IBM Maximo Asset Management
Version 7 Release 6

Developing Applications

Note

Before using this information and the product it supports, read the information in “Notices” on page 115.

This edition applies to version 7, release 6, modification 0, fix pack 7 of IBM® Maximo Asset Management and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Application development

You can develop new applications and modify existing applications using the Application Designer application. You can also use the Application Designer to export XML files which you can then modify using a text editor before importing them with the Application Designer.

Application framework

The application framework includes user interface components for developing applications and organizational components that provide access to them.

The user interface consists of controls, such as text boxes, buttons, and tabs, and each control has attributes that determine its behavior. Each application has apresentation.xml file that contains all of the information required to build the application user interface. When you create an application, it is automatically added to the navigation structure for modules and applications.

User interface controls

Controls are predefined components that are used to construct the elements of an application window. In Application Designer, you can select individual controls, view and modify the control properties, or drag new controls into the application from the **Control Palette**. You cannot change the code of a control, but you can modify its behavior by specifying appropriate values in the properties window for the control. For example, you can set a control to be a required field, associate a control with a lookup table, or link a control to an event.

Some controls, such as sections, tab groups, or button groups are container controls. Most container controls serve no purpose other than to contain other controls. Controls that are not containers must be placed within a container control. In Application Designer, you can drag controls from the **Control Palette** onto the application workspace, and then modify the control in the Control Properties window.

The positioning of controls is relative to other controls and is not based on pixels or a grid and provides for dynamic sizing of the width of a section and its columns. For example, if a section contains fields of various widths, the width of each field is automatically set to the width of the widest field in that section.

Application XML files

Every application has a presentation.xml file that contains all the information required to build the application user interface. All presentation.xml files are stored in the MAXPRESENTATION table in the database. Controls are predefined components that are used to construct the elements of an application window. In Application Designer, you can select individual controls, view and modify the control properties, or drag new controls into the application from the **Control Palette**. You cannot change the code of a control, but you can modify its behavior by specifying appropriate values in the properties window for the control. For example, you can set a control to be a required field, associate a control with a lookup table, or link a control to an event.

The presentation.xml file for an application includes a tag for each control used in the application user interface. Each control has a unique ID and attributes that determine how the control behaves when you view the application. The relative location of control tags in the presentation.xml file determine the order and sequence of the user interface elements that you see in an application window.

When a user starts an application, the presentation.xml file is retrieved from the database and is placed in the memory of the application server. The application framework retrieves the code for each control and creates a HTML description of it, based on the attributes specified in the presentation.xml. The rendering process is incremental, and the application server stores HTML descriptions for each control in memory until all controls in an application window are rendered. When all HTML elements are rendered, the application server passes the HTML to the user's web browser (client). The presentation.xml file is retained in the server cache, ready for the next time a user accesses the application.

When you open an application in the Application Designer, the presentation.xml file is loaded into memory. Any changes you make apply only to the stored version of the file until you save the application. When you save the presentation.xml, the modified information is committed to the database and you can then see the changes when you open the application.

Although you can perform most modifications to an application in Application Designer, you can edit the presentation.xml file if required. Sometimes, modifying the presentation.xml file is the most efficient approach to use, for example to change a specific term using search and replace. In an application with multiple tabs and dialog windows, this task requires that you open each individual window in Application Designer, which is both time-consuming and prone to error. Some applications use hidden tabs that are not visible in Application Designer. You cannot modify these tabs in Application Designer and must edit the presentation.xml file to change them.

Application organization

All applications are contained within modules. Applications are categorized into three different types although each type uses the same structure and components. When you create an application, you must specify its module and the type of application to create. The available application types are:

- A power application is the standard type of application that contains multiple tabs, a **Select Action** menu and toolbar buttons.
- A self-service application is used to create a record and does not include a List Table window or a toolbar.
- A single-page application is a non-tabbed application but it can include a **Select Action** menu and toolbar buttons.

After you create an application, it is added to **Go To** menu within the module that you specify when you create it.

Preparing the development environment

Before using the Application Designer, set up the development environment and familiarize yourself with the workspace and with the design guidelines.

Setting up the environment

Plan and prepare the development environment before you start working with Application Designer.

Procedure

1. Establish separate environments for the following activities to protect your production environment against configuration changes you make.
   a. Perform development activities on a dedicated workstation.
   b. Merge your configurations to a server for testing.
   c. Migrate the tested configurations to the production environment.
2. Set the screen resolution of your monitor to 1280x768 to optimize the amount of information that you can see without scrolling. With this screen resolution, it is easier to position the Control Palette and Properties windows outside of the application workspace where you perform configurations.

Exploring the workspace

When you open an application in the Application Designer, the Workspace tab opens, containing the layout of the controls that make up the application. Within the workspace, you can select individual controls to modify them and to position new controls in the application user interface.

About this task

The workspace opens the main tab of an application but some tabs, windows, controls, and configuration tools are not immediately visible. Before you start developing applications with the Application Designer, use the following procedures to familiarize yourself with the workspace.

Procedure

1. Open an application that you are familiar with in the Application Designer. The application opens in the workspace, typically on the application List tab.
2. Review the contents of the workspace when it opens, clicking on controls to select them.
3. In the **Select Action** menu, click **Toggle/Show All Controls**. You can use this action repeatedly to show or hide controls that are less frequently modified to conserve space on the workspace.
4. Select any control in the workspace and either right-click and select **Properties** or click the **Control Properties** icon on the toolbar. The properties window for the control opens where you can review the properties that are configured for the control.
5. In the Workspace, select a **Section** control and click the **Control Palette** icon on the toolbar.
6. Drag a control icon from the palette into the selected **Section** control. If the control cannot be placed within a **Section** control, it is not added to it. For example, you must place a **Tab** control within a **Tab Group** control and you must place a **Button** control within a **Button Group** control.
7. Click the **Edit Dialogs** icon on the toolbar to see a list of the dialog windows that are configured for the application.
8. Select a dialog to open it for review or modification in a temporary workspace.

9. Click on the Applications tab to close the workspace and do not save the changes when you are prompted to do so. Changes you make in the workspace are stored temporarily until you click **Save**, when the changes are committed to the database.

Design guidelines

Use the Application Designer and, if necessary, the application XML files, with an appropriate design process to develop applications that are easy and efficient to use.

Review user goals

Before you start developing or modifying an application, review the user goals for the application. Divide users into groups, based on the tasks that they perform. Instead of creating one large application to serve all users, consider creating smaller applications that serve specific needs. Give users only what they need, and nothing more.

Prioritize information

Consider the circumstances when users need access to secondary information and choose where to locate this information from the following options:

- On a separate tab.

  If you place secondary information on a separate tab, it does not load until the user clicks the tab, which makes start up performance faster. The information is available with one click when the user needs it.
- Below the screen area (fold line)

  Users can ignore information at the bottom of the screen and can scroll down to see it when it is necessary. Including the information on the lower screen can increase load-time, but vertical scrolling is easier, faster, and less error-prone than clicking.

When you design an application that requires a lot of data entry, arrange the input fields to optimize the use of the Tab key to navigate between fields. The tabbing order of a window goes from top to bottom and from left to right.

If you want to change the location of a number of data input fields in an application, consider duplicating the application and then changing the order of the input fields. When you are modifying an application, you can delete fields that are not needed for data entry, or move those fields to another screen.

Use appropriate controls

Controls are predefined components, such as text boxes, buttons, and tabs, that are used to build the application user interface. In the Application Designer, you can select a control from the Control Palette and drag it to the workspace to add it to the user interface. You then configure properties for the control, for example specifying an image file to use for a button control or setting a text control as a required field.

The following controls are the main layout controls:

- Presentation
- Page
- Dialog

The presentation control is the main control for an application, representing the `presentation.xml` file that defines all of the user interface components. Each application has just one page control, typically the main tab of the application. Use tab controls or dialog controls to create additional windows. These controls are the outermost containers for application components. Other container controls include:

- Section
- Tab group
- Table
- Push button group

Within a container control, you can place multiple controls such as placing multiple text boxes, combo boxes, and list boxes into a section, or placing multiple buttons into a button group. There are a number of benefits in grouping related content within the same container control. The grouping provides a logical order and convenience for the user and, if you move the container to another location, all of the controls that it contains move with it.

Organize navigation

The application framework includes a number of predefined navigation structures that you can use to provide quick and easy access to your application and application components. Use the following navigation structures:

- The **Go To** menu lists all of the product modules and their applications. When you create an application, you must specify its module. When you duplicate an application, the new application is automatically added to the same module as the application that you copied.
- You can add buttons and links to the application toolbar to provide quick access to application features.
- You can create an option in the Select Action menu for a common task. You can also restrict access to this option based on user group or conditions.

Data validation

The system validates data to ensure that users are working with the correct data. Data can be validated on the server-side or the client-side.

Server-side validation

When you enter information in an application, it is placed in the queue for server-side validation where the data is validated with data on the server. Server-side validation consists of asynchronous and synchronous validation. Asynchronous validation helps users to work faster because they can move from field to field, entering data without waiting for information to be validated.

Asynchronous validation is disabled automatically when a screen reader is being used.

In synchronous processing, values are validated when they are entered; asynchronous values remain in the queue until one of three conditions is met:

- When the threshold for the number of asynchronous values in the queue is reached. The default value is 2.
- When there is no action on the user interface, and the default timeout is reached. The default value is 10 seconds.
- When a user completes an action that triggers a synchronous validation.

When one or more of these conditions is met, all values in the queue are then sent to the server to be validated.

Asynchronous validation of user-entered data is enabled at the system level by default. You can disable asynchronous validation for the system, you can disable it for individual controls, and you can change the validation process for the system.

Table 1. Data validation levels

Data validation level	Where	How
Enable background validation at the system level	System Properties application	Disable or enable the **mxe.webclient.async** system property.
Configure background validation for individual controls	Application Designer application	Configure autofill properties: type-ahead, set values, and filters.
Modify the validation process system-wide	System Properties application	• Change the value for the **mxe.webclient.asyncrendertimelimit** system property. • Change the value for the **mxe.webclient.asyncerrortooltipwaitbeforeopen** system property. • Change the value for the **mxe.webclient.asyncrequestsbeforerender** system property.

Error notification during asynchronous validation

If an error occurs during asynchronous validation, an error indicator appears on the relevant field. When you click the error icon, a message opens. If there are errors on more than one tab, an error icon is also set at tab level for each tab that contains unresolved messages.

The threshold for the queue ensures that only asynchronous values are sent to the server for validation when the default value is passed.

Client-side validation

To improve performance, the system validates data on the client browser before validating on the server.

Autofill properties:

The autofill function populates fields based on the data that you enter in other fields. You can configure autofill properties so that fields are populated with cached data instead of server data to improve performance.

On a high latency network, the use of cached data can be more efficient. A high latency network is a network with a latency of more than 100 milliseconds.

You can configure autofill for a text box or a multipart text box. When you configure autofill properties, a data store is created in the user's browser. The data store on a browser is limited in size so configure autofill properties for the most important text boxes and multipart text boxes.

The autofill properties that you can configure are type-ahead, set values, and filters.

Type-ahead is an incremental search feature. As you type data in a field, one or more possible matches for the data is listed. You can configure type-ahead so that the list of possible matches includes one or more business objects. You must set the **mxe.webclient.ClientDataValidation** property to a value of *1* for type-ahead to work. You must configure type-ahead before you can configure set values and filters.

All fields that are associated with ALN, numeric, or synonym domains support type-ahead when you set the value of the **mxe.webclient.ClientDataValidation** property to *1*, but you can also configure autofill for additional fields. Simple client-side error validation is also enabled with the **mxe.webclient.ClientDataValidation** property.

You configure set values so that the data that a user enters in one field populates other fields. For example, if you configure a set value for the **Asset** field, when a user enters a value in the **Asset** field, other text boxes, multipart text boxes, combination boxes, or check boxes are populated with data.

You configure filters in a field to limit the type-ahead choices for a user based on information in another text box, multipart text box, combination box, or check box.

After you configure autofill properties, the information is validated with the data on the server. If the autofill configuration data is different from the server data, the fields are updated with the server data.

Configuring autofill properties:

To improve performance, fields in forms can be automatically filled with cached data. If your network latency is over 100 milliseconds, you can specify the fields to be automatically filled.

About this task

You can configure autofill for a text box or a multipart text box. The autofill properties that you can configure are type-ahead, set values, and filters. You can configure type-ahead so that the list of possible matches includes one or more business objects. You configure set values so that the data that a user enters in one field populates other fields. You configure filters in a field to limit the type-ahead choices for a user based on information in another text box, multipart text box, combination box, or check box.

Procedure

1. In the System Properties application, set the value of the **mxe.webclient.ClientDataValidation** property to *1*.
2. In the Application Designer application, select an application, highlight the field to configure, and open the properties sheet.
3. Click the **Advanced** tab. In the Autofill Properties section, configure the type-ahead feature.
   a. If a domain is not specified, complete the Domain field by identifying a domain in the Database Configuration application.
   b. Add the key domain attribute. To specify a domain attribute, you choose the primary key of that object as an attribute in the Attribute table that the domain is pointing to, and set the key Boolean value to `true`.
   c. Optional: To provide more information about the key attribute and to distinguish it from other type-ahead choices, add another attribute.

   d. Click **OK**.
4. Configure the set values.
   a. Specify an attribute that corresponds to the field whose value you want to set.
   b. Specify the domain attribute of the field where the set value comes from and click **OK**.
5. Configure filters.
   a. Specify the field to act as a filter for the type-ahead choices.
   b. Specify the domain attribute of the field with the type-ahead choices.
6. Save the application definition.

Example: Configuring autofill for the Affected Person field in service requests:

If you frequently open service requests for the same users, you can configure autofill on the Affected Person field. The next time you open a service request, as you type in the Affected Person field a list of one or more possible matches for the data is listed.

Procedure

1. In the System Properties application, set the value of the **`mxe.webclient.ClientDataValidation`** property to *1*.
2. In the Application Designer application, select the Create Service Request application, highlight the **Affected Person** field and open the properties sheet.
3. Click the **Advanced** tab. In the Autofill Properties section, click **Configure Type-Ahead**.
4. If a domain is not specified, go to the Database Configuration application to define a domain and complete the **Domain** field.
5. Add the domain attribute. To specify a domain attribute, you choose the primary key of that object as an attribute in the Attribute table that the domain is pointing to, and set the key Boolean value to `true`.
6. Optional: To provide more information about the key attribute and to distinguish it from other type-ahead choices, add another attribute.
7. Click **OK** and then save the application definition.

Error notification for client-side validation:

The values for data type fields are validated locally in the browser. If an error occurs, the user is notified immediately before a server round trip is needed.

For example, when you specify a value in an alphabetic field, the length of the value is validated in the browser. The error is added to the queue with the asynchronous values, but the error value is not counted in the threshold for the queue.

The error values are validated the next time the queue is validated, either when the threshold or timeout for asynchronous values is reached or when a synchronous event is triggered.

Error values for smart-fill fields and simple data entry values are also not included in the queue threshold.

Best practices for data validation types

In some circumstances, one type of data validation can be more efficient to use than another. All validation types work on textboxes, multiple-part textboxes, and check boxes. Synchronous and asynchronous validation also work on multiple-part text boxes and check boxes while client-side validation works on the first part of multiple-part check boxes.

Table 2. Data validation types and use cases

Validation type	Behavior setting	Configuration options	User interaction	Best practice	Interaction with smart-fill
Synchronous	Available if the asynchronous setting is turned off.	System Properties application or Application Designer application	Users must correct the errors in the current field before moving to next field.	Fields that affect the behavior of other fields. For example, if a particular entry in field A causes field B to become read-only, you can turn off asynchronous validation for field A. Users then cannot tab out to field B inadvertently before field A validates.	When users tab out of a field, the Select Value window opens containing filtered values. Users can specify the necessary value.
Asynchronous	Default behavior	System Properties application or Application Designer application	Errors and warnings are indicated at the field level. Users can resolve the errors and warnings at any time, in any order before saving.	In applications where users enter data quickly.	When users tab out of a field, the Smart Fill icon appears in the field. Users can click the icon and the Select Value windows opens containing filtered values. Alternatively, while in field users can press Ctrl+Enter on the keyboard to invoke the Select Value windows containing filtered values.

Table 2. Data validation types and use cases (continued)

Validation type	Behavior setting	Configuration options	User interaction	Best practice	Interaction with smart-fill
Client-side	Default behavior for data type validation	Asynchronous system setting must be turned on. Configurable for filters and set values in the Application Designer application.	Errors and warnings icons are indicated at the field level before server validation. Users can also see values in fields. If type-ahead is configured, users can see the information before server validation.	In applications where users experience network latency and for whom server round-trips for validation takes too long.	You can configure type-ahead to allow users to see a list of available options as drop-down values for a field. The available options in the list reduce as each successive letter is typed. The filtered Select Value window does not appear when a choice is made using the type-ahead list.

Property configurations

The behavior and characteristics of various user interface components can be determined by the values set as system properties or by values set for individual controls. Filter for `webclient` in the System Properties application to view and modify interface properties. Modify properties for an individual control in the Application Designer.

Common user interface (webclient) properties include properties that determine browser behavior when a connection is lost and properties that identify the location of login and log out pages. It can be practical to change these and other property values, depending on the type of application behavior you want to develop.

If the `webclient.smartfill` system property is enabled, when a user enters partial data in a field and an exact match is found in the database, the field updates with this value. If multiple possible matches are found, the user can select from a list of possible matches to update the field. The combined use of smartfill and asynchronous validation on a field can, in some cases, cause confusion. If required, you can disable smartfill for the system or for an individual control.

If you intend to use rich text editing capabilities that use custom fonts, you can add these fonts to the `webclient.richtext.fontlist` property. The Rich Text Editor control supports generic fonts by default, such as serif and sans-serif.

Developing applications

Before you create or modify applications, review the design guidelines and explore the workspace in the Application Designer.

Creating and modifying applications

You can either create an application or duplicate an existing application to use as a template. You can then modify the application to suit your business needs. You can also delete user-defined applications but you cannot delete predefined applications.

Duplicating applications

You can duplicate an existing application to use as a template for a new application. You can then extend and modify the functionality of the base application in the Application Designer.

Procedure

1. Open the application you want to duplicate in the Application Designer.
2. Select the **Duplicate Application Definition** action to open the Duplicate Application window.
3. In the **Application** field, specify a unique name for the application. Use a clear and consistent naming convention, such as using your company name as a prefix. Choose application names that are easy to identify in menus, and easy to recognize in log files and during a system upgrade.
4. In the **Description** field, specify a text description. This text shows in the navigation bar of the application.
5. Click **OK** and then click **Save**.

Results

The duplicated application is listed in the **Go To** menu in the same module as the original application. To view the new application, sign out of the system to refresh the list of applications in the **Go To** menu. When you sign back into the system, the new application is available in the **Go To** menu.

What to do next

Open the application in the Application Designer and make the changes needed to adapt the application to your business needs.

Creating applications

You can create three types of application in the Application Designer; a power application, a single-page application, or a self-service application. After you create an application, it is automatically added to the **Go To** menu.

Before you begin

Before you create an application, the main object for the application must exist in the database. If you intend to associate the application with a new object, define the object in the Database Configuration application before you create the application.

Procedure

1. Click **New Application Definition** in the toolbar of the Application Designer.
2. In the **Application** field, specify a unique name for the application. Use a clear and consistent naming convention, such as using your company name as a prefix. Choose application names that are easy to identify in menus, and easy to recognize in log files and during a system upgrade.
3. In the **Description** field, specify a text description. This text shows in the navigation bar of the application.

4. Specify a key attribute for the application.
5. Specify the name of the main object for the application.
   a. If the object is set at the SYSTEMORGSITE level, you must create a site text box field and an organization text box field.
6. Specify the name of the module that is to include the application. Every application must be located within a module.
7. Click the button that corresponds to the type of application that you want to create. The new application opens in the workspace.
8. Click **Save Application Definition**.

Results

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

What to do next

Filter for the new application in the List tab of the Application Designer to start working on the new application.

The default security setting for new applications is Read. All users who have access to the application in their security profiles can read records but cannot insert or delete information. Select **Add/Modify Signature Options** in the **Select Action** menu to define additional signature options for the new application.

Deleting an application

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

Before you begin

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

Procedure

1. Login to a database editor as an administrator.
2. Enter the following commands, and replace the variable *<APPLICATION NAME>* with the name of the application you want to delete. Enter the application name in all uppercase. For example, if the name of the application you want to delete is Test, replace APPLICATION NAME with TEST in all the commands.
   a. `delete from maxapps where app='<APPLICATION NAME>';`
   b. `delete from maxpresentation where app='<APPLICATION NAME>';`
   c. `delete from sigoption where app='<APPLICATION NAME>';`
   d. `delete from applicationauth where app='<APPLICATION NAME>';`
   e. `delete from maxlabels where app='<APPLICATION NAME>';`
   f. `delete from maxmenu where app='<APPLICATION NAME>' and menutype !='MODULE';`
   g. `delete from maxmenu where where elementtype='APP' and keyvalue='<APPLICATION NAME>';`

    h. `delete from appdoctype where app='<APPLICATION NAME>';`
    i. `delete from sigoptflag where app='<APPLICATION NAME>';`
3. Logout of the database editor.

Creating restrictions

To restrict the records that a user can see in an application, you add a query to the application. To restrict the data that users see for a control, you add the **`allowqualifiedrestriction`** property to the XML for a control in an application

Creating application restrictions:

You add a query to an application to restrict the records that a user can see. To hide inactive users in the Users application for example, you use the query "status != 'INACTIVE'". You create restrictions for power applications and single-page applications in Application Designer.

Procedure

1. Open the application in Application Designer.
2. In the **Select Action** menu, click **Toggle Show All Controls**.
3. Select the presentation control and click the **Control Properties** icon.
4. In the Presentation Properties window, enter the query in the **Application Restrictions** field.
5. Save the application definition.

Restricting data:

You add the **`allowqualifiedrestriction`** property to the XML for an application to restrict the data that is returned from the database. You can add the restriction when data is shown but not used for calculations. You can add the restriction for data that is used to route a process through steps. You can also restrict the records that are returned when using lookups.

Restricting the data returned from the database:

You restrict data by adding the **`allowqualifiedrestriction`** property. For example, you can restrict the classifications available to users who are classifying a service request.

About this task

You do not apply the restriction to child objects. You manually add the **`allowqualifiedrestriction`** property to the XML controls that create a data source. The controls that create a data source have an object, a relationship, or a bean class. The following controls are supported:

- datasrc
- section
- simpleheadertable
- tab
- table
- tree

Some controls inherit data sources from higher-level controls. When you add the property to the XML, you also make the data read-only. Editing a restricted data set can have unexpected results.

Procedure

1. Open the application XML and search for the control where the data source is needed.
2. Add `allowqualifiedrestriction= true` to the XML.
3. Import the modified XML.

Examples of restrictions on the data that users see:

You control the data that users see by restricting the data that is returned from the database. For example, you can restrict the person groups that users see when selecting an owner in the Change application.

Example: Restricting the person groups that users see when selecting an owner in the Change application

In this example, you set the **`allowqualifiedrestriction`** property into the XML for the tab in the window:

```
<dialog id="owner" label="Select Owner">
   <tabgroup id="owner_labor">

   <tab id="owner_labor_group" label="Person Groups" mboname="persongroup" allowqualifiedrestriction="true"
beanclass="psdi.webclient.beans.common.OwnerGroupBean">
```

Example: Restricting when users see the 'View Work Details' window in the Assets application

In this example, you set the **`allowqualifiedrestriction`** property into the XML for the tab in the table in the window:

```
   <dialog id="wopms_byasset" label="View Work Details" relationship="VIEWWOPMS"
  beanclass="psdi.webclient.beans.common.ViewWOPMsBeanAsset" width="975" height="400">

    <section id="viewWOandPMs_ViewWOandPMsFieldGridAsset" border="true">
      <multiparttextbox id="viewWOandPMs_ViewWOandPMsFieldGrid_2" dataattribute="assetnum" inputmode="readonly"
  descdataattribute="asset.description" longdescinputmode="readonly" descinputmode="readonly" />
      <checkbox id="viewWOandPMs_ViewWOandPMsFieldGrid_3a" dataattribute="showchildren" />
          <checkbox id="viewWOandPMs_ViewWOandPMsFieldGrid_3b" dataattribute="showparents" />

   <buttongroup id="viewWOandPMs_ViewWOandPMsFieldGridAsset99">
    <pushbutton id="viewWOandPMs_ViewWOandPMsFieldGridAsset98" label="Refresh" default="true" mxevent="doRefresh"
targetid="wopms_byasset"/>
   </buttongroup>
   </section>

   <tabgroup id="viewWOandPMs_viewWOandPMsAsset">

   <tab id="viewWOandPMs_viewWOandPMsAsset_ViewWO_tab" default="true" label="Work">
     <table id="viewWOandPMs_viewWOandPMsAsset_ViewWO" label="Work" relationship="VIEWWORK_BYASSET"
allowqualifiedrestriction="true" beanclass="psdi.webclient.beans.common.ViewWOByAssetBean" selectmode="multiple">

        <defaultvalue id="viewwoasset_historyflag_default" dataattribute="historyflag" value="N"
defaulttype="query" />
        <defaultvalue id="viewwoasset_istask_default" dataattribute="istask" value="N" defaulttype="query" />
```

Example: Restricting the classifications that users see when classifying a service request in the Service Request application

In this example, you set the **`allowqualifiedrestriction`** property into the XML for the table in the tree in the window:

```
 <dialog id="associateClassification" label="Classify">

  <tree mboname="CLASSSTRUCTURE" allowqualifiedrestriction="true" id="associateClassification_tree"
beanclass="psdi.webclient.beans.assetcat.AssociateSpecBean" width="600" height="250" openfirstlevel="false"
maxchildren="1000">
```

Defining signature options for an application

Signature options specify privileges for using applications, menu options, and toolbar items. You apply signature options to groups of users to control their access to data, to applications, and to controls. New applications are automatically assigned a READ signature option. You can add more signature options in the Application Designer.

Procedure

1. Open the application in the Application Designer.
2. Select the **Add Modify Signature Options** in the **Select Action** menu and click **New Row**.
3. Specify a signature option in the **Option** field.
4. Optional: Add additional signature options in the **Option Also Grants** field. Use commas to separate multiple entries if you want to enable more than one additional signature option. For example, if the signature option grants access to the People application, you can add signature options that grant access to a related application, such as the Users application.
5. Optional: In the **Option Also Revokes** field, specify any signature options that are revoked as a result of adding the primary option. For example, you add a signature option that allows read-only access to an application. You want to remove any write access to the application that users with the read-only signature option have. To remove the write access, you specify the signature option that grants write access to the application in the **Option Also Revokes** field.
6. Save the signature options.

Arranging the layout of controls

The layout of components is relative to the other components in a user interface. Use container controls, like sections and tables, to group associated controls. Arrange controls to make it easy for users to enter information in an orderly and efficient manner.

Laying out a new tab

In standard applications, the information is contained in multiple tabs, including the List tab, the application main tab, and additional tabs as required.

Procedure

1. Optional: To add a new tab, highlight the **Tab Group** control in the Application Designer and drag a **Tab** control to it from the Control Palette.
2. Highlight this new tab control and drag a **Section** control to it from the Control Palette. Repeat this step if you want to create multiple sections that create vertical division in the tab. A section control is a container for other controls and is not visible in the application.
3. Add section column controls to a section if you want to divide a section horizontally.
4. After you create horizontal and vertical sections, you can add data controls, such as text boxes or buttons, within each section or section column.

Merging classification attribute values into one column

Multiple columns are used to display classification attributes and their values which makes the display complex. You can merge the data for the three columns into one column to make more space available in the table for additional columns to be added. This size reduction ensures that the table is not so large that users have to scroll to see columns.

Before you begin

The Maximo® business object (MBO) that the Classification Specifications table is bound to must contain ALNVALUE, NUMVALUE, and TABLEVALUE attributes. If the MBO does not contain the attributes, you cannot merge classification attribute values.

The Classification Specifications table must also contain either a field class or a MBO validation class. The inclusion of either a field class or a MBO validation class ensures that all the value fields are read-only. The only exception is the value field associated with the data type of an attribute. For example, if the datatype is alphanumeric, the alphanumeric MBO attribute is read/write while the numeric and table value MBO attributes are read-only.

About this task

You can merge classification attribute values only for the classification specification attributes.

Procedure

1. In the Application Designer application, open the application that you want to change.
2. In the Specifications table on the **Specifications** tab, click the attribute that you want to change.
3. Delete the alphanumeric value column, the numeric value column, and the table value column by using the Context menu.
4. In the toolbar, click **Control Palette** and select the **Table Column** icon.
5. Drag the table column onto the Specifications table and close the Controls window.
6. Open the Context menu, click **Properties**, and specify the label name for the column. If you do not specify a name, the label from the alphanumeric value attribute is used.
7. Specify `{CLASSSPECVALUE}` as the attribute value, and clear the **Show Filter** check box.
8. Ensure that the **Sortable** check box and the **Filterable** check box are clear.
9. Save the changes.

Controlling the size of fields

You change the size of a single field by modifying the width value for that field in the `presentation.xml` file. You can also define a range of values for a data type in the `field-size-config.xml` file.

Specifying size ranges for data types:

You control the size range of a data type in the user interface by modifying the value in the `field-size-config.xml` file. The size of a field in the user interface depends on a setting for each data type and the length of the column in the database. This dependency creates a uniform appearance in the UI when fields of different lengths are beside or on top of each other in the same section.

About this task

Field sizes are defined in the `<root>\applications\maximo\properties\field-size-config.xml`. For example, for an ALN field:

```
<maxtype name="aln">
<range below="2" size="2"/>
<range upper="30" size="10"/>
<range upper="31" size="40"/>
</maxtype>
```

In this example if the ALN value is defined with a length of 1 or 2, the field displays 2 characters. If the value is defined from 3 to 30, the field displays 10 characters. If the field is defined with a size of 31 or greater, the field displays 40 characters.

Procedure

1. Open the `field-size-config.xml` file in a text editor.
2. Search for the data type and change the size range that is specified for the field.
3. Save the file.
4. Rebuild and redeploy the .EAR files on the application server for your changes to take effect.

Specifying the size of single fields:

You modify the width attribute in the `presentation.xml` file to change the width of a single field.

Procedure

1. Open the `presentation.xml` file.
2. Identify the field that you want to change the width of.
3. Check that the `description id` has a width attribute and add a width attribute if it is absent.
4. Change the width attribute in the `description id` to specify the size of the field. For example, you use the following code to change the width of a field to 100 pixels:

   ```
   <tablecol dataattribute="description" id="results_showlist_tablebody_3" width="100"/>
   ```
5. Save the `presentation.xml` file.

Adding images

The Application Designer includes controls that you can use to add images such as icons, to an application. The images directory contains existing images that you can reuse or you can add new images to the directory. Images must be in a file format supported by a browser.

Adding images to the images directory:

All images used in applications must be stored in the images directory. You can add new images to this directory.

Before you begin

Images must be in a format that is supported by a browser, such as .gif, .jpg, or .bmp formats.

Procedure

1. Copy an image (in a supported format) into the images directory, at `<`*`root`*`>\applications\maximo\maximouiweb\webmodule\webclient\images`.
2. If the image is .gif format, no further action is necessary before you use the image in the Application Designer.
3. If the image is not in .gif format, you must redeploy the .EAR files on the application server before you can use it.

Adding a record image to an application:

You can add a record image control to an application to create a placeholder for a record image, typically an expandable thumbnail image that a user can click to enlarge. When you add the record image to the application, the image displays automatically in the placeholders you created with the record image control.

Before you begin

The record image file must be located in the images directory before you can add it to an application. If the **Add Modify Image** option is not included in the **Select Action** menu, you must add this menu option before you can add the record image to the application.

About this task

You can add only one record image to an application but you can use the same image multiple times within the application by inserting multiple record image controls.

Procedure

1. Highlight a container control and drag the **Record Image** control to it from the Control Palette. You can place a record image control into any container control.
2. Select the **Record Image** control and click **Control Properties** to open the Record Image Properties window.
3. Optional: Deselect **Is Thumbnail** if you only want to display a full-size version of the image. The default behavior for record images is to display an expandable thumbnail image that a user can click to enlarge. If you do not want to provide a thumbnail version of the image, you must specify a value in the **Label** field that overwrites the default label, Click to enlarge.
4. Optional: You can specify a label for the image or you can leave this field blank to use the default label text, Click here to enlarge.
5. Specify appropriate values in the **Height** and **Width** fields, based on the size of the image you intend to use. The default values generate a thumbnail image of 100 x 100 pixels, appropriate for a square image. To avoid distortion of the image, ensure that the size values are proportionate to the size of the image.
6. Provide a text description of the image in the **ALT Text on Image** field. Alternative text is used as a substitute for the image when images are not supported, for example by screen readers.
7. Repeat the preceding steps if you want to add placeholders for the record image in other positions in the application, such as on another tab.
8. Select the **Add Modify Image** option in the **Select Action** menu to specify the record image to use for the application. The image can be in .gif or .jpg format and must be located in the images directory.

Configuring default values for controls

You can change the labels for fields, specify default values and filters for fields, and you can configure value lists for fields.

Assigning a default label to an attribute

You can specify a default value for an attribute and all text box controls that bind to the attribute inherit this value. The value is saved in the database

Before you begin

This procedure includes updating the database. Backup the database and make any necessary arrangements before you restart the database and the application server.

Procedure

1. Select the attribute in the Database Configuration application or click **New Row** to specify a new attribute.
2. Specify the value you want to use as the default label for the attribute in the **Title** field.
3. Click **Save**.
4. Run the `configdb` command to save the changes to the database. Sign out of the system and stop the application server.
5. In the Application Designer, drag a text box control from the Control Palette into a section control in the workspace.
6. Select the text box control and click **Control Properties**.
7. Specify the attribute that you configured. The window refreshes and the **Default Label** field contains the value that you configured.

Saving changes to the database

After you make changes to the database in the Database Configuration application, you must commit the changes to the database. The data on the server is then reloaded and the user interface (UI) is updated with the changes that you made. For example, new fields are added or field labels are updated.

Procedure

1. Select the **Apply Configuration Changes** option in the **Select Action** menu in the Database Configuration application.
2. Sign out of the system.
3. Stop the application server and wait one minute to ensure that the user session terminates.
4. Open a command prompt, change the directory to `install_home\tools\maximo`, and run the `configdb` command to add the new default value to the database.
5. Start the application server and login to the system.

Overriding a value in the default label field

You can specify a label for a text box that overrides the default label.

About this task

The label that you specify applies to this specific application and does not replace the default value that you configured for the attribute.

Procedure

1. Select the text box in the workspace and click **Control Properties**.
2. Specify the value that you want to use in the **Label** field. The label for the text box updates in the workspace.
3. Click **Save**.

Assigning a default value to an application field

You can assign a default value for a field that is specific to an application. The default value does not apply to fields in other applications that are associated with the same attribute.

Procedure

1. Select the section control that contains the field and click **Control Properties**.
2. Drag the **Default Value** control to the section from the **Control Palette**. If you do not see the Default Value control when the workspace refreshes, click **Toggle Show All Controls** in the **Select Action** menu.
3. Select the **Default Value** control and click **Control Properties**.
4. Specify an attribute in the **Attribute** field and specify its default value in the **Value** field. When you associate a text box in the section with the same attribute, the default value is set to this value.
5. Optional: To use the value you selected for each new record, specify Insert as the Default Type.
6. Optional: To inherit the default value from another data source and attribute, complete the **From Data Source ID** and **From Attribute** fields.

Setting default filters for a table window

You can define a default filter for a table window using the **Default Value** control in the Application Designer,

Procedure

1. Drag the **Default Value** control from the Control Palette into the section control that contains the table to filter. If the **Default Value** control is not visible after the workspace refreshes, click **Toggle Show All Controls** in the **Select Action** menu.
2. Specify the attribute to filter.
3. Specify one or more values to use as filters in the **Value** field. Use commas to separate multiple values. You can also use % as a wildcard or = to specify an exact match.
4. Select **Query** in the **Default Type** field.
5. Click **Save**.

Adding a value list to a text field

You can add a lookup icon to a text box control and configure a list of values that users can select from when they click the icon.

Before you begin

You can use an existing lookup value list or you can configure a new one. To create a value list, configure a domain that contains the values and associate the domain with an attribute. To use an existing lookup value list, export the system XML and locate the relevant Lookup ID in the **lookups.xml** file. When you assign lookups for a field that has a table domain, you use a lookup from the application that the field is in. You cannot use the VALUELIST lookup with table domains.

Procedure

1. Click **Add New Domain** > **Add New ALN Domain** in the Domains application,
2. In the ALN Domain window, specify values in the required fields. You must use precisely the same values when you configure the associated attribute in the Database Configuration application. The data type determines whether options in the list are displayed as uppercase, lowercase, or either. The length determines the width of the list. A length of 10 is adequate for most purposes.
3. Click **New Row**, specify a value, and click **OK**. This value becomes an item in the value list.
4. Repeat Step 3 for each additional value.
5. Click **Save**.
6. In the Database Configuration application, open the application to which you intend to add the value list.
7. In the **Attributes** tab, click **New Row** and configure the attribute by using the precise values you specified for the domain.
8. To save the changes to the database:
   a. Sign out of the system.
   b. Stop the application server and wait 1 minute.
   c. Open a command prompt, change the directory to `install_home\tools\maximo`, and run the `configdb` command to add the new default value to the database.
   d. Start the application server and login to the system.
9. In the Application Designer, drag a text box control from the Control Palette to a section in the workspace. An Invalid Binding message is displayed in the text box. When you associate the control with an attribute, the message no longer displays.
10. Select the text box control and click **Control Properties**.
11. Specify the same name for the attribute as the name you configured in the database.
12. Specify Valuelist as the value in the **Lookup** field. A lookup icon is displayed next to the text box control and the Invalid Binding message is no longer displayed.
13. Click **Save**.

Moving application controls

Use the Application Designer to move controls across tabs or across applications. You can also move controls and applications to a different server and export multiple applications.

Moving controls across tabs

You can move a control or a group of controls from one tab to another within an application.

About this task

When you move a container control, all of the controls it contains move with it.

Procedure

1. Select a control or a group of controls, typically a section.
2. Right click the mouse, and click **Copy** or **Cut**.

3. Click the tab where you want to copy or move the control.
4. Select the target control (usually a section).
5. Click **Paste**.

Moving controls across applications

You must edit the `presentation.xml` files to move a control or a group of controls from one application to another.

Procedure

1. Open the source application in the Application Designer and click **Export Application Definition**. The XML definition opens in the browser.
2. Save the XML definition to a file.
3. Open the target application and perform the same steps to export the XML definition and save it to a file.
4. Open the XML file for the source application in a text editor.
5. Locate the control ID or the section control ID that you want to move. You can use the unique control ID that is listed in the Control Properties window to search for the control in the XML file.
6. Select the relevant code and copy it to the clipboard.
7. Open the XML file for the target application in a text editor.
8. Navigate to where you want to insert the control and Paste the code that is stored on the clipboard.
9. Save both XML files.
10. Open the target application in the Application Designer and click **Import Application Definition**.
11. Specify the target application XML file that you modified, and click OK.

Results

When you copy a multipart text box control to a different application, the control must link to an attribute that exists in the main object table for both applications. If the attribute relationship does not exist, the multipart text box control in the target application does not contain any property values.

Moving components and applications across servers

You can move user interface components and applications across servers, for example to move from a test environment to a production environment. Different members of the organization can configure different applications and then move them to a centralized test environment.

Procedure

1. Create a backup directory for the XML files you are moving. You can use the backup files to restore application definitions in the future, if necessary.
2. Open the application on the source server, and click **Export Application Definition**.
3. Save the code as an XML file in the backup directory you created.
4. Repeat steps 2 and 3 for each application that you want to export.
5. Make a copy of the XML file(s) you created.
6. Paste the copy of the XML files onto the target server.

7. Open the Application Designer on the target server, click **Import Application Definition**, and navigate to the XML files. When you import an application, you overwrite the application in the database and not in the directory of `presentation.xml` files.

Exporting multiple applications

You can copy multiple applications at the same time to another server by exporting the presentation XML for each application to a single XML file.

Procedure

1. In the Applications tab of the Application Designer, press the Return key. The screen refreshes with a list of all applications.
2. Scroll to the end of the screen and click **Select Records** check box. The screen refreshes and includes a selection check box for each record.
3. Check each application whose XML presentation you want to export.
4. In the Workspace tab, click **Export Application Definition** from the **Select Action** menu.
5. An XML file opens in a browser window that includes the presentation code for each application you selected for export. You can find the start point for an application by searching for the `presentation id` string.
6. Save the XML file to a directory.

What to do next

To import the XML file into another system, in the Application Designer click **Import Application Definition** and navigate to the XML file. The import reads the beginning and end point for each application embedded in the XML file and loads the applications accordingly.

Adding navigation components

You can link to applications and to external URLs and you can add menus and buttons to organize application navigation.

Navigation components

There are several ways that you can integrate a new application into the existing navigation and you can add new navigation components if required.

The Application Designer provides several navigation components that you can modify to add a link to an application, an application tab, or an application popup window.

- **Go To** menu
- Application toolbar
- **Select Action** menu
- Menu templates
- Application bar menu in self-service applications

Go To menu

The **Go To** menu lists all of the modules and applications that a user can access. The menu is hierarchical and lists modules at the top level. When the user clicks a module, a submenu lists the applications in that module. When you create an application in the Application Designer, you specify its module and the application name is added under this module in the **Go To** menu. When you duplicate an application, the new application is added to the same module as the original application.

The contents of the **Go To** menu are generated automatically and you cannot customize them in the Application Designer.

Application toolbar

The application toolbar contains buttons that are arranged in a horizontal line at the top of the application window. You can add new buttons to the toolbar and modify existing ones in the Application Designer. Buttons can be specific to the toolbar in one application tab, or can be visible in all tabs.

Button images are stored in the images directory and you can add new images or reuse existing images.

Select Action menu

The **Select Action** menu lists application-specific actions. You can add new options to the **Select Action** menu and modify its existing options in the Application Designer.

Menu templates

You can reuse the code for existing menus as a template for a new menu, Use the Application Designer to export the `menus.xml` file, modify the file in a text editor, and then import the updated file back into the Application Designer.

You can use the XML code for any of the menus in the `menus.xml` file as a template for a new menu.

Application bar menu

Self-service applications do not have a Go To menu and provide access to applications in a menu at the top of the application window. You can export the `presentation.xml` file to add a navigation property to the application bar of a self-service application. If you customize the application bar menu, any preconfigured links are ignored.

Linking to an external URL

You can create a link to an external website from an application using a hyperlink control.

Procedure

1. Open the application in the Application Designer.
2. Select either a section control or a table control and drag a hyperlink control to it from the Control Palette.
3. Specify a value in the Label field. This value becomes the text link that a user clicks to go to the URL.
4. Specify a valid URL for the destination site.
5. Optional: Specify values in the **Image Filename** and **Image Alignment** fields if you want to associate an image with the hyperlink.
6. Optional: Specify a value in the **CSS Class Name** field if you want to change the appearance of the hyperlink.

Results

When a user clicks the link within the application, the destination website opens in a new browser window.

Linking to an application from a text field

You can configure a pull-down menu associated with a text field to have options that take the user to different applications.

Procedure

1. Open the application in the Application Designer.
2. Select a text box control and click **Control Properties**. You can also configure an application menu for multipart text box control and a table column control.
3. Add the name of an application that you want to link to in the **GOTO Application** field.
4. Specify a value in the **Menu Type** field that corresponds to the application you specified. Click **Export System XML** in the **Select Action** menu to export the `menus.xml` file. You can search this file for the appropriate menu type value.

Adding buttons to the application toolbar or Common Actions menu

You can include extra buttons on the application toolbar or in the Common Actions section of the side navigation menu. Toolbar buttons provide quick access to common actions, such as Save and Insert, and can be seen in all application tabs or can be limited to a specific tab.

About this task

If the side navigation menu is enabled, any buttons that you add to the toolbar are displayed in the Common Actions section of the side navigation menu.

Procedure

1. Open the application in the Application Designer.
2. In the **Select Action** menu, select the **Add/Modify Toolbar Menu** option.
3. Create a row and select an **Element Type**.
4. Select a **Key Value**. The key value represents the button action and is visible when the user moves the mouse over the button.
5. Specify an image to use for the button that is located in the images directory and is in a supported format.
6. Enter position values that determine the location of the button relative to the other buttons in the toolbar.
7. Specify the tabs where the option is visible.

Example

To add a **Change Password** button to the toolbar in the Main tab of the Person application:

1. Set the **Element Type** to App (application).
2. Select `ChgPassword` as the **Key Value**.
3. Set **Position** to `60` and **Subposition** to `0`.
4. Specify the `nav_icon_changepassword.gif` file in the **Image** field.
5. Select the Main tab option.

What to do next

You can add buttons that act as separators if you want to group several buttons together in a section.

Adding buttons next to text boxes

You can add buttons next to text boxes for the menu items that are most frequently used. You can add up to three buttons to a text box.

About this task

You can select existing menus from the menus.xml file, or you can define new menus in the menus.xml file. The menu items must contain the label, event, and image attributes, for example:

```
label="Select Value" event="selectvalue" image="menu_icon_find.gif
```

The following restrictions apply when you add buttons next to text boxes and multipart text boxes:

- **Select Value** menu items are not displayed if there is no lookup attribute defined for the text box. For example, **Select Value** menu items are not displayed next to **Long description** fields.
- If the button is a link to an application and the user does not have access to the application, the button is not displayed.
- If the menu item is defined with any conditional expressions, the button is not displayed.

Procedure

1. In Application Designer, select the field that you want to add buttons to and open the **Control Properties** window.
2. Specify the ID of the menu.
3. Save the application definition.

Example: Adding buttons next to a multipart text box:

You want to add buttons next to a multipart text box so that users do not have to search the **Select Action** or **Detail** menus. You identify the menu items that users require most frequently for the multipart text box. No existing menus include the items so you create a menu that is called ICONTEST1 in the menus.xml file.

When there are no restrictions on the text box or user, the first three menu items that are defined in the menu are displayed. When there are restrictions on the text box or user, a menu item might not be displayed and the next menu item is displayed instead.

You identified the five menu items that users require most often:

- Select Value
- Application link to the Configuration Items application
- Application link to the Assets application
- Classification search
- Attribute search

You create a menu that is called ICONTEST1 that specifies five menu items:

```
<menu id="ICONTEST1">
     <menuitem id="icontest1_mi1" label="Select Value" event="selectvalue" image="menu_icon_find.gif" />
     <menuitem id="icontest1_mi2" label="Go To" event="applink" image="menu_icon_link.gif" applinkid="ci"  />
     <menuitem id="icontest1_mi3" label="Go To" event="applink" image="menu_icon_link.gif" applinkid="asset"  />
     <menuitem id="icontest1_mi4" label="Classifications(w/Sigopt)" event="searchClassificationCI"
image="action_insert_launch.gif" sigoption="REFICONS" />
     <menuitem id="icontest1_mi5" label="Attributes" event="searchAttributeCI"  image="menu_icon_catalog.gif" />
</menu>
```

In the Control Properties window of Application Designer, you specify `ICONTEST1` as the ID.

Look up icons or **Select value** menu items are not shown next to **Long description** fields so the first button is not displayed. If the user group does not have access to the sigoption REFICONS, the fourth menu item is also not displayed. In this case, only the buttons for menu items 2, 3, and 5 are displayed as buttons next to the **Long description** field.

Adding options to the Select Action menu or More Actions menu

You can include options in the **Select Action** menu or in the More Actions menu. You add more options to give users access to the actions that they occasionally require.

About this task

If the side navigation menu is enabled, options that you add to the **Select Action** menu are displayed in the More Actions section of the side navigation menu.

Procedure

1. Open the application in the Application Designer.
2. In the **Select Action** menu, select the **Add/Modify Select Action Menu** option.
3. Create a row and select an **Element Type** and **Key Value**.
4. Enter position values that determine the location of the option in the menu.
5. Specify the tabs where the option is visible in the **Select Action** or **More Actions** menu.

Modifying links in a self-service application

To change the links in the application bar of a self-service application, you must configure a property in the `presentation.xml` file for the application. You can configure these links for a self-service application that you duplicated from an existing application or for a new self-service application that you create.

About this task

When you configure the `appbarlinks` property, any existing links that are found in the database are ignored, and only the customized links are displayed.

Procedure

1. Open the application in the Application Designer.
2. Select the **Export System XML** option in the **Select Action** menu.
3. Save the XML file to a local directory and open it in a text editor.
4. Search for the `appbar` tag in the XML file.
5. Add `appbarlinks=" "` to the `appbar` tag to add the navigation menu to the application bar control.
6. Insert the links, separated by semicolons.
7. For each link, you can specify up to three attributes, separated by commas.

   a. ID of the linked application
   b. Text description to display in the application bar
   c. Image to display next to the description

   For example:

```
appbarlinks="Application1,Description1,image1.gif:
Application2,Description2,image2.gif"
```

8. Save the changes to the XML file.
9. Import the file into the Application Designer to save the new menu information.

Configuring related information for fields

To see information related to a field without leaving the record, you add a hover window to the field. Hover windows are immediately enabled on some fields but you can also enable these existing hover windows on other fields or you can create a hover window depending on your requirements.

Example: Enabling an existing hover window

You can enable existing hover windows on fields to suit your needs. For example, you can add a hover window to show the contact details for the owners of service requests. In general, you can show related information on any fields that support lookup.

About this task

The following hover windows are immediately available on some fields but can be enabled on other fields:

Field	ID
Asset	*asset_recordhover*
Work Order	*wo_recordhover*
Person	*person_recordhover*
Item	*item_recordhover*

In a large company, a supervisor is responsible for ensuring that service requests are assigned to the correct responder. To get contact information for the owner, the supervisor typically navigates to the Person record, which takes time and decreases efficiency. The supervisor wants to be able to hover the mouse over the **Owner** field and see the details of the owner of a service request. You can enable a hover window on the **Owner** field in the Service Requests application so that the supervisor can hover over the field and see the full name, email address and telephone number of the owner of the service request, without leaving the record.

Procedure

1. In Application Designer, open the Service Requests application, click the **Owner** field and select **Control Properties**.
2. On the Advanced tab, specify `person_recordhover` as the ID of the hover window and save the application definition.

Example: Creating hover windows

If none of the existing hover windows suit your needs, you can create a hover window. For example, you can create a hover window to show the organizations where failure classes occur. You design the hover window and specify the fields that appear.

About this task

The following widget types are supported:

- **recordhoversection**
- **recordhoversectioncol**
- **recordhoversectionrow**
- **recordhovertextbox**
- **recordhovermultiparttextbox**

In a multinational company, an administrator is noting the organizations where failure classes are reported as assets seem to be failing more frequently in some organizations. The administrator wants to hover over the **Failure Class** field to see the name of the organization where the failure occurred.

You review the list of hovers and determine that none suit this scenario, so you create a hover for the **Failure Class** field.

Procedure

1. In Application Designer, select the **Export System XML** action and export the RECHOVERS.XML file.
2. Open the RECHOVERS.XML file in a text editor and add the following lines:

```
</recordhover>
<recordhover id="fc_recordhover">
 <recordhoversection id="fc_rhs_1" label="Failure Class Details">
  <recordhoversectioncol id="fc_rhsc_1">
   <recordhoversection id="fc_rhs_2">
    <recordhovertextbox dataattribute="ORGID" id="fc_rh_1" label="Organization">
   </recordhoversection>
  </recordhoversectioncol>
 </recordhoversection>
</recordhover>
```

3. Save the RECHOVERS.XML file and import it.
4. Open the Assets application, click **Failure Class** and select **Control Properties**.
5. In the **Advanced** tab, specify fc_recordhover as the ID of the hover window and save the application definition.

Providing user information

You can provide information to help users in the windows and tabs in the user interface. You can create links from windows and tabs, and also from field help, to topics in the product Knowledge Center. You can also add new error messages to an application.

Adding messages to the messages table

You can add new messages to an application that help the user to resolve issues when the application encounters errors. Messages are displayed to the user and are added to logs to help with error resolution. When you add a message in the Application Designer, it becomes available for translation into supported languages.

Procedure

1. Open the application in the Application Designer.
2. Select the **Messages** option from the **Select Action** menu.
3. Specify the identifier for the **Message Group**.
4. Specify a **Message Key** that is unique in this message group.
5. Select a Message ID.
6. Specify the text of the message in the **Value** field.
7. Complete any other relevant fields and click **OK**.

Linking to user information

You can add a link to a topic in the product Knowledge Center from a help grid control, or from the field help for any other control.

About this task

The URL for an Knowledge Center topic must have the following format: `com.ibm.mbs.doc,application/helpfile.html`

Procedure

1. To create a link from a help grid control, specify the URL for the topic in the **More information field** in the Help Grid Properties window.
2. To create a link from the field help for a control, specify the URL for the topic in the **More information** field in the control properties window When the user clicks Alt + F1, a **More information** link is available in the help window.

Results

When the user clicks a more information link, the relevant topic in the Knowledge Center opens in a new tab. You can add links from multiple controls to one topic, or you can link to multiple topics. You must add links manually to each individual control. If an Knowledge Center topic is moved or deleted, users can encounter broken links if you do not update each control.

Configuring dialog windows for an application

You can modify dialog windows that are specific to an application in the Application Designer. To modify dialog windows that are shared with other applications, you must modify a system XML file. You can also create a dialog and add it to an application.

Modifying an application dialog window

You can modify a dialog window that is specific to an application in the Application Designer.

About this task

This procedure applies to dialog windows that are specific to the current application. If the dialog window is not available in the Edit Dialogs window, the window is probably a system dialog window that is shared by other applications.

Procedure

1. In the Application Designer, click **Edit Dialogs** and select the dialog window that you want to modify.

2. In the Workspace, perform the modifications in the same manner as you modify an application.
3. Click **Save**.

Modifying a system dialog window

To modify a system dialog window, you must export the `library.xml` file and modify the XML file in a text editor.

Before you begin

Do not modify the `library.xml` file while other applications are using it.

About this task

System dialog windows, such as the Change Status window, are shared by multiple applications and any changes you make apply to all applications that use the dialog. If you do not want to modify the window for all applications, copy the XML code for the dialog element and use this as a template for a new window that is specific to an application.

Procedure

1. In the Application Designer, click the **Export System XML** option in the **Select Action** menu.
2. In the Export System XML window, select the Library option and click the **Export XML** icon. The `library.xml` file opens in a new browser window.
3. Save the `library.xml` file to a local file and open it in a text editor.
4. Locate the dialog element that you want to change in the XML file and modify as required.
5. Save the XML file.
6. In the Workspace of the Application Designer, click the **Import Application Definition** icon on the toolbar.
7. Browse to the local copy of the `library.xml` file that you changed and select it. The modified XML definition is saved to the database and the changes that you made are now available to all applications.

Creating a dialog window

You can add a new dialog window to a specific application or you can make it available for use by all applications. You must modify an XML file to create a dialog window and import the modified file into the Application Designer. After importing it, you can modify the new dialog window in the Application Designer.

Procedure

1. In the Application Designer, export one of the following XML files:
   a. In the List tab, click **Export System XML** in the **Select Action** menu to export the `library.xml` file if you want to add a new dialog for use by all applications.
   b. In the Workspace tab, click the **Export Application Definition** icon to export the `presentation.xml` file if you want to add a new dialog for use by a specific application.

   The XML definition opens in a new browser window.
2. Save the XML definition to a local file and open it with a text editor.

3. Add a code snippet for the dialog at the end of the XML file, just before the final closing tag, either the `</systemlib>` tag or the `</presentation>` tag. The following code creates a View History dialog window to the end of a `presentation.xml` file:

```
  <dialog id="newviewhist" label="View History">
  </dialog>
</presentation>
```

4. Save the changes to the XML file.
5. In the Application Designer, click the **Import Application Definition** icon, navigate to the modified XML file and import it.
6. Click the **Edit Dialogs** icon and select the dialog window you created.
7. In the Workspace tab, modify the dialog window as required.

Transferring data between applications

Linked objects and crossover domains are different configuration mechanisms that you can use to move data from one field in an application to another field in a different application.

Data transfer options

Although similar in purpose, there are distinct differences between the two configuration options that you can use to transfer data from a field in one application to a field in another. These data transfer options are based on either linked objects or on crossover domains.

Linked objects can pass data from a source application to a target application using database relationships. In the destination application, the data value can be read, or read/write, depending on how you configure the input field. If you configure the destination input field to be read/write, the application framework automatically updates the source application and parent object if you update a displayed value.

Crossover domains, pass data from one application to a different application using domains. A copy of the data value from a source application is passed to a destination application using the crossover domain as a trigger mechanism. You cannot edit the data value in the destination application and the copied data value is static.

Database relationships that link objects and crossover domains enable the export of information from a source object to a destination object using WHERE clauses to filter for list and validation values. You can use a database relationship or a crossover domain to accomplish the following objectives:

- Enter an asset number in the Items application, retrieve the serial number of the asset from the ASSET object, and copy this value into a field.
- Enter an asset number for a new work order, retrieve the parent asset number from the ASSET object, and insert this value into a field.
- Enter a user name in an application, retrieve the user's phone number and address from the PERSON object, and insert the information into two different fields.
- Enter the user name for a person that is creating a material receipt record, retrieve the name of the user's supervisor from the PERSON object, and insert it into a field.

Linking objects to share data between applications:

User interface controls are bound to database attributes that present data from the application parent (and child) objects in a browser window. Most applications present data that resides in other objects and you can use relationships, defined in the Relationships application, to indicate the connection between these objects. This scenario describes how to add user comments about an asset to a work order when you add the asset to a work order.

Adding a comment attribute to an asset:

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

Procedure

1. In the Database Configuration application, filter for the ASSET object and click the **Attributes** tab.
2. In the Details section, click **New Row**.
3. Specify values in the following required fields or accept default values where appropriate:

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

4. Click **Save**.
5. Perform the usual procedures to run the ConfigDB tool to update the database to enable these database changes to take effect.

Linking objects in different applications:

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

Procedure

1. In the Database Configuration application, select the Work Order application and then open the **Relationships** tab.
2. Click **New Row**.
3. In the Details window, specify the following values:

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

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

WHERE clause syntax	Description
:assetnum	Represents asset number value entered in work order screen
assetnum=	Look in Asset object for the asset number that matches the value entered in the work order screen
:siteid	The Site value entered in work order screen
siteid=	Look in Asset object for the Site value that matches the current value in the work order screen

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

4. Click **Save**.

What to do next

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

Adding a text box for comments:

Complete this procedure to add text box controls to the Assets application and to the Work Order Tracking application. The text box in the Assets application lets you input or edit a comment value for an asset. You can view these comments in the text box in the Work Order Tracking application.

Before you begin

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

Procedure

1. In the main tab of the Assets application in the Application Designer, click the section that holds the **Asset** field.
2. Click Control Palette and drag a text box control into the Asset section.
3. Open the Control Properties window for the text box control.
4. Specify the following information to bind the control to the assetcomment attribute:

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

5. Click **Save**.
6. In the Application Designer, open the Work Order Tracking application and then select the Asset section of the application.
7. Drag a text box control from the Control Palette to the workspace and position it between the **Asset** and the **Parent WO** fields.
8. Open the Control Properties for the text box and specify the following values:

Field input	**Value**	**Description**
Default Label	Defaults to Comment	When you specify assetcomment in the Attribute field, this field defaults to the default label value of the attribute (if available). You can assign your own label value for this attribute that is specific to the application in the **Label** field. For example, you might enter Asset Comments.

Field input	Value	Description
Label	Comments	The value you enter here, for example, Comments, overrides the Default Label setting within the application. The value you enter for Label is specific to the application and not stored in an object.
Attribute	asset2wolink.assetcomment	Type the name of the asset comment attribute in the Asset object. You must preface the assetcomment attribute with the name of the database relationship that binds the Asset and Workorder objects and a dot. For example, asset2wolink.assetcomment.
Input Mode	DEFAULT or READONLY	The **Default** option configures the **Input Mode** for the **Comment** field to be Read/Write or Edit mode. You can also configure the **Comment** field to be Read Only

Results

After this configuration, the **Comments** field is populated in the Work Order Tracking application whenever you add an asset record to a work order.

Crossing over domains to share data between applications

Using a crossover domain to display comments about an asset in the Work Order Tracking application is similar to using linked objects except that you use a crossover domain value to establish a link between the Work Order and Asset objects and not a database relationship. When using a crossover domain to copy data across applications, you cannot edit the asset comment value in the destination application.

Adding a comment attribute to an asset:

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

Procedure

1. In the Database Configuration application, filter for the ASSET object and click the **Attributes** tab.
2. In the Details section, click **New Row**.
3. Specify values in the following required fields or accept default values where appropriate:

Field label	Input value	Description
Attribute	Assetcomment	Attribute name

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

4. Click **Save**.
5. Perform the usual procedures to run the ConfigDB tool to update the database to enable these database changes to take effect.

Adding a comment attribute to a work order:

You must add an attribute (if one does not already exist) to the work order object.

Procedure

1. In the Database Configuration application, filter for the WORKORDER object and click the **Attributes** tab.
2. In the Details section, click **New Row**.
3. Specify values in the following required fields or accept default values where appropriate:

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

4. Click **Save**.
5. Perform the usual procedures to run the ConfigDB tool to update the database to enable these database changes to take effect.

Creating a crossover domain:

You can create a crossover domain in the Domains application.

Procedure

1. In the Domains application, click **Add New Domain** > **Add New CROSSOVER Domain** to open the Crossover Domain window.
2. Specify ASSET2WO as the domain name and Asset to Work Order Crossover Domain as the full description.

3. Click **New Row** in the Crossover Domain section to open the detail section.
4. Specify Asset in the **Object** field to define Asset as the source object of the crossover.
5. Enter assetnum=:assetnum and siteid=:siteid in the **Validation Where Clause** field.
6. Click **New Row** in the Crossover Fields section to open the detail section.
7. Use the lookup to select assetcomment from the Asset object in the **Source** field, then type woassetcomment in the **Destination** field.
8. Click **OK**.

Applying a crossover domain:

You must apply the crossover domain that you created to a field that triggers the crossover conditions.

Procedure

1. In the Database Configuration application, open the WORKORDER object and click the Attributes tab.
2. Click **Filter**, and specify assetnum in the **Attribute** field.
3. Press Enter to open the Assetnum details section, and specify asset2wo in the **Domain** field.
4. Click **Save**.

Adding a text box for comments:

Complete this procedure to add text box controls to the Assets application and to the Work Order Tracking application. The text box in the Assets application lets you input or edit a comment value for an asset. You can view these comments in the text box in the Work Order Tracking application.

Before you begin

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

Procedure

1. In the main tab of the Assets application in the Application Designer, click the section that holds the **Asset** field.
2. Click Control Palette and drag a text box control into the Asset section.
3. Open the Control Properties window for the text box control.
4. Specify the following information to bind the control to the assetcomment attribute:

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

5. Click **Save**.
6. In the Application Designer, open the Work Order Tracking application and then select the Asset section of the application.
7. Drag a text box control from the Control Palette to the workspace and position it between the **Asset** and the **Parent WO** fields.
8. Open the Control Properties for the text box and specify the following values:

Field input	Value	Description
Default Label	Defaults to Comment	When you specify assetcomment in the Attribute field, this field defaults to the default label value of the attribute (if available). You can assign your own label value for this attribute that is specific to the application in the **Label** field. For example, you might enter Asset Comments.
Label	Comments	The value you enter here, for example, Comments, overrides the Default Label setting within the application. The value you enter for Label is specific to the application and not stored in an object.

Field input	**Value**	**Description**
Attribute	asset2wolink.assetcomment	Type the name of the asset comment attribute in the Asset object. You must preface the assetcomment attribute with the name of the database relationship that binds the Asset and Workorder objects and a dot. For example, asset2wolink.assetcomment.
Input Mode	DEFAULT or READONLY	The **Default** option configures the **Input Mode** for the **Comment** field to be Read/Write or Edit mode. You can also configure the **Comment** field to be Read Only

Results

After this configuration, the **Comments** field is populated in the Work Order Tracking application whenever you add an asset record to a work order.

Configuring conditional user interface components

By applying conditions to control application behavior and to the user interface (UI), you can configure the UI to support your business processes. You can achieve the user interface behavior you require by configuring conditional data restrictions, by configuring conditional UI control behavior in Application Designer, or by configuring conditional security restrictions. For example, you can apply conditions to change the color of controls, the content of menus for different user groups, or the properties of the data shown.

Related information:

Technotes about conditional user interface features delivered with your product.

Conditional user interface

Conditions are clauses or classes that evaluate to true or false when they are evaluated against an object. Conditions can be applied to user interface (UI) controls to control how the UI is shown to users and to support your business processes. You can apply conditional data restrictions, signature options to conditionally control access to controls, or conditional control properties to configure the UI.

You can bind any UI control to a signature option to grant or revoke access to the control by groups of users. You can also apply conditions to the properties of a control so that the control looks or behaves differently for specified groups of users based on the evaluation of the applied conditions. For example, you can apply conditions to a control so that the color of the control changes depending on the value shown.

You can use the Application Designer to associate a control with a signature option and conditions. When a control is bound to a signature option or an attribute, you can take the following actions:

- Configure its behavior by setting a data restriction or conditional data restriction in Security Groups
- Grant the option in the Security Groups application
- Configure conditional properties in the Application Designer

When you are planning your conditional user interface system, you choose between the implementation methods listed. Data restrictions are useful for implementations where you want the restriction that you create to apply to every user interface element or application that uses an object or attribute. Security restrictions are useful when you want to restrict access to a control or to data to a specific group of users.

Example: Showing status options based on the status of the current record

XYZ Corporation requires that the work order status that is available to work order dispatchers is based on the current work order status. For this company, when a work order status is Waiting on Approval, only the values Approved and Canceled are available to the dispatchers.

To implement this business process, in the Condition Expression Manager, create the following conditional expression:

```
WOSTATUSAPPR :status = 'WAPPR'
```

Then in the Application Designer application, apply that condition to the domain values WOSTATUS|APPR and WOSTATUS|CAN. The status options shown on the UI reflect the status of the current work order.

Example: Changing historical data to read-only

ABC Inc. requires that all users see completed work orders from previous financial accounting periods as read-only. To implement this behavior, the application developer creates a condition that is true if the date checked is in the past. The date condition is evaluated against the date on completed work order records. The developer applies the condition to work order data in the Security Groups application for the security group that contains all users.

When the condition is evaluated, it returns a value of true if the work order date is from a previous financial period. All users see work order records from previous financial periods as read-only on the UI.

Hierarchy of restrictions:

The conditional restrictions that you define are applied hierarchically. For example, if you make an attribute read-only by using a data restriction, you cannot use the Application Designer application to make a field that is bound to the attribute editable.

The restrictions are applied as follows:

1. Database configurations
2. Data restrictions
3. Application configurations

Database configurations take priority over data restrictions and data restrictions take priority over application configurations.

If you define an object as read-only in the database, you cannot use data restrictions or the Application Designer application to make the object editable.

Data restrictions take priority over application configurations made in the Application Designer application. For example, if an attribute is made read-only using a data restriction, you cannot use the Application Designer application to make the attribute editable.

Configurations that are made with data restrictions apply everywhere an attribute is used. Application configurations made in Application Designer apply to that application only. For example, if you want to restrict access to a field that is in the header section of multiple tabs, you can put a data restriction on the field every instance of the field has the access restriction. If you configure the restriction in Application Designer, you must apply the restriction to each instance of the field on each tab.

Creating conditions

You can create conditions in the Conditional Expression Manager. Then, you can apply the conditions to fields, tabs, and other user interface controls in applications to control how the UI looks and behaves. You can also apply conditions to security groups to control access to applications, controls, and data.

Conditions overview:

A condition is a clause or class that evaluates to true or false when evaluated against an object. You can create and maintain a library of conditions. You can apply the conditions to fields, tabs, and other user interface controls to control the presentation. You can also apply conditions to security groups to control access to data, applications, and controls.

There are two types of conditions:

Expression-type condition

An expression-type condition is a clause that evaluates data and returns a result of true or false. The expression syntax is similar to structured query language (SQL) expressions, but it supports substitution variables. Use expressions to define simple conditions.

Class-type condition

For class-type conditions, the condition logic is contained in Java classes that are stored in a file under the root product folder. You can create the class files that contain the condition logic. When the class-type condition is evaluated, it returns a true or false result. Use a class-type condition if the logic to be evaluated is too complex to be accomplished with a simple expression.

Conditions are created and managed in the Conditional Expression Manager. Describe each condition that you create to help you identify the conditions to apply to user interface elements or security groups.

You can use the Application Designer to apply conditions to user interface elements. You can also apply conditions in the Security Groups application to control access to applications and to actions within applications.

You can modify expression-type conditions. When you modify the condition, access to all data and to all user interface controls that use the condition is modified. You can determine whether a condition is applied to user interface controls, security groups, or data restrictions by checking the **reference count** field. If the **reference count** field is zero then the condition is not applied.

Creating conditional expressions and classes:

You can create conditional expression-type and class-type conditions. Then you can apply the conditions to control access to data and to user interface controls.

Procedure

1. In the Conditional Expression Manager, add a row and change the auto-generated **Condition** identifier to a meaningful identifier. Enter a complete description of the condition in the **Description** field. A meaningful identifier and a complete description help you identify a condition later.
2. Select the condition type and enter the condition.
   a. If you selected expression type, enter an expression, or click **Condition Expression Builder** to build an expression in the Condition Expression Builder.
   b. If you selected class type, enter the class file name. The file must be in a folder under the root of the product folder, for example `Maximo_home\maximo\samples`
3. Save the condition.

Related concepts:

"Conditional expression syntax"
You enter conditional expressions using a syntax similar to Structured Query Language (SQL) with some additional variables.

"Sample conditional class files" on page 45
Sample condition class files are stored in a folder under the root directory of the product installation. You can use the sample files when you create class-type conditions.

Conditional expression syntax:

You enter conditional expressions using a syntax similar to Structured Query Language (SQL) with some additional variables.

When you create a conditional expression, use a colon (:) to define a variable. The colon is used to avoid ambiguity when you create expressions that relate the current record to a different, specific record.

You can have subselect in an expression,

```
exists (select 1 from workorder where wonum=:wonum).
```

The first *wonum* is the *wonum* attribute on the work order object. At run time, the second *wonum* is replaced with the value of the *wonum* attribute for the current record.

Replacement variables

Syntax	Description	Comments
`:yes`	true	Logically true, 1 if stored in the database
`:no`	false	Logically false, 0 if stored in the database
`:&date&`	Current date	
`:&datetime&`	Current date and time	
`:&user&`	Logged in user	For example, if a user is signed in as Smith, `:owner.id=&user&` converts to `:ownerid='SMITH'`
`:&personid&`	Person ID of logged in user	For example, if a user is logged in as Smith, `:reportby=&personid&` converts to `:reportby='SMITH'`
`:&appname&`	Application name	For example, in the Work Order Tracking application, `:&appname& = WOTRACK` converts to `WOTRACK = WOTRACK`. This variable is useful for setting different behavior for different copies of an application.
`:&mboname&`	Name of the current business object	For example, in the work order object, `object = :&mboname&` converts to `object = WORKORDER`
`:&ownername&`	Name of the owner business object	For example, in the Work Order Tracking application, `:&ownername &.jobplan.priority> &ownername&.priority` converts to `workorder.jobplan.priority >workorder.priority`

Bind variables

Syntax	Description	Examples
`:<relationshipname> .<attrname>`	Value of an attribute of a related business object of the current business object	`:location.description`

Syntax	Description	Examples
`:&owner&.<attrname>`	Value of an attribute of the owner business object	When you apply a job plan to a work order, the system copies the priority of the job plan to the child work order. The condition can be: `:&owner&.jobplan.priority> :&owner&.priority`. In this example, the system copies this information if the job plan has a higher priority than the parent work order.
`:&owner &.<relationship_name> .<attrname>`	Value of an attribute of the related business object of the owner business object	See the example for `:&owner&.<attrname>`
`:&old_<attrname>`	The initial value from the database of the attribute	For example, if you change the value of a field from 1 to 2 to 3 to 4, the original value is 1.

Conditional expressions examples

- `:wostatus='APPR'`
- `:type='EM'`
- `:ownerid=:&user&`
- `:supervisor!=:&personid&`
- `:asset.assettype = 'IT' and :&personid&=:owner`
- `:reportby=:&personid&`
- `:assetspec.classstructureid = 122`
- `:po.poline.receivedqty=0`
- `:&owner&.jobplan.priority>:&owner&.priority`
- `:owner&.po.$old_description like '%Turbin%'`

Sample conditional class files:

Sample condition class files are stored in a folder under the root directory of the product installation. You can use the sample files when you create class-type conditions.

To use a sample condition class, enter the file name in the **Class** field in the Conditional Expression Manager application.

Sample condition classes

- Evaluates to true: `psdi.common.condition.AlwaysTrue`
- Evaluates to false:`psdi.common.condition.AlwaysFalse`

Applying conditional security

You can limit access to data and to elements of the user interface to users that have the right level of access.

Conditional security:

You can apply these conditions to security groups, to enforce security measures, for example, you can limit access to certain elements of the user interface to users with the right level of access.

In the condition library, you can define conditions, either as expressions or as custom class files. You can control access to applications and to controls in applications by applying conditions to security groups. The conditions are applied to signature options (SIGOPTION) that are then granted to the security group. Conditional access is granted in the security groups application. If a user is in multiple security groups, the highest level of access is granted when the security groups are joined.

The following are some examples of the types of conditional access that you can set:

- Give read-only access to the information that is shown in a field.
- Give read/write access to the information that is shown in a field.
- Give a user group read-only access to a specific field in an application.
- Give all members of a user group read/write access to an application.
- Hide a field or tab in an application from certain users.
- Grant access to application options in the **Select Action** menu or side navigation menu for a security group.
- Configure any property in a control for a group, such as making a control hidden, masked, read-only, or required.
- Configure other properties, such as color, label, and application link, to be different for on the user group that is accessing the UI.
- Show or hide a data attribute globally or for a security group.

Defining signature options for an application:

Signature options specify privileges for using applications, menu options, and toolbar items. You apply signature options to groups of users to control their access to data, to applications, and to controls. New applications are automatically assigned a READ signature option. You can add more signature options in the Application Designer.

Procedure

1. Open the application in the Application Designer.
2. Select the **Add Modify Signature Options** in the **Select Action** menu and click **New Row**.
3. Specify a signature option in the **Option** field.
4. Optional: Add additional signature options in the **Option Also Grants** field. Use commas to separate multiple entries if you want to enable more than one additional signature option. For example, if the signature option grants access to the People application, you can add signature options that grant access to a related application, such as the Users application.
5. Optional: In the **Option Also Revokes** field, specify any signature options that are revoked as a result of adding the primary option. For example, you add a signature option that allows read-only access to an application. You want to remove any write access to the application that users with the read-only

signature option have. To remove the write access, you specify the signature option that grants write access to the application in the **Option Also Revokes** field.

6. Save the signature options.

Removing a signature option from a UI control:

To remove a signature option from a user interface (UI) control, you must first delete all security groups that are associated with the signature option.

Procedure

1. In the Application Designer workspace, select the control and click **Control Properties**.
2. In the Control Properties window, click **Configure Conditional Properties**.
3. Delete all rows in the Security Groups section and click **OK**.
4. Delete the signature option in the Control Properties window.
5. Click **Save**.

Results

The default behavior of the UI control is restored.

Applying signature options to show user interface controls:

You can grant access to a user interface (UI) control by associating it with a signature option. Then, users who belong to the security groups that are associated with this signature option are granted access to the control. Security groups that are not associated with the signature option do not have access to the control.

About this task

Multiple controls can be associated with one signature option. Then, access to the controls can be granted or revoked as a group.

Procedure

1. In the Application Designer workspace, select the application that you want to configure.
2. Select **Add/Modify Signature Options** and create a signature option.
3. Select the control that you want to configure and click **Control Properties**. Select the signature option that you created and close the window.
4. Click **Save**.
5. In the Security Groups application, create a group or select an existing group.
6. In the Options for Actions section of the **Applications** tab, select the **Grant Access** check box for the access types that you want to provide to the application.
7. Select the **Grant Access** check box for the signature option that you applied to the control in step 2.
8. Click **Save**.

Results

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

Showing controls to some users:

A user can be a member of many security groups. A user can have read-only access to an application in one group. As a member of another group the same user can have conditional access to the application. When the security groups combine, the user can access the application when the condition is true. When the condition is false, the user cannot access the application.

Procedure

1. In the Application Designer workspace, select the control and click **Control Properties**.
2. Select a signature option and close the window.
3. Click **Save**.
4. In the Conditional Expression Manager application, create a condition that grants access to a control when the condition is true.
5. In the Security Groups application, select the security group that contains all users. For example, select the MAXEVERYONE security group.
6. Click the **Applications** tab and filter for the application that you want to apply the condition to.
7. In the signature options row, check **Grant Access**, and specify the identifier of the condition you created in Step 4 in the **Condition** field.
8. On the **List** tab, select another security group. For example, select the PROJECTMANAGERS security group.
9. In the**Applications** tab, check **Grant Access** in the Read Access to Actions row.
10. Add the same user to both security groups.

Results

You have one group of users, PROJECTMANAGERS that has read-access to the application. The security group that contains all users, MAXEVERYONE, has conditional access to the control. When the two groups are combined, users have access to the control when the condition is true. Users who do not have read-access to the application in any group do not have access to the control.

Applying conditional data restrictions

To support your business processes, you can apply conditions to restrict access to data for a user or for a user group. The data can be hidden, read-only, or required.

Conditional data restrictions:

Data restrictions can be conditional or unconditional. You can restrict access to data for a user or user group. The data can be hidden, read-only, or required.

You can use these methods to restrict a user's access to data:

- Database configuration of objects and attributes
- Data restrictions on objects and attributes
- Application configurations

Data restrictions are configured in the Security Groups application. You can also set data restrictions for attributes in objects, either with or without a specified application. Data restrictions that are applied to an object do not apply to views of the object. You must create a separate restriction for the object view.

You can apply conditions to objects to hide them, to make them required, or to make them be read-only. You can apply the condition for every user, or just to a particular security group. Objects or objects within the context of an application can also be associated with a condition to determine what data is shown in the application. Only data that satisfies the condition is returned when the database is queried.

You can also apply conditions to attributes within objects, either with or without an associated application. When you view records in an application, the evaluation of conditions can change the presentation of controls bound to objects or attributes with conditional restrictions applied.

You associate objects or attributes with conditions in the Security Groups application.

If you apply multiple conditions, the conditions are joined by an OR operator when they are evaluated.

You can apply two or more conditions joined by an AND operator to an object or attribute. You can append the second and subsequent conditions to the first one with an AND function between them.

As an application developer, you are implementing a data restriction that causes information about an asset to be shown on the UI. The asset is shown if the following conditions are met:

- The asset is a calibration asset
- The asset has asset meter records

You create two conditions in the Condition Expression Manager:

- Condition 1 displays calibration-related fields for the asset if it is a calibration asset:
  ```
  iscalibration=1:
  ```
- Condition 2 checks for any asset meter records:
  ```
  exists (select 1 from assetmeter where assetnum = :assetnum and orgid =:orgid)
  ```

You append Condition 2 to Condition 1 with an AND operator:

```
iscalibration=1:and exists
(select 1 from assetmeter where assetnum = :assetnum and orgid =:orgid)
```

You apply the appended condition to the required asset object or attribute.

Related tasks:

"Specifying data restrictions for security groups"
To customize your security settings, you can specify restrictions on which records a security group can access. You can use a conditional expression or a conditional class file to define and apply these restrictions.

"Specifying global data restrictions for security groups" on page 50
Using global data restrictions, you can use a condition to specify restrictions on which records can be accessed by security groups.

Specifying data restrictions for security groups:

To customize your security settings, you can specify restrictions on which records a security group can access. You can use a conditional expression or a conditional class file to define and apply these restrictions.

Procedure

1. In the Security Groups application, select the group for which you want to set restrictions.
2. On the **Data Restrictions** tab, select the type of restriction:
   - To specify restrictions on objects, click **Object Restrictions**.
   - To specify restrictions on attributes, click **Attribute Restrictions**.
3. Click **New Row**.
4. In the **Object** field, specify the table or view on which to set the restriction.
5. Optional: If you are specifying an attributes restriction, specify the attribute that you want to restrict.
6. Optional: In the **Application** field, specify the application to which you are applying the restriction. Leave the field blank to apply the restriction to all applications that use the object or attribute.
7. Specify the type of restriction.
8. Optional: Specify the following options for restrictions:

Option	Description
Reevaluate	Select this option for the restriction condition to be reevaluated when the user tabs to another field. If you do not select this option, the restriction conditions are evaluated after the changes to a field are saved.
Condition	Specify a conditional expression.

9. Save your changes.

Related concepts:

"Conditional data restrictions" on page 48
Data restrictions can be conditional or unconditional. You can restrict access to data for a user or user group. The data can be hidden, read-only, or required.

Specifying global data restrictions for security groups:

Using global data restrictions, you can use a condition to specify restrictions on which records can be accessed by security groups.

Procedure

1. Optional: In the Conditional Expression Manager application, create one or more conditions to be evaluated to control access.
2. In the Security Groups application, select the **Global Data Restrictions** action.
3. In the Global Data Restrictions window, select the type of restriction, and specify the details of the restriction.

Related concepts:

"Conditional data restrictions" on page 48
Data restrictions can be conditional or unconditional. You can restrict access to data for a user or user group. The data can be hidden, read-only, or required.

Adding conditions to the properties of UI fields

You can control how the user interface (UI) is shown to users and to support your business processes. When a control is bound to a signature option or an attribute, you can configure conditional properties in the Application Designer

Adding conditions to determine access to UI controls:

You can conditionally modify the default access to user interface (UI) controls. UI controls include text boxes, check boxes, and radio buttons.

Procedure

1. In the Conditional Expression Manager application, create one or more conditions that grant access to a UI control when the condition is true.
2. In the Security Groups application, create or identify the security groups that can access the UI control.
3. In the Application Designer workspace, select **Add/Modify Signature Options**, and create a signature option.
4. Select the UI control that you want to configure, and select **Control Properties**.
5. Click **Configure Conditional Properties** and specify the following options:

Option	Description
Signature Option	Specify the signature option you created for the control.
Security Groups	Specify the groups that you want to provide with access to the control.
Conditions for a Security Group	Specify the conditions you created in step 1. The conditions apply only to the security groups that you specified.
Property Values for Condition	Specify the names and values of the properties to be controlled by the condition.

6. In the Security Groups application, grant the signature option to the security group.

Results

The users in the security groups that you specified can see the UI control when the conditions you apply are true. Other users cannot see the UI control.

Adding conditions to hide UI controls:

You can create and apply a signature option to a user interface (UI) control to deny access based on satisfying a condition. When the condition is true, access is granted to all of the controls who are associated with the signature option. Conditional access is granted to all users who belong to the security groups that are associated with the signature option.

Procedure

1. In the Conditional Expression Manager application, create one or more conditions that denies access to a UI control when the condition is true.
2. In the Security Groups application, create or identify the security groups that cannot access the UI control.
3. In the Application Designer workspace, select **Add/Modify Signature Options**, and create a signature option.
4. Select the UI control that you want to configure, and select **Control Properties**.
5. Click **Configure Conditional Properties** and specify the following options:

Option	Description
Signature Option	Specify the signature option you created for the control.
Security Groups	Specify the groups that you want to hide the UI control from.
Conditions for a Security Group	Specify the conditions you created in step 1. The conditions apply only to the security groups that you specified.
Property Values for Condition	Specify the names and values of the properties to be controlled by the condition.

6. In the Security Groups application, grant the signature option to the security group.

Results

The users in the security groups that you specified do not see the UI control when the condition you apply is true. Users in other security groups can see the UI control.

Application developer properties

Each control has its own set of properties that you can set. You can also set user interface properties to control the behavior of many aspects of user interface behavior.

Control properties

You can configure the properties for each individual user control in an application user interface.

Application bar control

The application bar (or appbar) control, is a menu bar that lists all applications in a module except the currently active application. An application bar is always open at the top of the window in a self-service application.

You can place an application bar inside the client area control. You can also place this control in a section control if you use the correct layout options.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	Specify a text value in this field if you want the application bar to have a label. This label is specific to the application and is not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Appbarlinks	Specify the links to add to the application bar.

Attribute name	Description
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Attachments control

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

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	You can specify a text value in this field to override the default label for the control, which is Attachment. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Blank line control

A blank line control inserts vertical white space in a section. Blank lines provide spacing between controls to improve readability or to visually separate different types of controls within a section.

A blank line control functions as a graphic element, is transparent in the user interface, and does not have configurable properties. Place a blank line control inside a section container control.

Breadcrumbs control

You can use the breadcrumb control to enhance navigational functionality in an application. Breadcrumb navigation tracks the steps you take in an application and helps you to retrace steps quickly. You can use a breadcrumb control by itself or in conjunction with another control, such as a tree control. You can place this control inside any container control.

A breadcrumb trail complements a hierarchical tree structure by showing a horizontal representation of where the user is in a menu structure or a service catalog. You can configure a delimiter character to separate items in the breadcrumb trail, such as the greater than character (>) or or a colon (:). Each phrase or identifier between delimiters represents one step in a hierarchy.

To synchronize a tree control and a breadcrumb controls, copy the **Data Source ID** property from the tree control to the breadcrumb control. The following table describes how a tree control and breadcrumb control behave in the user interface when they are used in combination.

When you click . . .	**This happens**
The label of any node	A new breadcrumb trail replaces any existing breadcrumb trail and: • Highlights the node label in the tree • Displays the node label in breadcrumb • All levels in the tree hierarchy above the highlighted node label appear in the breadcrumb (in descending order from left to right)
The plus sign in a tree	No effect.
The minus sign in a tree	Highlights the right-most breadcrumb, which is the same as the node label next to the minus sign.
The dot sign in a tree	The corresponding node label highlights in the tree and displays in the breadcrumb. All levels in the tree hierarchy above the highlighted node label appear in the breadcrumb (in descending order from left to right).
Any node label in the breadcrumb	Equivalent to clicking the node label in the tree. The tree contracts to the level of the node label and any breadcrumb links to the right of the clicked node label disappear.
A delimiter in the breadcrumb	No effect.

You can expect the following user interface behavior when navigating a tree or breadcrumb:

- Any time you highlight a node label in a tree or breadcrumb, the window displays information related to that label.
- Clicking a plus sign in a tree expands the node, but does not affect window content, label highlighting, or the breadcrumb.
- Clicking a minus sign in a tree does not affect window content, but does highlight the node label associated with the minus sign and the breadcrumb reflects the new hierarchy ending with the highlighted node.

- The highlighted node label in a tree always reflects the right-most element in the breadcrumb. The breadcrumb elements to the left reflect the intervening levels in the tree hierarchy.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	You can specify a text value to use as a label for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Bean Class	You can specify the name of a data bean class to use when creating the data source for the control. If not defined and the control is creating a data source, the default psdi.webclient.system.DataBean class is used. Typically, you use this property to create custom event methods or to override the manner in which the data source obtains its MboSet.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application.
Business Object	Specify the name of a MBO in this field to determine which MBO the control uses to show and process information. If a MBO is not specified, the control uses the data source of its parent control. When you set a MBO for this control a data source is created that can be referenced from other controls.
CSS Class Name	If you want to change the layout and appearance of the hyperlink, enter the name of the CSS class to use. For example, the “powerwhite” class name changes the font color of a hyperlink to white. You can search for CSS class names in the `<root>\applications\maximo\maximouiweb\webmodule\webclient` directory.
Separator	Specify the delimiter to use between breadcrumb trail entries, for example >.

Attribute name	Description
Relationship	If you specified a value in the Parent Data Source ID field, specify a value in this field that identifies the relationship between this control and the new data source. A relationship uses a WHERE clause to determine what to show and process from the data source. The control creates a datasource that is by default used by the child controls. You can search for relationship values in the **Database Configuration** > **Relationships** tab.
Breadcrumbs	The source ID of the breadcrumb control.
Source Method	Specify a source method to apply to the source
Align	Specify the alignment options for the breadcrumbs.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Button group control

A button group is a container control that holds push button controls. The button group defines the layout for the buttons it contains, including alignment within the container and the title for the group. Place this control within a section control or a table control.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	You can specify a text value in this field to provide a label for the button group. Most button groups do not have a label, because the control functions primarily as a container that holds a layout for its children push button controls. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.

Attribute name	Description
Text Alignment	If you specified a label for the control, you can set the label to align to the **Left**, **Center**, or **Right**. The text label always displays to the left of any push buttons you add to the group.
Button Alignment	Reserved for future use.
Show Container	If you select this property, a shaded background of horizontal lines is visible for the button group. You cannot show background shading for a button group control in a table container.

Check box control

With a check box control, a user can indicate a yes or no response to a question or select one or more items from a list. A check box is a text type control because it has a text field or a label that is visible to the user. Place this control inside a section control.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the label associated with the database attribute bound to this control. The label for a check box control ends with a question mark, for example **GL Account?**.
Label	You can specify a text value in this field to override the **`Default Label`** for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Attribute	You can associate this control with a database attribute. Not all attributes have a **`Default Label`** value.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application.

Attribute name	Description
Input Mode	Input mode options are: - `Blank:` An input mode with no value is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. - `Default:` This mode is used for quick entry of default table data in tables. - `Query`: This mode is read-only and shows a typed value. - `Readonly`: This mode is read-only and shows grayed-out text. - `Required`: This mode is read/write and the record cannot be saved unless the user enters a value in the field.

Advanced properties

Attribute name	Description
Data Change Event	You can configure a control to perform one of the following actions in the event of a data change to the control: - The `REFRESHTABLE` event refreshes the data source table for the control. - The `RESETCHILDREN` event deletes any child objects of the data bean of the control which then reacquires its `MBOSetRemote` references.
Always Synchronous	If deselected, the control can communicate with the database without pausing user activities. If cleared, the user cannot perform activities in other fields until validations for this field are completed. Synchronous activity is disabled by default. If there are dependencies between values specified for this control and other controls in the application, you can enable synchronous activity for the dependent controls.
Notify Portlets	If selected, this control can send notifications to a portlet, if communication with a Tivoli Integration Portlet is configured.

Attribute name	Description
More information	You can specify the values to construct a URL for an Knowledge Center topic using the following format: `com.ibm.mbs.doc,application/helpfile.html` The server address and other information for the URL are controlled by the mxe.help properties which you can review and configure in the System Properties application. If you specify a URL, a More Information link is added to the field help (Alt + F1) for the control. When the user clicks the link, the target topic opens in the Knowledge Center. The property is specific to the field ID in the presentation XML file. If you want to provide the same link in other controls or other applications, you must manually add the URL to each individual control.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Combo box control

With a combo box control, a user can select one or more options from a list drop down menu. Place this control inside a section control.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the label associated with the database attribute bound to this control.
Label	You can specify a text value in this field to override the **`Default Label`** for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.

Attribute name	Description
Attribute	You can associate this control with a database attribute. Not all attributes have a **Default Label** value.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, MAINRECORD is the data source because it is the primary data source for the application.
Display Attribute	Specify the attributes to show as options in the drop down menu for the combo box control. For example, if you create a combo box called Status, you can specify values such as **Waiting for Approval** and **Approved**. The values are bound to the data source and the attribute specified for the control.
Input Mode	Input mode options are: • Blank: An input mode with no value is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. • Default: This mode is used for quick entry of default table data in tables. • Query: This mode is read-only and shows a typed value. • Readonly: This mode is read-only and shows grayed-out text. • Required: This mode is read/write and the record cannot be saved unless the user enters a value in the field.
Width	The standard width for a combo box control is 120 pixels. To make the control narrower or wider, enter a numeric value, for example 60 (narrower) or 200 (wider).

Advanced properties

Attribute name	Description
Select Attribute	This value provides a synonym or abbreviation for the **Display Attribute** in the text box area of the combo box. For example, for the Status combo box, you can specify WAPPR and APPR as Select Attributes abbreviations for the Waiting for Approval and Approved display attributes.

Attribute name	Description
Data Change Event	You can configure a control to perform one of the following actions in the event of a data change to the control: • The REFRESHTABLE event refreshes the data source table for the control. • The RESETCHILDREN event deletes any child objects of the data bean of the control which then reacquires its MBOSetRemote references.
More information	You can construct a URL for an Knowledge Center topic using the following format: com.ibm.mbs.doc,application/helpfile.html If you construct a URL, a more information link is added to the field help (Alt + F1) for the control. When the user clicks the link, the target topic opens in the Knowledge Center. The property is specific to the field ID in the presentation XML file. If you want to provide the same link in other controls or other applications, you must manually add the URL to each individual control.
Notify Portlets	Reserved for future use.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Data source control

Use a data source control to override the default data source, or the current data source, for an application. This control is used primarily in dialog windows to reference a business object that controls in the window can use as a data source. A data source control is transparent and does not render any visible element.

You can place a data source control inside the following controls:

- Client area
- Dialog
- Page
- Section
- Tab

- Table

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Bean Class	You can specify the name of a data bean class to use when creating the data source for the control. If not defined and the control is creating a data source, the default psdi.webclient.system.DataBean class is used. Typically, you use this property to create custom event methods or to override the manner in which the data source obtains its MboSet.
Business Object	Specify the name of a MBO in this field to determine which MBO the control uses to show and process information. If a MBO is not specified, the control uses the data source of its parent control. When you set a MBO for this control a data source is created that can be referenced from other controls.
Parent Data Source ID	You can specify the ID of a data source to use as the parent object for a relationship, or to specify a parent object for a bean class if necessary. Then specify the relationship that uses this parent in the Relationship field. If the field is left blank, the control uses the data source ID of its parent control as the Parent Data Source ID
Relationships	If you specified a value in the Parent Data Source ID field, specify a value in this field that identifies the relationship between this control and the new data source. A relationship uses a WHERE clause to determine what to show and process from the data source. The control creates a datasource that is by default used by the child controls. You can search for relationship values in the **Database Configuration** > **Relationships** tab.
Order By	You can specify the order in which a SQL query fetches data for the data source for this control.
Where Clause	You can specify a WHERE clause that the data source for this control uses to retrieve data. This WHERE clause is used in addition to any default WHERE clause used by the business object of the data source.
Listeners	Enter the ID or IDs of one or more data sources that you want to refresh whenever there are changes to this control. Listener controls can be located anywhere within the application. Separate multiple values with a comma.

Default value control

You can use a default value control to speed up a database insert or query by restricting the action to a limited set of records. The default value control is transparent and does not render anything visible in the user interface. You can place the default value control in a section control, a table control, or in the presentation control.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Attribute	Select an attribute from a data source to update or query. You can set an update or query value using one of the following methods: • Specify a static value in the **Value** field. • Specify values in the **From Data Source ID** and the **From Attribute** fields to read a value from the database.
Value	The alphanumeric value in this field provides a static default value for an insert or query action for the specified **Attribute** value. In the Person application, for example, you can specify DEPARTMENT in the **Attribute** field and MAINTENANCE in the **Value** field, to restrict queries to people working in the maintenance department.
From Data Source ID	If you do not configure a static default value in the **Value** field, specify the data source ID to use as the data source for the **From Attribute** field.
From Attribute	If you do not configure a static default value in the **Value** field, specify a **From Attribute** that updates or queries the database with its value.
Default Type	Specify the appropriate type of action for this default query.

Help grid control

A help grid control contains HTML text that you can insert in the user interface, typically to provide brief instructions on how to enter data and perform tasks. For complex information, help grids can include hyperlinks to additional information in a Knowledge Center topic. You can place help grid controls within Section, Section Column, Tab, and Dialog controls.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.

Attribute name	Description
Label	You can specify a text value in this field if you want to associate a label with this control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
HTML Content	You can add text and a More Information link to a help grid control. A help grid control can contain HTML text for the user interface, typically to provide brief instructions about how to enter data and perform tasks. For complex information, help grids can include hyperlinks to additional information in a Knowledge Center topic. You can place help grid controls within Section, Section Column, Tab, and Dialog controls: 1. In Application Designer, select the help grid control to modify. Modify the HTML Content attribute to add display text. 2. Optionally, link from that text to a Knowledge Center topic by specifying the URL for the topic in the morehelp attribute. The URL is in this format: `<documentation plugin name>,<path and filename.extension>`. For example, a link from an assets tab to an overview topic about assets could take this form: `com.ibm.mam.doc,asset/c_asset_overview.html`.
More information	You can add a More information button and link to field help. A more information button and link in field help can provide technical or contextual details that do not fit in field help. 1. In Application Designer, select the control that you want to modify, such as a field or check box. 2. In the More information property, specify the URL to the help topic. The URL is in this format: `<documentation plugin name>,<path and filename.extension>`. For example, a link from an asset check box to a topic about classifying assets could take this form: `com.ibm.mbs.doc,assetcat/c_classify_item_desc.html`.
Signature Option	You can specify a signature option to restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click **Configure Conditional Properties**.

Attribute name	Description
Sig Option Data Source ID	A MBO must be used to evaluate whether a user has access to a signature option. Specify the value of the data source if it is different to the data source that is configured for this control.

Hover window control

A hover window is a read-only pop-up window that is applied to a user interface control to display more information about an attribute.

Properties

Attribute name	Description
Hover Window ID	Can be applied to the multiline text box, multipart text box, table column, and text box controls. At runtime, a circular icon with an i in the center indicates that a hover window is enabled for a control. Available hover windows are defined in the `RECHOVERS.XML` file in the system library. To apply one of these hover windows to a control, you specify the ID of the hover window as defined in the `RECHOVERS.XML` file.
Hover Window Attribute	By default, a hover window uses the data attribute of the field. Use this attribute when you want the hover window to use a different attribute than the data attribute of the field.

Related tasks:

Configuring related information for fields
To see information related to a field without leaving the record, you add a hover window to the field. Hover windows are immediately enabled on some fields but you can also enable these existing hover windows on other fields or you can create a hover window depending on your requirements.

Hyperlink control

You can configure text or a graphic as a hyperlink that opens a menu or window, or a destination inside or outside the application. Place this control inside a section control.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	If you specify a text value, it forms a link that the user clicks to initiate the hyperlink.

Attribute name	Description
Event Type	You can associate a hyperlink with an event and, when the user clicks the hyperlink, the event is triggered. To configure a hyperlink event, you must specify the event type, the event value, and, optionally, the ID of a control that processes the event. For example, to configure an event that opens the **Go To** menu, specify**showmenu** in the **Event Type** field, specify **goto** in the **Event Value** field, and specify**pageTitlebar** in the **Control Target ID** field. Events are processed in the following order: 1. If you specify a control target ID, the event is sent to this data source for processing. 2. If you do not specify a control target ID, the event is sent to the main JSP page of the application for processing. 3. The window associated with the specified event loads. To view all of the events defined in the system, export the system .XML files and search these files for `MXEVENT` entries.
Control Target ID	If you associate an event with the hyperlink, you can specify the ID of a target control to process the event that is different from the current control. The target ID for the Go To hyperlink, for example, is pageTitlebar.
Event Value	If you associate an event with the hyperlink, specify the value of the event. The event value for the Go To hyperlink, for example, is goto.
Text Alignment	Aligns the hyperlink label to the left, right, or center within a section.
Image Filename	You can associate an image with the hyperlink. Specify the name of an image file that is stored in the `<root>\applications\maximo\maximouiweb\web module\webclient\images` directory. It is not necessary to enter the path to the images directory.
Image Alignment	If the hyperlink includes an image, aligns the image to the left or right of the hyperlink label.
CSS Classname	If you want to change the layout and appearance of the hyperlink, enter the name of the CSS class to use. For example, the "powerwhite" class name changes the font color of a hyperlink to white. You can search for CSS class names in the `<root>\applications\maximo\maximouiweb\webmodule\webclient` directory.

Attribute name	Description
Access key	Specify a letter that is used in the hyperlink label that acts as a keyboard shortcut. For example, the access key for the Go To hyperlink is G and when the user presses ALT + G the **Go To** menu drops down. The access key is underlined in the user interface.
Separator	Reserved for future use.

Advanced properties

Attribute name	Description
Column	You can specify a number that positions a hyperlink in a specific column. Each section control has 7 columns that are not visible to the user. If you specify the value 3 in this field and set text alignment to center, the hyperlink is centered above or below column three. If the hyperlink is wider than its column, the width of the column extends automatically to contain the hyperlink.
URL	If you want the hyperlink to link to a URL, specify the value here.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Icon menu control

The icon menu control displays menu items next to a text box or multipart text box. You can use the `mxe.webclient.iconsToDisplay` system property to limit the number of menu items to display but the default is three.

Properties

Attribute name	Description
Icon Menu ID	Displays detail menu items as buttons next to a text box or the first part of a multipart text box.
Description Icon Menu ID	Displays detail menu items as buttons next to the second part of a multipart text box.

Related tasks:

Adding buttons next to text boxes
You can add buttons next to text boxes for the menu items that are most frequently used. You can add up to three buttons to a text box.

Image control

You can add an image control to create a visual marker to identify an application you can open or an action you can perform. You can configure the height, width, and alignment for the specified image file. You can place an image control with a section control.

Properties

The image files provided as application icons are in the `<root>\applications\maximo\maximouiweb\webmodule\webclient\images` directory. You can access the files in this directory using the image control.

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Filename	You can specify the filename of any file stored in the images directory at `<`*`root`*`>\applications\maximo\maximouiweb\webmodule\webclient\images`. You can reuse any existing images or you can add new images to the directory. Images can be in any format supported by a browser. When you add a .gif file, you can use it immediately. If you add other file formats, such as .jpg or .bmp files, you must redeploy the .EAR file on the application server to add the images to the user interface.
Height	Enter a numeric value in this field that will increase or decrease the height of the image. If you change the default dimensions, the image can become distorted.
Width	Enter a numeric value in this field that will increase or decrease the width of the image. If you change the default dimensions, the image can become distorted.
Align Image	Specify where to align the image, to the Left, Center, or Right of the section that contains it.
Column	You can use columns to manipulate the position of an image. A section has four invisible columns. You can specify a number (1-4) to position an image in the corresponding column. You can change the position of the image by moving the graphic into one of the columns.

Attribute name	Description
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Include control

You can use include controls to reuse dialog windows, layouts, and data tables from an existing application in the presentation.xml file of a new application. You can place an include control within a page, tab, or section control.

The include control is used with the presentation, page, and client area controls to create the presentation.xml file for an application. Include controls are typically used to obtain common application functions like header and footer from the `LIBRARY.XML` file. You can also use an include control within the client area to, for example, obtain table definitions for application log files from the `LIBRARY.XML` file.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Control ID to Clone	You can specify any Control ID from the library.xml file. This property enables you to include the same library element multiple times in a presentation.

List box control

A list box control displays a list of items in a box that is always open. Place this control inside a section control.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the text value that is associated with the database attribute bound to this control.
Label	You can specify a text value in this field to override the **`Default Label`** for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.

Attribute name	Description
Attribute	You can associate this control with a database attribute. Not all attributes have a **`Default Label`** value.
Business Object Name	Specify the name of a MBO in this field to determine which MBO the control uses to show and process information. If a MBO is not specified, the control uses the data source of its parent control. When you set a MBO for this control a data source is created that can be referenced from other controls.
Parent Data Source ID	You can specify the ID of a data source to use as the parent object for a relationship, or to specify a parent object for a bean class if necessary. Then specify the relationship that uses this parent in the Relationship field. If the field is blank, the control uses the data source ID of its parent control as the Parent Data Source ID.
Key Attribute	You can specify a different attribute of the business object to use as a key attribute for the list other than the key attribute specified for the MBO.
Key Value	You can specify an option in the list to display as the default value.
Input Mode	Input mode options are: • `Blank:` An input mode with no value is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. • `Default:` This mode is used for quick entry of default table data in tables. • `Query`: This mode is read-only and shows a typed value. • `Readonly`: This mode is read-only and shows grayed-out text. • `Required`: This mode is read/write and the record cannot be saved unless the user enters a value in the field.
Height	To make the control shorter or longer, enter a numeric value.
Width	To make the control narrower or wider, enter a numeric value.

Advanced properties

Attribute name	Description
Bean class	You can specify the name of a data bean class to use when creating the data source for the control. If not defined and the control is creating a data source, the default psdi.webclient.system.DataBean class is used. Typically, you use this property to create custom event methods or to override the manner in which the data source obtains its MboSet.
Where Clause	You can specify a WHERE clause that the data source for this control uses to retrieve data. This WHERE clause is used in addition to any default WHERE clause used by the business object of the data source.
Order By	You can specify the order in which a SQL query fetches data for the data source for this control.
Auto-Refresh	If selected, information in this control refreshes automatically when there is a change to the business object.
Always Synchronous	If deselected, the control can communicate with the database without pausing user activities. If cleared, the user cannot perform activities in other fields until validations for this field are completed. Synchronous activity is disabled by default. If there are dependencies between values specified for this control and other controls in the application, you can enable synchronous activity for the dependent controls.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Menu bar control

The menu bar control is similar to an application bar control but the menu options typically contain drop down menus with additional option choices. The menu bar at the top of the List tab in most applications is a typical example. You can place a menu bar control only in a tab control.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	You can specify a label for the menu bar. The label is not visible in the user interface. This label is specific to the application and is not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Source Method	Specify the name of the method used in the code to call the menu bar, including its signature options, and layout information. The method that calls the standard search menu bar is getAppSearchOptions.
Event Value	Specifies 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 text box control

A multiline text box control is a large text box that can display several lines of text or accept this text from user input. Text boxes are often linked to select value lookups and detailed menus. You can place a multiline text box control within a section control.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the text value that is associated with the database attribute bound to this control.
Label	If you enter a text value in this field, it overrides the **Default Label** for the control. This label is specific to the application and is not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Hide Label	By default, the label is visible in the user interface. Select this check box to hide the label.
Attribute	You can associate this control with a database attribute. Not all attributes have a **Default Label** value.
Rows	Enter the number of rows you want in the multi line textbox.
Columns	Most multiline text boxes have a default width of 35. To increase or decrease the width of the box, enter a different value.

Attribute name	Description
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application.
Lookup	You can specify a lookup ID to link this control to a pre-formatted lookup table that pulls records from the main table. A user can select a record from the lookup table to use in this control. Lookup IDs are stored in the `lookups.xml` file. To view the content of this file, export the system XML files.
Input Mode	Input mode options are: • `Blank:` An input mode with no value is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. • `Default:` This mode is used for quick entry of default table data in tables. • `Password`: This mode is read/write and displays asterisks. • `Password Readonly`: This mode is read-only and displays asterisks. • `Password Required`: This mode is read/write and displays asterisks. The record cannot be saved unless there is a value in the field. • `Query`: This mode is read-only and shows a typed value. • `Readonly`: This mode is read-only and shows grayed-out text. • `Required`: This mode is read/write and the record cannot be saved unless the user enters a value in the field.
Menu Type	You can specify the value of a menu ID in this field to link the control to a preformatted drop-down menu. Menu IDs are stored in the `menus.xml` file. To view the content of this file, export the system XML files.

Advanced properties

Attribute name	Description
Data Change Event	You can configure a control to perform one of the following actions if there is any data change to the control: • The REFRESHTABLE event refreshes the data source table for the control. • The RESETCHILDREN event deletes any child objects of the data bean of the control which then reacquires its MBOSetRemote references.
Always Synchronous	When deselected, the control can communicate with the database without pausing user activities. If selected, the user cannot perform activities in other fields until validations for this field are completed. Synchronous activity is disabled by default. If there are dependencies between values specified for this control and other controls in the application, you can enable synchronous activity for the dependent controls.
Notify Portlets	If selected, this control can send notifications to a portlet, if communication with a Tivoli Integration Portlet is configured.
More information	You can specify the values to construct a URL for a Knowledge Center topic using the following format: `com.ibm.mbs.doc,application/helpfile.html` The server address and other information for the URL are controlled by the mxe.help properties which you can review and configure in the System Properties application. If you specify a URL, a More Information link is added to the field help (Alt + F1) for the control. When the user clicks the link, the target topic opens in the Knowledge Center. The property is specific to the field ID in the presentation XML file. If you want to provide the same link in other controls or other applications, you must manually add the URL to each individual control.
Move to Data Source ID	Specify a record stored in a different object that the user can move to. You can select any data source ID from the presentation control. By default, the system sends the move to event to the ResultsBean, which contains a set of records displayed in the List tab. The move to event affects the Move to Data Source ID object.

Attribute name	Description
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Multipart text box control

A multipart text box contains two text boxes, positioned side-by-side, that can contain information from different sources. You can input information, populate a box by selecting a value from a list, or show an attribute description from a table. You can place a multipart text box within a section control.

Typically, a multipart text box has a visible label, a pull-down menu in one text box, and the description of an attribute in the second box. Labels describe the type of information displayed in the box. Text boxes are often linked to select value lookups and detailed menus. Most applications typically use the second part of a multipart text box to provide a description for the value contained in the first text box.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the text value that is associated with the database attribute bound to this control.
Label	If you enter a text value in this field, it overrides the **`Default Label`** for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Attribute	You can associate this control with a database attribute. Not all attributes have a **`Default Label`** value.
Attribute for Part 2	You can associate the second part of a multipart text box with a different attribute than the one specified in the **Attribute** field. Typically, you configure the second part of a multipart control to display the description information for the record selected in the first part.

Attribute name	Description
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application. In a multipart text box, the attributes for both parts must reference the same object.
Lookup	You can specify a lookup ID to link the first part of a multipart text box to a pre-formatted lookup table that pulls records from the main table. A user can select a record from the lookup table to use in this control. Lookup IDs are stored in the `lookups.xml` file. To view the content of this file, export the system XML files.
Lookup for Part 2	You can specify a lookup ID to link the second part of a multipart text box to a pre-formatted lookup table that pulls records from the main table. A user can select a record from the lookup table to use in this control. Lookup IDs are stored in the `lookups.xml` file. To view the content of this file, export the system XML files.
Go To Applications	You can associate a pull-down menu with this control and add application names to it that take the user to different applications.

Attribute name	Description
Input Mode	Input mode options are: • `Blank:` An input mode with no value is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. • `Default:` This mode is used for quick entry of default table data in tables. • `Password`: This mode is read/write and displays asterisks. • `Password Readonly`: This mode is read-only and displays asterisks. • `Password Required`: This mode is read/write and displays asterisks. The record cannot be saved unless there is a value in the field. • `Query`: This mode is read-only and shows a typed value. • `Readonly`: This mode is read-only and shows grayed-out text. • `Required`: This mode is read/write and the record cannot be saved unless the user enters a value in the field.
Input Mode for Part 2	In most cases, where the second part of a multipart text box contains a full description, the property value will be read-only. In other cases, standard input mode options can apply.
Menu Type	You can specify the value of a menu ID in this field to link the control to a preformatted drop-down menu. Menu IDs are stored in the `menus.xml` file. To view the content of this file, export the system XML files.

Advanced properties

Attribute name	Description
Change Event	You can configure a control to perform one of the following actions in the event of a data change to the control: • The `REFRESHTABLE` event refreshes the data source table for the control. • The `RESETCHILDREN` event deletes any child objects of the data bean of the control which then reacquires its `MBOSetRemote` references.

Attribute name	Description
Turn Smart Fill Off?	Check this option if you want to disable smart fill which is turned on by default. Smart fill enables a user to enter a partial value in a text box which the system attempts to match to a valid value, or values, in the database. When a matching value is found, the value is entered in the text box. If the system finds more than one matching value, the user can select one from a list of matching values. If you turn smart fill off, user text is not validated during entry.
Readonly Long Description	If the value of **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), users can edit the long description.
Always Synchronous	If deselected, the control can communicate with the database without pausing user activities. If selected, the user cannot perform activities in other fields until validations for this field are completed. Synchronous activity is disabled by default. If there are dependencies between values specified for this control and other controls in the application, you can enable synchronous activity for the dependent controls.
Part 2 Always Synchronous	If deselected, the control can communicate with the database without pausing user activities. If selected, the user cannot perform activities in other fields until validations for this field are completed. Synchronous activity is disabled by default. If there are dependencies between values specified for this control and other controls in the application, you can enable synchronous activity for the dependent controls.
Notify Portlets	If selected, this control can send notifications to a portlet, if communication with a Tivoli Integration Portlet is configured.

Attribute name	Description
More information	You can specify the values to construct a URL for a Knowledge Center topic using the following format: `com.ibm.mbs.doc,application/helpfile.html` The server address and other information for the URL are controlled by the mxe.help properties which you can review and configure in the System Properties application. If you specify a URL, a More Information link is added to the field help (Alt + F1) for the control. When the user clicks the link, the target topic opens in the Knowledge Center. The property is specific to the field ID in the presentation XML file. If you want to provide the same link in other controls or other applications, you must manually add the URL to each individual control.
Move to Data Source ID	Specify a record stored in a different object that the user can move to. You can select any Data Source ID from the presentation control. By default, the system sends the Move To event to the `ResultsBean`, which contains a set of records displayed in the List tab. The Move To event affects the Move to Data Source ID object.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Parameter value control

A parameter value control can display dynamic values in the label of a section control or a table control.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.

Attribute name	Description
Parameter Position	This property points to the Parameter Values Property field you want to replace. See the Parameter Values Property field for more information.
Attribute	You can associate this control with a database attribute to display as the Label property in a table or section control.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, MAINRECORD is the data source because it is the primary data source for the application.

Parameter values control

A parameter values control is a container control that can contain a parameter value to show a dynamic value that represents a replaceable value in the label of a Section Header control.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Property	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.

Presentation control

The presentation control manages the appearance of the user interface display and all other controls are contained by this control. The presentation control contains only one page control. The page control specifies the main page of an application and uses the entire client area of the browser. All other application windows are dialogs that open as popup windows. You can add the definition for a dialog window to the `PRESENTATION.XML` file or you can include the definition of an existing dialog from the `LIBRARY.XML` file.

Rules

The presentation control is the outermost control for all presentations and can contain all other controls.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.

Attribute name	Description
Business Object	Specify the name of a MBO in this field to determine which MBO the control uses to show and process information. If a MBO is not specified, the control uses the data source of its parent control. When you set a MBO for this control a data source is created that can be referenced from other controls.
App Bean Class	You can specify the name of a data bean class to use when creating the data source for the control. If not defined and the control is creating a data source, the default psdi.webclient.system.DataBean class is used. Typically, you use this property to create custom event methods or to override the manner in which the data source obtains its MboSet.
Order By	You can specify the order in which a SQL query fetches data for the data source for this control.
Where Clause	You can specify a WHERE clause that the data source for this control uses to retrieve data. This WHERE clause is used in addition to any default WHERE clause used by the business object of the data source.
Application Restrictions	You can set restrictions for the application at the level of the presentation.xml.
Results Table ID	You can specify the ID of the results table.
Application help	You can specify the URL that takes the user to context-specific help for this application in an Knowledge Center.

Advanced properties

Attribute name	Description
Viewport	If this application is optimized to be viewed in a viewport other than the standard viewport, specify that viewport in this field.
Is Mobile	A Boolean value that indicates whether the application is designed for access on a mobile device.
Always Synchronous	If deselected, the control can communicate with the database without pausing user activities. If cleared, the user cannot perform activities in other fields until validations for this field are completed. Synchronous activity is disabled by default. If there are dependencies between values specified for this control and other controls in the application, you can enable synchronous activity for the dependent controls.
Debug	You can set a debug level for the application.

Push button control

A push button control initiates an event such as opening a window with a list of records or opening an empty row where a user can enter a record. Place this control inside a button group control.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	You can specify a label that identifies the control to the user. If you do not enter a label, the button is blank. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Event	You can associate a push button with an event and, when the user clicks the button, the event is triggered. To configure a push button event, specify the event and, optionally, the ID of a control that processes the event. Events are processed in the following order: 1. If you specify a control target ID, the event is sent to this data source for processing. 2. If you do not specify a control target ID, the event is sent to the main JSP page of the application for processing. 3. The window associated with the specified event loads. To view all of the events defined in the system, export the system .XML files and search these files for `MXEVENT` entries.
Target ID	If you associate an event with the hyperlink, you can specify the ID of a target control to process the event that is different from the current control. The target ID for the Go To hyperlink, for example, is pageTitlebar.
Value	Reserved for future use.
Default Button	If selected, this button becomes the default button in a button group. You can configure the most frequently used button, for example, **New Row**, as the default button.
Menu Type	You can specify the value of a menu ID in this field to link the control to a preformatted drop-down menu. Menu IDs are stored in the `menus.xml` file. To view the content of this file, export the system XML files.

Advanced properties

Attribute name	Description
Image Filename	You can associate an image with the push button. Specify the name of an image file that is stored in the `<root>\applications\maximo\maximouiweb\web module\webclient\images` directory. It is not necessary to enter the path to the images directory.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Radio button control

A radio button is a control that allows users to choose one of a predefined set of options. Radio buttons are organized in groups of two or more. Radio buttons typically appear as little circles that are either empty (for unselected) or contain a dot (selected). When you select a radio button, the previously selected button in the same group is deselected.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the label defaults to the database attribute bound to the radio button group control.
Label	You can specify a label that identifies the control to the user. If you do not enter a label, the button is blank. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Button Value	Specify 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.

Attribute name	Description
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Radio button group control

A radio button group is a container control that holds radio buttons. The radio button group defines the layout for the buttons it contains, including a group title, text alignment for button labels, and whether or not to show a border. You can place a radio button group control within a section control.

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 the user makes 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, you can then edit the group in the Database Configuration application and in the Domains application.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the text value that is associated with the database attribute bound to this control.
Label	You can specify a text value in this field to override the **`Default Label`** for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Hide Label	By default, the label value always appears in the user interface. Select this check box if you want the label to be invisible.
Attribute	You can associate this control with a database attribute. Not all attributes have a **`Default Label`** value.

Attribute name	Description
Value Attribute	Optionally, if you do not automatically populate a radio button group with a list of buttons, you can specify a 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 can define a **Value Attribute** of WAPPR. When the user clicks the radio button, WAPPR is written to the database and Waiting for Approval is shown in the user interface.
Description Attribute	Optionally, you can specify a data attribute that displays a text label for the **Value Attribute** in the user interface.
Show Border	Select this check box to show a rectangular border around the control.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application.

Advanced properties

Attribute name	Description
Text Alignment	Select Left, Center, or Right to align button text relative to button images. The label for the radio button group always aligns to the left of any buttons that you add to the group.

Attribute name	Description
Input Mode	Input mode options are: - `Default:` The default input mode is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. - `Password`: This mode is read/write and displays asterisks. - `Password Readonly`: This mode is read-only and displays asterisks. - `Password Required`: This mode is read/write and displays asterisks. The record cannot be saved unless there is a value in the field. - `Query`: This mode is read-only and shows a typed value. - `Readonly`: This mode is read-only and shows grayed-out text. - `Required`: This mode is read/write and the record cannot be saved unless the user enters a value in the field.
Change Event	You can configure a control to perform one of the following actions in the event of a data change to the control: - The `REFRESHTABLE` event refreshes the data source table for the control. - The `RESETCHILDREN` event deletes any child objects of the data bean of the control which then reacquires its `MBOSetRemote` references.
Always Synchronous	If deselected, the control can communicate with the database without pausing user activities. If cleared, the user cannot perform activities in other fields until validations for this field are completed. Synchronous activity is disabled by default. If there are dependencies between values specified for this control and other controls in the application, you can enable synchronous activity for the dependent controls.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Record image control

You can use the record image control to associate an image with a record, including an expandable thumbnail image. You can then add this image to an application from within the application user interface. A user can click on a thumbnail image to view its full size and can drag the mouse over the image to see a text description. You can place a record image control in any container control.

When you add a record image to an application, you are inserting a placeholder but not the actual image. You can add only one record image to an application but you can use the same image multiple times within the application. For example, you can show an image in more than one application tab.

After creating a record image control, open the application and click the Add Modify Image option from the Select Action menu. If necessary, use an `autodb` script to add the Add Modify Image option to the Select Action menu item. You can add images that have a .jpg or .gif extension.

You can configure the record image control to view images associated with other records that are not part of the current application. Use View Image in the Details menu to open an image in a popup window. To add View Image to a Details menu, edit the `menu.xml` file. When you are finished, use Import Application Definition in the Application Designer to import the `menu.xml` file.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	You can specify a text value in this field to override the **`Default Label`** for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application.
CSS Class Name	You can specify the name of a CSS class to use if you want to change the layout or appearance of the record image. You can search for CSS class names in the `<root>\applications\maximo\maximouiweb\webmodule\webclient` directory.
Height	Enter a numeric value in this field that will increase or decrease the height of the image. If you change the default dimensions, the image can become distorted.

Attribute name	Description
Width	Enter a numeric value in this field that will increase or decrease the width of the image. If you change the default dimensions, the image can become distorted.
Is Thumbnail	Identifies whether or not the image is an expandable thumbnail image.
ALT Text on Image	You can specify a text description of the image that is visible when the user drags a mouse over the image. An ALT text description provides a text substitute for the image to users with a visual impairment that use screen readers.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Rich text editor control

The rich text editor control provides the ability to add rich text formatting to a field with long description characteristics.

General properties

The rich text editor control can be placed inside standard container controls, including sections, tabs, and tables. A rich text editor control provides users with a toolbar that helps them to apply rich text formats to the text entered in the text area. Rich text formatting is not appropriate for a field that is the subject of routine searches or for fields that contain a large quantity of information, because the addition of HTML tags to the text can exceed the character limit set for the field in the database.

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the text value that is associated with the database attribute bound to this control.
Label	If you enter a text value in this field, it overrides the **`Default Label`** for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.

Attribute name	Description
Hide Label	By default, the label value always appears in the user interface. Select this check box to make the label invisible.
Attribute	You can associate this control with a database attribute. Not all attributes have a **`Default Label`** value.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application.
Lookup	Specify a lookup ID to link this control to a pre-formatted lookup table that pulls records from the main table. A user can select a record from the lookup table to use in this control. Lookup IDs are stored in the `lookups.xml` file. To view the content of this file, export the system XML files.
Input Mode	Input mode options are: • `Blank:` An input mode with no value is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. • `Default:` This mode is used for quick entry of default table data in tables. • `Query`: This mode is read-only and shows a typed value. • `Readonly`: This mode is read-only and shows grayed-out text. • `Required`: This mode is read/write and the record cannot be saved unless the user enters a value in the field.
Menu Type	You can specify the value of a menu ID in this field to link the control to a preformatted drop-down menu. Menu IDs are stored in the `menus.xml` file. To view the content of this file, export the system XML files.
Width	You can enter a numeric value in this field to change the width of the editor area.
Height	You can enter a numeric value in this field to change the height of the editor area.

Advanced properties

<table>
<tr><th>Attribute name</th><th>Description</th></tr>
<tr><td>Plug-ins</td><td>The plug-ins that you specify determine the items that are displayed in the editor toolbar. The default value for plug-ins in the toolbar is
<code>['undo','redo','| ','cut','copy','paste','|</code>
<code>','bold','italic','underline','strikethrough','|</code>
<code>','insertOrderedList','insertUnorderedList','indent','outdent','|</code>
<code>','justifyLeft','justifyRight','justifyCenter','justifyFull']</code>
You can remove, reorder, or add additional plug-ins. To remove all these items from the toolbar, set the value to <code>[]</code>.
For more information about available plug-ins, go to the Dojo Toolkit website (http://www.dojotoolkit.org) and search for the reference guide.</td></tr>
<tr><td>Signature Option</td><td>You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click <b>Add/Modify Signature Options</b> in the <b>Select Action</b> menu. To specify a condition, click the <b>Configure Conditional Properties</b> button.</td></tr>
<tr><td>Sig Option Data Source ID</td><td>An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.</td></tr>
</table>

Rich text viewer control

The rich text viewer control provides the ability to view rich text formatting in a field with long description characteristics.

General properties

The rich text viewer control can be placed inside standard container controls, including sections, tabs, and tables.

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	If you enter a text value in this field, it overrides the **`Default Label`** for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Hide Label	By default, the label value always appears in the user interface. Select this check box to make the label invisible.
Attribute	You can associate this control with a database attribute. Not all attributes have a **`Default Label`** value.

Attribute name	Description
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application.
Width	You can enter a numeric value in this field to change the width of the viewer area.
Height	You can enter a numeric value in this field to change the height of the viewer area.

Section control

A section control is one of the core container controls that you use to organize the layout of an application. Container controls occupy space on a page and provide storage for other controls, such as tables, buttons, and text fields.

Rules

Section controls without labels do not have visible borders and are transparent in an application page. A section control with a label includes a visible horizontal blue bar with a minimize icon at the top of the section. You start all new application layouts by inserting a section container. Most applications have multiple pages with many sections and columns.

You can place the following controls in a section:

- Attachments
- Blank Line
- Button Group
- Checkbox
- Combobox
- Data Source
- Default Value
- Hyperlink
- Image
- Include
- Listbox
- Multiline Textbox
- Radio Button Group
- Section Header
- Section Row
- Static Text
- Textbox

You can place a section inside the following container controls:

- Dialogs
- Tab
- Table
- Section Columns

General properties

Attribute name	**Description**
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	This value identifies the control to the user. If you do not enter a label, the section does not have a title. The label you enter here is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Show Border	Select this option to add a horizontal rule to the top and bottom of the section. If you define a label for the section, the horizontal rules are not added.
Collapsed	If you check this box, the section is collapsed by default in the user interface and the user must click to maximize the section.
Description	You can specify text to show under the label in the title bar of the section.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application.
Business Object	Specify the name of a MBO in this field to determine which MBO the control uses to show and process information. If a MBO is not specified, the control uses the data source of its parent control. When you set a MBO for this control a data source is created that can be referenced from other controls.
Parent Data Source ID	You can specify the ID of a data source to use as the parent object for a relationship, or to specify a parent object for a bean class if necessary. Then specify the relationship that uses this parent in the Relationship field. If the field is left blank, the control uses the data source ID of its parent control as the Parent Data Source ID

Attribute name	Description
Input Mode	Input mode options are: • `Default:` The default input mode is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. • `Password`: This mode is read/write and displays asterisks. • `Password Readonly`: This mode is read-only and displays asterisks. • `Password Required`: This mode is read/write and displays asterisks. The record cannot be saved unless there is a value in the field. • `Query`: This mode is read-only and shows a typed value. • `Readonly`: This mode is read-only and shows grayed-out text. • `Required`: This mode is read/write and the record cannot be saved unless the user enters a value in the field.

Advanced properties

Attribute name	Description
Listeners	Enter the ID or IDs of one or more data sources that you want to refresh whenever there are changes to this control. Listener controls can be located anywhere within the application. Separate multiple values with a comma.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Section column control

A section column control is a vertical column that you can use to divide a section row into columns with headings. A section column is a container that can hold child section controls and is also a child to the parent section control. When you move a section control, all of its child controls 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 and is transparent in the user interface. A child section can hold, for example, a section header that lets you assign column headings for a multicolumn section. Except for tables, all graphical elements with multiple columns have a transparent section column within a section row, all contained within a parent section.

A section column control has only one property, the control ID property.

Section header control

Use a section header control to display a text heading for a section or a table. The text in the heading can be dynamic, based on variables you configure in parameter and parameter values controls. A section header control is a container control and is transparent in the user interface.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	The label contains a variable that displays dynamic values inherited from the parameter value and values controls in the section control.
Plain Text	If selected, the CSS (cascading style sheet) class definition for the label is removed so that it looks like plain text in the user interface.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Section row control

A section row control is a layout control that contains section column controls. A section row creates a horizontal row that you can partition into columns that, in , hold child sections. A section row is a child to a parent section control. When you move a section, all of its child controls automatically move with it.

You can insert section columns into section rows. A section row does not have a label and is transparent in the user interface. A section row control has only one property, the control ID.

Static text control

You can use a static text control to add text to a section control or to specify a database attribute that you can configure and position within a section as if it were unbound text. Static text is, typically, used in Date search windows in applications, where To and From labels are static text controls.

You can also configure the text to function as a hyperlink that takes the user to a URL when clicked. When configured as a URL, the static text displays in a larger font and is styled to look more like a URL string than plain text. You can place static text within a section control.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the text value that is associated with the database attribute bound to this control.
Label	If you enter a text value in this field, it overrides the **`Default Label`** for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Attribute	You can associate this control with a database attribute. Not all attributes have a **`Default Label`** value.
Text Alignment	Select Left, Center, or Right to align static text in a section control. Typically, you associate static text with a control in a section. You can place the text above or below the control.
Span	You can specify a number in this field if you want the static text to span more than one column.

Advanced properties

Attribute name	Description
Column	You can specify a number in this field to position static text in a specific column. Each section control has seven columns that are transparent to the user. If you enter the value 3 in this field and select the Center value for **Text Alignment**, the static text is centered above or below column three. If the text is wider than its column, the width of the column extends automatically to contain it.
Display as Link	By default, static text does not display as a URL link. Select this option if you want to configure the control to display as a URL link
Display as Plain Text	Selecting this check box removes the CSS (cascading style sheet) class definition so that text is displayed as plain text in the user interface.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Tab properties

A tab control is a container in which you place other controls such as tables and sections. Formatting tabs is, typically, the starting point for designing the structure and layout of a new application or extending the layout of an existing application. Place this control inside a tab group control.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	Specify the label for the tab control. This label is specific to the application and is not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.

Attribute name	Description
Type	Some tabs have a type attribute that you must set: • Set the type as "list" if the tab is a List tab. • Set the type as "insert" if the tab is a Main tab. • Leave this field blank for all other tabs.
Default	Select this option if you want this tab to be the default tab that opens when a user opens an application. The default tab can be a list tab or an insert tab except in a power application, when the default tab must be a list tab.
Bean Class	You can specify the name of a data bean class to use when creating the data source for the control. If not defined and the control is creating a data source, the default psdi.webclient.system.DataBean class is used. Typically, you use this property to create custom event methods or to override the manner in which the data source obtains its MboSet.
Business Object	Specify the name of a MBO in this field to determine which MBO the control uses to show and process information. If a MBO is not specified, the control uses the data source of its parent control. When you set a MBO for this control a data source is created that can be referenced from other controls.
Parent Data Source ID	You can specify the ID of a data source to use as the parent object for a relationship, or to specify a parent object for a bean class if necessary. Then specify the relationship that uses this parent in the Relationship field. If the field is left blank, the control uses the data source ID of its parent control as the Parent Data Source ID
Relationship	If you specified a value in the Parent Data Source ID field, specify a value in this field that identifies the relationship between this control and the new data source. A relationship uses a WHERE clause to determine what to show and process from the data source. The control creates a datasource that is by default used by the child controls. You can search for relationship values in the **Database Configuration** > **Relationships** tab.

Advanced properties

Attribute name	Description
Where Clause	You can specify a WHERE clause that the data source for this control uses to retrieve data. This WHERE clause is used in addition to any default WHERE clause used by the business object of the data source.
Order By	You can specify the order in which a SQL query fetches data for the data source for this control.
Listeners	Enter the ID or IDs of one or more data sources that you want to refresh whenever there are changes to this control. Listener controls can be located anywhere within the application. Separate multiple values with a comma.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Tab group control

A tab group control is a container control that holds tabs. Tab groups are part of the template for power applications. When you create a power application, a tab group control is provided with the standard List and Workspace tabs.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Format	Leave blank (the default) or specify one of the following layout options: • CARDDECK • DEFAULT • WIZARD The wizard option replaces the standard tab layout with arrows that point to the tab headings. The cardeck option appears in single-page applications that do not use tabs.

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 control

The table control is a container with the basic layout elements of a table window, including a toolbar with filter and search functionality and icons for previous and next row and page. You can place a table control within a section, tab, or window control and you can place a table column control or a section control within a table control.

The table control layout also includes a table column control and a details box where you can enter a section. The table control is similar to the table window in the List tab of most applications. The tables enable the user 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.

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

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Label	Enter a text value for the label. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Input Mode	Input mode options are: • `Blank:` An input mode with no value is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. • `Default:` This mode is used for quick entry of default table data in tables. • `Query`: This mode is read-only and shows a typed value. • `Readonly`: This mode is read-only and shows grayed-out text. • `Required`: This mode is read/write and the record cannot be saved unless the user enters a value in the field.
Select Mode	The select mode controls whether a table shows a single record or multiple records.

Attribute name	Description
Bean Class	You can specify the name of a data bean class to use when creating the data source for the control. If not defined and the control is creating a data source, the default psdi.webclient.system.DataBean class is used. Typically, you use this property to create custom event methods or to override the manner in which the data source obtains its MboSet.
Order By	You can specify the order in which a SQL query fetches data for the data source for this control.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, `MAINRECORD` is the data source because it is the primary data source for the application. Each application has a presentation file that defines the main MBO for the application, the `MAINRECORD`. To inherit data from a different MBO for this control (or group of controls) specify an alternative Data Source ID in this field.
Business Object	Specify the name of a MBO in this field to determine which MBO the control uses to show and process information. If a MBO is not specified, the control uses the data source of its parent control. When you set a MBO for this control a data source is created that can be referenced from other controls.
Parent Data Source ID	You can specify the ID of a data source to use as the parent object for a relationship, or to specify a parent object for a bean class if necessary. Then specify the relationship that uses this parent in the Relationship field. If the field is left blank, the control uses the data source ID of its parent control as the Parent Data Source ID
Relationship	If you specified a value in the Parent Data Source ID field, specify a value in this field that identifies the relationship between this control and the new data source. A relationship uses a WHERE clause to determine what to show and process from the data source. The control creates a datasource that is by default used by the child controls. You can search for relationship values in the **Database Configuration** > **Relationships** tab.

Attribute name	Description
Width	Specify a number to change the default width of the table. The default width for a table is 1000. You can decrease the width of a table to 400, the minimum width required to maintain the contents of the toolbar, including the table title and icons.
Description	Text in this field displays under the label in the title bar of the table.
Title Attributes	Reserved for future use.
Start Empty	If you select this option, the table always opens without displaying records. If you do not select the option (the default), the table automatically shows records that match the search query when opened.
Filterable	If you select this option, a filter row is provided to enable the user to search the table. The **Filterable** property is enabled by default and the setting automatically applies to table columns in the table. This option must be selected if you want to use the **Filter Expanded** property.
Collapsed	If you select this check box, the section is collapsed by default in the user interface and the user must click to maximize the section.

Advanced properties

Attribute name	Description
Collapsed Empty Label	You can specify a value that will override the table title (label property) when the table is collapsed and empty.
Collapsed Label	You can specify a value that will override the table title (label property) when the table is collapsed.
Parent Empty Label	If this table is a child of a parent table, and that table is empty, this value overrides the title of the parent table.
Row Details Expanded	Select this check box if you want a table to display the expanded detail sections for each record in the table. The default setting is not to expand the detail sections for a table row.
Application Restrictions	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.
Listeners	Enter the ID or IDs of one or more data sources that you want to refresh whenever there are changes to this control. Listener controls can be located anywhere within the application. Separate multiple values with a comma.

Attribute name	Description
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Table column control

The table column control is a container you can use to add columns and column headings to a table. You can place a table column control within a table control.

At runtime, a table column dynamically generates either a text box control or a check box control to show its data. A check box is used if the data type for the field is YORN. If the data type for the field is not YORN, a text box is used.

You can add a text box to a table column if you want to bind a field with a YORN data type to a text box, instead of to a check box. Unless you want to override the use of a check box for a field with a YORN data type, do not add a control to a table column control.

You can use the standard filter feature to display only records that meet specified criteria.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the text value that is associated with the database attribute bound to this control.
Label	If you enter a text value in this field, it overrides the **`Default Label`** for the control. This label is specific to the application and is not stored in the `MAXATTRIBUTE` or `APPFIELDDEFAULTS` tables.
Attribute	You can associate this control with a database attribute. Not all attributes have a **`Default Label`** value.

Attribute name	Description
Type	A table column can trigger an event or provide a link. If you select **Link**, the value, such as a record number, is underlined. When the user clicks this link, the record opens. If you select **Event**, enter appropriate values for the **Event Type**, **Event Description**, and **Event Icon** properties. For example, is you specify the TOGGLEDETAILSTATE event, the user can click a **View/Show** icon to open or close the details section for a selected record.
Sortable	If you select this option, the user can sort records in a table by clicking on the label of a column. The first click sorts the records in ascending order. The second click a label sorts the records in descending order.
Turn Smart Fill Off	Select this option to disable smart fill which is turned on by default. Smart fill enables a user to enter a partial value in a text box which the system attempts to match to a valid value, or values, in the database. When a matching value is found, the value is entered in the text box. If the system finds more than one matching value, the user can select one from a list of matching values. If you turn smart fill off, user text is not validated during entry.
Show Filter	If you select the **Filterable** and **Show Filter** properties, a filter text box for this column is displayed in the filter row of the table.
Filterable	If you set the **Filterable** property for a table, the setting also applies to each column in the table. You can override the **Filterable** setting for a table at the table column level. When you select this option, the user can search on the attribute associated with the label of the table column.
Event	If you select **Event** as the **Type** property, specify the event in this field. There is no directory of events but you can search for events and event descriptions in the system XML files. To export the files, in a main tab in the Application Designer, choose **Select Action** > **Export System XML**. This action exports the `library.xml`, `lookups.xml`, and `menus.xml` files to the <*root*>`\resources\presentation\system` directory. You can search these files for `MXEVENT` and `mxevent_desc` entries. You can also create your own events.
Target ID	Enter a target control ID, if different from the current control, where an event (if defined) is sent for processing.

Attribute name	Description
Event Description	This property provides a tool tip for an event (if defined).
Event Icon	This property can point to a .gif file of an image that represents an event. The icons in the user interface are stored in the `<root>\applications\maximo\maximouiweb\webmodule\webclient\images` directory. The image control in the Application Designer provides access to the files in the images directory.
Change Event	You can configure a control to perform one of the following actions in the event of a data change to the control: • The `REFRESHTABLE` event refreshes the data source table for the control. • The `RESETCHILDREN` event deletes any child objects of the data bean of the control which then reacquires its `MBOSetRemote` references.
CSS Class Name	If you want to change the layout and appearance of the control, enter the CSS class name to use. You can search for CSS class names in the `applications\maximo\maximouiweb\webmodule\webclient` directory.
Lookup	You can specify a lookup ID to link this control to a pre-formatted lookup table that pulls records from the main table. A user can select a record from the lookup table to use in this control. Lookup IDs are stored in the `lookups.xml` file. To view the content of this file, export the system XML files.
Go To Applications	You can associate a pull-down menu with this control and add application names to it that users can click to go to different applications.

Attribute name	Description
Input Mode	Input mode options are: - `Blank`: An input mode with no value is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. - `Default`: This mode is used for quick entry of default table data in tables. - `Password`: This mode is read/write and displays asterisks. - `Password Readonly`: This mode is read-only and displays asterisks. - `Password Required`: This mode is read/write and displays asterisks. The record cannot be saved unless there is a value in the field. - `Query`: This mode is read-only and shows a typed value. - `Readonly`: This mode is read-only and shows grayed-out text. - `Required`: This mode is read/write and the record cannot be saved unless the user enters a value in the field.

Advanced properties

Attribute name	Description
Label Attributes	You can enter a comma separated list of data attributes used to create the **Default Label** for this control. **Label Attributes** and **Label Source ID** enable you to display dynamic values in the **Default Label** for a table column.
Label Source ID	Enter an object name that is the data source for the Label Attributes property.
Title Attributes	Reserved for future use.
URL Attribute	Enter a URL value if you set the **Type** property for this control to **`Link`**. The URL value becomes a hyperlink.
Is LD Readonly	By default, the system sets a long description associated with a table column attribute to editable. Select this field to make the long description read-only.
Always Synchronous	If deselected, the control can communicate with the database without pausing user activities. If cleared, the user cannot perform activities in other fields until validations for this field are completed. Synchronous activity is disabled by default. If there are dependencies between values specified for this control and other controls in the application, you can enable synchronous activity for the dependent controls.

Attribute name	**Description**
Linked Controls ID	You can enter a control ID that is automatically updated whenever there is a change to the attribute value in this column. The control ID must exist somewhere within the application.
More information	You can specify the values to construct a URL for a Knowledge Center topic using the following format: `com.ibm.mbs.doc,application/helpfile.html` The server address and other information for the URL are controlled by the mxe.help properties which you can review and configure in the System Properties application. If you specify a URL, a More Information link is added to the field help (Alt + F1) for the control. When the user clicks the link, the target topic opens in the Knowledge Center. The property is specific to the field ID in the presentation XML file. If you want to provide the same link in other controls or other applications, you must manually add the URL to each individual control.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Text box control

A text box control is a single-line text box that displays information the user either enters or selects from a list. Text boxes have labels that are typically visible in the user interface. Labels are text fields that describe the type of information displayed in the box. You can place a text box within a section control.

Text boxes 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 text box linked to a detailed menu can contain diverse options, such as a Select Value window, jumps to other applications, or views of different records. The Menu Type property you configure for a text box controls the format and content of the detailed menu.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Default Label	This read-only value is the text value that is associated with the database attribute bound to this control.
Label	If you enter a text value in this field, it overrides the **Default Label** for the control. This label is specific to the application and is not stored in the MAXATTRIBUTE or APPFIELDDEFAULTS tables.
Hide Label	By default, the label value always appears in the user interface. Select this check box to make the label invisible.
Turn Smart Fill Off	Select this option to disable smart fill which is turned on by default. Smart fill enables a user to enter a partial value in a text box which the system attempts to match to a valid value, or values, in the database. When a matching value is found, the value is entered in the text box. If the system finds more than one matching value, the user can select one from a list of matching values. If you turn smart fill off, user text is not validated during entry.
Attribute	You can associate this control with a database attribute. Not all attributes have a **Default Label** value.
Data Source ID	Specify a data source ID if you want this control to use a different data source than the parent. If the control is creating a data source and the Data Source ID property is not set, the ID defaults to the ID of the control. If the data source is blank, the control inherits the data source from the parent. If no parent controls have a data source specified or if the parent is the application, MAINRECORD is the data source because it is the primary data source for the application.
Lookup	You can specify a lookup ID to link this control to a pre-formatted lookup table that pulls records from the main table. A user can select a record from the lookup table to use in this control. Lookup IDs are stored in the lookups.xml file. To view the content of this file, export the system XML files.
Go To Applications	You can associate a pull-down menu with this control and add application names to it that users can click to go to different applications.

Attribute name	Description
Input Mode	Input mode options are: • `Blank:` An input mode with no value is read/write. Any information entered in this field is stored in the database column specified in the **Attribute** field. • `Default:` This mode is used for quick entry of default table data in tables. • `Password`: This mode is read/write and displays asterisks. • `Password Readonly`: This mode is read-only and displays asterisks. • `Password Required`: This mode is read/write and displays asterisks. The record cannot be saved unless there is a value in the field. • `Query`: This mode is read-only and shows a typed value. • `Readonly`: This mode is read-only and shows grayed-out text. • `Required`: This mode is read/write and the record cannot be saved unless the user enters a value in the field.
Menu Type	You can specify the value of a menu ID in this field to link the control to a preformatted drop-down menu. Menu IDs are stored in the `menus.xml` file. To view the content of this file, export the system XML files.

Advanced properties

Attribute name	Description
Detail Image	Reserved for future use.
QBE Prepend	If you set the **Input Mode** to **`Query`**, you can filter the search by adding a text prefix that acts as filter. For example, the More Search Fields window uses **QBE Prepend** to add date search filters similar to ">=". You can specify the value here to save key strokes when searching for records that meet specific date criteria.
Change Event	You can configure a control to perform one of the following actions in the event of a data change to the control: • The `REFRESHTABLE` event refreshes the data source table for the control. • The `RESETCHILDREN` event deletes any child objects of the data bean of the control which then reacquires its `MBOSetRemote` references.

Attribute name	Description
Always Synchronous	If deselected, the control can communicate with the database without pausing user activities. If cleared, the user cannot perform activities in other fields until validations for this field are completed. Synchronous activity is disabled by default. If there are dependencies between values specified for this control and other controls in the application, you can enable synchronous activity for the dependent controls.
Readonly Long Description	By default, the system sets a long description associated with a table column attribute to editable. Select this field to make the long description read-only.
Display Type	Reserved for future use.
More information	You can specify the values to construct a URL for a Knowledge Center topic using the following format: `com.ibm.mbs.doc,application/helpfile.html` The server address and other information for the URL are controlled by the mxe.help properties which you can review and configure in the System Properties application. If you specify a URL, a More Information link is added to the field help (Alt + F1) for the control. When the user clicks the link, the target topic opens in the Knowledge Center. The property is specific to the field ID in the presentation XML file. If you want to provide the same link in other controls or other applications, you must manually add the URL to each individual control.
Move To Data Source ID	Specify a record stored in a different object that the user can move to. You can select any Data Source ID from the presentation control. By default, the system sends the Move To event to the `ResultsBean`, which contains a set of records displayed in the List tab. The Move To event affects the Move to Data Source ID object.
Signature Option	You can specify a signature option that can restrict access to this control or change the properties of this control based on a condition that you specify. To create a signature option, click **Add/Modify Signature Options** in the **Select Action** menu. To specify a condition, click the **Configure Conditional Properties** button.

Attribute name	Description
Sig Option Data Source ID	An MBO must be used to evaluate whether or not a user has access to a signature option. Specify the value of the data source that performs this evaluation if it is different to the data source that is configured for this control.

Tree control

Use a tree control as a container for tree node and tree attribute controls to build dynamic, hierarchical tree structures with specific business objects that support hierarchical data. You can place a tree control within a dialog control or a tab control. You can place a tree node control into a tree control.

The Classifications application uses tree controls to construct classification hierarchies and the Users application uses tree controls to construct the security profile hierarchy

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

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Height	Enter a numeric value in this field that will increase or decrease the height of the image. If you change the default dimensions, the image can become distorted.

Attribute name	Description
Width	Enter a numeric value in this field that will increase or decrease the width of the image. If you change the default dimensions, the image can become distorted.
Bean Class	You can specify the name of a data bean class to use when creating the data source for the control. If not defined and the control is creating a data source, the default psdi.webclient.system.DataBean class is used. Typically, you use this property to create custom event methods or to override the manner in which the data source obtains its MboSet.
Business Object	Specify the name of a MBO in this field to determine which MBO the control uses to show and process information. If a MBO is not specified, the control uses the data source of its parent control. When you set a MBO for this control a data source is created that can be referenced from other controls.
Relationship	If you specified a value in the Parent Data Source ID field, specify a value in this field that identifies the relationship between this control and the new data source. A relationship uses a WHERE clause to determine what to show and process from the data source. The control creates a datasource that is by default used by the child controls. You can search for relationship values in the **Database Configuration** > **Relationships** tab.
Use Data Source	Reserved for future use.
Order By	You can specify the order in which a SQL query fetches data for the data source for this control.

Tree attribute control

The tree attribute control describes the data that appears in each tree node. For example, the tree attributes in the Classifications application display classification IDs and descriptions in the user interface. You can place a tree attribute control within a tree node control.

Properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Display	Select this option to display the Attribute value in the tree node.

Attribute name	Description
Attribute	You can associate the values that display in a tree node with a database attribute. To create a classification example, 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.DESCRIPTION" to display the Classification Description value.

Tree node control

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 Tree Attribute control in this container control.

You can place this control inside the Tree container control.

General properties

Attribute name	Description
Control ID	Each control in a presentation is assigned a unique Control ID that is read-only.
Image	You can specify the file name of any file stored in the images directory at `<`*root*`>\applications\maximo\maximouiweb\webmodule\webclient\images`. You can reuse any existing images or you can add new images to the directory. Images can be in any format supported by a browser. You must redeploy the .EAR file on the application server to add the images to the user interface.
Display Key Attribute	Select this property to display the **Key Attribute** as part of the tree node label.
Display Object Name	Select this property to display the **Object Name** as part of the tree node label.
Key Attribute	You can specify a different attribute of the business object to use as a key attribute, other than the key attribute specified for the MBO.
Key Value	You can specify an option to display as the default value.
Return Attribute	Enter an attribute name whose value gets returned to the browser when you click the icon of a tree node line item.

Attribute name	Description
Enable Return	Select this property to enable the **Return Attribute** property functionality.

Notices

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

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

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

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

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

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

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

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

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

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

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

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

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

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

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

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

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

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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

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

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

Printed in USA