width	Enter a numeric value to increase or decrease the width of this control.
Data Source ID	If you want this control to use a different Data Source ID than the default of MAINRECORD, then enter that value here.
Is Thumbnail	Identifies whether or not the image is an expandable thumbnail.
ALT Text on Image	Identifies the record description during a mouse over or ALT Text action.
Label	The name you assign to the Record Image control. The label value identifies the control to the user. If you do not enter a label, the title defaults to "Click to enlarge" in the user interface.  The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Signature Option	Reserved for future use.



Attribute Name	Description
Data Source ID	<p>If you want this control to use a different Data Source ID than the default of MAINRECORD, then enter that value here.</p> <p>Each application has a presentation file that represents the graphical layout of the application pages displayed in a browser window. Each presentation file has a Presentation ID tag that can include any number of controls, dialogs, and data sources.</p> <p>The Presentation ID tag defines the main MBO for the application. For example, the MBO for the Work Order application is WORKORDER.</p> <p>See the Data Source ID description for the Combobox control for additional information on data sources.</p>
Application	Reserved for future use.

## Section



A Section is a container type of graphical control. Containers occupy real estate on a page and function as storage for other types of controls. A Section is a parent to children controls placed inside it. When you move a Section or other type of container control, its children automatically move with it.

Section controls without labels do not have visible borders and appear transparent in an application page. Sections with labels have a horizontal blue bar with a minimize icon that appear on top of the Section. At a minimum, you start all new application layouts by inserting a Section. Most applications have multiple pages with many sections and columns. See the Section Example for information on how to populate a Section control with multiple columns.

Sections are one of the primary components of the user interface. A Section is typically an application page that you can expand or minimize. Sections contain groupings of fields that hold similar or related data. Additionally, most tabs have many parent and children Sections that allow you to layout and define the user interface. Controls within Sections display information or accept user input.

### Rules

You can place the following controls in this container control:

- |                |                      |
|----------------|----------------------|
| ▼ Attachments  | ▼ Include            |
| ▼ Blank Line   | ▼ Listbox            |
| ▼ Button Group | ▼ Multiline Textbox  |
| ▼ Checkbox     | ▼ Radio Button Group |

- 
- ▼ Combobox
  - ▼ Data Source
  - ▼ Default Value
  - ▼ Hyperlink
  - ▼ Image
  - ▼ Section Header
  - ▼ Section Row
  - ▼ Static Text
  - ▼ Textbox
- 

You can place this control inside the following container controls:

- ▼ Dialogs
- ▼ Tab
- ▼ Table
- ▼ Section Columns

1 A Section container with a Textbox.



## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	<p>The name you assign to the Section control. The label value identifies the control to the user. If you do not enter a label, the Section is transparent in the user interface and does not have a title.</p> <p>The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.</p>
Show Border?	If you have added a Label, selecting this box also adds blue horizontal rules to the top and bottom of the Section. The border property does not work without a defined label.
Collapsed?	<p>Selecting this box collapses a Section and its contents if you have defined a Label for the Section. You see the collapsed Section when you run the application, but not in the Application Designer workspace.</p> <p>To expand a section in the user interface, click <b>Maximize Section</b>.</p>
Description	Text you enter in this field displays under the label in the title bar of the Section.

Attribute Name	Description
Data Source ID	<p>If you want this control to use a different Data Source ID than the default of MAINRECORD, then enter that value here.</p> <p>Each application has a presentation file that represents the graphical layout of the application pages displayed in a browser window. Each presentation file has a Presentation ID tag that can include any number of controls, dialogs, and data sources.</p> <p>The Presentation ID tag defines the main MBO for the application. For example, the MBO for the Work Order application is WORKORDER.</p> <p>The example below shows the first line of XML text in the Work Order Tracking presentation.</p> <pre data-bbox="842 764 1390 961"> &lt;presentation id="wotrack" resultstableid="results_showlist" mboname="WORKORDER" orderby="WONUM" beanclass="psdi.webclient.beans.workorder. WorkorderAppBean" version="6.0.0"&gt;  &lt;page id="mainrec"&gt; </pre> <p>Although the controls within an application can inherit data from different data sources, the default data source is always "MAINRECORD." In Work Order Tracking, any control whose Data Source ID is MAINRECORD pulls data from the WORKORDER table. MAINRECORD is the Parent Data Source ID for the presentation. To have individual controls inherit data from a different source, you must define a different Parent Data Source ID.</p> <p>See the Data Source ID description for the Combobox control on page 60 for additional information on data sources.</p>

Attribute Name	Description
Business Object	<p>The main MBO reference for the application. For example, Work Order Tracking uses WOTRACK and Assets uses ASSET.</p> <p>Every application works with a MBO and MBOSet. The set of work orders displayed on the List tab in the Work Order Tracking application, for example, is a MBOSet. Each individual work order is a MBO. The presentation tag of an application defines the main MBO for the application. MBOs are responsible for the following types of functions:</p> <ul style="list-style-type: none"> <li>▼ Field validation</li> <li>▼ Updating the database</li> <li>▼ Business processing logic</li> </ul> <p>By default, the presentation tag for an application defines the main MBO to use when saving or displaying data through the user interface. All controls in the application share the same MBO reference via the Parent Data Source ID property. Unless overridden, all controls added to an application bind to the Parent Data Source ID. If you wish an individual control to display and process information from a different MBO, you can enter a different Parent Data Source ID.</p>
Parent Data Source ID	<p>If you wish this Section control to inherit data from a source other than MAINRECORD, enter that value in this field. You might, for example, want to define a new parent data source in an application like Work Order Tracking where a table needs to pull data from an object other than WORKORDER. The Parent Data Source ID lets you build relationships with a different data source using Where Clauses.</p>

Attribute Name	Description
Input Mode	<p>Select one of the options listed below for the Input Mode:</p> <ul style="list-style-type: none"> <li>▼ Default</li> <li>▼ Password</li> <li>▼ Password Readonly</li> <li>▼ Password Required</li> <li>▼ Query</li> <li>▼ Readonly</li> <li>▼ Required</li> </ul> <p>The Default option sets the Input Mode for all controls to "edit." If you do not select an Input Mode for a control, the system automatically sets the mode to read/write so that you can both enter or display information in the field. If you enter information in a read/write text field, it posts to the database column specified by the Attribute property.</p> <p>Input Mode Option Descriptions:</p> <p><b>Default</b> – sets control to "edit" so you can display or input values and mode is read/write</p> <p><b>Password</b> – sets control to display asterisks (blanked out) for displayed or input values and mode is read/write</p> <p><b>Password Readonly</b> – sets control to display asterisks and mode is read only</p> <p><b>Password Required</b> – sets control to display asterisks and the control is required</p> <p><b>Query</b> – sets control to display a typed value and mode is read only</p> <p><b>Readonly</b> – sets control to display grayed-out text and mode is read only</p> <p><b>Required</b> – sets control to display a typed value and record cannot be saved unless there is a value in the control</p>
Listeners	<p>Enter the ID or IDs of one or more controls whose display values will be refreshed based on changes to this data source. Control ID or IDs can reside anywhere inside the application. You must separate multiple values with a comma.</p>

# Section Column



A Section Column is a vertical column control that lets you divide a Section Row into headings. A Section Column is a container that can hold child sections. A Section Column is a child to a parent section. When you move a Section, all of its children automatically move with it.

You can insert Section columns into Section Rows. When you drag a Section Column into a Section Row, it automatically creates two Section Columns. A Section Column does not have a label. Its only function is to hold child sections. A child section can hold, for example, a Section Header that lets you assign column headings for a multi-column section.

Sections Columns like Section Rows are transparent in the user interface. Except for tables, all graphical elements in the system that have multiple columns have a transparent Section Column within a Section Row that is within a parent section.

## Rules

You can place the following control in this container control:

- ▼ Section

You can place this control inside the following container control:

- ▼ Section Row

## Example

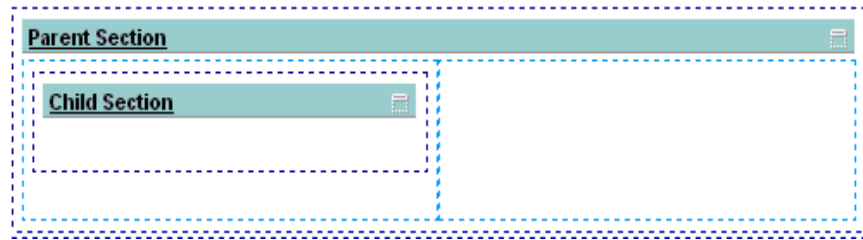
To create a multi-column section layout, complete the following steps:

- 1 Insert a Section.
- 2 Insert a Section Row.
- 3 Insert a Section Column.

To create a multi-column section layout with a child section, complete the following steps:

- 1 Insert a Section.
- 2 Insert a Section Row.
- 3 Insert multiple Section Columns.
- 4 Drop a new section into each Section Column.

- 5 Add controls to each new Section.



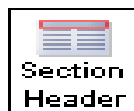
**NOTE** To view the page tag in the Application Designer, select **Toggle Show All Controls** from the **Select Action** menu.

- 1 You can drop Section Columns into Section Rows, but not Sections.
- 2 You use Section Columns to subdivide a Section, but you cannot drop controls into a Section Column.

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.

## Section Header



A Section Header is a control that can contain dynamic values in its label. The Plans tab in the Work Order application, for example, uses the Section Header and related Parameter and Parameter Values controls to display children and tasks of work orders. The Section Header is a container control.

Use this control to assign a column or table heading for a Section or Table. A Section Header can contain replaceable values depending on how you configure the Parameter Value and Parameter Values controls. For example, the label attribute for the Section Header in the task table in Work Order Tracking > Plans tab has a variable value similar to "(0)," where the variable value is defined by the Parameter Value control. When you move a Section, all of its children, including Section Header and related controls, automatically move with it.

Sections Headers are transparent in the user interface. To view a grouping of Section Header, Parameter Values, and Parameter Value controls in the Application Designer, you must choose **Select Action > Toggle Show All Controls** from a tab. Section Headers along with the Parameter Value and Values controls enable you to display dynamic values in the "Tasks for Work Order xxxx" label for the table in the previous example. The settings you configure in the Section Header and Parameter Values set "xxx" in the header to the current WONUM (i.e., 1000).

## Rules

You can place the following control in this container control:

- ▼ Parameter Value
- ▼ Parameter Values

You can place this control inside the following container controls:

- ▼ Section
- ▼ Table

### Example (Work Order Tracking > Plans >Task Table)

To create a Section Header with dynamic values, complete the following steps:

- 1 Insert a Section Header control into a Section or Table control.
- 2 Drag the Parameter Values control into the Section Header container.
- 3 A Parameter Value control automatically appears with the Parameter Values control.

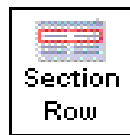
## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	The label contains a variable that displays dynamic values inherited from the Parameter Value and Values controls in the Table or Section control.
Plain Text	Any text you enter in this field removes the CSS (cascading style sheet) class definition for a label so that it looks like plain text in the user interface.



Attribute Name	Description
Data Source ID	<p>If you want this control to use a different Data Source ID than the default of MAINRECORD, then enter that value here.</p> <p>Each application has a presentation file that represents the graphical layout of the application pages displayed in a browser window. Each presentation file has a Presentation ID tag that can include any number of controls, dialogs, and data sources.</p> <p>The Presentation ID tag defines the main MBO for the application. For example, the MBO for the Work Order application is WORKORDER.</p> <p>See the Data Source ID description for the Combobox control for additional information on data sources.</p>

## Section Row



A Section Row is a container type of graphical control that holds Section Columns. A Section Row is a horizontal row that lets you partition a Section into columns that in turn hold child sections. A Section Row is a child to a parent section. When you move a Section, all of its children automatically move with it.

You can insert Section Columns into Section Rows. A Section Row does not have a label. Its only function is to hold Section Columns. Section Rows are transparent in the user interface. Except for tables, all graphical elements in the system that have multiple columns have a transparent Section Row within a Section.

### Rules

You can place the following control in this container control:

- ▼ Section Column

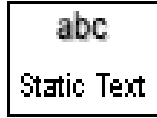
You can place this control inside the following container control:

- ▼ Section

### Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.

# Static Text



Static Text is a text control that lets you enter and configure free form text in a Section. It also lets you specify a database attribute that you can configure and position within a Section as if it were unbound text. You can position static text using the Text Alignment, Column, or Span properties. You typically see applications use static text in Date search windows. The static text functions as a To or From label in a Date Search window like the Search More window in Work Order Tracking.

In addition to configuring Static Text to function as a plain text label, you can configure the plain text to behave as a hyperlink that will take you to a URL when clicked. When configured as a URL, the Static Text uses a larger font and looks more like a URL string than plain text.

## Rules

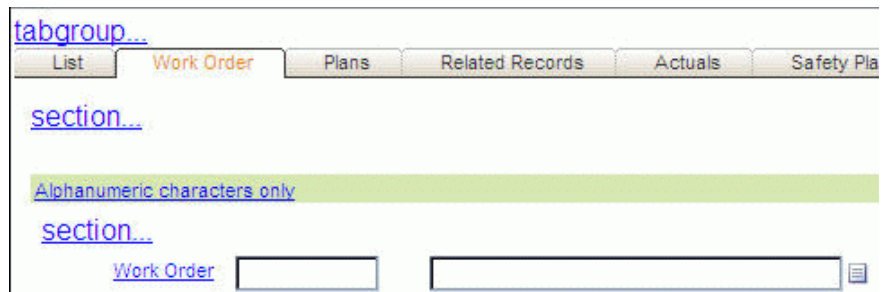
This is not a container control.

You can place this control inside the following container control:

- ▼ Section

## Example

- 1 Static Text example (“Alphanumeric characters only”) as it appears in the Application Designer workspace.



## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Default Label	This label defaults to the text value associated with the database attribute you bound to the Listbox. This is a read only field.

Attribute Name	Description
Label	You can enter a text value in this field that overrides the Default Label value. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Text Alignment	Select Left, Center, or Right to align Static Text within this control. Typically, you associate static text with a control that resides in a Section. You can place the text above or below the control.
Column	Enter a numeric value in this field that will position the static text at the beginning, middle or end of a column depending on the Text Alignment setting. By default, each control has seven columns that are transparent to the user. For example, if you enter "3" in this field and "Center" for Text Alignment, then the system centers the Static Text above or below column three.  If the Static Text you want to display is wider than its column, you cannot use the Text Alignment property and the system automatically extends the column width to contain the Static Text.
Span	Enter a numeric value in this field that will enable the Static Text to bridge one or more columns.
Attribute	You can associate static text with a database attribute by using the Select Value window. If the attribute you select has a default label value, it will display in the Default Label field.
Display as Link?	By default, Static Text does not display as a URL link. Select this field to configure the control as a URL link.
Display as Plain Text?	Any text you enter in this field removes the CSS (cascading style sheet) class definition for a label so that it looks like plain text in the user interface.

## Tab



Tab is a container control. Formatting a Tab or Tabs is the starting point for designing the structure and layout of a new application or extending the layout of an existing application. You would typically add a Section control as the first element in a Tab and then add other controls to complete your window layout. For example, you can break the layout of the page vertically by adding additional Section controls. Also, you can break a section horizontally by adding Section Columns.

## Rules

You can place the following controls in this container control:

- ▼ Data Source
- ▼ Menubar
- ▼ Table
- ▼ Section
- ▼ Help Grid
- ▼ Include
- ▼ Tree

You can place this control inside the following container controls:

- ▼ Tabgroup

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	The name you assign to the Tab control for a specific application. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Type	Select the type of tab you want to add to the Tabgroup. Valid values are List or Insert. Choosing List formats the tab as a standard search tab with a filter table. Choosing Insert formats the tab as an empty workspace where you can build an application window by inserting and configuring controls.
Default?	This property indicates which tab, either a List or Insert, to enable when you first open an application and display a record. When you create a new power application, List tab is the default.
Bean Class	Reserved for future use.

Attribute Name	Description
Business Object	<p>The main MBO reference for the application. For example, Work Order Tracking uses WOTRACK and Assets uses ASSET.</p> <p>Every application works with a MBO and MBOSet. The set of work orders displayed on the List tab in the Work Order Tracking application, for example, is a MBOSet. Each individual work order is a MBO. The presentation tag of an application defines the main MBO for the application. MBOs are responsible for the following types of functions:</p> <ul style="list-style-type: none"> <li>▼ Field validation</li> <li>▼ Updating the database</li> <li>▼ Business processing logic</li> </ul> <p>By default, the presentation tag for an application defines the main MBO to use when saving or displaying data through the user interface. All controls in the application share the same MBO reference via the Parent Data Source ID property. Unless overridden, all controls added to an application bind to the Parent Data Source ID. If you wish an individual control to display and process information from a different MBO, you can enter a different Parent Data Source ID.</p>
Parent Data Source ID	<p>If you wish the Tab control to inherit data from a source other than MAINRECORD, enter that value in this field. You might, for example, want to define a new parent data source in an application like Work Order Tracking where the tab and its children controls pull data from an object other than WORKORDER. The Parent Data Source ID lets you build relationships with a different data source using Where Clauses.</p>
Relationship	<p>You enter a relationship value in this field. You can search for values in the Database Configuration &gt; Relationships tab. If you enter a relationship, it is automatically inherited by any input child controls within the tab. Relationships enable controls to display or write to different business objects.</p>
Order By	<p>Reserved for future use.</p>
Where Clause	<p>Reserved for future use.</p>
Listeners	<p>Enter the ID or IDs of one or more controls whose display values will be refreshed based on changes to this data source. Control ID or IDs can reside anywhere inside the application. You must separate multiple values with a comma.</p>

# Tab Group



A Tabgroup is a container type of control that holds tabs. Tabgroups are part of the template for power applications. When you create a new power application, you will see a Tabgroup control with the standard List and Workspace tabs. You can add additional tabs as required by your business

application.

## Rules

You can place the following control in this container control:

- ▼ Tab

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Format	Leave blank (the default) or choose one of the following layout options: <ul style="list-style-type: none"> <li>▼ CARDECK</li> <li>▼ DEFAULT</li> <li>▼ WIZARD</li> </ul> <p>The wizard option replaces the standard tab layout with arrows that point to the tab headings. The cardeck option appears in single-page applications that do not employ tabs.</p>
Style	The Style option defaults to FORM, which is a style attribute for the control. The Form style defines the appearance of the main tab group.
Change Event	Reserved for future use.

# Table



The Table control is a container control with the basic layout elements of a table window, including a toolbar with filter and search functionality and icons for previous and next row and page. The basic Table layout also includes a Table Column control and a details box where you can enter a Section. The

Table control is similar to the table window in the List tab of most applications. The system tables enable you to search for and display records from one or more tables in a columnar format. You can use the standard filter feature to display only records that meet specified criteria.

## Rules

You can place the following controls in this container control:

- ▼ Table Column
- ▼ Section

You can place this control inside the following container controls:

- ▼ Section Column
- ▼ Tab
- ▼ window

**NOTE** You can insert a Section control into the details section of a Table window. The details section is an empty, rectangular box at the bottom of the window.

## Example

- 1 The Table Properties window in the Application Designer.

The screenshot shows the 'Table Properties' dialog box with the following fields and values:

- Control ID:** new\_table\_2
- Label:** (empty)
- Width:** (empty)
- Description:** (empty)
- Title Attributes:** (empty)
- Start Empty?:**
- No Row Message:** (empty)
- No Row Message Attributes:** (empty)
- Filter Expanded?:**
- Row Details Expanded?:**
- Collapsible?:**
- Collapsed?:**
- Collapsed Empty Label:** (empty)
- Collapsed Label:** (empty)
- Parent Empty Label:** (empty)
- Bean Class:** (empty)
- Select Mode:** (empty)
- Input Mode:** (empty)
- Data Source ID:** MAINRECORD
- Business Object:** (empty)
- Parent Data Source ID:** (empty)
- Relationship:** (empty)
- Order By:** (empty)
- Application Restrictions:** (empty)
- Listeners:** (empty)
- Display Rows Per Page:** (empty)
- Filterable?:**
- Customizable?:**

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Label	The text value you enter becomes the title of the table and displays in the toolbar. The system stores this value in the MAXLABELS table. You can edit this value. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Width	Enter a numeric value to resize the width of a table. The default window size for a table is approximately 1000, or the full width of a page. You can decrease the width of a table to approximately 400. The table control needs a width of 400 to maintain the contents of the toolbar, including table title and icons.
Description	Text you enter in this field displays under the Table label in the title bar.
Title Attributes	Reserved for future use.
Start Empty?	Select this box if you want the table to always open without displaying records. If you leave this box cleared (the default), the system automatically populates the table with records that match the search query when opened.
No Row Messages	By default, you see the message "No rows to display" when you open a table without rows. To override the default text message, enter a value in this field.
No Row Message Attributes	Reserved for future use.
Filter Expanded?	Select this box if you want to display an empty query filter row when you open a table. If the default is cleared, the filter row does not display with an open table. See also Filterable?.
Row Details Expanded?	Select this box if you want a table to display the expanded detail sections for each record in the table. The default is not to expand the detail sections for a table row.
Collapsible?	Select this box to enable the Show/Hide icons in the table toolbar. You must select this field to use the "Collapsed?" property. By default, this property is cleared.



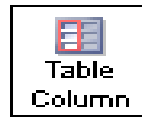
Attribute Name	Description
Collapsed?	If you selected "Collapsible?", then you can select this box to display only the table title and not any row details when you open a table in an application. By default, this property is cleared.
Collapsed Empty Label	When a table is collapsed and empty, the value you enter here will override the table title (i.e., Label property) in the user interface.
Collapsed Label	When a table is collapsed, the value you enter here will override the table title in the user interface.
Parent Empty Label	If this table is a child of a parent table, and that table is empty, the value you enter here will override the title of the parent table .
Bean Class	Reserved for future use.
Select Mode	You can restrict a table to displaying one or multiple records from a database table. You can choose either Multiple or Single. The Work Order Tracking list table, for example, has this field set to "MULTIPLE."

Attribute Name	Description
Input Mode	<p>Select one of options listed below for the Input Mode:</p> <ul style="list-style-type: none"> <li>▼ Default</li> <li>▼ Password</li> <li>▼ Password Readonly</li> <li>▼ Password Required</li> <li>▼ Query</li> <li>▼ Readonly</li> <li>▼ Required</li> </ul> <p>The Default option sets the Input Mode for all controls to "edit." If you do not select an Input Mode for a control, the system automatically sets the mode to read/write so that you can both enter or display information in the field. If you enter information in a read/write text field, it posts to the database column specified by the Attribute property.</p> <p>Input Mode Option Descriptions:</p> <p><b>Default</b> – sets control to "edit" so you can display or input values and mode is read/write</p> <p><b>Password</b> – sets control to display asterisks (blanked out) for displayed or input values and mode is read/write</p> <p><b>Password Readonly</b> – sets control to display asterisks and mode is read only</p> <p><b>Password Required</b> – sets control to display asterisks and the control is required</p> <p><b>Query</b> – sets control to display a typed value and mode is read only</p> <p><b>Readonly</b> – sets control to display grayed-out text and mode is read only</p> <p><b>Required</b> – sets control to display a typed value and record cannot be saved unless there is a value in the control</p>

Attribute Name	Description
Data Source ID	<p>The Data Source ID for this control. It points to the main MBO or object that the control uses for entering and displaying data. Each application links to a main object that represents the Parent Data Source for all child controls in the application. Individual or groups of controls within an application can inherit attributes from objects that are not the primary data source.</p> <p>See the Data Source ID description for the Combobox control for additional information on data sources.</p>
Business Object	<p>The main MBO reference for the application. For example, Work Order Tracking uses WOTRACK and Assets uses ASSET.</p> <p>Every application works with a MBO and MBOSet. The set of work orders displayed on the List tab in the Work Order Tracking application, for example, is a MBOSet. Each individual work order is a MBO. The presentation tag of an application defines the main MBO for the application. MBOs are responsible for the following types of functions:</p> <ul style="list-style-type: none"> <li>▼ Field validation</li> <li>▼ Updating the database</li> <li>▼ Business processing logic</li> </ul> <p>By default, the presentation tag for an application defines the main MBO to use when saving or displaying data through the user interface. All controls in the application share the same MBO reference via the Parent Data Source ID property. Unless overridden, all controls added to an application bind to the Parent Data Source ID. If you wish an individual control to display and process information from a different MBO, you can enter a different Parent Data Source ID.</p>
Parent Data Source ID	<p>The Parent Data Source ID points to the main MBO associated with an application, for example, WOTRACK. Unless they are overridden, all controls inherit the value stored in the presentation tag of an application.</p>

Attribute Name	Description
Relationship	<p>You enter a relationship value in this field. You can search for values in the Database Configuration &gt; Relationships tab. If you enter a relationship, it is automatically inherited by any input child controls within the table.</p> <p>Relationships enable a table to display information from different business objects. For example, if you entered SPAREPART, the table will be able to display information based on the defined relationship. When you enter a relationship in the Table control, it automatically prefixes any child column attributes added to the table. So, for SPAREPART, you only have to enter the column attribute name and not the full relationship of SPAREPART &lt;attribute&gt;.</p>
Order By	<p>Enter the name of an attribute to use when sorting information in the table. For example, you can sort on QUANTITY to list records according to their numeric value. If you enter ASC QUANTITY, the table displays records in ascending order from the lowest to the highest numeric value. If you enter DSC QUANTITY, the table displays records in descending order from the highest to the lowest numeric value.</p>
Application Restrictions	<p>Enter a Where Clause that filters information displayed in the table. If you enter a restriction, it modifies the overall Where Clause for the application. You cannot change the restriction at runtime.</p>
Listeners	<p>Enter the ID or IDs of one or more controls whose display values will be refreshed based on changes to this data source. Control ID or IDs can reside anywhere inside the application. You must separate multiple values with a comma.</p>
Display Rows Per Page	<p>The maximum number of rows you wish to display on a page.</p>
Filterable?	<p>This field adds a filter row for searching to the table. It defaults to enable. This setting automatically passes to the Filterable? property in all Table Column controls. You must select this box to use the Filter Expanded? property.</p> <p>See also the Filter Expanded? attribute description.</p>
Customizable?	<p>Reserved for future use.</p>

# Table Column



The Table Column control is a container type control that lets you define additional column headings for a table. The columns let you display or insert data from tables. You can use the standard filter feature to display only records that meet specified criteria.

## Rules

You can place the following controls in this container control:

- ▼ Checkbox
- ▼ Textbox

You can place this control inside the following container controls:

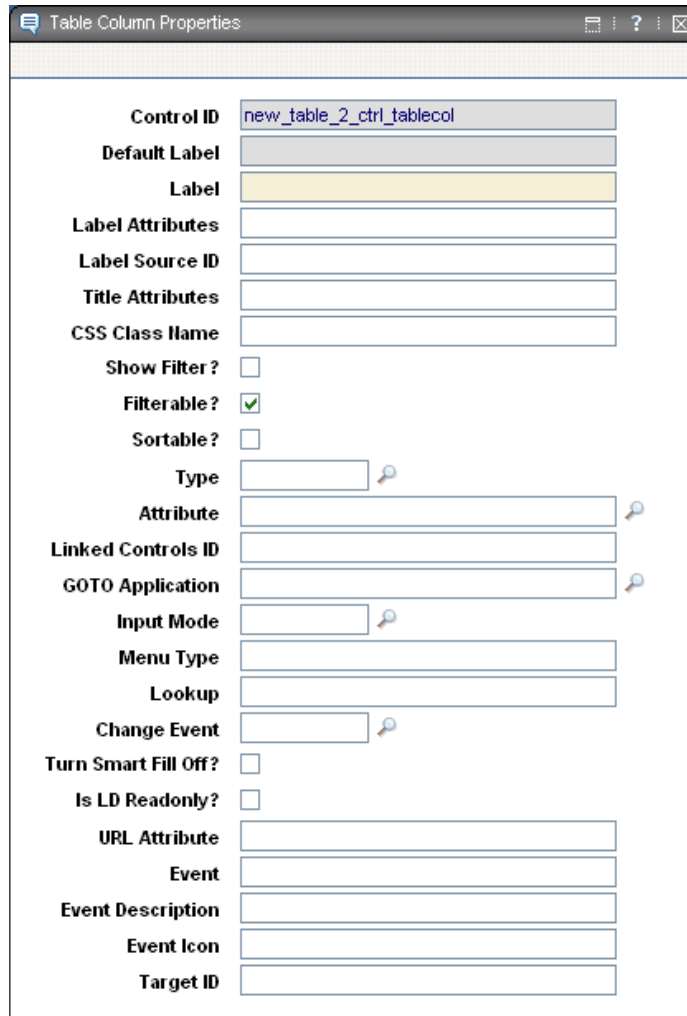
- ▼ Table

## Example

- 1 A Table Column control dropped into a table.



2 The Table Column Properties window in the Application Designer.



Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Default Label	This label, which becomes the Table Column heading, defaults to the text value associated with the database attribute entered in the Attribute property. For example, if you enter ASSETNUM as the attribute, the Default Label becomes Asset. This is a read only field.

Attribute Name	Description
Label	The text value you enter here replaces the Default Label as the Table Column heading. The system stores this value in the MAXLABELS table. You can edit this value. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Label Attributes	You can enter a comma separated list of data attributes to use when building the Default Label for this control. The Label Attributes and Label Source ID properties enable you to display dynamic values in the Default Label for a Table Column.
Label Source ID	Enter an object name that is the data source for the Label Attributes property.
Title Attributes	Reserved for future use.
CSS Class Name	Enter the class name for a cascading style sheet. The value you enter can change the layout and appearance of the Table Column. Typically, you would not use this field with a Table Column control.
Show Filter?	If you select the Filterable and Show Filter properties, you will see the filter Textbox control for this column in the filter row of the table.
Filterable?	If you select the Filterable property for a Table control, it automatically passes this setting to the Filterable property in each Table Column. You can select or clear Filterable at the table column level. Filterable, when selected, indicates that you can search on the attribute associated with the Table Column label.
Sortable?	If you select Sortable, you can sort the records in the table by clicking on the label of a column. The first time you click on a label it sorts the records in ascending order for the values of that column. The second time you click a label, it sorts the records in descending order.

Attribute Name	Description
Type	<p>You can link a Table Column to events and links. If you select Event, it maps the column to an event type, description, and icon. BOOKMARK and TOGGLEDETAILSTATE are event examples. Link adds an underline to a value, usually a record number, which causes the detail record to open when you select the value.</p> <p>In the Related Work Orders tab there is an event column before the Work Order Number column called "toggedetailstate" with a description of "Show Detail." This column places a <b>View/Show Details</b> icon next to the work order number column that lets you open or close the details section for a selected work order.</p> <p>The Work Order Number column in the Work Order List tab has a Type value of Link. Link adds an underline to all work order numbers that appear in the column.</p>
Attribute	<p>You can associate a Table Column with a database attribute by using the Select Value window. For example, if you enter ASSETNUM in the Attribute field, the system populates the Default Label with "Asset." This column displays asset numbers and "Asset" displays as the Table Column heading. Not all attributes you select will have a database default label value.</p>
Linked Controls ID	<p>You can enter a Control ID here that gets automatically updated whenever there is a change to the attribute value for this column. The Control ID must reside somewhere within the application.</p>
GOTO Application	<p>You can configure a pull-down menu associated with a Table Column to have options that will take the user to different applications. Use the Select Value window for this property to choose application names that will appear in the pull-down menu.</p>



Attribute Name	Description
Input Mode	<p>Select one of options listed below for the Input Mode:</p> <ul style="list-style-type: none"> <li>▼ Default</li> <li>▼ Password</li> <li>▼ Password Readonly</li> <li>▼ Password Required</li> <li>▼ Query</li> <li>▼ Readonly</li> <li>▼ Required</li> </ul> <p>The Default option sets the Input Mode for all controls to "edit." If you do not select an Input Mode for a control, the system automatically sets the mode to read/write so that you can both enter or display information in the field. If you enter information in a read/write text field, it posts to the database column specified by the Attribute property.</p> <p>Input Mode Option Descriptions:</p> <p><b>Default</b> – sets control to "edit" so you can display or input values and mode is read/write</p> <p><b>Password</b> – sets control to display asterisks (blanked out) for displayed or input values and mode is read/write</p> <p><b>Password Readonly</b> – sets control to display asterisks and mode is read only</p> <p><b>Password Required</b> – sets control to display asterisks and the control is required</p> <p><b>Query</b> – sets control to display a typed value and mode is read only</p> <p><b>Readonly</b> – sets control to display grayed-out text and mode is read only</p> <p><b>Required</b> – sets control to display a typed value and record cannot be saved unless there is a value in the control</p>

Attribute Name	Description
Menu Type	<p>You can enter a value in the Menu Type property that links the Table Column to a pre-formatted drop-down menu. The value, called a menu ID, defines the XML presentation content and format of a drop-down menu. WORKORDER, for example, is the Menu Type for the Parent WO textbox in Work Order Tracking. Menu IDs reside in a system XML file called menus.xml.</p> <p>You cannot edit this file from the Application Designer. However, you can view the content of this file or other system .xml files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the</p> <pre data-bbox="842 699 1329 751">&lt;root&gt;\resources\presentation\ system directory.</pre>
Lookup	<p>You can enter a value in the Lookup property that links the Table Column to a pre-formatted lookup table. The lookup table pulls records from the main table. Users can populate the Table Column by selecting a record from the table. The value, called a lookup ID, defines the XML presentation content and format of a select value table. WORKORDERISTASK, for example, is the Lookup property for the Parent WO textbox in Work Order Tracking. Lookup IDs reside in a system XML file called lookups.xml.</p> <p>You cannot edit system XML files from the Application Designer. However, you can view the content of these types of files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the</p> <pre data-bbox="842 1381 1318 1434">&lt;root&gt;\resources\presentation\system directory.</pre>
Change Event	<p>You can configure an input control to perform either of the following actions based on any data change to the control:</p> <ul style="list-style-type: none"> <li data-bbox="890 1587 1139 1619">▼ REFRESHTABLE</li> <li data-bbox="890 1619 1158 1650">▼ RESETCHILDREN</li> </ul> <p>Choosing REFRESHTABLE forces the table that is the data source for the input control to refresh its data. Choosing RESETCHILDREN forces any children of the control's data bean to delete and reacquire their MBOSetRemote references.</p>

Attribute Name	Description
Turn Smart Fill Off?	By default, the system enables the "Smart Fill" feature. To disable, clear the check box. The smart fill feature enables you to enter a partial value in a text box field. The system attempts to match the partial text string entered to a valid value or values in the database. If the system finds one matching value, it automatically populates the field so that you can continue entering information. If the system finds more than one matching value, it displays a Select Value window with a list of matching values. If you clear the smart fill property, the system does not validate the data you enter in the textbox.
Is LD Readonly?	By default, the system sets a Long Description associated with a Table Column attribute to editable. Select this field to make the Long Description read only.
URL Attribute	Enter a URL value if you set the Type property for this control to "Link." The URL value becomes a hyperlink.
Event	<p>If you select "Event" as the Type property for a Table Column, the system performs the action defined in this field. Continuing the example for the Type property above, entering "toggledetailstate" lets you view or close the details section of a work order. Users can also define their own events.</p> <p>Virtually every action you perform in the system, such as clicking Next or saving a record, triggers an event. Events defined in a presentation file are tied to a method in the code or an event tag in the JSP pages. The system is an event driven product. When you perform an action, the system sends an "Event" to the back end which invokes and runs a method associated with the event. There is no directory listing of events, however, you can search for events and event descriptions in presentations using the search words "mxevent" and "mxevent_desc."</p> <p>See also the Type attribute description.</p>
Event Description	<p>Indicates the tool tip that appears above a selected field or icon. For example, "Mark row for deletion" or "View Details."</p> <p>See also the Event attribute description.</p>

Attribute Name	Description
Event Icon	<p>Point to an image (.gif file) that represents an event. The image icons you see in the user interface reside in the following directory:</p> <pre>&lt;root&gt;\applications\maximo\ maximouiweb\webmodule\webclient\images</pre> <p>You can access the graphic files in the images directory using the Image control in the Application Designer.</p>
Target ID	<p>Enter a Control ID, if different from the current Table Column control, to which the system sends the Event property (if defined) for processing.</p>

## Textbox



A Textbox control is a single line text input box that displays information you enter or select from a list. Textboxes have labels that are usually visible in the user interface. Labels are text fields that describe the type of information displayed in the box.

Textboxes are often linked to select value lookups and detailed menus. A select value lookup lets you select and then display a single value from a list of records. A Textbox linked to a detailed menu can contain diverse options, such as a Select Value window, jumps to other applications, or views of different records. The Menu Type property you configure for a Textbox controls the format and content of the detailed menu.

Textbox controls have a Default Label property that enables the control to inherit a read-only label from the attribute you configure for the control. The Default Label value comes from the database object. Alternatively, you can override an inherited Default Label by entering a different value in the Label property. This value is local to the presentation file of the application. Labels you enter at the presentation level are not stored in the MAXATTRIBUTE OR MAXLABELS table.

Textboxes display object information from the Data Source ID and attribute values you specify in the properties window.

### Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Section

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Default Label	This label defaults to the text value associated with the database attribute you bound to the textbox. This is a read only field.
Label	You can enter a text value in this field that overrides the Default Label value. The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Attribute	You can associate a textbox with a database attribute by using the Select Value window. If the attribute you select has a default label value, it will display in the Default Label field.
Menu Type	<p>You can enter a value in the Menu Type property that links the textbox to a pre-formatted drop-down menu. The value, called a menu ID, defines the XML presentation content and format of a drop-down menu. WORKORDER, for example, is the Menu Type for the Parent WO textbox in Work Order Tracking. Menu IDs reside in a system XML file called menus.xml.</p> <p>You cannot edit menus.xml from the Application Designer. However, you can view the content of this file or other system .xml files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the</p> <pre>&lt;root&gt;\resources\presentation\system directory.</pre>

Attribute Name	Description
Lookup	<p>You can enter a value in the Lookup property that links the textbox to a pre-formatted lookup table. The lookup table pulls records from the main table. Users can populate the textbox by selecting a record from the table. The value, called a lookup ID, defines the XML presentation content and format of a select value table.</p> <p>WORKORDERISTASK, for example, is the Lookup property for the Parent WO textbox in Work Order Tracking. Lookup IDs reside in a system XML file called lookups.xml.</p> <p>You cannot edit system XML files from the Application Designer. However, you can view the content of these types of files by selecting a main tab from the Application Designer and choosing <b>Select Action &gt; Export System XML</b>. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;root&gt;\resources\presentation\system directory.</p>
Detail Image	Reserved for future use.

Attribute Name	Description
Input Mode	<p>Select one of options listed below for the Input Mode:</p> <ul style="list-style-type: none"> <li>▼ Default</li> <li>▼ Password</li> <li>▼ Password Readonly</li> <li>▼ Password Required</li> <li>▼ Query</li> <li>▼ Readonly</li> <li>▼ Required</li> </ul> <p>The Default option sets the Input Mode for all controls to "edit." If you do not select an Input Mode for a control, the system automatically sets the mode to read/write so that you can both enter or display information in the field. If you enter information in a read/write text field, it posts to the database column specified by the Attribute property.</p> <p>Input Mode Option Descriptions:</p> <p><b>Default</b> – sets control to "edit" so you can display or input values and mode is read/write</p> <p><b>Password</b> – sets control to display asterisks (blanked out) for displayed or input values and mode is read/write</p> <p><b>Password Readonly</b> – sets control to display asterisks and mode is read only</p> <p><b>Password Required</b> – sets control to display asterisks and the control is required</p> <p><b>Query</b> – sets control to display a typed value and mode is read only</p> <p><b>Readonly</b> – sets control to display grayed-out text and mode is read only</p> <p><b>Required</b> – sets control to display a typed value and record cannot be saved unless there is a value in the control</p>
QBE Prepend	<p>If you set Input Mode to "Query," then you can filter the record search by adding a text string prefix that acts as filter. For example, the More Search Fields window uses the QBE Prepend feature to add date search filters similar to "&gt;=". You enter the value here to save key strokes when searching for records that meet specific date criteria.</p>

Attribute Name	Description
Change Event	<p>You can configure an input control to perform either of the following actions based on any data change to the control:</p> <ul style="list-style-type: none"> <li>▼ REFRESHTABLE</li> <li>▼ RESETCHILDREN</li> </ul> <p>Choosing REFRESHTABLE forces the table that is the data source for the input control to refresh its data. Choosing RESETCHILDREN forces any children of the data bean of the control to delete and reacquire their MBOSetRemote references.</p>
Turn Smart Fill Off?	<p>By default, the system enables the "Smart Fill" feature. To disable, clear the check box. The smart fill feature enables you to enter a partial value in a text box field. The system attempts to match the partial text string entered to a valid value or values in the database. If the system finds one matching value, it automatically populates the field so that you can continue entering information. If the system finds more than one matching value, it displays a Select Value window with a list of matching values. If you clear the smart fill feature, the system does not validate the data you enter in the textbox.</p>
Readonly Long Description?	<p>By default, the system sets a Long Description associated with a textbox attribute to editable. Select this field to make the Long Description read only.</p>
GOTO Application	<p>You can configure a pull-down menu associated with a textbox to have options that will take the user to different applications. Use the Select Value window for this property to choose application names that will appear in the pull-down menu.</p>
Display Type	<p>Reserved for future use.</p>
Move to Data Source ID	<p>The value you enter lets you move to a record stored in a different object. You can select any Data Source ID from the presentation. Refer to the Data Source ID definition below.</p> <p>By default, the system sends the Move To event to the ResultsBean, which contains a set of records displayed in the List tab. The Move To event affects the Move to Data Source ID object.</p>



Attribute Name	Description
Data Source ID	<p>The Data Source ID for this control. It points to the main MBO or object that the control uses for entering and displaying data. Each application links to a main object that represents the Parent Data Source for all child controls in the application. Individual or groups of controls within an application can inherit attributes from objects that are not the primary data source.</p> <p>See the Data Source ID description for the Combobox control on page 60 for additional information on data sources.</p>

## Tree



The Tree control is a container type of control that works with the Tree Node and Tree Attribute controls. You can use these controls to build dynamic, hierarchical tree structures in the system with specific MBOs, such as CLASSSTRUCTURE, that support hierarchical data. The Classifications application uses the Tree controls and the CLASSSTRUCTURE MBO to build classification hierarchies. The Security Profile hierarchy in the Users application is another example of how the system employs the Tree controls.

The following table describes how a Tree control behaves in the user interface.

Clicking...	What happens...
Plus sign	Expands the tree node without highlighting the corresponding node label. The plus sign becomes a minus sign. If you highlighted a node label prior to clicking a plus sign, it remains highlighted.
Minus sign	Contracts the tree node and highlights the corresponding node label.
Dot sign	(If node label is not selected) Highlights the node label and removes any existing highlights.
Dot sign	(If node label is selected) No effect.
Node label	(Next to a plus sign) Highlights the node label and removes any existing highlights. Highlighted node does not expand. window displays hierarchical levels below the highlighted node.
Node label	(Next to a minus sign) Highlights the node label and removes any existing highlights. Highlighted node does not contract. window displays hierarchical levels below the highlighted node.

Clicking...	What happens...
Node label	(Next to a dot sign) Highlights the node label and removes any existing highlights. Highlighted node does not expand or contract. window displays any items or information related to the node label.

When you highlight a node label, the text color changes from black to blue.

## Rules

You can place the following controls in this container control:

- ▼ Tree Node

You can place this control inside the following container controls:

- ▼ Dialogs
- ▼ Tab

## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Height	Enter a numeric value in this field that will increase or decrease the height of this control.
Width	Enter a numeric value to increase or decrease the width of this control.
Bean Class	Reserved for future use.
Business Object	<p>The main MBO reference for the application. For example, Work Order Tracking uses WOTRACK and Assets uses ASSET. The primary MBO for the Classifications application is CLASSSTRUCTURE.</p> <p>Every application works with a MBO and MBOSet. The set of work orders displayed on the List tab in the Work Order Tracking application, for example, is a MBOSet. Each individual work order is a MBO. The presentation tag of an application defines the main MBO for the application. MBOs are responsible for the following types of functions:</p> <ul style="list-style-type: none"> <li>▼ Field validation</li> <li>▼ Updating the database</li> <li>▼ Business processing logic</li> </ul>

Attribute Name	Description
Relationship	You can enter a relationship value in this field. The system relationships reside in the Database Configuration > Relationships tab. If you enter a relationship, it is automatically inherited by any input child controls within the tree. Relationships enable controls to display or write to different business objects.
Use Data Source?	Reserved for future use.
Order By	Reserved for future use.

## Tree Attribute



The Tree Attribute control describes the type of data that appears in each Tree Node. For example, the tree attributes in the Classifications tree structure are responsible for displaying Classification IDs and Descriptions in the user interface. The following XML shows how the Tree Attribute controls point to ID and Description data attributes in the Classifications presentation:

```
<treeattribute id="selectParentClassification_tree_node1_attr1" dataattribute="classificationid" display="true" />
```

```
<treeattribute id="selectParentClassification_tree_node1_attr2" dataattribute="classification.description" display="true" />
```

### Rules

This is not a container control.

You can place this control inside the following container control:

- ▼ Tree Node

### Example

See the preceding Tree control example.

### Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Display?	Select this box to display the defined Attribute value in the Tree Node.

Attribute Name	Description
Attribute	You can associate what values display in the tree node structure with a database attribute by using the Select Value window. To create the example for this control, you can enter "=classstructure" in the Object field to display all the attributes in the CLASSSTRUCTURE table. Select the "CLASSIFICATIONID" attribute to display Classification ID as one of the Tree Node properties. In the Tree example, you also define another Tree Attribute property called "CLASSIFICATION.DESCRPTION" to display the Classification Description value.

## Tree Node



The Tree Node control is a container type of control that works with the Tree and Tree Attribute controls. A Tree Node defines the structure of a hierarchical tree.

### Rules

You can place the following controls in this container control:

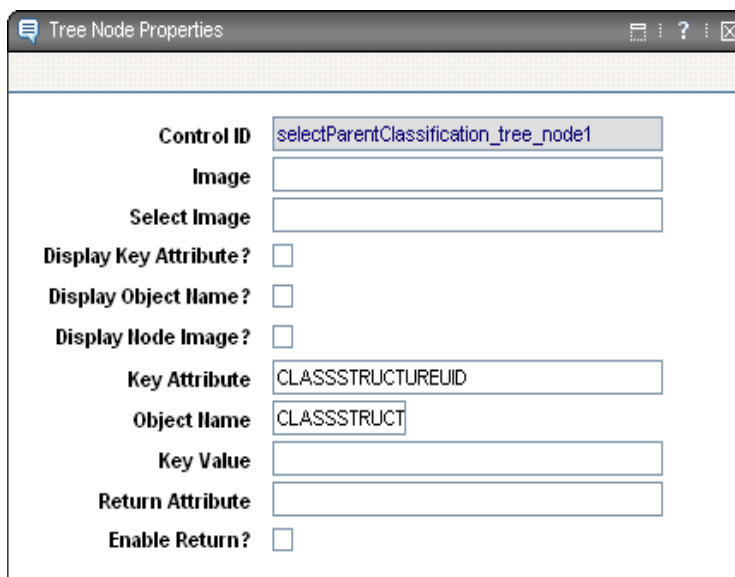
- ▼ Tree Attribute

You can place this control inside the following container control:

- ▼ Tree

## Example

- 1 The Tree Node Properties window for Select Parent Classification in the Application Designer workspace.



## Properties

Attribute Name	Description
Control ID	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
Image	<p>To add a graphic next to the Tree Node, enter a file name in this field. This field defaults to the path of the images directory so that you do not have to enter the full path for a graphic file.</p> <p>You can specify any type of graphic that a browser is capable of displaying, such as .gif, .jpeg, or .bmps. Some graphic files may require a plug-in.</p> <p><b>NOTE</b> You can place any graphic file into the system library of images. However, if you insert a graphic file without a .gif extension into the system, then you need to rebuild/ deploy the .EAR file.</p>
Select Image	Enter the name of a graphic file here that will replace the typical blue box that appears to the right of the + (open) and – (close) images when you open a Tree Node structure.
Display Key Attribute?	Select this property to display the Key Attribute as part of the Tree Node label.

<b>Attribute Name</b>	<b>Description</b>
Display Object Name?	Select this property to display the Object Name as part of the Tree Node label.
Display Node Image?	Select this property to display the Node Image as part of the Tree Node label.
Key Attribute	Reserved for future use.
Object Name	Reserved for future use.
Key Value	Reserved for future use.
Return Attribute	Enter an attribute name whose value gets returned to the browser when you click the blue square next to a Tree Node line item.
Enable Return?	Select this property to enable the Return Attribute property functionality (i.e., clicking the blue square).

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation  
2Z4A/101  
11400 Burnet Road  
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.



# Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM  
AIX  
DB2  
developerWorks  
Everyplace  
ibm.com  
Lotus  
Maximo  
Notes  
QuickPlace  
Tivoli  
WebSphere

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

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

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

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

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

UNIX is a registered trademark of The Open Group in the United States and other countries.



## A

- application bar (appbar)
  - properties table 51
  - reference information 51
- Application Designer
  - overview 1
- application dialogs
  - modifying 21
- application from a text field
  - linking to 23
- application restrictions
  - creating 11
  - filtering records with Application Designer 11
- applications
  - creating 13
  - deleting 15
  - duplicating 12
  - exporting with Application Designer 23
  - how to link from a text field 23
- ASSET object
  - adding assetcomment attribute 28
- assetcomment
  - add textbox using Application Designer 31, 32
- attachments
  - properties table 52, 55, 98
  - reference information 52
- attributes
  - adding 35
  - adding assetcomment textbox to Assets 31
  - adding assetcomment textbox to Work Order Tracking 32
  - adding assetcomment to ASSET object 28
  - adding textbox for assetcomment in Assets 28
  - adding woassetcomment textbox to Work Order Tracking 37
  - adding woassetcomment to WORKORDER object 35

## B

- blank line
  - properties table 53
  - reference information 52
- breadcrumbs
  - reference information 53
- button group
  - properties table 57
  - reference information 56
- buttons
  - adding 16

## C

- checkbox
  - properties table 58
  - reference information 57
- combobox
  - properties table 61
  - reference information 60
- conditional user interface, control properties 41
- configDB
  - configuring the database 29
  - using 29
- configurations
  - moving across servers 22
- configuring
  - the database 29
- control properties
  - configuring 41
- controls
  - application bar (appbar) 51
  - attachments 52
  - blank line 52
  - breadcrumbs 53
  - button group 56
  - checkbox 57
  - combobox 60
  - data source 63
  - default value 66
  - help grid 68
  - hiding 42
  - hyperlink 69
  - image 72
  - include 74
  - layout 5
  - list of 2
  - listbox 75
  - menubar 78
  - moving across applications 21
  - moving across tabs 21
  - multiline textbox 78
  - multipart textbox 82
  - parameter value 87
  - parameter values 89
  - pushbutton 91
  - radio button 93
  - radio button group 94
  - record image 97
  - reference information 51
  - section 99
  - section column 104
  - section header 105
  - section row 107
  - showing 42
  - static text 108
  - tab 109
  - tab group 112
  - table 112

## Index

- table column 119
- textbox 126
- tree 131
  - tree attribute 134
  - tree node 135
- controls reference 51
- crossover domain
  - scenario 26
- crossover domains
  - adding 36
  - applying 35, 37
  - creating 35
  - documentation scenario 26
  - examples 25
  - overview 25
  - user scenario overview 34
  - using to display information 34

## D

- data fields
  - optimizing placement 16
- data source
  - properties table 64
  - reference information 63
- database
  - configuring 29
- database relationships
  - examples 25
  - linking objects 30
- default filter for a table
  - defining 20
- default label value
  - assigning 18
  - overriding 18
- default values
  - defining for fields 19
  - defining with Application Designer 19
  - properties table 67
  - reference information 66
- dialogs
  - modifying with Application Designer 21

## E

- environment, setting up 7
- examples
  - crossover domains and database relationships 25
- external URL
  - linking to 23

## F

- field sizes
  - controlling 15
- fields, optimizing placement 15

## G

- Go To menu
  - modifying contents 24
  - using with Application Designer 24

## H

- help grid
  - properties table 68
  - reference information 68
- how to
  - add a value list to a field 9
  - add buttons and menu items 16
  - assign or override the default label value 18
  - create a new tab 14
  - create application restrictions 11
  - create new applications 13
  - define a default filter for a table 20
  - define field default values 19
  - define security options for an application 17
  - delete a user-created application 15
  - duplicate an application 12
  - export applications 23
  - link to an application from a text field 23
  - link to an external URL 23
  - modify application dialogs 21
  - move controls across applications 21
  - move controls across tabs 21
  - move user interface configurations across servers 22
  - optimize field placement for data entry 16
  - restrict field sizes 15
- hyperlink
  - properties table 70
  - reference information 69

## I

- image
  - properties table 73
  - reference information 72
- include
  - properties table 75
  - reference information 74
- input fields
  - placement 16
  - restricting size 15
- interface, conditional 41

## L

- linked objects
  - overview 25
  - user scenario overview 27
  - using to display information 27
- linking
  - WORKORDER and ASSET objects 30

- listbox
  - properties table 76
  - reference information 75

## M

- menu items
  - adding 16
- menubar
  - properties table 78
  - reference information 78
- messages
  - adding 24
- monitor
  - recommended resolution 7
- multiline textbox
  - properties table 79
  - reference information 78
- multipart textbox
  - properties table 83
  - reference information 82
- multiple applications
  - exporting 23

## O

- objects
  - linking 30
  - linking via database relationships 30
- overview
  - Application Designer 1
  - controls reference information 51
  - crossover domains scenario 34
  - linked objects and crossover domains 25
  - linked objects user scenario 27

## P

- parameter value
  - properties table 88
  - reference information 87
- parameter values
  - properties table 90
  - reference information 89
- preparation 7
- presentation XML
  - defined 3
  - editing 4
  - exporting 4
  - importing 4
  - modifying 4
  - overview 3
- properties table
  - application bar (appbar) 51
  - attachments 52, 55, 98
  - blank line 53
  - button group 57
  - checkbox 58

- combobox 61
- data source 64
- default value 67
- help grid 68
- hyperlink 70
- image 73
- include 75
- listbox 76
- menubar 78
- multiline textbox 79
- multipart textbox 83
- parameter value 88
- parameter values 90
- pushbutton 92
- radio button 94
- radio button group 95
- section 100
- section column 105
- section header 106
- section row 107
- static text 108
- tab 110
- tab group 112
- table 114
- table column 120
- textbox 127
- tree 133
- tree attribute 134
- tree node 136
- pushbutton
  - properties table 92
  - reference information 91

## R

- radio button
  - properties table 94
  - reference information 93
- radio button group
  - properties table 95
  - reference information 94
- record image
  - reference information 97
- reference information
  - controls 51
- rules and recommendations 7

## S

- scenarios
  - for linked objects and crossover domains 26
- section
  - properties table 100
  - reference information 99
- section column
  - properties table 105
  - reference information 104
- section header
  - properties table 106
  - reference information 105
- section row
  - properties table 107

## Index

- reference information 107
- security
  - adding or modifying options 17
  - list of signature options 47
- Select Action menu items, adding 16
- servers
  - moving user interface configurations 22
- signature options
  - common 47
  - defining 17
  - list 47
- static text
  - properties table 108
  - reference information 108

## T

- tab group
  - properties table 112
  - reference information 112
- table
  - defining a default filter 20
  - properties table 114
  - reference information 112
- table attribute
  - properties table 134
- table column
  - properties table 120
  - reference information 119
- tabs
  - laying out 14
  - properties table 110
  - reference information 109
- text field
  - linking to an application 23
- textbox
  - adding 32
  - properties table 127
  - reference information 126
- textbox control
  - assigning a default label value 18
  - overriding the default label value 18, 19
- textboxes
  - adding 31, 37
- tree
  - properties table 133
  - reference information 131
- tree attribute
  - reference information 134
- tree node
  - properties table 136
  - reference information 135

## U

- URL
  - linking from an application 23
- usage rules and recommendations 7
- user interface
  - architecture 1
  - conditional 41
- user scenarios

- for linked objects and crossover domains 26

## W

- woassetcomment
  - add textbox using Application Designer 37
- WORKORDER object
  - adding woassetcomment attribute 35

## X

- XML
  - presentation 3