

# maximo enterprise suite

## Application Designer Quick Start Guide

July 2006

**Maximo**

Release 6.0

**mro software**

make it *all* count

This document and its publication do not constitute or create a contract. MRO Software, Inc. makes no warranties, express or implied, as to the accuracy or completeness of this document or with respect to the related software.

© 2006 MRO Software, Inc. All rights reserved. This document contains confidential and trade secret information of MRO Software, Inc. Use, transfer, disclosure, or copying without MRO Software, Inc.'s express written permission is strictly forbidden.

**Patents:** United States Patent Nos. 6,324,522 B2, 6,519,588 B1, and Aust. Pat. No. 758001. Multiple foreign patents pending.

**U.S. Restricted Rights:** If Customer is a government agency, Customer acknowledges and agrees that the Licensed Software is provided with RESTRICTED RIGHTS. Subparagraph (c)(1)(ii) of The Rights in Technical Data and Computer Software clause at 252.227-7013 of the Department of Defense FAR Supplement and FAR clause 52.227-19 entitled Commercial Computer Software Restricted Rights, apply and use, duplication, or disclosure by the Government is subject to restrictions as set forth in this Agreement. The aforementioned restrictions shall prevail over any similar "Rights" provisions under the laws of any country. Contractor/Manufacturer: MRO Software, Inc., 100 Crosby Drive, Bedford, MA 01730.

**Trademarks:** Maximo® is a registered trademark of MRO Software, Inc. The following table contains a list of MRO Software's trademarks and service marks:

Maximo® Enterprise	Maximo® SLA Manager	Maximo® OCS
Maximo® Enterprise/SP	Maximo® Navigator	Maximo® Mobile Suite
Maximo® Enterprise IT	Maximo® Project Manager	Maximo® Mobile Auditor
Maximo® Asset Center	Maximo® Calibration	Maximo® Mobile Inventory Manager
Maximo® Service Center	Maximo® Enterprise Adapter	Maximo® Mobile Work Manager
Maximo® Discovery	Maximo® Fusion	Maximo® Mobile Calibration
Maximo® Enterprise IT/SP		

IBM® and WebSphere® are registered trademarks of IBM Corporation. WebLogic® is a registered trademark of BEA Systems, Inc. Broadvision® and related marks are registered trademarks or trademarks of Broadvision, Inc. webMethods® is a registered trademark of webMethods, Inc. Snowbound™ and RasterMaster™ are trademarks of Snowbound Software Corporation. Syclo® and Agentry® are registered trademarks of Syclo, LLC.

Other products and brand names are trademarks or registered trademarks of their respective companies.

**Third-Party Technology:** Certain MRO Software, Inc. products contain technology provided under license from third parties, as noted in the following table:

MRO Software Products	Third-Party Information
Maximo	Portions © 1995-2004 Actuate Corporation. Portions © 2003 BEA Systems, Inc. BEA WebLogic® Server™ provided by BEA Systems, Inc. Portions © 1996-2004 IBM Corporation. IBM® WebSphere® provided by IBM Corporation. Portions © 1996-2005, i-net software GmbH.
All Products	Portions © 1996-2003 Visual Mining, Inc. Visual Mining™ NetCharts Server™ provided by Visual Mining, Inc.
Maximo® Discovery	©1988-2004 Centennial Software Limited. MSDE © Microsoft Corporation.
Maximo Navigator	Portions © 1993-2002 Snowbound Software Corporation. RasterMaster™ Raster imaging technology provided by Snowbound Software Corporation. Portions © 1989-1998 Cimmetry Systems, Inc.
Maximo Mobile Suite	Portions © 2002-2003 Syclo LLC.

**Open Source:** Maximo contains computer software obtained from the public domain, known as "Open Source." A complete listing of all Open Source contained in Maximo may be viewed at <http://www.mro.com/support/opensource>, ownership of which is attributed as follows: Portions © 2005, International Business Machines Corporation and others. Portions © 2002, Steve Souza (admin@jamonapi.com). Portions © 2000 by Jef Poskanzer (jef@acme.com). Portions © 2000-2004 Jason Hunter & Brett McLaughlin. Portions © 2004-2005, The Apache Software Foundation (<http://www.apache.org/>). All Rights Reserved.

# Contents

<b>About This Guide .....</b>	<b>1</b>
Why Read This Guide?.....	1
Who Should Read This Guide? .....	1
Related Documentation.....	1
Support.....	1
<b>What is the Application Designer? .....</b>	<b>3</b>
Overview .....	3
<b>Application Technology .....</b>	<b>4</b>
User Interface Architecture.....	4
Controls .....	4
Presentation XML.....	6
Process of modifying the presentation XML .....	7
Exporting and importing presentation XML's.....	7
Control layout.....	7
Editing XML files.....	8
<b>Before using the Application Designer .....</b>	<b>9</b>
Setting up an environment .....	9
Usage rules and recommendations .....	10
<b>Using the Application Designer to edit Maximo applications .....</b>	<b>11</b>
How to add a value list to a field .....	11
How to create and apply a crossover domain .....	13
Creating the crossover domain .....	14
Applying the crossover domain .....	14
How to display data in Maximo applications from linked objects .....	15
How to create application restrictions .....	17
How to duplicate an application.....	18
How to create a new application .....	18
How to layout a new tab .....	20
How to delete a user-created application.....	20
How to control the size of fields.....	21
How to optimize the placement of input fields for data entry.....	22
How to add buttons and select action menu items.....	22
How to define signature options for an application .....	23
How to assign and override the default label value for a textbox control .....	23
Assigning a value to a default label field .....	24
Overriding a value in the default label field.....	24
How to define default values for fields.....	25
How to define a table's default filter .....	25

How to modify application dialogs .....	26
How to move controls across tabs.....	27
How to move controls across applications .....	27
How to move user interface configurations across servers.....	28
How to export multiple applications .....	28
How to link to an external URL .....	29
How to link to an application from a text field.....	29
How to modify contents of the Go To menu .....	30
<b>Appendix A – List of Signature Options.....</b>	<b>31</b>
Common Signature Options .....	31
Specific Signature Options .....	32
<b>Appendix B – Controls Reference Information.....</b>	<b>34</b>
Appbar.....	34
Attachments.....	35
Blank Line.....	37
Button Group.....	38
Checkbox .....	39
Combobox.....	43
Data Source.....	47
Default Value.....	51
Help Grid.....	53
HyperLink.....	55
Image.....	59
Include.....	61
Listbox.....	63
Menubar .....	66
Multiline Textbox .....	68
Multipart Textbox .....	72
Parameter Value .....	77
Parameter Values.....	78
Pushbutton.....	81
Radio Button.....	83
Radio Button Group .....	85
Section .....	89
Section Column.....	93
Section Header.....	95
Section Row.....	97
Static Text.....	98
Tab.....	101
Tab Group .....	104
Table.....	105
Table Column.....	111
Textbox.....	118
Tree.....	122
Tree Attribute .....	127
Tree Node .....	128

# About This Guide

## Why Read This Guide?

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

## Who Should Read This Guide?

Any user involved with configuring Maximo can benefit from reading this guide. All users should read the Application Designer's online help content before performing any of the steps outlined in this guide.

## Related Documentation

- ▼ Online help for Application Designer
- ▼ *Maximo System Administrator's Guide*

## Support

To find the nearest corporate office, visit the company contact page at the MRO Web site:

[www.mro.com](http://www.mro.com)

MRO Software, Inc. users with a valid Annual Customer Support Plan (ACSP) can obtain product support online at:

[support.mro.com](http://support.mro.com)

**Note** To use the MRO Software Support Online Web site, you must provide your contact information and your MRO Software product serial/license number.

The MRO Software Support Online Web site includes information on product releases, software patches, and documentation updates. To find the most current version of a document, refer to the Support Web sites Knowledge Base.

# What is the Application Designer?

This section provides overview information for the Application Designer application introduced in Maximo Enterprise Suite (MXES).

## Overview

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

The easy-to-use Application Designer interface enables you to change the appearance of Maximo 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 a selected application's user interface:

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

Advanced capabilities include:

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

Although you can create a new or duplicate an existing application with this tool, you may 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

# Application Technology

## User Interface Architecture

The following sections discuss the user interface (UI) and general architecture of the Application Designer, including the following topics:

- Controls
- Presentation XML (Extensible Markup Language)
- Process of modifying the presentation XML
- Exporting and importing presentation XML's
- Control layout
- Editing XML files

## Controls

Maximo uses “controls,” numerous pre-fabricated elements like tables and textboxes, to build the user interface for its applications. The individual controls are part of Maximo’s 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 Maximo business object (MBO) attribute. Attributes store values from the Maximo database that are referenced in a control.

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

Control Name	Description
Tabgroup	A container for adding tabs.
Tab	An individual tab in a tab group.
Section	A container for adding controls such as section columns, section rows, fields, buttons, combo boxes, and check boxes.
Section Column	A column in a section. This control can be only

Control Name	Description
	be used in a row within a section.
Section Row	A row in a section.
Section Header	A container to label a section or a table window, with the capability to display dynamic values in its label.  This control contains the parameter values control which is a container for the parameter value control.
Parameter Values	A container for the parameter value control to be used in sections and table windows.
Parameter Value	Dynamic values that reside within a section header.
Combobox	A combo box with drop-down functionality.
Checkbox	An HTML check box.
Blank Line	Inserts white space in a section.
Help Grid	Help text displayed in dialog boxes.
Image	Allows you to insert an image.
Listbox	An open, drop-down menu which displays dynamic content.
Multiline Textbox	A text input box with multiple lines.
Multipart Textbox	Two text boxes next to each other; one for displaying text and the other a description.
Static Text	Informational text displayed in an application.
Textbox	A text input box with a single line.
Table	A container for a table window.
Table Column	A container for a column in a table window.
Tree	A container for a hierarchical structure.
Tree Attribute	A value to describe the tree node control within the tree control.
Tree Node	A container for defining the child elements within the tree control.
AppBar	A container that displays all the applications, belonging to the current module, except for the currently displayed application.
Menubar	A menu bar that contains signature options for the Search, Query, and Bookmark menus.
Hyperlink	A hyperlink that supports both text and

Control Name	Description
	graphics.
Button Group	A container for adding buttons.
Pushbutton	Individual buttons that belong to the button group.
Radio Button Group	A container for radio buttons.
Radio Button	An individual radio button that belongs to the radio button group.
Default Value	Forces a default value for a control that resides within a container.
Attachments	An attachments icon that allows you to attach different types of information to a Maximo record, including images, spreadsheets, and Web pages.

## Presentation XML

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

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

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

## Process of modifying the presentation XML

When you display an application in the Application Designer, Maximo loads its presentation XML in memory. When you edit the application by moving, adding, and changing attribute values and deleting controls, the Application Designer edits the XML in memory to reflect 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.

**Note** Controls that represent user interface elements carried over from versions of Maximo earlier than 6.0 have an ID that loosely corresponds to the JSP tag structure in the first Web-based version of the product. The difference in control ID's from 5.x to 6.0 will make it easier to identify user interface elements when comparing 5.X to 6.0 screens.

## Exporting and importing presentation XML's

You have to 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 screen using the Application Designer as the user interface editor.

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

MRO recommends that you limit XML edits to simple changes, such as modifying labels, to ensure that changes migrate with subsequent upgrades of Maximo.

## Control layout

The positioning of controls in the Maximo user interface is completely relative. The user interface is neither pixel based nor is it grid based as in 5.x releases. 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 a section's columns and overall width. For example, a section that contains fields of various widths will automatically size itself relative to the largest field (widest) in that section.

## Editing XML files

There are times when editing the XML file of a presentation is more efficient than using the Application Designer to make changes. In some cases, the only way for you to make a desired change to a presentation is to edit system level XML code.

If you want to make a presentation change by editing XML code, you must first use the export function in the Application Designer to export the desired application's XML code to a file. Once exported, you must save the file to a local directory, and then edit that file with a simple text editor. Once completed, you can import the modified XML file using the import function of the Application Designer.

# Before using the Application Designer

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

## Setting up an environment

Before using 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 floating dialog boxes, like the Control Palette and Control Properties, outside of the application's configurable screen area.

- 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, you migrate the changes to your production environment so that they are available to all users.

Employing a multi-stage model insulates your production environment from configuration changes made in development and staging.

## Usage rules and recommendations

You should familiarize yourself with the following rules and recommendations when using the Application Designer:

1. Only one user should edit an application on the same server.
2. MRO recommends that you create the data source and associated objects for a new application in Database Configuration before creating screens with the Application Designer.
3. You can only delete user-created or duplicated applications. You cannot delete applications shipped with the product.
4. 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.
5. You can drag most controls into container type controls, such as a section or table.
6. The package key you use when installing Maximo determines what applications you can edit with the Application Designer.

# Using the Application Designer to edit Maximo applications

This section provides information on how to perform common application configuration tasks using the Application Designer and other Maximo configuration applications, such as Database configuration and domains.

## How to add a value list to a field

Maximo uses domains to limit the choice of values that you can enter in a field or to filter what 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 Database Configuration 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 Database Configuration.

The following example shows how you 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. Open the Domains application by choosing Go To > Configuration > Domains.
2. Click Add New Domain > Add New ALN Domain.
3. Enter "Assetstatus" in the Domain field, "Asset Status" in the Description field, "UPPER" in the Data Type field, and "8" for the Length.

4. In the ALN Domain screen, click New Row, and enter “ACTIVE” in the Value and Description fields.
5. Repeat Step 4, and enter values of “PENDING” and “INACTIVE.”
6. Click OK, and then click Save.
7. Open the Database Configuration application by choosing Go To > Configuration > Database Configuration.
8. Type “ASSET” in the Object field, and press Enter.
9. Click on “ASSET” to open the object record.
10. Click the Attributes tab, and then click New Row.
11. 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
12. Click Save.
13. To save the changes to your database, complete the following:
  - Sign out of Maximo.
  - Shut down the application server (Ctrl + C), and then wait a minute.
  - Open a command shell (Start > Run > cmd), and navigate to the <maximo\_root>/tools/maximo directory.
  - Run configdb.bat.  
  
Refer to the *Maximo System Administrator’s Guide* for additional details on configuring the database.
  - Restart the application server, and login to Maximo.
14. Open the Application Designer application by choosing Go To > Configuration > Application Designer.
15. Type “ASSET” in the Application field, and press Enter.
16. Click on ASSET to open the Assets application in the AD workspace.
17. Click the Asset tab.

18. Click on the Control Palette icon.
19. Click on the Section label that holds the Asset and Status fields. The section background turns green.
20. 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.
21. Select the textbox control (the background turns green), and click the Control Properties icon.
22. 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).
23. Type “VALUELIST” in the Lookup field, and press Tab. Note that the AD 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.
24. Click Save.
25. Open the Assets application by choosing Go To > Assets > Assets 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 <maximo\_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 to import the presentation file into the database by using the Import Application Definition button, which is available from both the Applications and Workspace tabs of the Application Designer.

## How to create and apply a crossover domain

A crossover is the mechanism that Maximo uses to copy data from a child object to a main object based on a value you enter in an application’s user interface. 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 (assetnum) to a work order after you enter the asset number in a new work order record.

Creating a crossover is a multi-procedure process. The first procedure is to create the crossover using the Domains application. The second procedure is to apply the crossover using the Database Configuration application.

To illustrate how to create and apply a crossover, let's describe how to use a crossover to copy the parent asset number to a new work order record.

The WORKORDER object manages work order records. To create a work order record, you generally enter an asset number. Maximo stores that value as part of the work order record in WORKORDER. Additionally, the ASSET object maintains information about assets, including the parent asset number, in its attributes. The PARENT attribute points to the parent asset number.

The crossover enables Maximo to copy and save the parent asset number of an asset once you create a work order and enter an asset number.

## Creating the crossover domain

Follow the steps below to create the crossover:

1. Open the Domains application by choosing Go To > Configuration > Domains.
2. Click Add New Domain > Add New Crossover Domain.
3. Enter "ASSET2WO" as the Domain name.
4. Click New Row in the Crossover Domain section.
5. Enter "ASSET" in the Object field. This step defines ASSET as the source object of the crossover.
6. Enter "assetnum=:assetnum and siteid=:siteid" in the Validation Where Clause field.
7. Click New Row in the Crossover Fields section.
8. Enter "PARENT" in the Source Field and "PARENTASSET" in the Destination Field.

## Applying the crossover domain

1. Open the Database Configuration application by choosing Go To > Configuration > Database Configuration.
2. Type "WORKORDER" in the Object field, and click Enter.
3. Click WORKORDER to open the Work Order table.
4. Click the Attributes tab, and then click New Row.
5. Enter "PARENTASSET" in the Attribute field, and click Save.

6. Click Filter, and enter “ASSETNUM” in the Attribute field.
7. Press Enter to open the ASSETNUM details section, and enter “ASSET2WO” in the Domain field.
8. Click Save.
9. Configure the database by running configdb.bat.

Refer to the *Maximo System Administrator's Guide* for additional details on configuring the database.

In summary, by completing the preceding procedures you:

- Created a crossover domain that retrieves asset information when a user enters an asset number in a work order record.
  - The crossover 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 PARENT attribute (the source) of the matching record in the ASSET object and copies it to PARENTASSET attribute (the destination) in the WORKORDER object.
  - The Parent Asset Number now appears in the work order record, but the data is static. Static means that the parent asset number data will not change in the work order record even if someone subsequently modifies the parent asset value in the ASSET object.

## How to display data in Maximo applications from linked objects

Maximo binds the presentation XML for each of its applications to a single main object that represents a data table in the Maximo database. The Work Order Tracking application, for example, uses WORKORDER as its main object. The user interface for a Maximo application uses controls that are bound to database attributes of the main object to present data in a browser window. The Work Order Tracking application, for example, displays information from the WORKORDER object.

Additionally, because most applications need to present data that resides in other objects, you need to use the Database Configuration application to define “relationships” between objects. The relationships you create between objects enables 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. Complete the following steps to link the ASSET object to the WORKORDER object so that Asset Number and other ASSET

table information can display in Work Order Tracking. This example assumes that you are adding a multipart textbox control to Work Order Tracking to display Asset Number and Description.

1. Open the Database Configuration application by choosing Go To > Configuration > Database Configuration.
2. Type “WORKORDER” in the Object field, and click Enter.
3. Click WORKORDER to open the Work Order table.
4. Click the Relationships tab, and then click New Row. To create a binding, parent to child relationship between the WORKORDER and ASSET objects, type the following information:
  - “ASSET” in the Relationship field
  - “ASSET” in the Child Object field
  - “assetnum=:assetnum and siteid=:siteid” in the Where Clause field

The Where Clause value ensures that lookups in the Asset table return a unique record value.

5. Click Save.

Now that you have created a relationship between the WORKORDER and ASSET objects, you can add controls to the Work Order Tracking application that bind to attributes in the Asset table. For example, the following steps show how you can add a multipart textbox control that displays Asset Number and Description.

6. Open the Application Designer by choosing Go To > Configuration > Application Designer.
7. Open Work Order Tracking in the Workspace tab.
8. Select the Work Order tab.
9. Click a section where you want to add the multipart textbox control.
10. Click the Control Palette button.
11. Drag the multipart textbox control into the selected section, and then click on the control.
12. Click the Control Properties button to open the Multipart Textbox Properties dialog box. To assign the control a field label and to bind to the Asset Number and Description attributes, type the following information:
  - “Asset” in the Default Label field
  - “ASSETNUM” in the Attribute field
  - “ASSETMAIN” in the Menu Type field

- “ASSET” in the Lookup field
- “ASSET” in the GOTO Application field
- “ASSET.DESCRPTION” in the Attribute for Part 2 field
- “READONLY” in the Input Mode for Part 2 field

13. Click Save.

Refer to the Database Configuration section in the *Maximo System Administrator's* guide for more information on defining relationships.

## How to create application restrictions

You can use the Application Designer to define restrictions on what records can display in an application. You can filter what data a user can see when opening an application by entering a query in the application's presentation control. You might want to filter what records a specific application displays when it and several other applications use the same data source, but not all of the data records in the set are applicable to all of the applications.

To define an application restriction, complete the following steps:

1. Open the Application Designer application by choosing Go To > Configuration > Application Designer.
2. Type the name of the application you wish to open, or press Enter to display a list of applications.
3. As an example, open the Assets application in the Workspace tab.
4. Click Select Action > Toggle Show All Controls.
5. Click the “presentation...” control.
6. Click the Control Properties button to open the Presentation Properties dialog box.
7. In the Where Clause field, enter “inactive='0’”. The Application Designer interprets the query as “select \* from asset where inactive='0’”.
8. Click Save.

When you open the Assets application, you will only see records where the inactive attribute has a value of “0”.

## How to duplicate an application

You can use the Application Designer to duplicate an existing Maximo application. In previous releases, duplicating an application was referred to as “cloning.” If you duplicate an application, you create a copy of the application’s presentation and application definition.

To duplicate an existing application, complete the following steps:

1. Open the Application Designer application by choosing Go To > Configuration > Application Designer.
2. Type the name of the application you wish to open, or press Enter to display a list of applications.
3. Open the application you want to duplicate in the Workspace tab.
4. Open the Duplicate Application dialog box by choosing Select Action > Duplicate Application Definition.
5. 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.
6. Click OK to open the new application in the Workspace tab.
7. Click Save.

The new application appears in the same module as the original application. To view the new application:

- Sign out of Maximo (refreshes the list of applications in the “Go To” menu)
- Sign in to Maximo
- Navigate to the new application

## How to create a new application

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

- Standard power application
- Single page
- Self service

Maximo 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. The template used to create a power application lets you build an application with 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 Maximo application

A single page application uses a variation of the power application's template. 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, complete the following steps:

1. Open the Application Designer application by choosing Go To > Configuration > Application Designer.
2. Click New Application Definition in the toolbar to open the Create a New Application dialog box.
3. Enter a unique name for the new application in the Application field.
4. Enter a brief description of the new application in the Description field. This name becomes the title of the application in the Maximo navigation bar and in the Go To menu.
5. Enter the name of the main object (primary database table) for the application in the Main Object field.

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

6. Enter the name of the Maximo module for this application. If needed, click the select value icon to display a list of modules.
7. Click the button for the type of application you want to create:
  - Power App
  - Single Page App
  - Self Service App

The new application opens in the Workspace tab.

8. Click Save Application Definition to save your new application to the database.

Maximo 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 the Add/Modify Signature Options select action. Refer to Appendix A for a complete list of signature options.

You can use the Controls and the Properties floating dialog boxes to add or modify controls (such as tabs, buttons, and table windows) in the new application.

To view your new application in the Maximo “Go To” menu, sign out and then back into Maximo.

## How to layout 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 make a section display a bar and a minimize/maximize button by completing the Section Properties dialog box.

Once you create the basic layout of a screen with the section controls, you can begin adding additional controls that display data such as the textbox control.

## How to delete a user-created application

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

**Note** Deleting an application does not remove any custom tables, fields, or domains that you might have created to support the application. Additionally, you must shut down the Maximo application server before you delete an application created with the Application Designer.

To delete an application that you created or duplicated, 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 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.

Executing the above statements delete the application from the Maximo database.

## How to control the size of fields

Maximo 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 this file:

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

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

## How to optimize 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, MRO recommends that you duplicate an existing application and then change the order of the input fields.

## How to add buttons and select 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 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 dialog box for entering attributes.

## How to define signature options for an application

The Application Designer enables you to add or modify signature options for a new or existing application. Signature options, which are listed in Appendix A, specify privileges for using Maximo applications, menu options, and toolbar items.

To define signature options, complete the following steps:

1. Open the Application Designer application by choosing Go To > Configuration > Application Designer.
2. Open an application in the Workspace tab.
3. Choose Select Action > Add/Modify Signature Options to open the signature options dialog box.
4. Click New Row.
5. Enter a valid signature value, for example INSERT or SAVE, in the Option field.
6. Enter a description for the signature option in the Description field.
7. You can enter additional signature options in the Option Also Grants field that Maximo will automatically grant 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.
8. In the Option Also Revokes field, enter any options that will be automatically revoked as a result of adding the primary option.
9. Click OK.

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

## How to assign and override 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 the Control Properties dialog box, 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. Maximo 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. Open the Application Designer application by choosing Go To > Configuration > Application Designer.
2. Open an application in the Workspace tab, for example, the Assets application.
3. Click the Asset tab, and select a section control.
4. Click the Control Palette button to open the Controls dialog box.
5. Drag the Textbox control into the highlighted section.
6. Select the Textbox control, and click the Control Properties button.
7. In the Textbox Properties dialog box, click the lookup icon for the Attribute field, and select ORGID.

Notice that the field label in the workspace and the Default Label field in the dialog box 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.

8. 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. Open the Application Designer application by choosing Go To > Configuration > Application Designer.
2. Open an application in the Workspace tab, for example, the Assets application.
3. Click the Asset tab, and select a textbox control.
4. Select the Textbox control, and click the Control Properties button.
5. In the Textbox Properties dialog box, 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.”

6. Click Save.

## How to define 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 you would 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 by completing the following steps:

1. Open the Application Designer by choosing Go To > Configuration > Application Designer.
2. Open an application in the Workspace tab.
3. Open the Control Palette.
4. 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.
5. Open the properties dialog box for the control, and enter the following:
  - The name of the attribute bound to the field to which you are assigning the default value
  - Enter the desired default value in the Value field
  - Select ‘Insert’ for Default Type
6. Click Save.

## How to define a table’s default filter

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,

Maximo 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 will only display 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`, Maximo displays item records with only `MOTOR` in the item description.

Complete the following steps:

1. Open the Application Designer by choosing `Go To > Configuration > Application Designer`.
2. Open an application in the Workspace tab.
3. 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`.
4. Open the properties sheet for the control, and enter the following:
  - The name of the attribute of the column to which you are assigning the default query
  - Enter 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
5. Click Save.

## How to modify application dialogs

You can use the Application Designer to edit application-specific dialog boxes that Maximo stores within the XML presentation of the application.

Complete the following steps:

1. Open the Application Designer by choosing `Go To > Configuration > Application Designer`.
2. Open an application in the Workspace tab.
3. Click the Edit Dialogs button to open a list of dialog boxes specific to the application.

You can edit the dialog content in the same way that you would edit any other user interface element in the workspace.

4. The Maximo user interface also provides general purpose dialogs that are used by more than one application. To edit these dialogs, such as Change Status, you must use a text editor to update the `library.xml` file located in the following directory:

<maximo\_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 the Import Application Definition button in the Application Designer. The import functionality is available in both the Applications and Workspace tabs.

## How to move controls across tabs

Use the Copy or Cut and Paste buttons 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 on Copy or Cut. Click the desired tab, and then select the target control (usually a section). Click the Paste button in the toolbar.

## How to move controls across applications

You can move an individual control or a section container of controls across applications by editing the XML presentation code and using the import/export buttons of the Application Designer.

Complete the following steps:

1. Open the Application Designer by choosing Go To > Configuration > Application Designer.
2. Open the source application in the Workspace tab.
3. Click the Export Application Definition button.
4. Choose File > Save As to save the XML definition to a file.
5. Open the destination application in the Workspace tab.
6. Repeat Steps 3 and 4.
7. Open the saved source and destination XML files in a text editor, such as WordPad.
8. 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(s) of code.

Use the unique ID of a control in the Application Designer's property sheet to simplify searching in XML presentation files.

9. Open the destination file, and search for the location where you want to copy the code from the source file. For example, choose Ctrl+V or choose Edit > Paste in a Windows environment.

10. Save both XML files.
11. In the Application Designer, open the destination application in the Workspace tab.
12. Click the Import Application Definition button, and enter the destination XML file name with your edit.
13. 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 Maximo 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 copied to the source destination application does not contain any property values.

## How to move user interface configurations across servers

You can move your user interface configurations to other servers using the export and import functionality of the Application Designer. These features enable you to easily move Maximo 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(s) to a backup directory. If you want to restore this instance of the user interface in the future, use the Import Application Definition feature. 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 `<maximo_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 Maximo using the Application Designer.

## How to export multiple applications

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

Complete the following steps:

1. Open the Application Designer by choosing Go To > Configuration > Application Designer.
2. In the Applications tab, click Select Records.
3. Check each application whose XML presentation you want to export.
4. Click the Export Application Definition button.
5. A new browser window opens a single XML file with the presentation code for each application you exported.

Note that the Import Application Definition function 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.

6. Choose File > Save As to save the XML file to a directory.

## How to link to an external URL

You can create a link to an external Web site from a Maximo 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 control’s property sheet.

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

## How to link to an application from a text field

Many text fields in Maximo 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.

Complete the following steps:

1. Open the Application Designer by choosing Go To > Configuration > Application Designer.
2. Open an application in the Workspace tab.
3. Select a textbox control, and click the Control Properties button.
4. Enter values in the GOTO Application and Menu Type fields.

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

For example, if you select the Asset field in Work Order Tracking, the properties dialog box has a value of “ASSET” in the GOTO Application field and a value of “ASSETMAIN” in the Menu Type field.

## **How to modify 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 creating a new application, the Application Designer lets you specify in which module you would like the new application to appear. When duplicating applications, the newly created application will be available in the same module as the originating application.

# Appendix A – List of Signature Options

This appendix lists the signature options available to new and existing applications. See “How to define signature options for an application” for more information.

## Common Signature Options

Control Name	Description
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
SEARCHVMQR	View/Manage Queries
SEARCHWHER	Where Clause

## Specific Signature Options

Control Name	Description	Example
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 SLAs	
ASTLOCCOMM	Associate Services	
AUTHSITES	Authorize Sites	
COMMCODE	Add/Modify Commodity Codes	
COMP	Complete	Complete Work Order
CONVERSION	Add/Modify Conversions	
CREATE	Create	CREATEWO – Create Work Order
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	
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	

Control Name	Description	Example
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	
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	

# Appendix B – Controls Reference Information

This appendix lists the controls available in the Application Designer control palette. Each control contains an introduction, rules, example, 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 Maximo module except the currently active application. An application bar appears in Maximo self service applications like Desktop Requisitions and Service Requests that do not use tabs.

### Rules

**This is not a container control.**

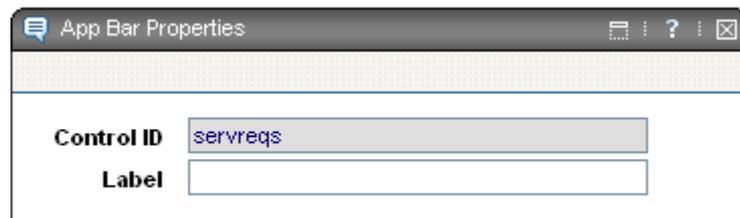
**You can place this control inside the following container control:**

- Client area

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

### Example

1. **AppBar Properties dialog box configured in the Application Designer.**



2. Appbar as it appears in the Application Designer workspace.



3. Appbar as it appears in the Create Service Request application.



**Properties**

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	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 dialog boxes associated with Attachments by dropping the control into any Section container in a Maximo or custom application. However, you cannot access the functions used to build the Attachments control through the Application Designer interface.

The Attachments control lets you change the label for Attachments that appear in various Maximo 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

## Example

1. Attachments Properties dialog box configured in the Application Designer.



2. Attachments control as it appears in the Application Designer workspace.



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	<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 Maximo user interface.

## Rules

**This is not a container control.**

**You can place this control inside the following container control:**

- Section

## Example



## Properties

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

# Button Group



A 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:

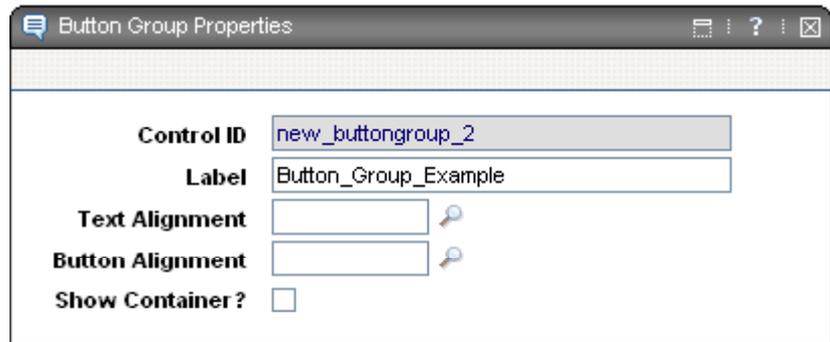
- Pushbutton

You can place this control inside the following container control:

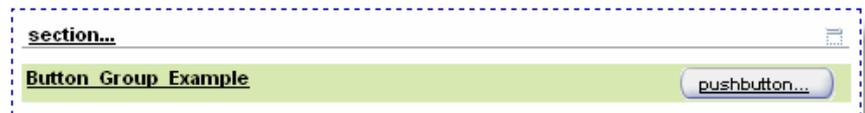
- Section
- Table

## Example

1. Button Group Properties dialog box configured in the Application Designer.



2. Button group as it appears in the Application Designer workspace.



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	<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 Maximo, 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>
<b>Text Alignment</b>	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.
<b>Button Alignment</b>	Reserved for future use.
<b>Show Container?</b>	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 dialog boxes 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 application's presentation file. Labels you enter at the

presentation level are not stored in the MAXATTRIBUTE OR MAXLABELS table.

Checkboxes appear in most applications and, depending on 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:

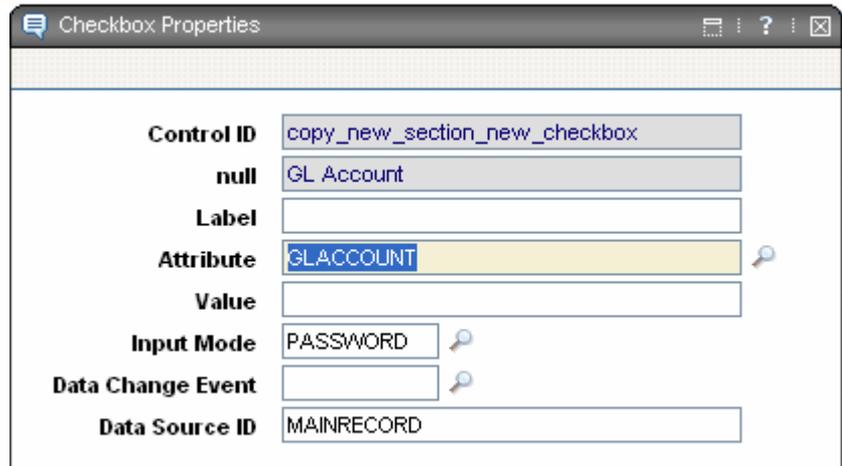
- Section

## Example

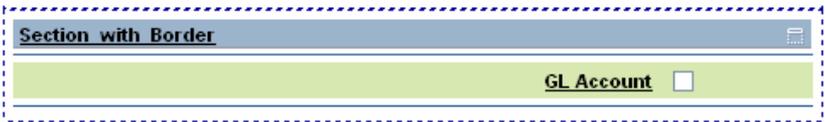
1. Checkbox control dropped into section without defined properties.



2. Checkbox Properties dialog box.

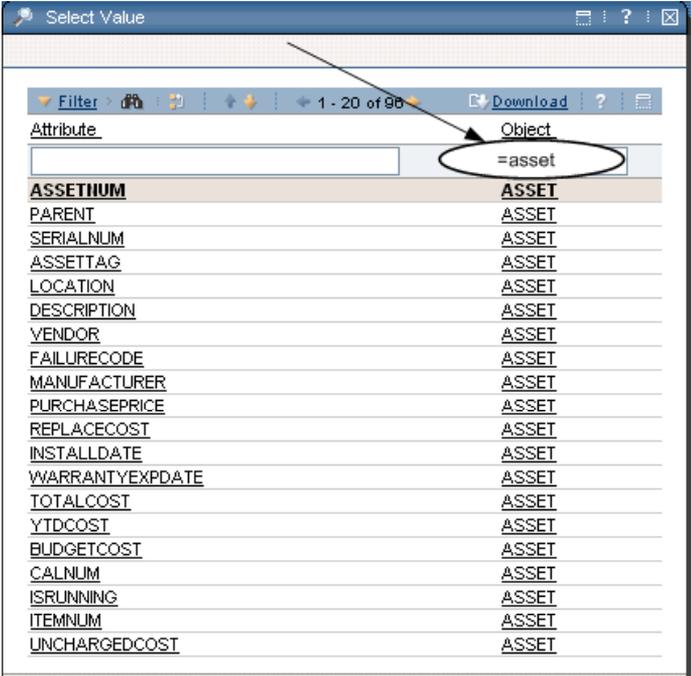


3. Checkbox with defined properties.



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

## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Null (aka Default Label)</b>	This label defaults to the text value associated with the database attribute you bound to the Checkbox. This is a read only field.
<b>Label</b>	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.
<b>Attribute</b>	<p>You can associate a Checkbox with a database attribute by using the Select Value dialog box. To create the example for this control, you can enter “=asset” in the Object field to display all the attributes in the ASSET table.</p>  <p>Select the “GLACCOUNT” attribute to inherit the “GL Account?” label for the Checkbox. Not all attributes you select will have a database default label value.</p>
<b>Value</b>	Reserved for future use.
<b>Input Mode</b>	<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</li> <li>• Query</li> <li>• Readonly</li> <li>• Required</li> </ul>

Attribute Name	Description
	<p style="text-align: center;">Readonly</p> <ul style="list-style-type: none"> <li>• Password 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, Maximo 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>
<p><b>Data Change Event</b></p>	<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>
<p><b>Data Source ID</b></p>	<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 dialog boxes 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 application's presentation file. 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 Maximo 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

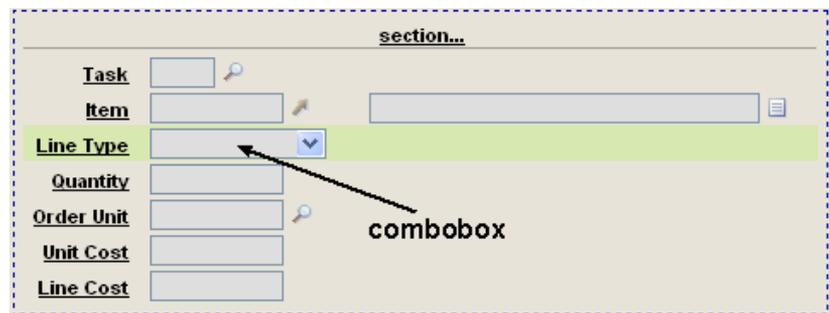
**This is not a container control.**

**You can place this control inside the following container control:**

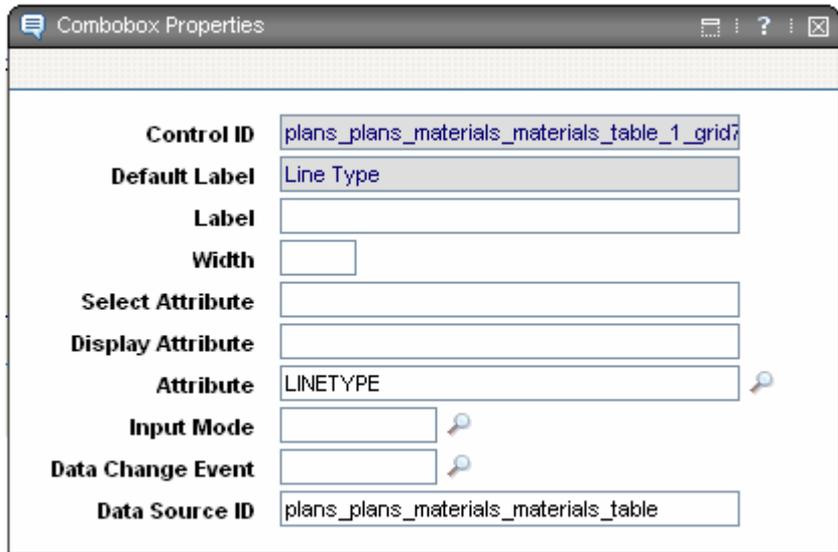
- Section

## Example

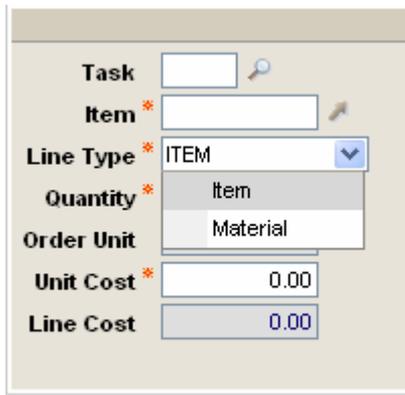
1. Combobox control in Details section of the Materials subtab in Work Order Tracking > Plans. The control label is Line Type. The application is Application Designer.



2. The Line Type Combobox Properties dialog box in the Application Designer.



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



Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control. You cannot edit this value.
<b>Default Label</b>	This label defaults to the text value associated with the database attribute you bound to the Combobox. This is a read only field.
<b>Label</b>	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.
<b>Width</b>	The Width field enables you to enter a numeric value to

Attribute Name	Description
	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.
<b>Select Attribute</b>	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.
<b>Display Attribute</b>	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.
<b>Attribute</b>	You can associate a Combobox with a database attribute by using the Select Value dialog box. 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.
<b>Input Mode</b>	<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, Maximo 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>

Attribute Name	Description
	<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>
<p><b>Data Change Event</b></p>	<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>
<p><b>Data Source ID</b></p>	<p>The ID of the data source for this control. A data source connects a control to a business object that displays and edits data using the control. Every presentation must minimally define at least one data source named MAINRECORD, which represents the main object for the application. In Work Order Tracking, for example, MAINRECORD links to the WORKORDER object. Similarly, in Purchasing, MAINRECORD links to the PO object.</p> <p>The “mboname” attribute that typically appears in the first line of the presentation file defines the link between the MAINRECORD data source and a business object. The MAINRECORD data source might be the only data source defined in a presentation. If other data sources exist, they are typically children of the MAINRECORD.</p> <p>The presentation defines child data sources via relationships using a “parentdatasrc” attribute (if the parent is not MAINRECORD) and a “relationship” attribute. For example, the Work Order Lock Out table in Work Order Tracking has a parentdatasrc=”WOTagOut_table” and a relationship=”WOTAGLOCK.” If a Data Source ID property is not explicitly defined for a control, the control looks at its parent (and up the presentation hierarchy to its parent’s parent) until it can find a data source that it</p>

Attribute Name	Description
	<p>can bind to. If there is not a control that defines an alternate data source, the control uses the MAINRECORD ID as its data source.</p> <p>The Materials subtab in Work Order Tracking &gt; Plans, for example, uses a Table control for entering planned material against a work order. The Materials table in Application Designer has a Relationship property value of "SHOWPLANMATERIAL." Additionally, if you open Database Configuration &gt; WORKORDER &gt; Relationships, you can view the details of SHOWPLANMATERIAL. This relationship uses the WPMATERIAL table to show the planned material for a given work order. Because all the children controls of the Materials table pull data from WPMATERIAL and not WORKORDER, they do not have MAINRECORD as the Data Source ID. Instead, the Materials Table control and all its children have a Data Source ID of "plans_plans_materials_materials_table" that captures the shared relationship between the Materials Table and its children controls.</p>

## Data Source



A Data Source is a miscellaneous type of control that is mostly used in Maximo 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

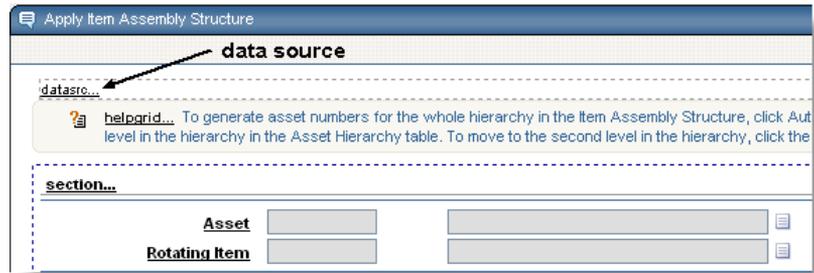
**This is not a container control.**

**You can place this control inside the following container controls:**

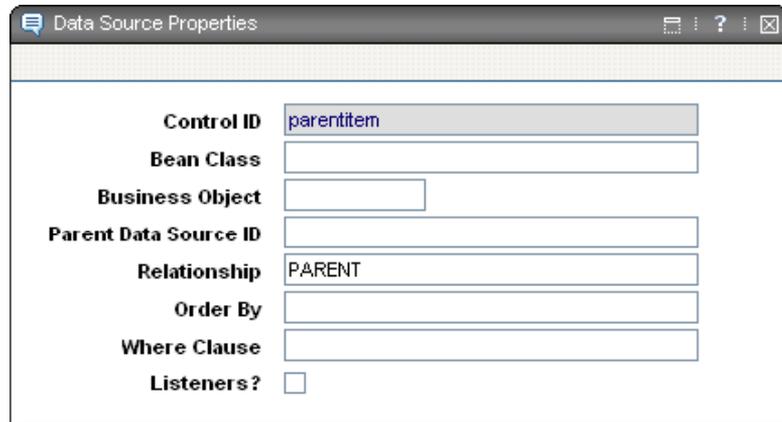
- Section
- Tab
- Table
- Clientarea

## Example

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



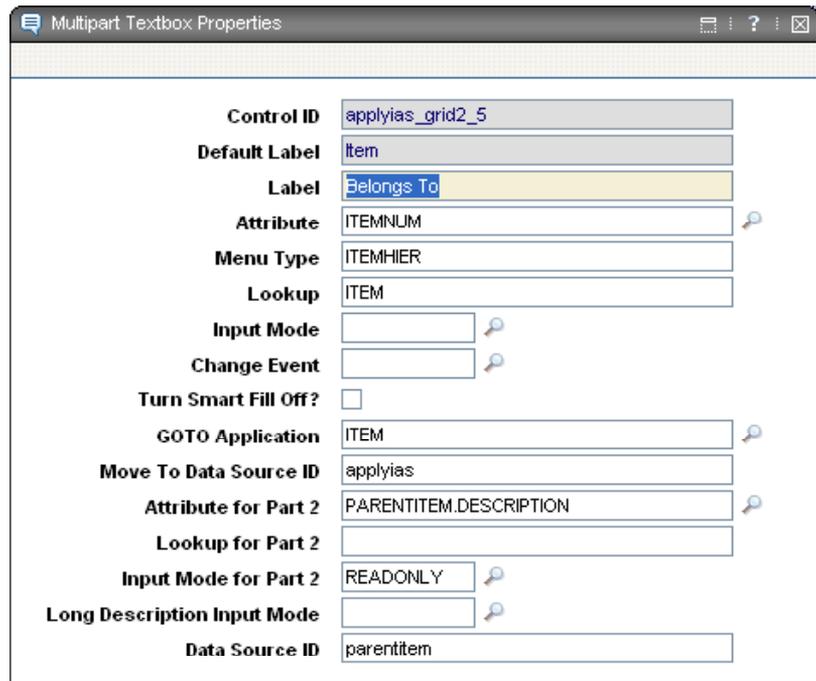
2. Data Source Properties configured in the Application Designer for the Apply Item Assembly Structure dialog.



3. Parent relationship defined in Database Configuration for Asset.



4. Belongs To field in the Application Designer > Apply Item Assembly Structure dialog configured to use “parentitem” defined in Step 2 to identify a parent asset number.



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Bean Class</b>	Reserved for future use.
<b>Business Object</b>	<p>The main MBO reference for the application. For example, Work Order Tracking uses WOTRACK and Assets uses ASSET.</p> <p>Every Maximo 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 application’s presentation tag 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
	<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>
<p><b>Parent Data Source ID</b></p>	<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>
<p><b>Relationship</b></p>	<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 Maximo 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>
<p><b>Order By</b></p>	<p>Reserved for future use.</p>
<p><b>Where Clause</b></p>	<p>Reserved for future use.</p>
<p><b>Listeners</b></p>	<p>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.</p>

# 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. Maximo 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 Maximo configures the Default Value Properties dialog box 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.**

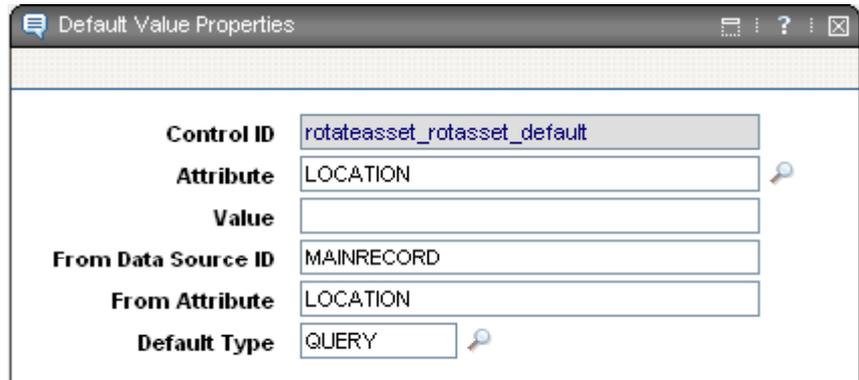
**You can place this control inside the following container controls:**

- Section
- Table
- Presentation

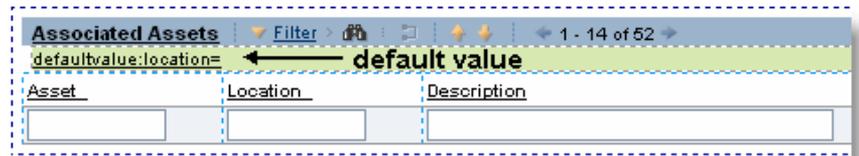
**Note** To view the presentation tag in Application Designer, select Toggle Show All Controls from the Select Action menu.

## Example

1. **Default Value Properties dialog box configured in the Application Designer for Inventory > Inventory > Rotating Assets.**



2. **Portion of Associated Assets table with Default Value control displayed in Application Designer.**



When you perform the query, Maximo populates the Location field for the Query by Example (QBE) with the value of the attribute defined by the From Data Source and From Attribute properties. For example, it copies the value from the Location attribute in the MAINRECORD (Asset Table) to the Location field in the QBE.

## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Attribute</b>	<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 dialog box to select an attribute from the MAINRECORD or any other table in Maximo. If you enter “=” before the object name, the dialog box displays only attributes for the table you</p>

Attribute Name	Description
	specify. You can also enter wildcard characters like “%” before or after a text string in the Attribute or Object field to filter your search.
<b>Value</b>	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.
<b>From Data Source ID</b>	Enter the Data Source ID for the From Attribute, if configured.
<b>From Attribute</b>	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.
<b>Default Type</b>	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 Maximo dialog box. 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 dialog boxes within the Maximo user interface. To view dialog boxes in the Application Designer, open an application and click the Edit Dialogs button in the toolbar.

### Rules

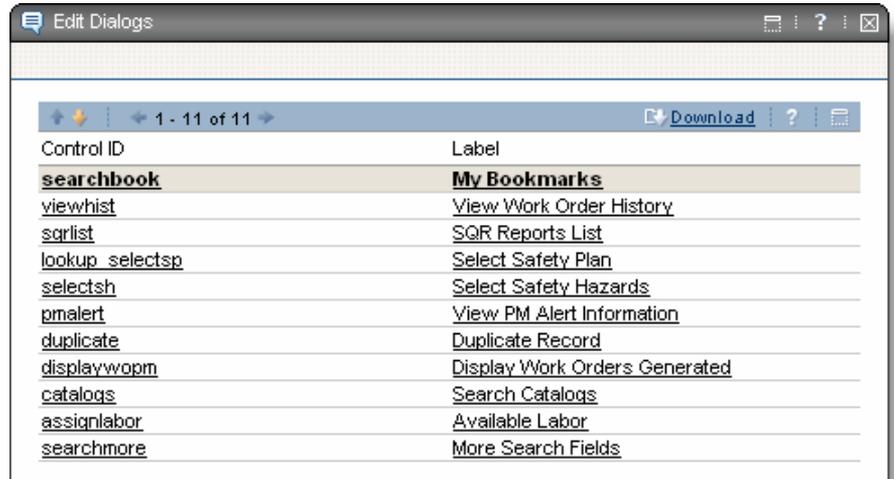
**This is not a container control.**

**You can place this control inside the following container controls:**

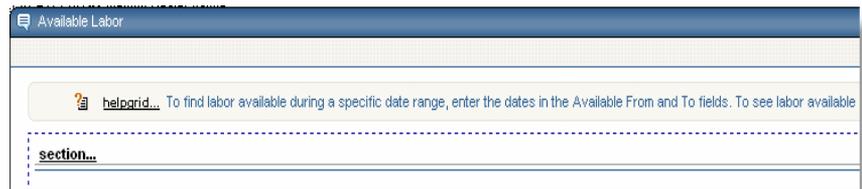
- Section
- Section Column
- Tab
- Dialogs

## Example

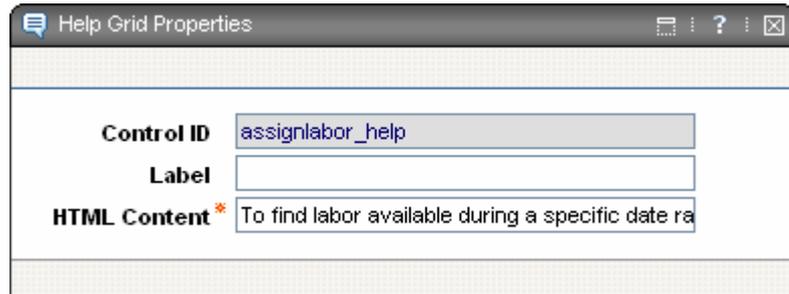
1. Open WOTRACK in the Application Designer, then click Edit Dialogs in the toolbar to open Edit Dialogs.



2. Open the "Available Labor" dialog. A portion of the Help Grid text appears below.



3. Select the "helpgrid..." control and the Control Properties icon to open Help Grid Properties.



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	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.
<b>HTML Content</b>	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 dialog box, 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 Maximo navigation bar are examples of hyperlinks.

### Rules

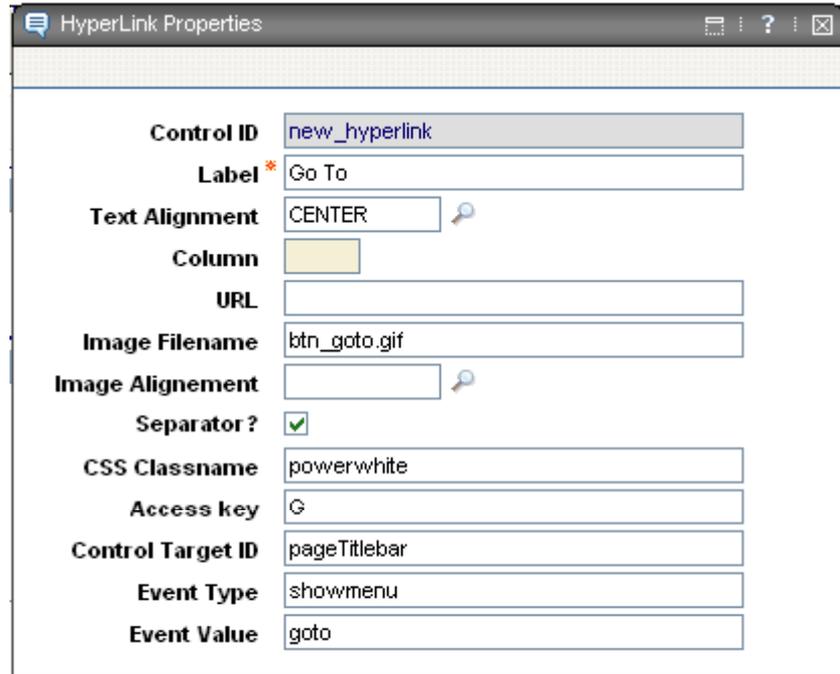
**This is not a container control.**

**You can place this control inside the following container control:**

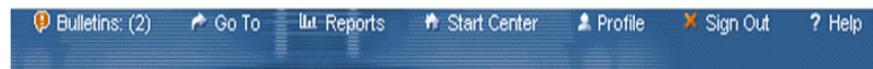
- Section

## Example

1. HyperLink Properties dialog box configured to display the Maximo Go To menu.



2. Hyperlinks as they appear in the navigation bar for all Maximo applications.



3. 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
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	<p>The name you assign to the Hyperlink control. This value displays in the user interface. You click the label to initiate the link.</p> <p>The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.</p>
<b>Text Alignment</b>	Select Left, Center, or Right to align the hyperlink label within a section.
<b>Column</b>	<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 Maximo 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 and</p>

Attribute Name	Description
	Maximo automatically extends the column width to contain the hyperlink.
<b>URL</b>	If you want the hyperlink to take you to a URL, enter the value here.
<b>Image Filename</b>	<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 Maximo library of images. However, if you insert and apply a graphic file without a .gif extension, then you need to rebuild/deploy the Maximo .EAR file.</p>
<b>Image Alignment</b>	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.
<b>Separator?</b>	Reserved for future use.
<b>CSS Classname</b>	<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>&lt;maximo_root&gt;\applications\maximo\maximouiweb\ webmodule\webclient</pre>
<b>Access Key</b>	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 <enter> to drop down the Maximo main menu. Maximo underlines the access key in the user interface.
<b>Control Target ID</b>	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.”
<b>Event Type</b>	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 Maximo Go To menu.

Attribute Name	Description
	<p>To view all of the events defined in Maximo, you must export the system .xml files by selecting a main tab from the Application Designer and choosing Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_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 application’s JSP main page 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 dialog box associated with the specified event.</li> </ol>
<b>Event Value</b>	Defines the event called when you select a hyperlink. The event value for the Go To menu is “goto.”

## Image



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

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

c:\maximo\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

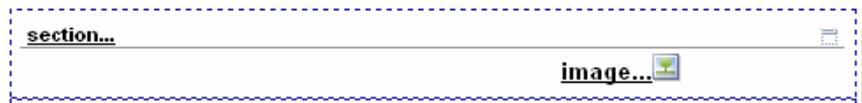
This is not a container control.

You can place this control inside the following container control:

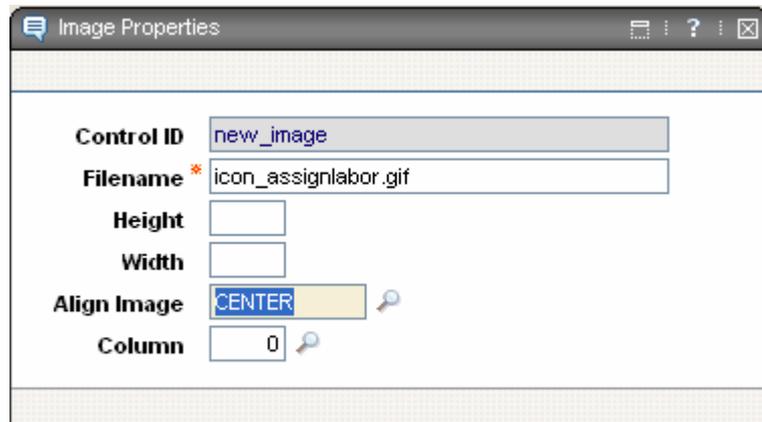
- Section

## Example

1. Image control dropped into a section in the Application Designer.



2. Image Properties dialog box configured to display the Assign Labor icon.



3. Assign Labor image displayed in a section of the Application Designer.



## Properties

Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Filename</b>	<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 Maximo library of images. However, if you insert a graphic file without a .gif extension into Maximo, then you need to rebuild/deploy the Maximo .EAR file.</p>
<b>Height</b>	Enter a numeric value in this field that will increase or decrease the height of the image.
<b>Width</b>	<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>
<b>Align Image</b>	Choose Left, Center, or Right to align an image with the left, center, or right margin of a section.
<b>Column</b>	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 Maximo uses together with presentation, page, and clientarea 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 browser's entire clientarea, from the application's dialogs. 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 clientarea control differentiates the body of a page or dialog from its header and footer. The clientarea must contain one or more Sections, a Table, and/or a Tabgroup 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 Maximo presentations.

The following example is an extract from the Work Order Tracking application presentation that shows how the include control imports the standard Maximo 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.**

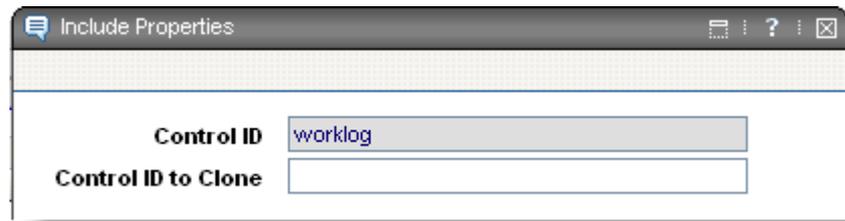
**You can place this control inside the following container controls:**

- Page
- Tab
- Section

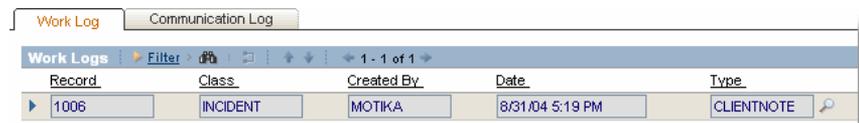
**Note** To view the page tag in Application Designer, select Toggle Show All Controls from the Select Action menu.

## Example

1. Include Properties dialog box configured in the Application Designer for Incidents > Log > Work Log.



2. Work Log table format inherited from “comlog” in library.xml.



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Control ID to Clone</b>	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 application’s presentation file. 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 dialog box. 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

## Example

No current example for this control. Not widely used in Maximo applications.

## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Default Label</b>	This label defaults to the text value associated with the database attribute you bound to the Listbox. This is a read only field.
<b>Label</b>	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.
<b>Height</b>	Enter a numeric value in this field that will increase or decrease the height of the Listbox.
<b>Width</b>	Enter a numeric value in this field that will increase or decrease the width of the Listbox.
<b>Bean Class</b>	Reserved for future use.
<b>Attribute</b>	You can associate a Listbox with a database attribute by using the Select Value dialog box. If the attribute you select has a default label value, it will display in the Default Label field.
<b>Input Mode</b>	Select one of 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, Maximo 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>
<b>Business Object Name</b>	Reserved for future use.
<b>Parent Data Source ID</b>	Reserved for future use.
<b>Order By</b>	Reserved for future use.
<b>Where Clause</b>	Reserved for future use.
<b>Auto-Refresh?</b>	Select this box if you want the information in the Listbox to automatically refresh whenever there is a change to the MBO.
<b>Key Attribute</b>	Reserved for future use.
<b>Key Value</b>	Reserved for future use.

# Menubar



The Menubar, short for menu bar, is similar in appearance to the Appbar. The bar that appears at the top of the List tab in most Maximo 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

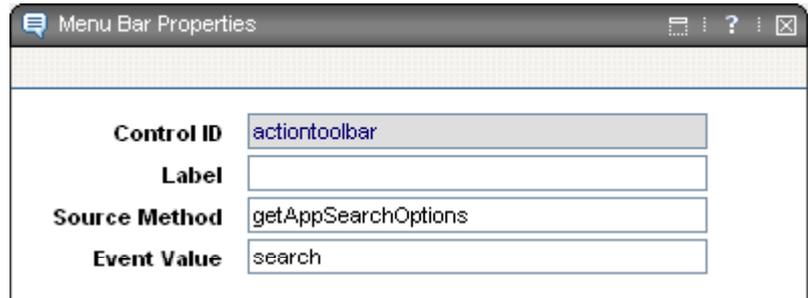
This is not a container control.

You can place this control inside the following container control:

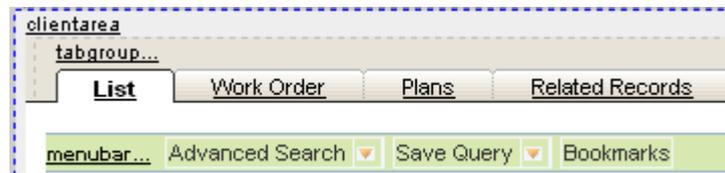
- Tab

## Example

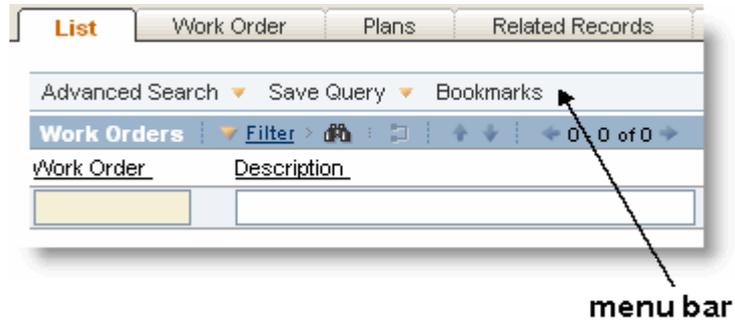
1. Menubar Properties dialog box configured in the Application Designer.



2. Menubar as it appears in the Application Designer workspace.



3. Menubar as it appears in Work Order Tracking.



**Properties**

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	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.
<b>Source Method</b>	The piece of code functionality that calls the menu bar, its signature security options, and layout. The method that calls the standard Maximo search menu bar is “getAppSearchOptions.”
<b>Event Value</b>	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 application's presentation file. 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 dialog box.

## Rules

**This is not a container control.**

**You can place this control inside the following container control:**

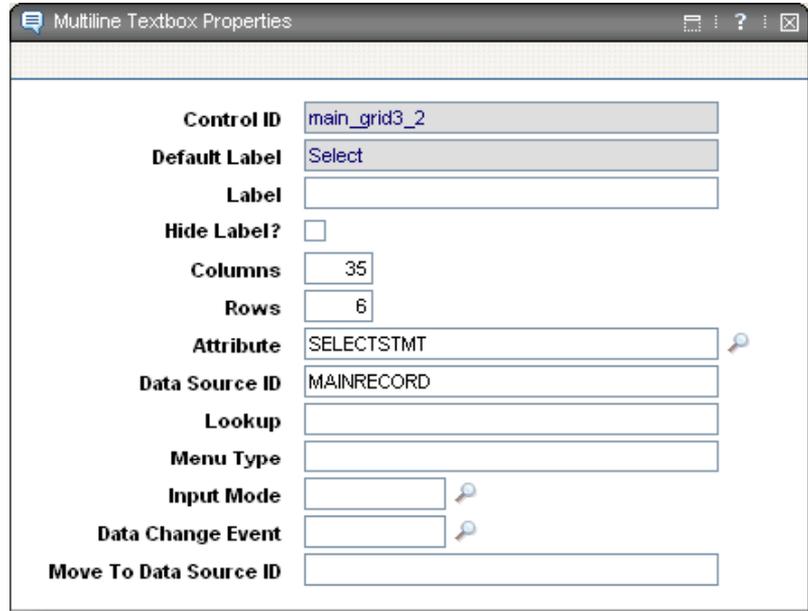
- Section

## Example

1. **Multiline textbox example in Reporting > KPI Manager. The textbox is in the KPI tab and part of the Select Query table.**

Select \*

2. The Multiline Textbox Properties dialog box for the preceding KPI example.



**Properties**

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Default Label</b>	This label defaults to the text value associated with the database attribute you bound to the textbox. This is a read only field.
<b>Label</b>	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.
<b>Hide Label?</b>	By default, the label value always appears in the user interface. Select this box to hide the label.
<b>Columns</b>	The number of columns in the Multiline Textbox. Most Multiline Textboxes in Maximo, 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.”
<b>Rows</b>	Enter an integer for the number of rows you want in the Multiline Textbox.
<b>Attribute</b>	You can associate a Textbox with a database attribute by using the Select Value dialog box. 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

Attribute Name	Description
	SELECTSTMT from the KPIMAIN table.
<b>Data Source ID</b>	<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 MAINRECORDED is KPIMAIN.</p> <p>See the Data Source ID description for the Combobox control for additional information on data sources.</p>
<b>Lookup</b>	<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 ID's 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 Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_root&gt;\resources\presentation\system directory.</p>
<b>Menu Type</b>	<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 ID's 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 Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_root&gt;\resources\presentation\system directory.</p>
<b>Input Mode</b>	<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</li> <li>• Query</li> <li>• Readonly</li> <li>• Required</li> </ul>

Attribute Name	Description
	<p style="text-align: center;">Readonly</p> <ul style="list-style-type: none"> <li>• Password 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, Maximo 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>
<p><b>Data Change Event</b></p>	<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>
<p><b>Move to Data Source ID</b></p>	<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, Maximo 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</p>

Attribute Name	Description
	ID object.

## 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 in Maximo 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 dialog box, 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 application's presentation file. 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 Maximo 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 dialog box.

### Rules

**This is not a container control.**

**You can place this control inside the following container control:**

- Section

## Example

1. Asset Number Multipart Textbox in the Work Order tab for the Work Order Tracking application.

The screenshot shows a form with the following fields:

- Work Order: 1004
- Location: BR230
- Asset: 11230
- Parent WO: (empty)
- Classification: (empty)
- Description: (empty)

To the right of these fields is a multipart textbox containing a list of items:

- Generator Overhaul
- Boiler Room Emergency Generator
- Emergency Generator

An arrow points from the text "multiparttextbox" to the list of items.

2. Asset Number Multipart Textbox Properties dialog box in the Application Designer.

The screenshot shows the "Multipart Textbox Properties" dialog box with the following settings:

- Control ID: main\_grid3\_10
- Default Label: Asset
- Label: (empty)
- Attribute: ASSETNUM
- Menu Type: ASSETMAIN
- Lookup: ASSET
- Input Mode: (empty)
- Change Event: (empty)
- Turn Smart Fill Off?:
- GOTO Application: ASSET\_IPC
- Move To Data Source ID: (empty)
- Attribute for Part 2: ASSET.DESCRPTION
- Lookup for Part 2: (empty)
- Input Mode for Part 2: READONLY
- Long Description Input Mode: (empty)
- Data Source ID: MAINRECORD

## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Default Label</b>	This label defaults to the text value associated with the database attribute you bound to the textbox. This is a read only field.
<b>Label</b>	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.
<b>Attribute</b>	You can associate a textbox with a database attribute by using the Select Value dialog box. 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.
<b>Menu Type</b>	<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 ID's 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 Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_root&gt;\resources\presentation\system directory.</p>
<b>Lookup</b>	<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 ID's 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 Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_root&gt;\resources\presentation\system directory.</p>

Attribute Name	Description
<p><b>Input Mode</b></p>	<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, Maximo 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>
<p><b>Change Event</b></p>	<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>
<p><b>Turn Smart</b></p>	<p>By default, Maximo enables the “Smart Fill” feature. To</p>

Attribute Name	Description
<b>Fill Off?</b>	disable, clear the check box. This feature enables you to enter a partial value in a text box field. Maximo attempts to match the partial text string entered to a valid value(s) in the database. If Maximo finds one matching value, it automatically populates the field so that you can continue entering information. If Maximo finds more than one matching value, it displays a Select Value dialog box with a list of matching values. If you clear the smart fill feature, Maximo does not validate the data you enter in the textbox.
<b>GOTO Application</b>	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 dialog box 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.
<b>Move to Data Source ID</b>	<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, Maximo 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>
<b>Attribute for Part 2</b>	You can associate the second part of a Multipart Textbox with a database attribute by using the Select Value dialog box. 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.
<b>Lookup for Part 2</b>	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.
<b>Input Mode for Part 2</b>	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.
<b>Long Description Input Mode</b>	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.
<b>Data Source ID</b>	The Data Source ID for this control. It points to the main MBO or object that the control uses for entering and

Attribute Name	Description
	<p>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(s) controls, automatically move with it.

A Parameter Value control works with the following:

- Parameter Values
- Section Header

A Parameter Value is transparent in the Maximo user interface. Section headers along with the Parameter Value and 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:

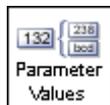
1. Insert Section Header into a Section or Table control
2. Drag Parameter Values into the Section Header container
3. Parameter Value automatically appears with the Parameter Values control

**Example (See Section Header Example)**

**Properties**

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Parameter Position</b>	This property points to the Parameter Values Property field you want to replace. See the Parameter Values Property field for more information.
<b>Attribute</b>	<p>Select a data attribute from a data source to display as the Label property in the Table or Section Header.</p> <p><b>Note:</b>                      You can use the Select Value dialog box to select an attribute from the MAINRECORD or any other table in Maximo. If you enter “=” before the Object name, the dialog box 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>
<b>Data Source ID</b>	<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(s) controls, automatically move with it.

A Parameter Values control works with the following:

- Parameter Value
- Section Header

Parameter Values are transparent in the Maximo user interface. Section Headers along with the Parameter Value and 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 Section Header into a Section or Table control
2. Drag Parameter Values into the Section Header container
3. Parameter Value automatically appears with the Parameter Values control

## Example (See Section Header Example)

### Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Property</b>	<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</p>

Attribute Name	Description
	<p>“Children of Work Order (0)” heading with work order numbers.</p> <pre data-bbox="730 317 1453 932"> &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.SelectW O RecordsBean"&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> <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>

Attribute Name	Description
	Dataattribute = wonum  Defines the data value (e.g., work order number) to display in the Table Header

## Pushbutton



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

### Rules

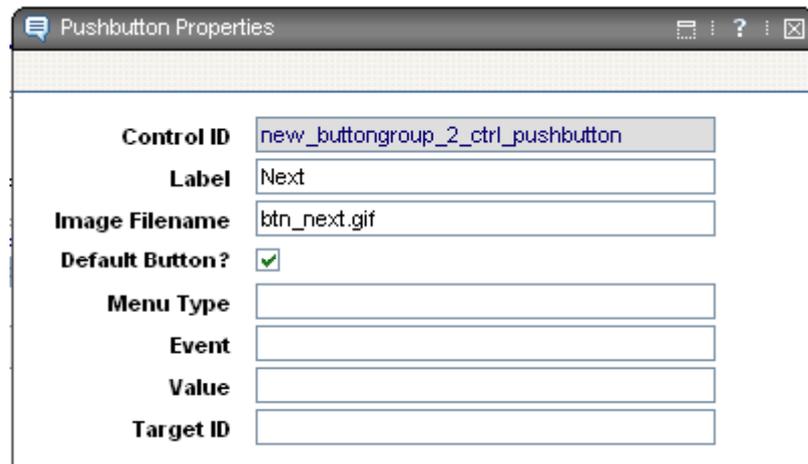
This is not a container control.

You can place this control inside the following container control:

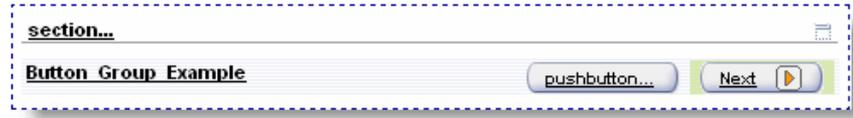
- Button Group

### Example

1. Pushbutton Properties dialog box configured in the Application Designer.



2. Pushbutton as it appears in the Button Group in the Application Designer workspace.



Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	<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>
<b>Image File name</b>	<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 Maximo that reside in the following directory:</p> <p>c:\maximo\applications\maximo\maximouiweb\webmodule\webclient\images</p>
<b>Default Button?</b>	Select this field to make a button the active button in a Button Group. Maximo highlights the label text of the active button in the user interface. You may want to make the most frequently used button (e.g., New Row) the default button.
<b>Menu Type</b>	<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 ID's 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 Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_root&gt;\resources\presentation\system directory.</p>
<b>Event</b>	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.

Attribute Name	Description
	<p>To view all of the events defined in Maximo, you must export the system .xml files by selecting a main tab from the Application Designer and choosing Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_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 application’s JSP main page 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 dialog box 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 “Select Assets.” When you press this button, a Select Assets dialog box opens that displays a list of asset records. The dialog displays because Maximo processes the “selectallssets” event specified for this button using the logic described above.</p>
<b>Value</b>	Reserved for future use.
<b>Target ID</b>	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. When you select a Radio Button, the previously selected button in the same group is deselected.

## Rules

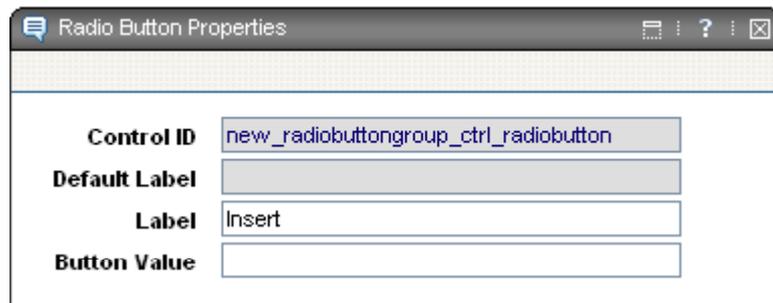
This is not a container control.

You can place this control inside the following container control:

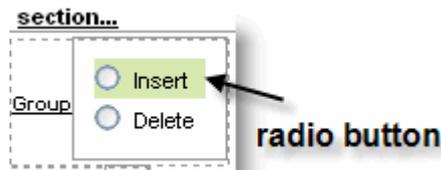
- Radio Button Group

## Example

1. Radio Button Properties dialog box configured in the Application Designer.



2. Radio Button as it appears in the Application Designer workspace.



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Default Label</b>	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.
<b>Label</b>	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.

Attribute Name	Description
	The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
<b>Button Value</b>	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. 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. When you select a Radio Button, the previously selected button in the same group is deselected. Radio buttons typically appear as little circles that are either empty (for unselected) or contain a dot (selected).

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. Maximo binds most Radio Button Groups to an attribute.

### Rules

**You can place the following control in this container control:**

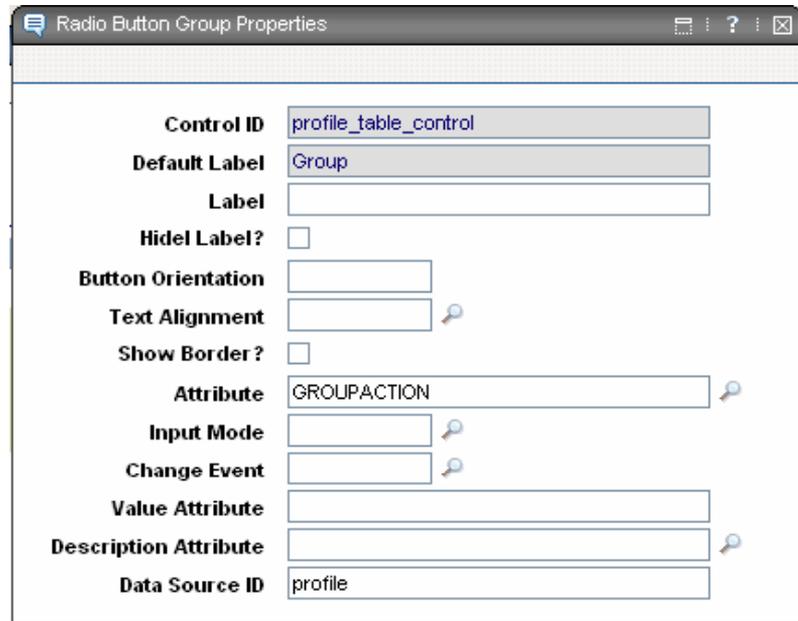
- Radio Button

**You can place this control inside the following container control:**

- Section

## Example

1. Radio Button Group Properties dialog box for the Set Security Profile dialog. To display, select Go To > Configuration > Application Designer and open the Users application. Choose Edit Dialogs in the toolbar, then select Select Security Profile from the list. Scroll down to the Update Groups section and select Group. This control binds to the GROUPACTION attribute.



2. Radio Button Group as it appears in the Application Designer workspace. This group inherits its buttons and layout from the GROUPACTION attribute.



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Default Label</b>	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.
<b>Label</b>	<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>You can enter a text value in this field that overrides the Default Label value.</p> <p>The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.</p>
<b>Hide Label?</b>	Select this field to hide the Label or Default Label value in the user interface.
<b>Button Orientation</b>	Enter a value of Vertical or Horizontal to align the radio buttons in a group either vertically or horizontally.
<b>Text Alignment</b>	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.
<b>Show Border?</b>	Select this field to draw a rectangular border around the Radio Button Group.
<b>Attribute</b>	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.
<b>Input Mode</b>	<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</li> <li>• Query</li> <li>• Readonly</li> <li>• Required</li> </ul>

Attribute Name	Description
	<p style="text-align: center;">Required</p> <p>The Default option sets the Input Mode for all controls to “edit.” If you do not select an Input Mode for a control, Maximo 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>
<b>Change Event</b>	<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>
<b>Value Attribute</b>	<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.” Maximo writes “WAPPR” to the database and displays “Waiting for Approval” in the user interface.</p>

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

## 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 Maximo applications have multiple pages with many sections and columns. See the Example for information on how to populate a Section control with multiple columns.

Sections are one of the primary components of the Maximo 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 Maximo 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
- Blank Line
- Button Group
- Checkbox
- Combobox
- Default Value
- Hyperlink
- Image
- Include
- Listbox
- Radio Button Group
- Section Header
- Section Row
- Static Text
- Textbox

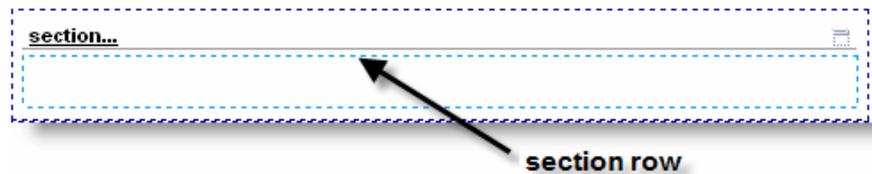
- Data Source
- Multiline Textbox

You can place this control inside the following container controls:

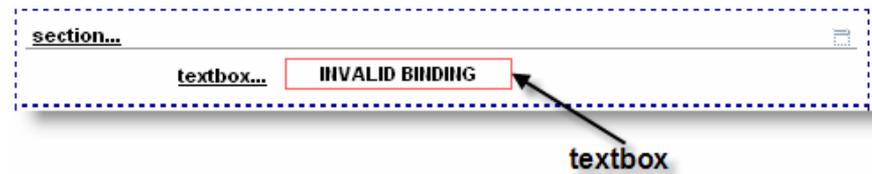
- Dialogs
- Tab
- Table
- Section Columns

### Example

1. Section container with Section Row.



2. Section container with a Textbox.



### Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	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.  The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
<b>Show Border?</b>	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.
<b>Collapsed?</b>	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

Attribute Name	Description
	<p>Application Designer workspace.</p> <p>To expand a section in the user interface, click the Maximize Section icon in the application.</p>
<b>Description</b>	<p>Text you enter in this field displays under the Section's label in the title bar.</p>
<b>Data Source ID</b>	<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 Maximo 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 Maximo Business Object (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> presentation id="wotrack" resultstableid="results_showlist" mboname="WORKORDER" orderby="WONUM" beanclass="psdi.webclient.beans.workorder.WorkorderAppBean" version="6.0.0"&gt;  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 for additional information on data sources.</p>
<b>Business Object</b>	<p>The main MBO reference for the application. For example, Work Order Tracking uses WOTRACK and Assets uses ASSET.</p> <p>Every Maximo 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 application's presentation tag defines the main MBO for the application. MBOs are responsible for the following types of functions:</p>

Attribute Name	Description
	<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>
<b>Parent Data Source ID</b>	<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>
<b>Input Mode</b>	<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, Maximo 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>

Attribute Name	Description
	<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>
<b>Listeners</b>	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.

## 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 Maximo user interface. Except for tables, all graphical elements in Maximo 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

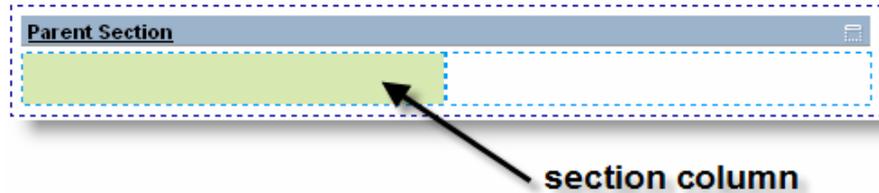
**You can place this control inside the following container control:**

- Section Row

## Example

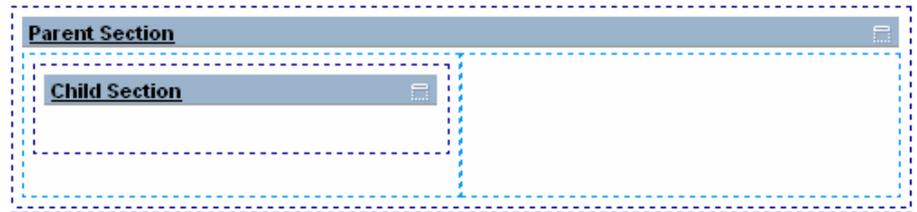
To create a multi-column section layout, complete the following steps:

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



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

1. Insert Section
2. Insert Section Row
3. Insert Section Columns (to subdivide the section into columns)
4. Drop a new section into each Section Column
5. Add controls to each new Section



**Note** To view the page tag in 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
<b>Control ID</b>	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 Maximo 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 “xxxx” 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:**

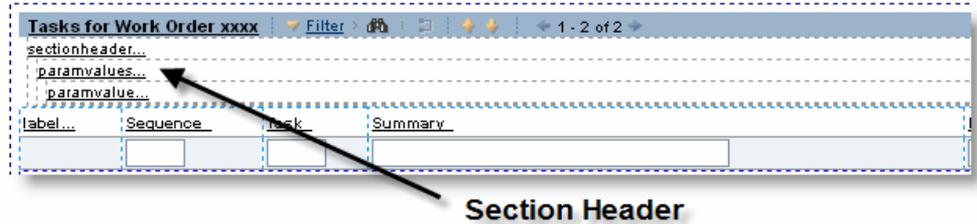
- Section
- Table

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

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

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

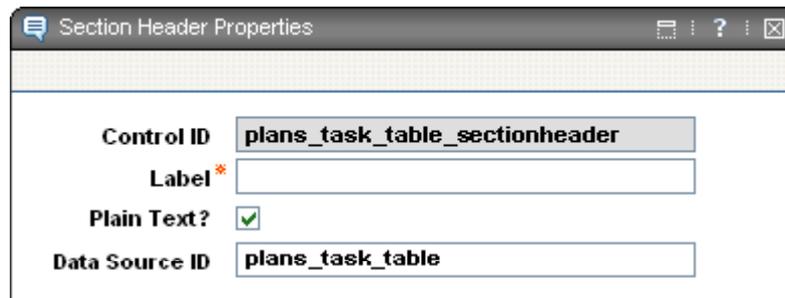
**Work Order Tracking (WOTRACK) Section Header opened in Application Designer:**



**Work Order Tracking Section Header as it appears in Maximo:**

Sequence	Task	Summary
<input type="text"/>	10	Relocate guard rails to allow fork truck access
	20	Relocate associated electrical conduit

**Work Order Tracking Section Header dialog box as it appears in Application Designer:**



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	The label contains a variable that displays dynamic values inherited from the Parameter Value and Values controls in the Table or Section control.
<b>Plain Text</b>	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.
<b>Data Source ID</b>	<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 Maximo 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 Maximo Business Object (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. Sections Rows are transparent in the Maximo user interface. Except for tables, all graphical elements in Maximo that have multiple columns have a transparent Section Row within a Section.

## Rules

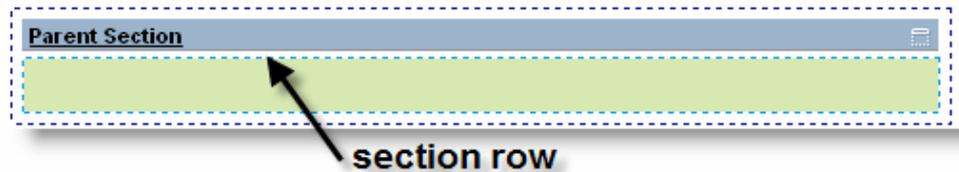
You can place the following control in this container control:

- Section Column

You can place this control inside the following container control:

- Section

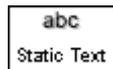
## Example



## 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 dialog boxes. The static text functions as a To or From label in a Date Search dialog box like the SEARCHMORE dialog 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

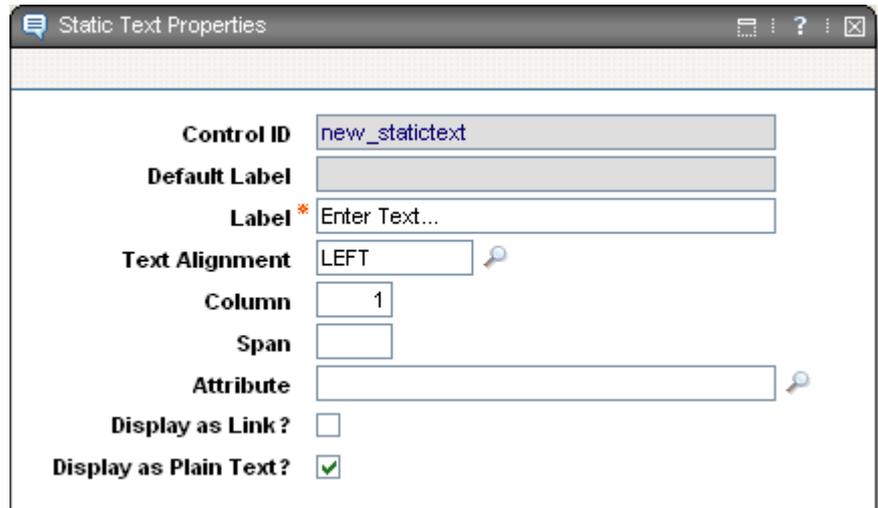
This is not a container control.

You can place this control inside the following container control:

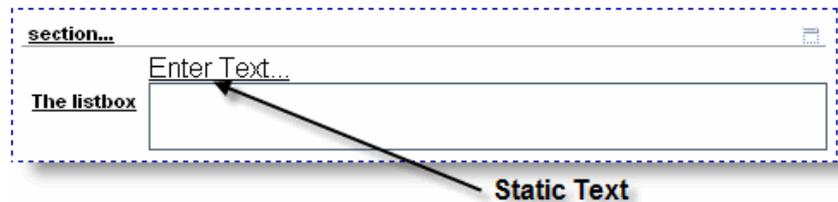
- Section

## Example

1. Static Text Properties dialog box configured in the Application Designer.



2. Static Text example as it appears in the Application Designer workspace.



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Default Label</b>	This label defaults to the text value associated with the database attribute you bound to the Listbox. This is a read only field.
<b>Label</b>	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.
<b>Text Alignment</b>	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.
<b>Column</b>	<p>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 Maximo centers the Static Text above or below column three.</p> <p>If the Static Text you want to display is wider than its column, you cannot use the Text Alignment property and Maximo automatically extends the column width to contain the Static Text.</p>
<b>Span</b>	Enter a numeric value in this field that will enable the Static Text to bridge one or more columns.
<b>Attribute</b>	You can associate static text with a database attribute by using the Select Value dialog box. If the attribute you select has a default label value, it will display in the Default Label field.
<b>Display as Link?</b>	By default, Static Text does not display as a URL link. Select this field to configure the control as a URL link.
<b>Display as Plain Text?</b>	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



A Tab is a container control. Formatting a Tab(s) 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 screen 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:**

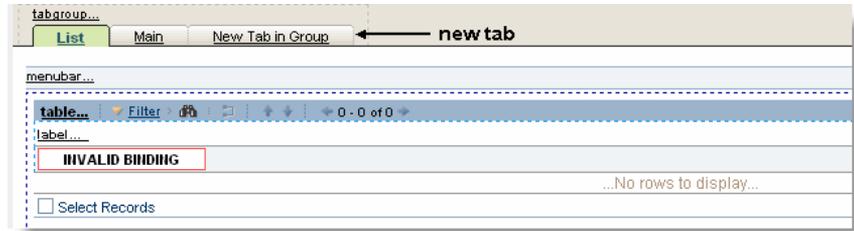
- Tabgroup

## Example

1. Tab Properties dialog box configured in the Application Designer.

<b>Control ID</b>	new_tab_2
<b>Label</b>	New Tab in Group
<b>Type</b>	INSERT
<b>Default?</b>	<input type="checkbox"/>
<b>Bean Class</b>	
<b>Business Object</b>	
<b>Parent Data Source ID</b>	
<b>Relationship</b>	
<b>Order By</b>	
<b>Where Clause</b>	
<b>Listeners</b>	

2. New Tab as it appears in the Application Designer workspace.



Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	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.
<b>Type</b>	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 Maximo search tab with a filter table. Choosing Insert formats the tab as an empty workspace where you can build an application screen by inserting and configuring controls.
<b>Default?</b>	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.
<b>Bean Class</b>	Reserved for future use.
<b>Business Object</b>	The main MBO reference for the application. For example, Work Order Tracking uses WOTRACK and Assets uses ASSET.  Every Maximo 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 application's presentation tag defines the main MBO for the application. MBOs are responsible for the following types of functions:

Attribute Name	Description
	<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>
<b>Parent Data Source ID</b>	<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>
<b>Relationship</b>	<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 Maximo business objects.</p>
<b>Order By</b>	<p>Reserved for future use.</p>
<b>Where Clause</b>	<p>Reserved for future use.</p>
<b>Listeners</b>	<p>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.</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 Maximo List and Main (workspace) tabs. You can add additional tabs as required by your business application.

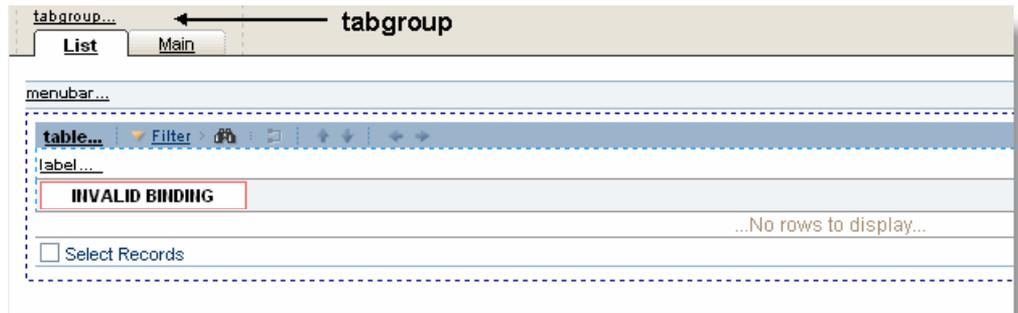
## Rules

You can place the following control in this container control:

- Tab

## Example

The Tabgroup control as it appears in the Application Designer when you create a new power application.



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Format</b>	<p>Leave blank (the default) or choose one of the following layout options:</p> <ul style="list-style-type: none"> <li>• CARDECK</li> <li>• DEFAULT</li> <li>• WIZARD</li> </ul> <p>The wizard option replaces the standard Maximo tab layout with arrows that point to the tab headings. The cardeck option appears in single-page applications that do not employ tabs.</p>

Attribute Name	Description
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/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 Maximo applications. Maximo 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

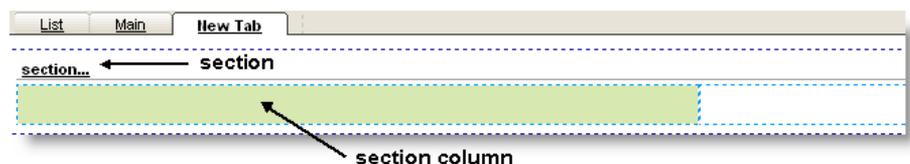
You can place this control inside the following container controls:

- Section Column
- Tab
- Dialog Box

**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

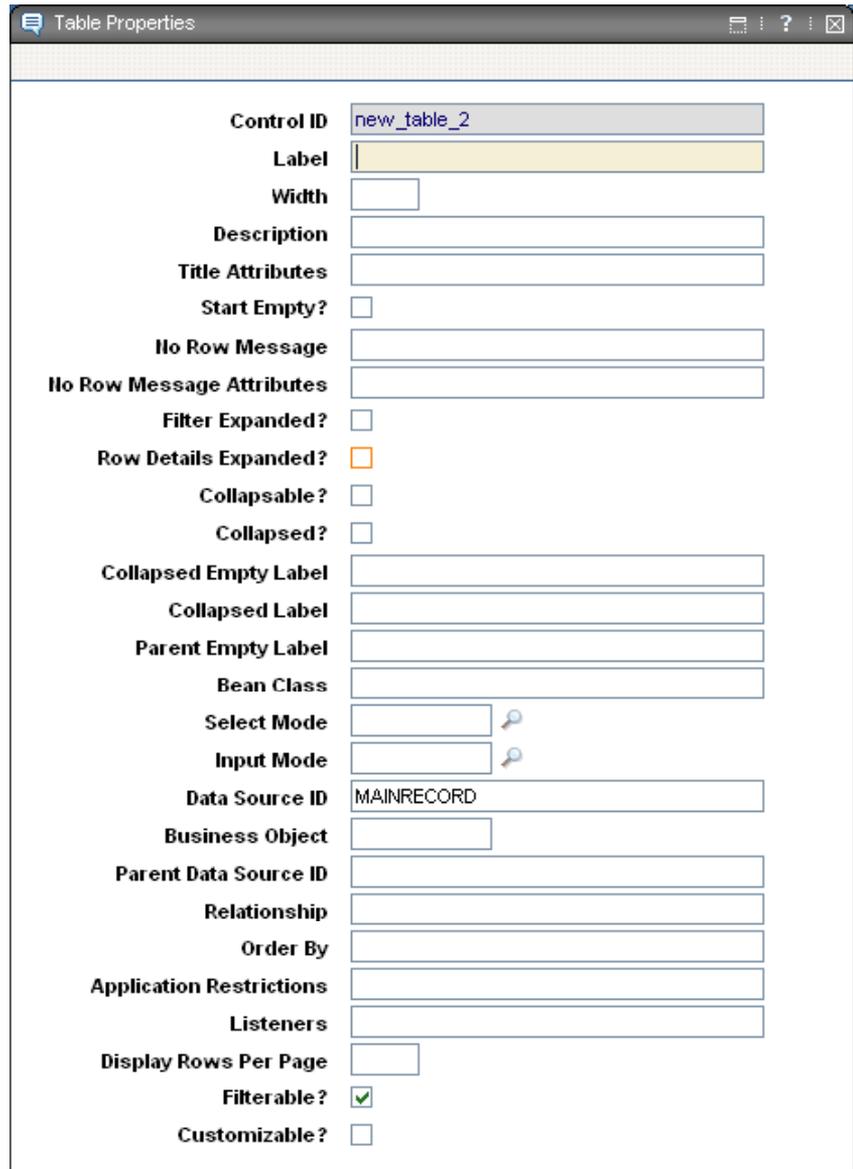
1. Section control dropped into a Tab container. Section Row dropped into the Section container. Section Column dropped into the Section Row.



2. Table control dropped into the Section Column.



3. Table Properties dialog box in the Application Designer.



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Label</b>	<p>The text value you enter becomes the title of the table and displays in the toolbar. Maximo stores this value in the MAXLABELS table. You can edit this value.</p> <p>The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.</p>
<b>Width</b>	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.
<b>Description</b>	Text you enter in this field displays under the table's label in the title bar.
<b>Title Attributes</b>	Reserved for future use.
<b>Start Empty?</b>	Select this box if you want the table to always open without displaying records. If you leave this box cleared (the default), Maximo automatically populates the table with records that match the search query when opened.
<b>No Row Messages</b>	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.
<b>No Row Message Attributes</b>	Reserved for future use.
<b>Filter Expanded?</b>	Select this box if you want to display an empty query filter row when you open a table. If cleared, the default, the filter row does not display with an open table. See also Filterable?.
<b>Row Details Expanded?</b>	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.
<b>Collapsible?</b>	Select this box to enable the Show/Hide icons in the table toolbar. You must select this field to

Attribute Name	Description
	use the “Collapsed?” property. By default, this property is cleared.
<b>Collapsed?</b>	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.
<b>Collapsed Empty Label</b>	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.
<b>Collapsed Label</b>	When a table is collapsed, the value you enter here will override the table title in the user interface.
<b>Parent Empty Label</b>	If this table is a child of a parent table, and that table is empty, the value you enter here will override the parent’s table title.
<b>Bean Class</b>	Reserved for future use.
<b>Select Mode</b>	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.”
<b>Input Mode</b>	<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 Table control, the input mode defaults to read/write so that you can both display or add new records to the table.</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</p>

Attribute Name	Description
	<p>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>
<b>Data Source ID</b>	<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>
<b>Business Object</b>	<p>The main MBO reference for the application. For example, Work Order Tracking uses WOTRACK and Assets uses ASSET.</p> <p>Every Maximo 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 application's presentation tag 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</p>

Attribute Name	Description
	different MBO, you can enter a different Parent Data Source ID.
<b>Parent Data Source ID</b>	The Parent Data Source ID points to the main MBO associated with an application, for example, WOTRACK. Unless overridden, all controls inherit the value stored in the application's presentation tag.
<b>Relationship</b>	<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 Maximo 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>
<b>Order By</b>	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.
<b>Application Restrictions</b>	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.
<b>Listeners</b>	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.
<b>Display Rows Per Page</b>	The maximum number of rows you wish to display on a page.
<b>Filterable?</b>	This field adds a filter row for searching to the table. It defaults to enable. This setting automatically passes to the Filterable? property

Attribute Name	Description
	in all Table Column controls. You must select this box to use the Filter Expanded? property.  See also Filter Expanded?
Customizable?	Reserved for future use.

## Table Column



The Table Column control is a container type control that lets you define additional column headings for a table. Maximo 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:

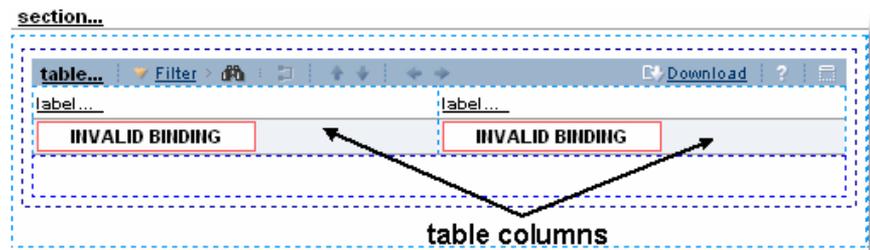
- Checkbox
- Textbox

You can place this control inside the following container controls:

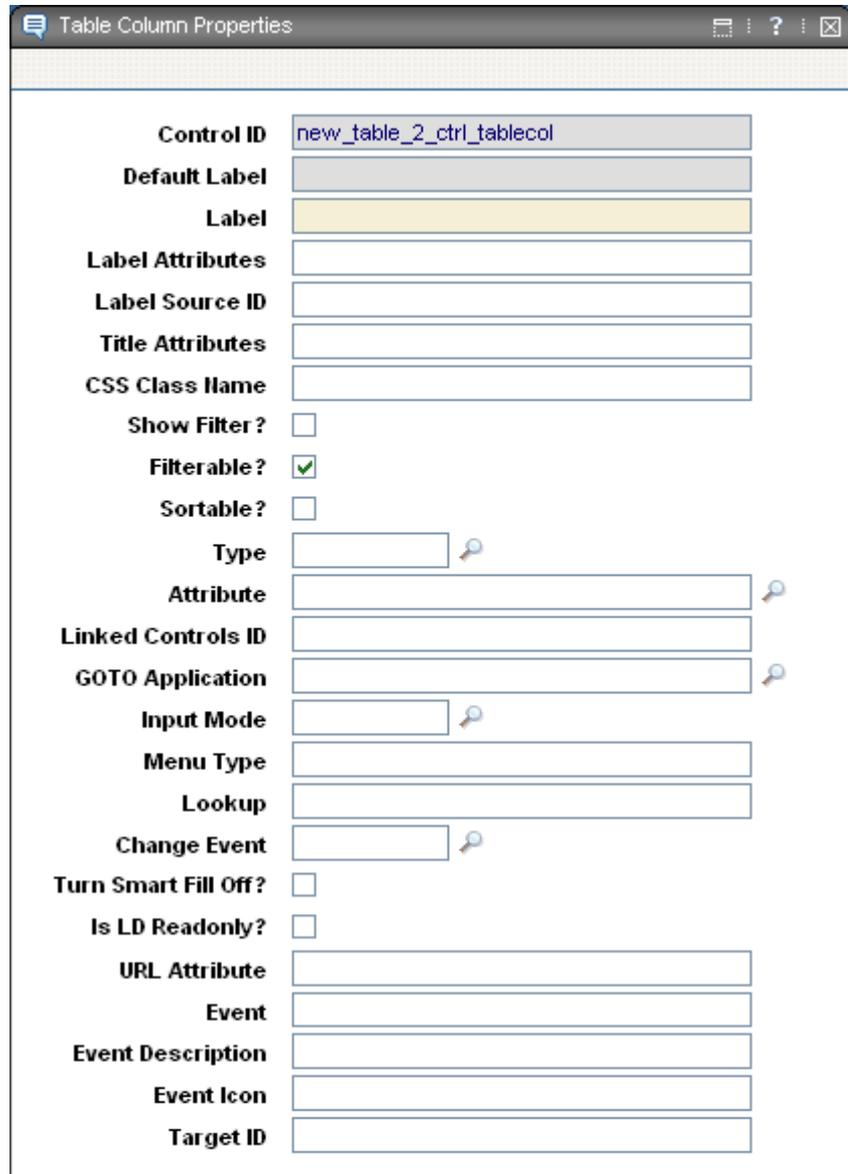
- Table

### Example

1. Table Column dropped into a table.



2. Table Column Properties dialog box in the Application Designer.



**Properties**

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Default Label</b>	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

Attribute Name	Description
	becomes Asset. This is a read only field.
<b>Label</b>	<p>The text value you enter here replaces the Default Label as the Table Column heading. Maximo stores this value in the MAXLABELS table. You can edit this value.</p> <p>The label you enter here is specific to the application and not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.</p>
<b>Label Attributes</b>	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.
<b>Label Source ID</b>	Enter an object name that is the data source for the Label Attributes property.
<b>Title Attributes</b>	Reserved for future use.
<b>CSS Class Name</b>	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.
<b>Show Filter?</b>	If you select the Filterable? and Show Filter? properties, you will see the filter Textbox control for this column in the table's filter row.
<b>Filterable?</b>	<p>If you select the Table control's Filterable? property, it automatically passes this setting to the Filterable? property in each Table Column. You can select or clear Filterable? at the table column level.</p> <p>Filterable?, when selected, indicates that you can search on the attribute associated with the Table Column label.</p>
<b>Sortable?</b>	If selected, you can sort the records in the table by clicking on the column's label. The first time you click on a label it sorts the records in ascending order for that column's values. The second time you click a label, it sorts the records in descending order.
<b>Type</b>	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

Attribute Name	Description
	<p>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 “toggledetailstate” with a description of “Show Detail.” This column places a View/Show Details 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>
<b>Attribute</b>	<p>You can associate a Table Column with a database attribute by using the Select Value dialog box. For example, if you enter ASSETNUM in the Attribute field, Maximo 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>
<b>Linked Controls ID</b>	<p>You can enter a Control ID here that gets automatically updated whenever there is a change to this column’s attribute value. The Control ID must reside somewhere within the application.</p>
<b>GOTO Application</b>	<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 dialog box for this property to choose application names that will appear in the pull-down menu.</p>
<b>Input Mode</b>	<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, Maximo automatically sets</p>

Attribute Name	Description
	<p>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>
<b>Menu Type</b>	<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 ID’s 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 Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_root&gt;\resources\presentation\ system directory.</p>
<b>Lookup</b>	<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</p>

Attribute Name	Description
	<p>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 ID's 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 Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_root&gt;\resources\presentation\system directory.</p>
<b>Change Event</b>	<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>
<b>Turn Smart Fill Off?</b>	<p>By default, Maximo enables the "Smart Fill" feature. To disable, clear the check box. This feature enables you to enter a partial value in a text box field. Maximo attempts to match the partial text string entered to a valid value(s) in the database. If Maximo finds one matching value, it automatically populates the field so that you can continue entering information. If Maximo finds more than one matching value, it displays a Select Value dialog box with a list of matching values. If you clear the smart fill property, Maximo does not validate the data you enter in the textbox.</p>
<b>Is LD Readonly?</b>	<p>By default, Maximo sets a Long Description associated with a Table Column attribute to editable. Select this field to make the Long Description read only.</p>
<b>URL Attribute</b>	<p>Enter a URL value if you set the Type property for this control to "Link." The URL value becomes a hyperlink.</p>

Attribute Name	Description
<p><b>Event</b></p>	<p>If you select “Event” as the Type property for a Table Column, Maximo performs the action defined in this field. Continuing the example for the Type property above, entering “togglestate” 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 Maximo, 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. Maximo is an event driven product. When you perform an action, Maximo 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 Type.</p>
<p><b>Event Description</b></p>	<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 Event.</p>
<p><b>Event Icon</b></p>	<p>Point to an image (.gif file) that represents an event. The image icons you see in the Maximo user interface reside in the following directory:</p> <p>c:\maximo\applications\maximo\maximouiweb\webmodule\webclient\images</p> <p>You can access the graphic files in the images directory using the Image control in the Application Designer.</p>
<p><b>Target ID</b></p>	<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 dialog box, 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 application's presentation file. 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 dialog box.

## Rules

**This is not a container control.**

**You can place this control inside the following container control:**

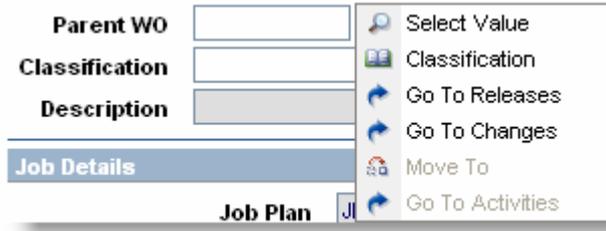
- Section

## Example

1. Parent WO textbox field in the Work Order Tracking application.



2. Detailed menu for the Parent WO field.



3. Select value table row for the Parent WO field.

Work Order	Description	Class	Location	Asset	Status	Is Task?
2003	Scale Calibration on Dock Mis-reading	WORKORDER	SHIPPING	WAPPR		false

Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Default Label</b>	This label defaults to the text value associated with the database attribute you bound to the textbox. This is a read only field.
<b>Label</b>	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.
<b>Attribute</b>	You can associate a textbox with a database attribute by using the Select Value dialog box. If the attribute you select has a default label value, it will display in the Default Label field.
<b>Menu Type</b>	<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 ID's 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</p>

Attribute Name	Description
	<p>Designer and choosing Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_root&gt;\resources\presentation\system directory.</p>
<p><b>Lookup</b></p>	<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 ID's 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 Select Action &gt; Export System XML. This action exports the library.xml, lookups.xml, and menus.xml files to the &lt;maximo_root&gt;\resources\presentation\system directory.</p>
<p><b>Detail Image</b></p>	<p>Reserved for future use.</p>
<p><b>Input Mode</b></p>	<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, Maximo 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</p>

Attribute Name	Description
	<p>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>
<b>QBE Prepend</b>	<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 dialog 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>
<b>Change Event</b>	<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>
<b>Turn Smart Fill Off?</b>	<p>By default, Maximo enables the “Smart Fill” feature. To disable, clear the check box. This feature enables you to enter a partial value in a text box field. Maximo attempts to match the partial text string entered to a valid value(s) in the database. If Maximo finds one matching value, it automatically populates the field so that you can continue entering information. If Maximo finds more than one matching value, it displays a Select Value dialog box with a list of</p>

Attribute Name	Description
	matching values. If you clear the smart fill feature, Maximo does not validate the data you enter in the textbox.
<b>Readonly Long Description?</b>	By default, Maximo sets a Long Description associated with a textbox attribute to editable. Select this field to make the Long Description read only.
<b>GOTO Application</b>	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 dialog box for this property to choose application names that will appear in the pull-down menu.
<b>Display Type</b>	Reserved for future use.
<b>Move to Data Source ID</b>	<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, Maximo 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>
<b>Data Source ID</b>	<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>

## 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 Maximo 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 Maximo employs the Tree controls.

## Rules

You can place the following controls in this container control:

- Tree Node

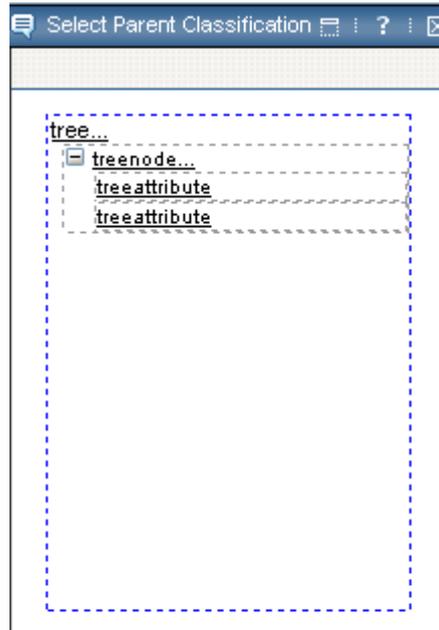
You can place this control inside the following container controls:

- Dialogs
- Tab

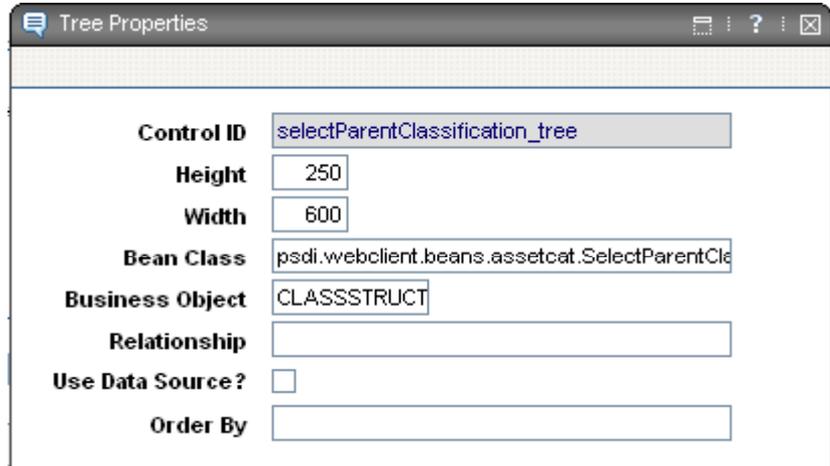
## Example

The following Select Parent Classification tree example illustrates how the combination of Tree, Tree Node and Tree Properties controls work together to graphically display an existing hierarchical data structure in Maximo.

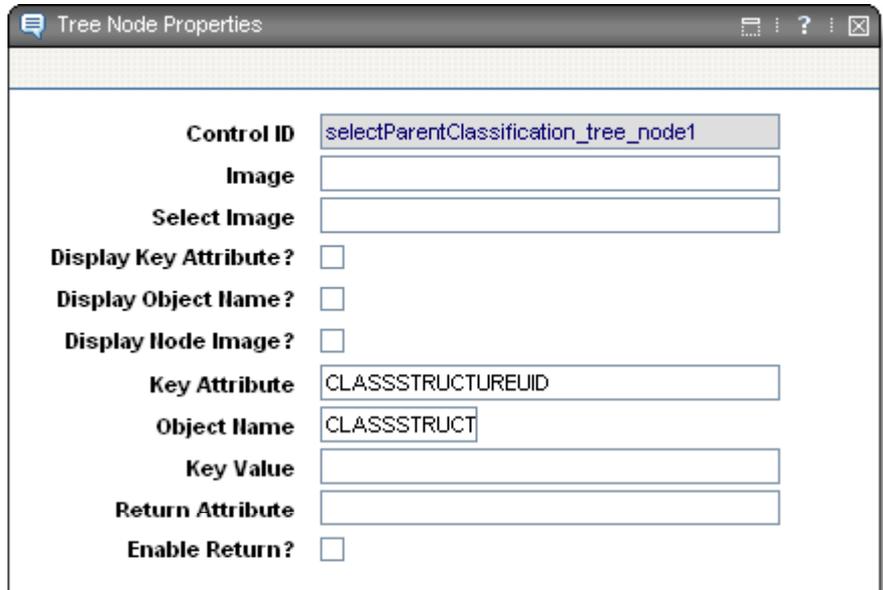
1. **The Select Parent Classification tree structure in the Application Designer workspace. To view this dialog box select Go To > Configuration > Application Designer > ASSETCAT > Select Parent Classification dialog.**



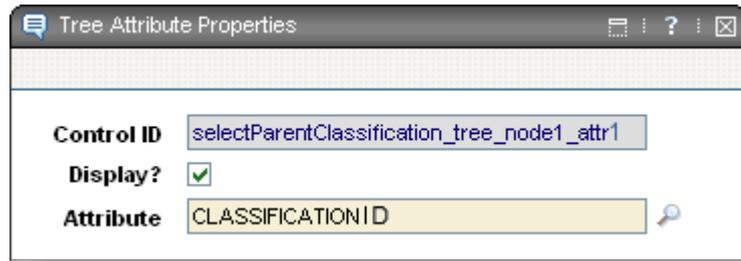
2. Tree Properties dialog box for Select Parent Classification in the Application Designer workspace.



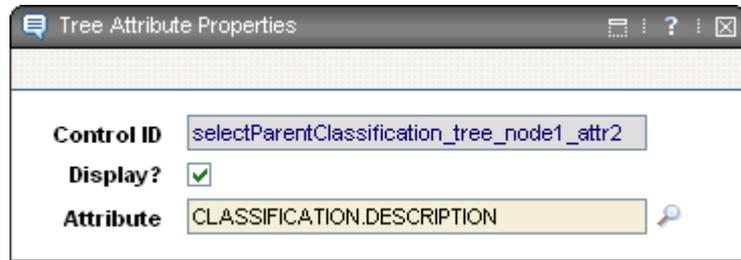
3. Tree Node Properties dialog box for Select Parent Classification in the Application Designer workspace.



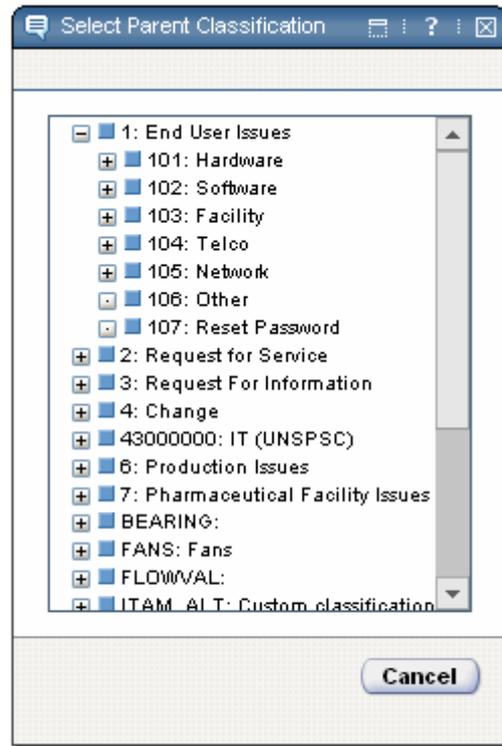
4. **Tree Attribute Properties dialog box for Select Parent Classification with CLASSIFICATION attribute in the Application Designer workspace.**



5. **Tree Attribute Properties dialog box for Select Parent Classification with CLASSIFICATION.DESRIPTION attribute in the Application Designer workspace.**



6. **The Administration > Classifications > Parent Classification > Select Parent Classification dialog box with data structure in the Maximo user interface.**



## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Height</b>	Enter a numeric value in this field that will increase or decrease the height of this control.
<b>Width</b>	Enter a numeric value to increase or decrease the width of this control.
<b>Bean Class</b>	Reserved for future use.
<b>Business Object</b>	<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 Maximo 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 application's presentation tag 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>
<b>Relationship</b>	You can enter a relationship value in this field. Maximo 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 Maximo business objects.
<b>Use Data Source?</b>	Reserved for future use.
<b>Order By</b>	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
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Display?</b>	Select this box to display the defined Attribute value in the Tree Node.
<b>Attribute</b>	You can associate what values display in the tree node structure with a database attribute by using the Select Value dialog box. 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

See the preceding Tree control example.

## Properties

Attribute Name	Description
<b>Control ID</b>	A unique, alphanumeric identifier assigned to each control in a presentation. You cannot edit this value.
<b>Image</b>	<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 Maximo library of images. However, if you insert a graphic file without a .gif extension into Maximo, then you need to rebuild/deploy the Maximo .EAR file.</p>
<b>Select Image</b>	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.
<b>Display Key Attribute?</b>	Select this property to display the Key Attribute as part of the Tree Node label.

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