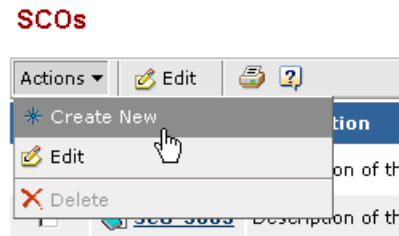

Building Toolbars

The configurable toolbar displays a drop-down menu of the actions that can be performed on a table page, form page, portal page, or a list of objects. This drop-down menu can replace action links and action bars for table and form pages. Because the links are hidden until the user clicks the menu, the configurable toolbar takes up much less space than the action links.



In future versions, the configurable toolbar will also replace the My Desk and Action tabs and the main toolbar on the right side of the page.

This section explains how to build and configure toolbars.

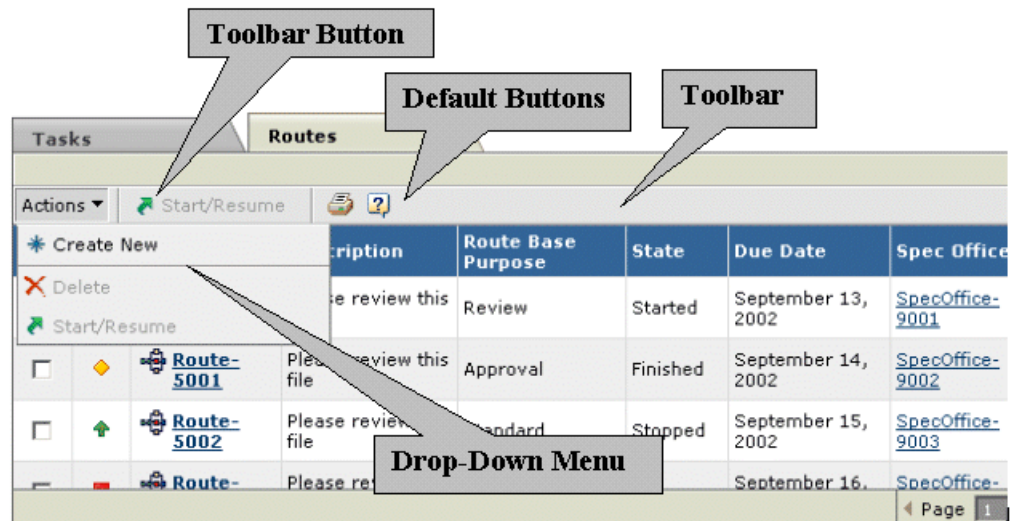
- [Overview of the Configurable Toolbar](#)
- [Drop-Down Menus Can Display to the Right](#)
- [Building a Configurable Toolbar](#)
- [Parameters and Settings for Toolbar Menu Objects](#)
- [Parameters and Settings for Toolbar Link Command Objects](#)
- [URL Parameters Accepted by emxPortal.jsp](#)
- [Implementing a Toolbar in a JSP](#)

Overview of the Configurable Toolbar

The configurable toolbar contains two main components:

- the *Toolbar* itself, which is the horizontal grey bar that contains default buttons such as Help and Printer-Friendly, links, and drop-down menus with additional links
- *Drop-Down Menus*, which expand when click to list links configured for the menu

The following figures shows the configurable toolbar within the context of a table page.



Toolbar

This graphic shows a toolbar that contains all the main components that a toolbar can have.



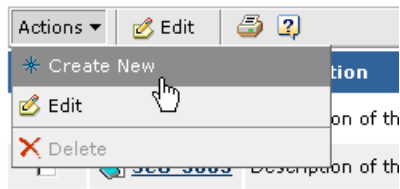
The toolbar can contain:

- Default links such as the Help and Printer-Friendly links shown above.
The toolbar component relies on querystring parameters to determine whether to display the Printer-Friendly link, the Export to Excel links, and the TipPage. By default, these buttons display, unless they are explicitly turned-off.
The Help link always displays. If no valid help marker is passed in, or if a valid help marker is passed in but no matching marker is defined in the source help files, it defaults to the main help page for the specific application.
- Links configured for the toolbar, such as the Edit link shown above.
When the user clicks such a link, the configured page for it is called.
- Top-level menu items that contain a list of drop-down actions, such as the Actions menu item shown above.
When the user clicks the menu item, a list of drop-down actions displays.
- Vertical separators that separate links on the toolbar.
The above graphic has 2 vertical dividers. The dividers can be placed as needed. For example, you can configure the toolbar to have a divider for every two buttons. Or, a divider can be added to create a logical grouping of links, much like the grouping of new, open, save buttons on the Windows toolbar.

Drop-Down Menus

The drop-down contains the list of actions that can be performed from the current page. This graphic shows a drop-down menu.

SCOs



Drop-down menus can contain:

- Top-level menu buttons that activate the drop-down menu. This is the Actions button in the graphic shown above.

The title of the top-level menu button is configurable.

The key corresponding to the display name is found in the system.properties file by the name: emxFramework.UIActionbarMenu.Label. This properties key in turn points to a key in the frameworkstringresource.properties file. By default the value is set to emxNavigator.UIMenuBar.Shortcuts. To view or change the value for the display actions name, open the emxFrameworkStringResource.properties file and look for the emxNavigator.UIMenuBar.Shortcuts key and change its value to the desired string.

- Links in the drop-down menu.

When users click the links, the page configured for that link is called.

Both the drop-down menu links and the toolbar links can be configured with the following characteristics:

- Icons/Text/Both—For example, the Edit link shown above is configured as an icon with text and the Printer-Friendly and Help links are shown above as icons only.
- Activation—Toolbar buttons and drop-down menus can be enabled and disabled. For example, if the action requires that at least one item is selected on the list page, then the action is disabled until at least one item is selected.
- Form Submission—Support for form submission through toolbar buttons/drop-down options.
- Horizontal separators.
From historical perspective, the divider represents the separation of the top and bottom action links used in UI Level 3.

The configurable toolbar replaces the top and bottom action bars. When top and bottom action bars are used, they are each configured separately and must both be passed in as URL parameters to the configurable page. With the configurable toolbar, only a toolbar menu is required and it is passed to the configurable page. The top actions now display on top of the drop down menu and the actions that would previously appear on the bottom actionbar appear below the horizontal separator.

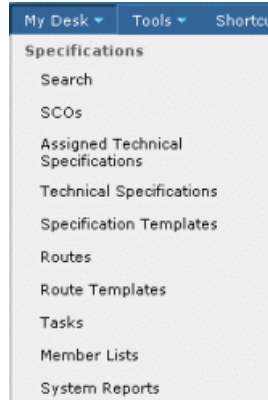
For Netscape 4.7 and Macintosh IE 5.2 browsers, the toolbar is displayed as Action Links in UI Level 3 environment. All the toolbar menu items, buttons and sub-menu items are displayed as the top action bar at the same level.

For both the toolbar and the drop-down menus, there are no set limits for the number of items in the toolbar or the drop-down menus. This new behavior replaces the pagination behavior currently used with the actionbar in UI Level 3.

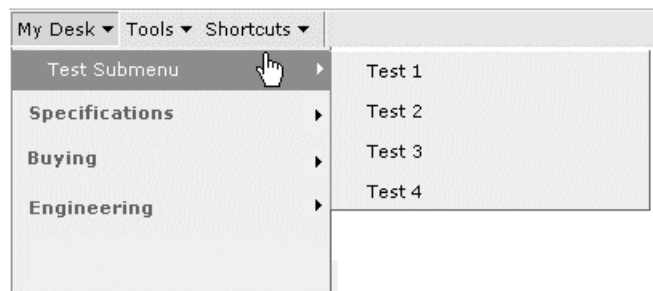
Drop-Down Menus Can Display to the Right

Using a setting for the menu object that represents the drop-down menu, you can configure a drop-down menu so the links in it display directly under the menu or to the right of it. This option to pull right is available only for the second-level drop-down menus. By default, the second-level drop-down menus pull to the right.

Here is an example of a drop-down menu that is not pulled to the right.



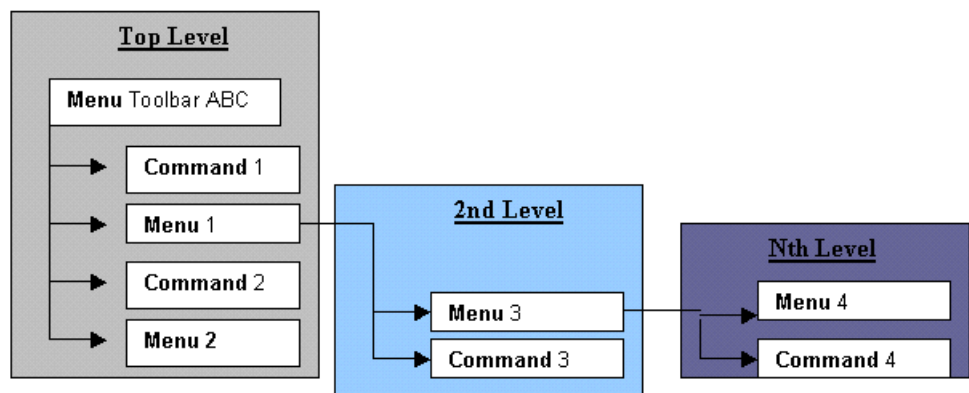
Here is an example of a menu that is pulled to the right.



Building a Configurable Toolbar

Mechanisms that Define a Configurable Toolbar

The configurable toolbar is configured using menus and commands. The toolbar supports an unlimited number of levels, but it is recommended that the number of levels be kept at 2 or 3. The following image depicts one possible structure.



The Top-Level section contains the toolbar menu and its connected menus and commands. The to-level menu, Menu Toolbar ABC in the above graphic, is an anchor for the toolbar

structure. As such, it contains no settings. The connected menus and commands display as toolbar buttons on the top-level toolbar as shown below.

Command1	Menu 1	Command 2	Menu 2
----------	--------	-----------	--------	-------

So to display a standard link in the toolbar, one that does not contain a drop-down menu, you connect the command for the link to the top-level menu object. To display a drop-down menu on the toolbar, you connect the menu object for the drop-down menu, such as Menu 1, to the top-level menu object. Then connect the commands for the links within the drop-down menu to Menu 1.

How the Toolbar is Linked to an Application

To use the configurable toolbar on a table, form, or portal page, the URL that calls the page needs to add the following parameter:

```
toolbar=TOOLBARMENU
```

Where TOOLBARMENU is the name of the toolbar menu. Below is an example URL that includes a toolbar on a table page:

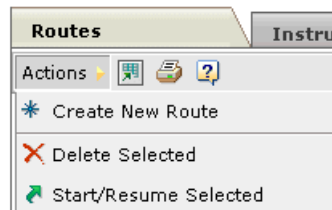
```

${COMMON_DIR}/
emxTable.jsp?program=emxAEFCollection:getObjects&table=AEFColle
ctionItems&toolbar=AEFCollectionsToolBar&selection=multiple

```

Steps for Building a Toolbar and Linking it to an Application

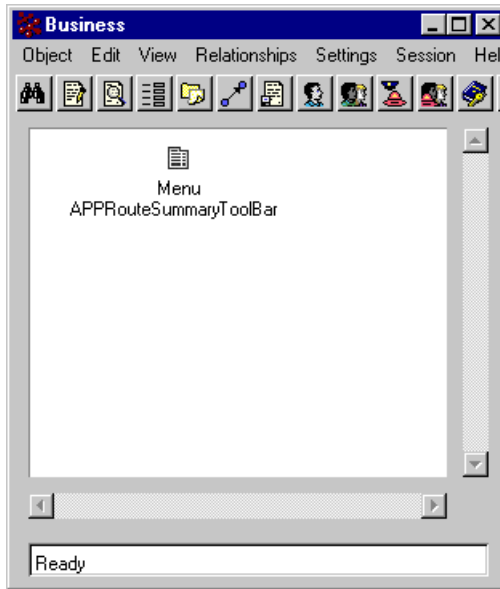
The following procedure lists the main steps needed for creating a toolbar and including it on a page. The graphics included in the procedure represent the administrative objects needed to build a toolbar for a page that lists routes, as shown in this graphic. This sample toolbar has 1 drop-down menu and 3 default links. It has no links that display on the toolbar and that are added using command objects. The drop-down menu contains 3 links plus a separator.



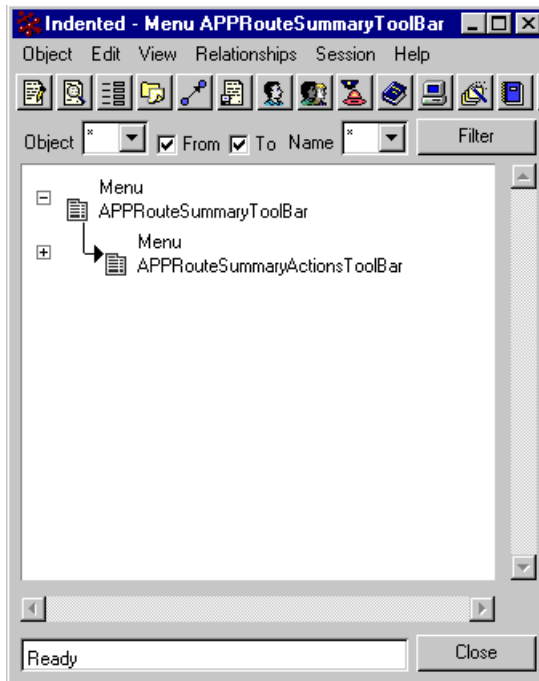
To create a toolbar and add it to a page in an application

1. Create a menu object to represent the top-level menu.

There are no settings needed for the top-level menu. For naming conventions, see [Naming Conventions for UI Administrative Objects](#).



2. Create a command object for each link that users should be able to access for the page, including those that should be within a drop-down menu.
 For a description of how to fill in parameters and settings for toolbar link command objects, see [Parameters and Settings for Toolbar Link Command Objects](#). For naming conventions, see [Naming Conventions for UI Administrative Objects](#).
3. Create lower-level menu objects for each drop-down menu and assign the lower-level menu objects to the top-level menu object. For help understanding the structure, see [Mechanisms that Define a Configurable Toolbar](#).
 For a description of how to fill in parameters and settings for toolbar menu objects, see [Parameters and Settings for Toolbar Menu Objects](#). For naming conventions, see [Naming Conventions for UI Administrative Objects](#).

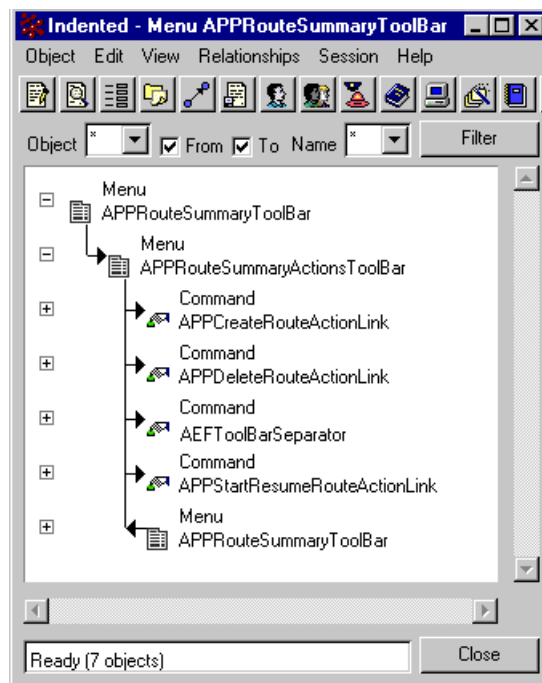


- For each link that you want to appear on the toolbar, and not in a drop-down menu, assign the command to the top-level menu object that you created in Step 1. (The sample route toolbar shown before this procedure does not toolbar links except the default links. So no commands will be connected to the top-level menu for the sample toolbar.)

Remember to assign commands to represent separators as needed. For each separator that you want in the toolbar, assign the command `AEFToolBarSeparator`, which is installed by default with the Framework, in the appropriate order for the top-level menu. When a separator command is added to the toolbar, it appears as a vertical line.

- For each link that you want to appear in a drop-down menu, assign it to the appropriate menu. A link can appear in both the toolbar and in a drop-down menu.

Remember to assign commands to represent separators as needed. For each separator that you want in the drop-down menu, assign the command `AEFToolBarSeparator`, which is installed by default with the Framework, in the appropriate order for the menu. When a separator command is added to a drop-down menu, it appears as a horizontal line.



- In the URL that calls the table page, form page, or portal page that you want to include the toolbar on, include the parameter:

```
toolbar=TOOLBARMENU
```

Where `TOOLBARMENU` is the name of the toolbar menu. Below is an example URL that includes a toolbar on a table page:

```

${COMMON_DIR}/
emxTable.jsp?program=emxAEFCollection:getObjects&table=AEFCollectionItems&toolbar=AEFToolBarMenu&selection=multiple

```

Make sure you pass in the parameters as needed for the default links. The Help link always displays. If you don't pass in any parameters, the Printer-Friendly and Export links display by default and the Tip and Conversion links do not. You can pass in parameters to have these link display or not display.

```
PrinterFriendly=false/true
```

export=false/true

TipPage=false/true

CurrencyConverter=false/true

7. If you are working with the Web-based user interface as you are making changes and want to see your changes in the user interface, click the **Reload Cache** tool in the toolbar and click the browser **Refresh** button.

The cache is refreshed automatically when the component age expires. This setting is in emxSystem.properties.

Only persons assigned to the Administration Manager role have access to the Reload Cache tool.

Parameters and Settings for Toolbar Menu Objects

This table describes how to fill in the parameters for menu objects that represent the top-level menu for a toolbar and the lower-level menus. For specific instructions on how to create menu objects using Business Modeler or MQL, refer to the *Business Modeler Guide* or *MQL Guide*.

Parameter	Description	Accepted Values/Examples
Alt	The text that should display in a Tooltip when the user mouses over the toolbar.	—
Commands (specified in the Items tab in Business Modeler)	The command objects that represent the links in the toolbar for the top-level menu and that represent links in the drop-down menu for lower-level menus. The order of the commands is the order the links appear on the toolbar and in the drop-down menu.	Names of command objects, such as: ENCCreateRevisionToolBarActionLink TMCEditProfileToolBarActionLink
Href	Not applicable to menu objects for toolbar menus.	—
Icon	Not applicable to menu objects for toolbars.	The name of an image file, such as Part.gif.
Label	Not applicable for top-level menus. The text that should be displayed on the toolbar menu. This is the text that appears on the “button” that users click to access the drop-down menu, such as Action. Either a string resource ID for the text string or the actual text string that should appear. To internationalize the text, you must use a string resource ID. See Internationalizing Dynamic UI Components . The system first looks for a string resource ID that matches the entered value. If it finds one, it uses the value for the ID. If it doesn't find one, it displays the entered text.	Create New Part emxFramework.common.Actions If the label text is not found and no image is defined, the toolbar component displays an error icon to indicate that a configuration error has occurred.
Menus (specified in the Items tab in Business Modeler)	The menu objects that represent drop-down menus. The order of the menus is the order they appear in the toolbar or drop-down menu.	Names of menu objects, such as: ECBOMListToolBar TMCToolBar
Settings	Additional settings that define the behavior and appearance of the toolbar menu objects.	Name/value pairs, as defined in the table below.

This table lists and describes the settings for menu objects that represent drop-down toolbar menus. There are no settings for the top-level menu. Note that the name and value for each setting are case sensitive.

Setting	Description	Accepted Values/Examples
Maximum Length	Limit the number of characters that are displayed on drop-down menu label. The Title property of the button is used as the Alt text to display the full label.	Any number of characters, such as 25 If this setting is not set, the toolbar displays the entire label.
Image	The image used for the menu. This is optional if a label has been defined but required if there is no label.	\${COMMON_DIR}/ iconSmallOrganization.gif

Setting	Description	Accepted Values/Examples
*Registered Suite	<p>The application the toolbar menu belongs to. The system looks for files related to the menu in the registered directory for that application, which is specified in emxSystem.properties.</p> <p>Based on the application name, the system passes the following parameters in the Href URL:</p> <ul style="list-style-type: none"> • suiteKey • emxSuiteDirectory • StringResourceId 	<p>The value must be set without any spaces, for example, EngineeringCentral or Framework. The value must be set to the suite name as defined in the key eServiceSuites.DisplayedSuites within emxSystem.properties. If the suite name starts with “eServiceSuite” then this prefix can be skipped and assign the remaining text to the setting. For example, if the suite name in emxSystem.properties is “eServiceSuiteEngineeringCentral”, then the word “EngineeringCentral”, can be assigned as “Registered Suite”.</p> <p>In the Href URL that is called when the menu is clicked, the system passes a parameter called “suiteKey”. The value for the parameter is the property name from emxSystem.properties that maps to the setting’s value.</p>
Pull Right	<p>Determines whether the drop-down links are displayed in the same level as the menu or pulled right to display the links in the next level. Applies only to second-level drop-down menus. For examples, see Drop-Down Menus Can Display to the Right.</p>	<p>True (default) False</p>
Access Expression Access Function Access Mask Access Program	<p>All these settings can be used to control access to tabs just as they can be used to control access to other UI components. For details, see Controlling User Access to User Interface Components.</p>	<p>—</p>
<p>*Required Setting</p>		

Parameters and Settings for Toolbar Link Command Objects

This table describes how to fill in the parameters for command objects used for toolbar links. For specific instructions on how to create command objects using Business Modeler or MQL, refer to the *Business Modeler Guide* or *MQL Guide*.

Parameter	Description	Accepted Values/ Examples
Icon	Icon for the object within the Matrix core. This setting does not display an image in the UI.	The name of an image file, such as Part.gif.
Label	The text that should be displayed on the toolbar link. You can configure a link to have a text label, an image, or both. If you don't specify an image using the Image setting, you must specify a label. Either a string resource ID for the text string or the actual text string that should appear. To internationalize the text, you must use a string resource ID. See Internationalizing Dynamic UI Components . The system first looks for a string resource ID that matches the entered value. If it finds one, it uses the value for the ID. If it doesn't find one, it displays the entered text.	Create New Part emxFramework.common.Actions If the label text is not found and no image is defined, the toolbar component displays an error icon to indicate that a configuration error has occurred.
Href	The URL to call when a user clicks the link.	<code>\${COMPONENT_DIR}/emxComponentsPage.jsp</code> , <code>\${SUITE_DIR}/emxTeamWorkspaceDetailsFS.jsp</code>
Alt	The Alt text to appear over the link when the user mouses over it.	—
Access	The persons, roles, and groups who can access the link. To make the link available to all users, regardless of role/group assignments, choose All. Note that if no users are assigned access, the system assumes all users have access. If a user doesn't have access to any links in a menu, the menu is not displayed. Also see Controlling User Access to User Interface Components .	Names of group, role, person administrative objects. or All (default)
Settings	Additional settings that define the behavior and appearance of the links. For a list of the accepted settings, see the table below.	Name/value pairs, as defined in the table below.

This table lists and describes the settings for command objects used for toolbars. Note that the name and value for each setting are case sensitive.

Setting	Description	Accepted Values/Examples
Access Expression Access Function Access Mask Access Program	All these settings can be used to control access to toolbars just as they can be used to control access to other UI components. For details, see Controlling User Access to User Interface Components .	—
Action Type	Tells the configurable toolbar that the command is a separator instead of a link. If the setting is not included, the toolbar treats the command as a standard link.	Separator—A line that separates one or more links. If the command is assigned to the top-level menu and is therefore on the toolbar itself, it is considered a vertical separator. Otherwise, it is considered a horizontal separator.
Maximum Length	Limit the number of characters that are displayed on toolbar link. The Title property of the button is used as the Alt text to display the full label.	Any number of characters, such as 25 If this setting is not set, the toolbar displays the entire label.
Image	The image used for the link. This is optional if a label has been defined but required if there is no label.	<code>\${COMMON_DIR}/ iconSmallOrganization.gif</code>
*Registered Suite	The application the command belongs to. The system looks for files related to the link in the registered directory for that application, which is specified in <code>emxSystem.properties</code> . Based on the application name, the system passes the following parameters in the Href URL: <ul style="list-style-type: none"> • suiteKey • emxSuiteDirectory • StringResourceId 	The value must be set without any spaces, for example, <code>EngineeringCentral</code> or <code>Framework</code> . The value must be set to the suite name as defined in the key <code>eServiceSuites.DisplayedSuites</code> within <code>emxSystem.properties</code> . If the suite name starts with “eServiceSuite” then this prefix can be skipped and assign the remaining text to the setting. For example, if the suite name in <code>emxSystem.properties</code> is “eServiceSuite EngineeringCentral ”, then the word “ EngineeringCentral ”, can be assigned as “Registered Suite”. In the Href URL that is called when the tree category is clicked, the system passes a parameter called “suiteKey”. The value for the parameter is the property name from <code>emxSystem.properties</code> that maps to the setting’s value.
Popup Modal	If the setting Target Location is set to popup, the window can be configured as modal or non-modal.	true—the popup window is modal false—the popup window is non-modal If the setting is not specified, the window is modal.
*Required Setting		

Setting	Description	Accepted Values/Examples
Confirm Message	<p>Provides a JavaScript confirmation message when users click on the link. For example, a custom delete message can be configured for the onClick event to display a JavaScript confirm message to the user. The Confirmation dialog has the “OK” and “Cancel” button. Clicking OK proceeds with the processing and Cancel cancels the operation.</p> <p>You can include the number of selected table rows in the message using a macro. See <i>Other Macros</i>.</p>	<p>The actual text to display in the confirmation message or a string resource property key. To internationalize the message, a string resource key must be used: Are you sure you want to delete this object? emxFramework.common.alertMsg</p>
Row Select	<p>Used only for links on table pages. Specifies whether the JSP specified for the Href expects one, at least one, or no rows in the table (Middle Frame) to be selected.</p> <p>If the Row Select setting is set to “single” or “multi”, the setting Submit must be set to “true” in order to get the details of the selected item(s) from the table.</p> <p>If the setting is not specified, it is assumed to be none.</p> <p>A JavaScript alert message is displayed on error conditions when the onClick event occurs. The error conditions are explained in the column to the right.</p>	<p>single—Appropriate only for links that appear on table pages that have radio buttons or check boxes for each row. The JSP specified for the Href expects exactly one row in the table (middle frame) to be selected. If more than one item is selected, a JavaScript alert message is displayed. The default message is set to the value of the following key in emxFrameworkStringResource.properties: emxFramework.Common.PleaseSelectOneItemOnly. Currently, the message is set to “Please Select One Item Only”. If no item is selected, then the user will see the following message: “Please select an Item”. This message corresponds to this key value in the emxFrameworkStringResource.properties file: emxFramework.Common.PleaseSelectitem</p> <p>multi—Appropriate only for links that appear on table pages that have check boxes or radio buttons for each row. The JSP specified for the Href expects at least one row in the table (middle frame) to be checked. If no item is selected, then the user will see the following message: “Please select an Item”. This message corresponds to the following key value in the emxFrameworkStringResource.properties file: emxFramework.Common.PleaseSelectitem</p> <p>none (default)—The JSP specified for the Href expects no row in the table to be selected. If a row is selected, the JSP does not use it.</p>
*Required Setting		

Setting	Description	Accepted Values/Examples
Submit	Specifies whether the system should send the object ID(s) (and relationship ID, if applicable) for the current page to the JSP specified in the Href parameter. For form pages, the submitted data is just the object ID for the business object the page is about. For table pages, the submitted data is the ID for each selected item in the table, which can include object and relationships. In either case, the IDs are sent using the relBusIdList parameter. If the setting is not specified, it is assumed to be false.	true—Submits the IDs to the URL specified in the Href parameter. false—The URL specified in the Href parameter is directly called from the link and no IDs from the current page are used. Use this value for links that do not require the business object IDs, such as links for creating a new object.
Target Location	Controls where the page specified in the Href parameter appears.	content—The page replaces the content frame. popup—Page appears in new window. The modality of the window is set using the Popup Modal setting. _top—Page replaces the entire body of the browser window. listHidden—Submits the table frame and carries out any background processing. hiddenFrame—Targets the Navigator hidden frame to perform any background processing. This frame should not be used within the context of the configurable table. Use listHidden instead. fromViewHidden—Used within the context of the configurable form. Any valid frame name that is available to the current frame.
Window Height	The window height for the new popup window. This value is used only when the setting for Target Location is set to “popup”.	400 600 (default) 800
Window Width	The window width for the new popup window. This value is used only when the setting for Target Location is set to “popup”.	400 600 (default) 800
*Required Setting		

URL Parameters Accepted by emxPortal.jsp

This table lists the parameters that emxPortal.jsp can use. You can add these parameters to the Href parameter for the component that calls the portal page. For example, when you specify the emxPortal.jsp to be called from a tree category, you can add these parameters to the Href parameter for the command object.

Parameter	Description	Accepted Input Values
portal	Specifies the menu admin object that represents the portal page.	Name of menu admin object defined to represent a portal page.
toolbar	Specifies the menu admin object that represents the action bar, which appears in the page header.	Name of menu admin object defined to represent a portal page.

Parameter	Description	Accepted Input Values
header	The content of the heading that appears at the top of the portal page.	Any alphanumeric text or a string resource ID. header=SummaryView header=emxSpecificationCentral.S COSummaryView.SummaryView
HelpMarker	Specifies the name of the help marker to call for context-sensitive help. For information about implementing help, see Implementing Context-Sensitive Help .	String The naming convention for help markers “emxhelp” followed by the object or feature and then the action, for example, emxhelproutecreate and emxhelpprojectedit. The marker is all lowercase with no spaces.
TipPage	Specifies whether the page should include the Tip Page tool and call a specific html or jsp when a user clicks the tool. If this setting is not included, the Tip tool is not included on the page.	Name of a custom html or JSP page, including any path. The starting point for the directory reference is the content directory. For example, if you want to call an html file in ematrix/doc/customcentral and the content directory is ematrix/customcentral, you would add this parameter to the table.jsp: TipPage=../doc/customcentral/tippage.html

Implementing a Toolbar in a JSP

To implement the toolbar with a JSP page

1. Configure the required toolbar (admin menu object) and action links (admin command objects) and connect them appropriately as described in [Building a Configurable Toolbar](#).
2. Include these stylesheets in the jsp page:
emxUIDefault.css
emxUIMenu.css
emxUIToolbar.css
3. Include these Javascript files in the jsp page:
emxUIConstants.js
emxUICore.js
emxUICoreMenu.js
emxUIToolbar.js
4. Insert the following section of JSP code into the JSP page.

```
// *****  
<%
```

```

// String sToolBarName = "<Name of the Toolbar Bar - menu
object>";
String sToolBarName = "ENCRoutesToolBar";
if ( (sToolBarName!= null))
{
%>
<jsp:include page = "emxToolBar.jsp" flush="false">
    <jsp:param name="toolbar" value="<%=sToolBarName%>"/>
    <jsp:param name="objectId" value="<%= objectId %>"/>
.....
</jsp:include>
<%
}
%>
//
*****

```

In the above code, the main parameters to be assigned are:

- sToolBarName: This is JSP page variable assigned to the name of the menu object, which is configured for the toolbar.
- toolbar: This is jsp:include parameter, assigned to the name of the Toolbar menu object.

Access changes

[Add the following to the descriptions of Access Mask and Access Expression in the section on “Controlling User Access to User Interface Components”]

Administrators need to make sure that an objectId is available before configuring this setting for configurable toolbar menus and commands. If no objectId is available, the toolbar uses the Person objectId to determine access.

Commands and Menus

[add the following row to the commands table in the schema definitions chapter]

Command	Description	Configured As:
AEFToolBarSeparator	Dividing line between toolbar links and drop-down menu links.	Separator for the configurable toolbar links

[add the following row to the menus table in the schema definitions chapter]

Menu	Description
AEFPortalHomeToolbar	Toolbar for the home portal page, PowerView. The toolbar is just a starting point for applications to add to and therefore has no settings or commands assigned to it.

Naming Convention

[add the following rows to the existing section on naming conventions for configurable components]

Admin Object Type, Usage	Convention	Examples
Menu objects for toolbars	3-letter standard abbreviation for application name (see table below), followed by the name of the action or feature the toolbar is used for, followed by “ToolBar”, all with no spaces.	ECBOMListToolBar TMCProjectsToolBar APPRouteSummaryToolBar SPCTechSpecActionsToolBar
Command objects for toolbar items	3-letter standard abbreviation for application name (see table below), followed by the name of the action or feature the link performs, followed by “ActionLink”, all with no spaces.	SPCCopySpecificationActionLink APPDeleteRouteActionLink

