







**Note**

Before using this information and the product it supports, read the information in Appendix A, "Notices," on page 25.

**First Edition (May 2005)**

This edition applies to version 2.5 of IBM Workplace Collaboration Services API Toolkit (product number L-GHUS-653JTG) and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright International Business Machines Corporation 2005. All rights reserved.**

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

---

# Contents

## Chapter 1. IBM Workplace Client

### Technology user interface design. . . . 1

User interaction in the IBM Workplace rich client . . .	1
The IBM Workplace personality . . . . .	1
Application look and feel . . . . .	2
Adding your application to the user interface . . . .	2
Testing your application . . . . .	3
User interface components. . . . .	4

### Chapter 2. User interface layout . . . . 5

Title bar . . . . .	5
Menu bar . . . . .	6
Menu contributions . . . . .	7
Global menu contributions. . . . .	7
File menu . . . . .	7
Preferences . . . . .	8
Edit menu . . . . .	9
View menu . . . . .	9
Actions menu . . . . .	9
Tools menu . . . . .	10
Window menu . . . . .	10
Help menu . . . . .	10
Part-associated action set menu contributions . .	11

Context menus . . . . .	11
Icons in menus . . . . .	12
Search bar . . . . .	12
Search results. . . . .	12
Switcher bar . . . . .	13
Data area . . . . .	14
Folder list . . . . .	15
Accessible views. . . . .	15
Tabs . . . . .	16
Action bar. . . . .	17
Document modes . . . . .	18
Toolbars . . . . .	18
Status bar . . . . .	19

### Chapter 3. Capitalization and punctuation guidelines. . . . . 21

### Chapter 4. Messages. . . . . 23

### Appendix A. Notices . . . . . 25

### Appendix B. Trademarks . . . . . 27



---

## Chapter 1. IBM Workplace Client Technology user interface design

These user interface design and interaction guidelines are intended for designers and developers who use IBM Workplace Client Technology to build applications for the IBM Workplace rich client. The guidelines provide information and examples of the common user interface components used to build the user interface of the rich client products provided with IBM Workplace 2.5.

IBM Workplace Client Technology is based on Eclipse, which has its own user interface guidelines that you can refer to in addition to these guidelines. To view the Eclipse User Interface Guidelines, go to:

<http://www.eclipse.org/articles/Article-UI-Guidelines/Contents.html>.

### Related concepts

“User interaction in the IBM Workplace rich client”

“The IBM Workplace personality”

“User interface components” on page 4

Chapter 2, “User interface layout,” on page 5

Chapter 3, “Capitalization and punctuation guidelines,” on page 21

Chapter 4, “Messages,” on page 23

---

## User interaction in the IBM Workplace rich client

The IBM Workplace rich client user interface is similar in appearance and action to other user interfaces, such as Microsoft Windows, Macintosh, and Motif. Users employ a “selection/action” model to interact with it.

In a selection/action model, selection pertains to each view and is independent of other selections in other views. The view retains what is called selection memory. For example, when a user selects one or more items in a list in one view, and goes to a different view (in a different plug-in or the same one), and then returns to the list in the original view, the selected items are still selected. This selection model helps users remember where they were and what they were doing in the view.

Inactive selection refers to the display of a previous selection at the same time that the active selection is displayed. All actions only take place on active selections, but continuing to display the inactive selection helps users remember the choices they have made.

### Related concepts

Chapter 1, “IBM Workplace Client Technology user interface design”

---

## The IBM Workplace personality

The layout of your application is defined by a perspective, which is the Eclipse equivalent of a window. You set the perspective you want the IBM Workplace rich client to use by specifying a personality. The personality of an application defines the framework the platform uses to determine what perspectives, menus, toolbars, action bar items, and status line controls display when the application starts, as well as what services are available and what sequence of events or what life cycle should be applied to the objects associated with that application.

The client provides a default IBM Workplace personality. In order for applications to plug into the IBM Workplace rich client environment, have a consistent look and feel with other rich client offerings, be available from the Switcher bar, and participate seamlessly with the life cycle events of the platform, you must specify the IBM Workplace personality. You instruct the client to start up using the IBM Workplace rich client personality by doing one of the following actions:

- Append the following parameter to the rich client start command:  
`<client_home>\rcp\richclient -personality com.ibm.workplace.personality`
- Edit the `plugin_customization.ini` file for your plug-in to include the following key:  
`com.ibm.rcp.platform/DEFAULT_PERSONALITY_ID="com.ibm.workplace.personality"`

where `com.ibm.workplace.personality` is the ID of the IBM Workplace personality.

#### **Related concepts**

Chapter 1, “IBM Workplace Client Technology user interface design,” on page 1

---

## **Application look and feel**

The look and feel of an IBM Workplace application is defined by the `LWPPresentationFactory` class. The `LWPPresentationFactory` class is based on the Eclipse presentation factory extension point `org.eclipse.ui.workbench.presentationFactories`. It defines the graphics to use for branding, the size of the title bar, the coloring to use for the title bar and tab background, the title bar and tab text, and the folder view background to associate with an application’s views.

The `LWPPresentationFactory` is implemented, by default, by the IBM Workplace personality. The IBM Workplace personality instantiates it using the following code:

```
public void preWindowOpen(IWorkbenchWindowConfigurer configurer) {  
    // set the presentation factory  
    configurer.setPresentationFactory(new LwpPresentationFactory());  
}
```

When you implement the IBM Workplace personality, the formatting of the user interface is handled for you by the presentation factory.

---

## **Adding your application to the user interface**

To add your application to the IBM Workplace rich client user interface, you must deploy your application to a WebSphere Portal server and provision it to the rich client from there.

You create portlets on WebSphere Portal to represent your application’s views. You define the layout of the views within the perspective of your application when you add the portlets to a WebSphere Portal page and define the layout of the portlets on WebSphere Portal. You can think of an Eclipse view as being equivalent to a WebSphere Portal portlet and an Eclipse perspective as being equivalent to a WebSphere Portal page. When you create these components to represent your application on WebSphere Portal and set the page to be read as Rich Client Markup Language (RCPML), WebSphere Portal produces an RCPML-formatted file to represent the page. The `com.ibm.rcp.pagebuilder` plug-in included with the rich client platform, interprets the RCPML in the resulting file that defines the page and builds the perspective and views on the client accordingly.

For details on the steps required to deploy your application to WebSphere Portal, see the *IBM Workplace Client Technology API Toolkit User's Guide*.

You cannot programmatically define the perspective for your views; you must define the application views in RCPML if you want IBM Workplace to host them, unless the application is built to be accessed as a stand-alone application using a custom personality.

---

## Testing your application

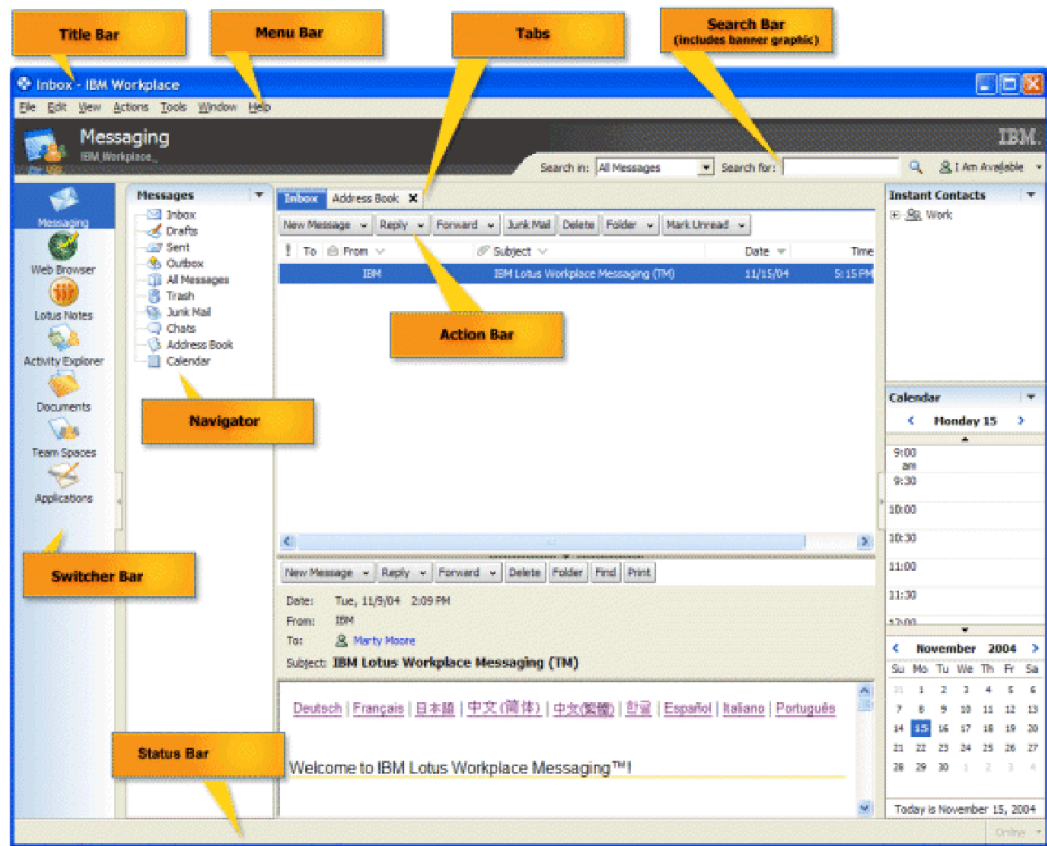
If you are building an application to run on the IBM Workplace rich client, you use the IBM Workplace personality that you define through RCPML to specify the layout of your application. After deploying your application to WebSphere Portal and provisioning it to the client machine, you can test the provisioned application.

To test your provisioned application:

1. Populate your application by defining its views using portlets and its perspective using a WebSphere Portal page on WebSphere Portal.
2. Provision the client from the server. See the IBM Workplace Collaboration Services information center.
3. Copy the cached RCPML in `<workspace>\.metadata\.plugins\com.ibm.rcp.pagebuilder` to another folder, such as `c:\rcpml`
4. In the Program Arguments field of the Eclipse runtime environment page, add `"-pbcache c:\rcpml."` Now you can run your application in the integrated development environment (IDE) and make changes to the local RCPML files to test minor changes without having to repeat the provisioning process.

## User interface components

This figure of the IBM Workplace rich client highlights the user interface components discussed in this guide.



The following parts of the IBM Workplace rich client user interface are displayed on every page, by default, unless the user chooses to hide them:

- Menu bar
- Search bar, displaying only the banner graphic if search is not available.
- Status bar
- Switcher bar

### Related concepts

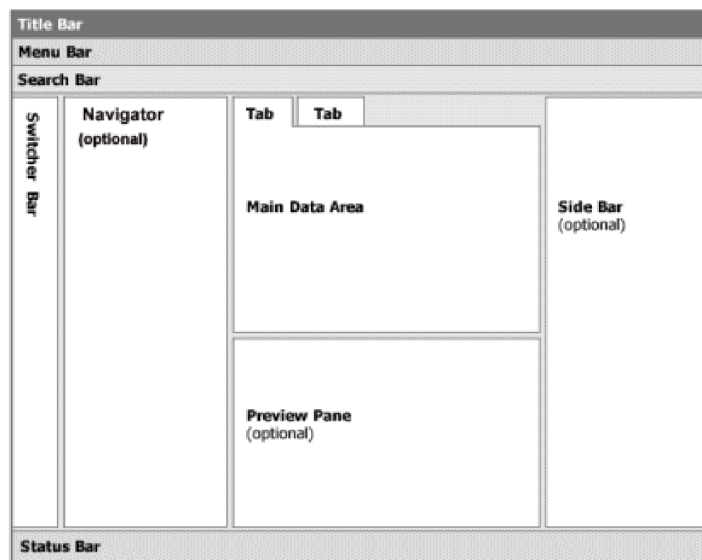
Chapter 1, "IBM Workplace Client Technology user interface design," on page 1

---

## Chapter 2. User interface layout

The IBM Workplace rich client products are built on IBM Workplace Client Technology. These products, such as IBM Workplace Messaging and IBM Workplace Documents, implement the IBM Workplace rich client user interface layout.

The following figure illustrates the basic components of the IBM Workplace rich client user interface. This guide describes how to build user interface components to contribute to the layout, starting from the top of the user interface layout and ending at the bottom.



### Related concepts

Chapter 1, "IBM Workplace Client Technology user interface design," on page 1

"Title bar"

"Menu bar" on page 6

"Search bar" on page 12

"Switcher bar" on page 13

"Data area" on page 14

"Folder list" on page 15

"Tabs" on page 16

"Action bar" on page 17

"Document modes" on page 18

"Toolbars" on page 18

"Status bar" on page 19

---

## Title bar

All applications built for the IBM Workplace rich client have a title bar. The following format is applied to the title bar:

<Application icon> <Name of view that has focus> <dash(-)> <Application name>

For example, the following name is displayed in the title bar for the IBM Workplace Messaging rich client product, when the Inbox view is displayed:

- Inbox - IBM Workplace Messaging

The title bar derives its content from individual contributing views. The only information that you provide to the title bar is the content of the <Name of the view that has focus> value. The <Name of the view that has focus> displays the view title you assign to a view in the name attribute of the view's plugin.xml file, if the view has a title. For example, the title of the Inbox view as it is defined in its plugin.xml file is:

```
%str.client.Inbox.label
```

The plugin\_en.properties file translates this key to "Inbox."

#### **Related concepts**

Chapter 2, "User interface layout," on page 5

---

## Menu bar

All features of an application built to work with IBM Workplace rich client products must be available from the menu items on the menu bar. They also can be available from buttons or context menus. Context menus are displayed when a user right-clicks the background of a user interface component, such as a view or document.

In general, menus should display all menu items that are applicable to the current view.

If	Then
A menu item is not applicable to what is currently selected, but the user can take an action while in the same tab or window to enable the menu item,	that menu item should appear dimmed.
A menu item is not applicable to what is in the view,	that menu item should be hidden.

Pull-right menus should always be enabled, even if none of the items in the pull-right menu are available. Users should be able to view the contents of the pull-right menu, even if none of the actions are available.

#### **Related concepts**

Chapter 2, "User interface layout," on page 5

"Menu contributions" on page 7

"Preferences" on page 8

"Part-associated action set menu contributions" on page 11

"Context menus" on page 11

"Icons in menus" on page 12

#### **Related reference**

"Global menu contributions" on page 7

"File menu" on page 7

"Edit menu" on page 9

"View menu" on page 9

"Actions menu" on page 9

- “Tools menu” on page 10
- “Window menu” on page 10
- “Help menu” on page 10

## Menu contributions

You can make a feature you have created available to users by adding it to the menu bar as a menu item. You can create and contribute menu items by implementing one of the following contribution types:

- Global menu contributions -- These menu contributions persist across every view of a specific personality in the client. These menu items are universal and can be used from any context. These menu items are also “retargetable,” so your application can write code to retarget these global menu items, such as Cut, Copy, and Paste, for application-specific purposes.
- Part-associated action set contributions -- These menu contributions are specific to a single view. By default, most menu items are local, meaning that if the view associated with those menu items is not displayed, those menu items are not displayed. Contributing view-specific menu items enables you to create discrete menu items for a page of your application that are not shared with other products on the client.

For more information on contributing to menus, refer to the *IBM Workplace Client Technology API Toolkit User’s Guide*.

### Related concepts

- “Menu bar” on page 6

## Global menu contributions

When you implement the IBM Workplace personality, the following standard global menu item contributions are implemented automatically, in the following order:

<u>F</u> ile	<u>E</u> dit	<u>V</u> iew	<u>A</u> ctions	<u>T</u> ools	<u>W</u> indow	<u>H</u> elp

The underlined letters represent access keys that are defined for the menu options to enable the use of keyboard shortcuts.

### Related concepts

- “Menu bar” on page 6

## File menu

The global items on the File menu are listed below. The underlined letter in the menu item name identifies the access key for the menu item.

Close

Work offline

—

Print

Properties

Preferences

## Exit

### Placement of menu items

The guidelines for the placement of menu items contributed to the File menu are as follows:

- Close appears in the first section of the menu after menu items such as New, Open, and Import when an application includes them in the File menu.
- Work offline appears after Close, but other menu items can be placed between them.
- Print, Properties, Preferences, and Exit are the last four menu items in the menu, unless the application displays a list of the most recently used windows. If the application displays one, place the list of most recently used windows between Preferences and Exit.

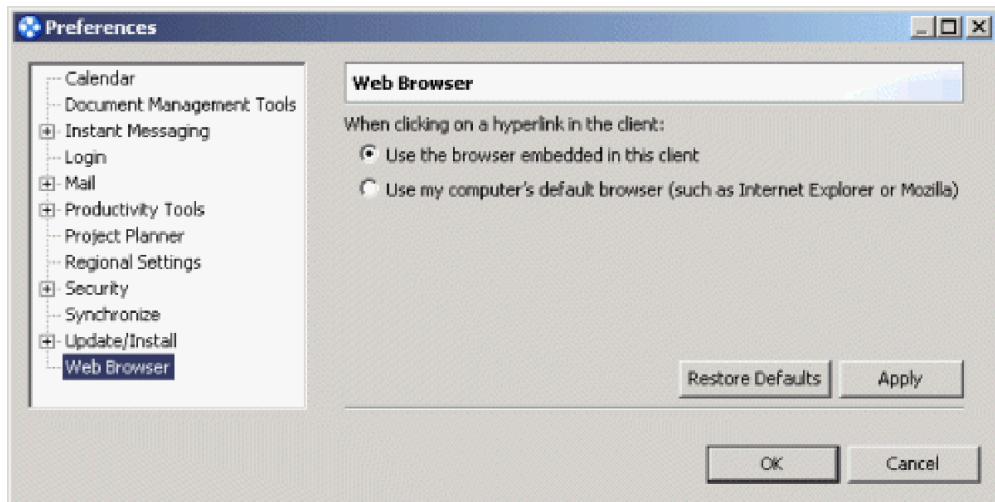
#### Related concepts

“Menu bar” on page 6

## Preferences

The rich client provides a preferences interface and implementation to store and provide preferences for individual plug-ins.

Plug-in preferences are key/value pairs, where the key describes the name of the preference, and the value is one of several different types, including boolean, double, float, int, long, and string. The Eclipse platform provides support for storing plug-in preferences and showing them to the user on pages in the workbench Preferences dialog box. The following graphic illustrates the Web Browser pane of the Preferences dialog box.



You do not have to define a menu item to display the Preferences dialog box. A global **File** → **Preferences** menu option is provided with the IBM Workplace rich client personality.

For more information, see the *IBM Workplace Client Technology API Toolkit User's Guide*. When you write the code that defines the content of the preference page, follow these formatting rules:

- Use group boxes to separate areas, if you feel that grouping is necessary. Capitalize only the first letter of the first word of the group box heading.
- Begin each preferences page with a sentence that describes what the user can do.
- Add a colon after field labels.
- Always provide the Restore Defaults and Apply buttons. You can add other command buttons as necessary.

**Related concepts**

“Menu bar” on page 6

## Edit menu

There are no global menu items in the Edit menu, although Undo, Cut, Copy, Paste, and Select All should be on this menu if your application supports such actions.

**Related concepts**

“Menu bar” on page 6

## View menu

The View menu contains a Show menu item. The Show menu item is a pull-right menu that allows users to show or hide the basic parts of the user interface. The following table lists the menu items available from the Show pull-right menu. The underlined letters identify the access keys.

Show

Search Bar

Preview Pane

Status Bar

Side Bar

Switcher Bar

Labels on the Switcher Icons

### Placement of menu items

You can add other menu items to the Show pull-right menu, before or after these menu items.

**Related concepts**

“Menu bar” on page 6

## Actions menu

The global menu item is:

**Maximize Current Tab**

This is a toggle menu item; clicking or double-clicking it maximizes the tab. Clicking or double-clicking it again, minimizes it.

### Placement of menu items

- Maximize Current Tab must be the last menu item in the menu.

- Include menu items in the Actions menu to trigger the actions you contribute to action bars in your application.

**Related concepts**

“Menu bar” on page 6

## Tools menu

The global menu item is:

**Synchronize Now**

### Placement of menu items

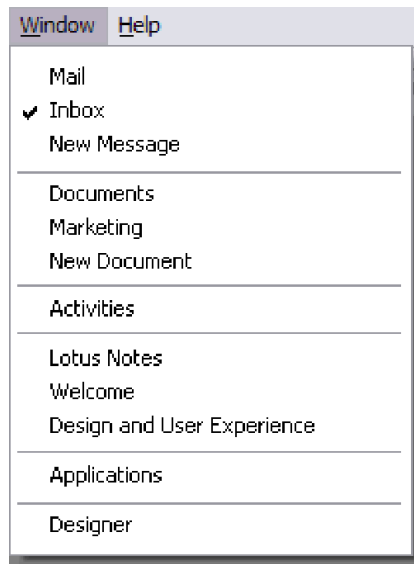
Synchronize Now should be the first menu item.

**Related concepts**

“Menu bar” on page 6

## Window menu

The Window menu lists each possible perspective and each open view within each perspective, as shown in the following figure.



### Placement of menu items

The Window menu should be adjacent to the Help menu item, so that it is the next-to-last menu item on the menu bar.

**Related concepts**

“Menu bar” on page 6

## Help menu

The global Help menu items are:

**Help Contents**

The accelerator key for this menu item is F1.

## About IBM Workplace

### **Related concepts**

“Menu bar” on page 6

## **Part-associated action set menu contributions**

In addition to having the global menus, each product (for example, Mail or Document Management) contributes its own menu items on the existing File, Edit, View, Actions, Tools, Window, and Help menus. A product can also contribute entire menus to the menu system.

The View, Actions, and Tools menus change based on the application that has focus. For example, when a Document Manager user views a list of documents in a library, the View menu includes a **Sort By** menu item for sorting documents by author or by file name. When the user opens a document, the View menu hides the **Sort By** menu item.

You can contribute entire menus to the menu bar. For example, the IBM spreadsheet editor contributes a Format menu to the menu bar when it has focus and is in edit mode. The administrator can use a user policy to make groups of features available or unavailable. For example, the administrator can exclude the calendar from mail. In this case, all menu items associated with the calendar would be hidden when the user is using mail.

Conforming to the part-associated menu contribution rules enables the IBM Workplace rich client to act as a pluggable environment and supports the user policy mechanism that enables administrators to pick and choose the applications to enable for an organization. If an administrator sets a policy to make an application, such as Document Management, unavailable, all the menu items associated with Document Management are automatically hidden from the menu.

### **Related concepts**

“Menu bar” on page 6

## **Context menus**

A context menu is the menu that is displayed when a user right-clicks the background of a user interface component, such as a view or document. Context menus should repeat pertinent menu items that are available on the menu bar or pertinent actions on a dialog box triggered from a menu item. The contents of the context menu must change to be appropriate to the object that has focus or the object that is selected.

A context menu item should not be the only way a user accesses a piece of functionality. All context menu items must have access keys for accessibility. Indicate the access key by underlining the key text in the menu item label. If possible, use the same access key as the one used for the menu item in the menu bar.

Give the following objects context menus:

- Objects in the navigator view, such as mail folders or document libraries.
- Items selected in a list.

### **Related concepts**

“Menu bar” on page 6

## Icons in menus

If a menu item is represented by a toolbar button with an icon, include that icon on the menu. If your application does not have a toolbar or the toolbar does not include icons, do not include an icon on the menu. Include icons to the left of the menu items.

The IBM productivity tools provide an example of including icons in the menu appropriately. The tools display a toolbar within each tabbed window that corresponds to the specific tool. The menus for each productivity tool have menu items with icons that correspond to each toolbar button.

### Related concepts


“Menu bar” on page 6

---

## Search bar

The design goal is that all IBM Workplace rich client products are searchable.

The search bar consists of the following elements:

- Label of the tab that has focus -- The tab label that displays when the tab has focus should be the same label that is displayed in the Switcher bar.
- Search controls -- The search controls consist of:
  - **Search in** drop-down list. Include items in this drop-down list by extending the Search bar extension point. You should include the areas that users can search in for the active application. Some areas are always searchable. For example, People Finder, which is not tied to a specific product, is available at all times for users to search for people in the organization.
  - **Search for** drop-down list.
  - **Search icon** .

For example:



- Instant messaging controls.

If your application does not implement search capabilities, display the branding graphic and title, but hide the search controls within the search bar.

### Related concepts

Chapter 2, “User interface layout,” on page 5

“Search results”

## Search results

When users choose what to search, type text into the search bar, and click the Search button, the client runs the search code you implemented using the searchBarAdapters extension point. You must create a view that returns the list of results to the user in one of the following ways:

- If the user chooses to search a fully integrated product, such as mail, return the search results in a list as a new tabbed view in your application.
- If the user chooses to search an area that is not fully integrated, such as Google, return the results in one of the following ways:
  - In a tabbed window that displays in the embedded browser.

- In a tabbed window that displays in a browser window that runs separately from the IBM Workplace Client Technology platform.

If possible, display the search results in order of relevance.

For more information on extending the `searchbarAdapater` extension point, see the *IBM Workplace Client Technology API Toolkit User's Guide*.

#### Related concepts

“Search bar” on page 12

## Switcher bar

The switcher bar is a vertical bar on the left side of the client window that lists the available applications, each represented by an icon, from which users can select the service they want to start.

**Note:** The switcher bar is located on the right of the client window in bidirectional environments.

When a user clicks an icon, the client opens the corresponding application in the main data area. Each application represented on the switcher bar has a perspective defined for it. The perspective defines how to populate the window. The switcher bar provides a way for a user to switch from one perspective to another. It prevents multiple applications from having to share a single perspective. As a result, it allows for a cleaner, less cluttered user interface.

In addition to using the switcher bar, users can switch between open applications using the Window menu on the menu bar and through the keyboard shortcuts **Ctrl+F8** or **Shift + Ctrl + F8**.

When you provision an application to a client machine, the RCPML that defines the application also adds the application to the switcher bar. You specify the icon to display on the switcher bar to represent your application by providing an `iconpath` parameter value on the WebSphere Portal page that represents your application.

The following is a sample of the XML code produced by WebSphere Portal for the Call Center sample application that is provided to the client RCPML:

```
<content_node>
  ...
  <supported-markup markup="html" update="set"/>
  <supported-markup markup="rcpm1" update="set"/>
  <localedata locale="en">
    <title>RCPSDK Call Center</title>
  </localedata>
  <parameter name="bookmarkable" type="string" update="set">Yes</parameter>
  <parameter name="activities" type="string"
  update="set">com.ibm.workplace.demo.callcenter.activity</parameter>
  <parameter name="default_folder" type="string" update="set">main</parameter>
  <parameter name="iconpath" type="string"
  update="set">/lwpupdate/images/callcenter.png</parameter>
  ...
  <component>
    ...
  </component>
</content_node>
```

The following is an example of the resulting RCPML that defines what to display on the switcher bar:

```
<rcp>
  <config count="3">
    <page name="Content Root"
      url="/lwp/myworkplace!/ut/p/.cmd/cs/.ce/7_0_A/.s/7_0_A/_s.7_0_A/7_0_A"
      markups="html,rcpml,html,wml">
      <page name="WorkplaceRCPPages"
        url="/lwp/myworkplace!/ut/p/.cmd/cs/.ce/7_0_A/.s/7_0_LT/_s.7_0_A/7_0_LT"
        markups="rcpml">
        <parameters>
          <param name="lwprootpage">Yes</param>
        </parameters>
        <page name="RCP SDK Call Center"
          url=
            "/lwp/myworkplace!/ut/p/.cmd/cs/.ce/7_0_A/.s/7_0_18L/_s.7_0_A/7_0_18L"
          markups="rcpml">
          <parameters>
            <param name="bookmarkable">Yes</param>
            <param name="activities">
              com.ibm.workplace.demo.callcenter.activity
            </param>
            <param name="default_folder">main</param>
            <param name="iconpath">/lwpupdate/images/callcenter.png</param>
          </parameters>
        </page>
      </page>
    </page>
    ...
  </config>
</rcp>
```

The icons on the switcher bar are displayed in a flat list only; no nesting is supported.

#### **Related concepts**

Chapter 2, "User interface layout," on page 5

---

## **Data area**

The primary data area of the IBM Workplace rich client user interface consists of at least one open tabbed window. When setting the WebSphere Portal parameters for your application, design your application so that it contains a folder view. Associate with this folder at least one view that is set to be "fixed" and "visible." The result of these WebSphere Portal settings is a tabbed view that cannot be closed.

A tabbed window is either a document or view window, built using an Eclipse view. The following rules apply to all the windows that are displayed in the data area:

- Each window must have a tab with a tab label and an action bar.
- A new tabbed window must open for each document opened from a list view.
- List views must open in a new tabbed window on top of other tabbed views in the data area. The tab of each view remains visible but only the view associated with the active tab is displayed. For example, the Messaging plug-in opens a tabbed window for the Inbox view and opens other Mail views, such as the Draft view, on top of the Inbox view when requested by a user. The Draft view appears stacked on top of the Inbox view. The Inbox view's tab remains visible so that a user can click it to switch back to the Inbox view.

#### **Related concepts**

## Folder list

A folder list is a tabbed list view that is displayed in the data area of the user interface layout. You create the folder that the folder lists are displayed in by creating a new portlet on WebSphere Portal and specifying the following parameters for it:

- Title -- A name that identifies this portlet as a folder. Main Folder is an example of a title.
- folderid -- The ID that any tabbed views that are displayed in this folder use to associate themselves with this folder. main is an example of a folderid value.
- ratio -- The percentage of the window that the folder should occupy. .25 is an example of a ratio value.

To make a view display as a tabbed view in this folder, create a new portlet to represent the view. Set the folder portlet's folderid value as the folderid parameter for the view portlet. See the *IBM Workplace Client Technology API Toolkit User's Guide* for more information on installing a feature on WebSphere Portal.

Be sure also to specify a default folder for your application. The folder you define as the default folder is the folder that any view that is programmatically started appears in. For example, if you define the main folder as the default folder and set the main folder to display in the data area when you define the page layout, when you double-click to open a selected document in a list view, the document opens in the data area. See the *IBM Workplace Client Technology API Toolkit User's Guide* for more information on adding parameters to a WebSphere Portal page.

Each folder list should include an action bar and a view from which users can perform the following tasks:

- Resize each column by clicking a cell divider and dragging it.
- Sort the view by column, such as by date or by person.
- Display a ToolTip for each column head.
- Select one or more items in the list and take action on them. For example, select several documents and move, copy, delete, or open them.
- Select an item in the list and display it in the Preview window.

### **Related concepts**

Chapter 2, "User interface layout," on page 5

"Accessible views"

## Accessible views

You can increase the accessibility of your views by providing keyboard shortcuts. Keyboard shortcuts are combinations of keys, such as Ctrl + T, that initiate actions that are available from a menu or action bar. These shortcuts provide an alternate way to trigger an action.

Implement the following keyboard shortcuts in your views:

- Up arrow -- Moves the cursor up the list view.
- Down arrow -- Moves the cursor down the list view or drop-down list activated from the action bar.

- Tab key -- Cycles clockwise through the user interface components in the view, including the action bar and its buttons, the view column heads, and the view list entries.
- Shift+Tab -- Cycles counterclockwise through the user interface components in the view, including the view list entries, the view column heads, and the action bar and its buttons.
- Enter -- Opens an item selected in a list or triggers an action in an action bar or action bar drop-down list.
- Tab -- Moves the cursor from the action buttons to the column headers to the items in the list view.
- Home -- Gives focus to the first item in the list view.
- End -- Gives focus to the last item in the list view
- Ctrl+F7 -- Displays a pop-up menu that lists the currently open views and cycles through them on consecutive uses.
- SHIFT+Ctrl+F7 -- Cycles through the pop-up menu that lists the currently open views from bottom to top.
- Shift +F10 -- Displays the context menu for the view.

#### Related concepts

“Folder list” on page 15

---

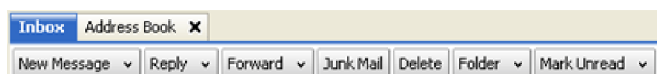
## Tabs

Standard views that you add to the rich client data area must be contributed as tabbed views. You create a tabbed view by associating your view plug-in to a folder when you create a portlet to represent it in Websphere Portal. Create a folder portlet and create at least one view portlet and associate it with the folder by specifying its ID in the view portlet’s folderid attribute. Set the view portlet to be “visible” and “fixed” to establish it as the default tabbed view. The default view’s tab is called the primary tab. The view that contains the primary tab is the view that is displayed when a user first opens or switches to your application’s perspective. Users should not be able to close the primary tab view. Use the following method of the Eclipse IViewLayout interface to prevent the tabbed view from containing a Close icon:

```
setFixed()
```

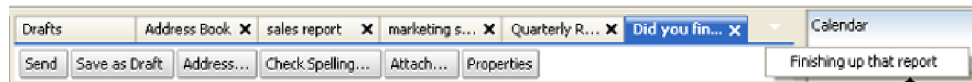
You can set additional views to display as tabbed views in the folder by creating view portlets for each of them and setting the portlet folderid attribute to the ID of the folder.

For example, the Inbox view tab in the graphic below is the primary tab. The Address Book view, which has a close icon on its tab indicating that the view can be closed, is contributed to the same folder as the Inbox view.



The tabs that display on views you associate with a folder are automatically formatted by the LWPPresentationFactory, which is a class that defines the look and feel of IBM Workplace rich client products. The LWPPresentationFactory class applies the following formatting to the tabs automatically:

- Tab width -- (the tab maximum width is approx. 35 characters and the minimum is 3). An overflow control appears automatically if necessary, as shown below.



- Hover behavior -- (hover text should show the full name of the item that the tab represents)
- Double-click behavior -- Users can double-click the tab to maximize the tab window to fill the entire area of the Workplace client window. Double-clicking the title bar of that maximized window returns the window to a tab.

You must apply the following formatting to the tabs explicitly:

Create a Close menu item in the File menu and the Context menu for all views and documents that users can close.

The tab label displays the title of the list or document view that the tab is associated with. By default, the title is defined by the name attribute of the view's plugin.xml file. You can also specify a title programmatically using the following method of the Eclipse IViewSite interface:

```
setTitle()
```

Use the setTitle() method to define the view and document titles according to the following guidelines:

- If the tab is associated with a new document that has not been saved, format the tab label as New <insert descriptor> . The following tabs are examples:
  - New Message
  - New Meeting
  - New Spreadsheet
- If the tab is associated with a document that has been saved, or received if it is a mail message, format the tab label as <subject or name of the document>. Apply the following rules for formatting the tabs:
  - An open mail message contains the beginning of the mail message's Subject field.
  - An open meeting invitation contains the beginning of the meeting's Subject field. Set the tab label to Canceled:<subject field> if a meeting is canceled.
  - An open address book entry contains the entry's Name field.
  - A saved document contains the file name of the document.

#### **Related concepts**

Chapter 2, "User interface layout," on page 5

---

## **Action bar**

The action bar provides quick access to commonly used actions and commands. Action buttons should trigger actions that users can also access by selecting menu items on the Actions menu. If the action bar buttons cannot fit on the visible area of the action bar, an overflow button appears. When clicked, the additional actions are displayed in a drop-down list.

Use the ActionBarManager API to create an action bar in each open document, folder view, or list view. Apply the following formatting to the action bar:

- Set spacing between each button to 2 pixels using the following method:

```
actionBarManager.setSpaceWidth(2);
```
- Set the rounded button renderer using the following method:

```
actionBarManager.setCustomRenderer(new RoundedButtonRenderer());
```

- Set the background to white using the following method:

```
actionBarManager.setControlBackground  
(Display.getCurrent().getSystemColor(SWT.COLOR_WHITE));
```
- If the action bar includes a Properties command, add it as the last button on the action bar.

The following illustration is an example of a IBM Workplace action bar:



You can create the following types of action buttons:

- A simple button -- Clicking the button executes the action.



- A drop-down list -- Clicking the down arrow button displays a menu. Clicking a menu item executes the action.



- A combined drop-down list and button -- Clicking the button activates the default command, while clicking the down arrow button displays a menu with other options.



#### **Related concepts**

Chapter 2, “User interface layout,” on page 5

---

## **Document modes**

Users can access documents in the following modes:

- Read mode -- Contains properties fields, an optional set of fixed fields and an optional rich text area. A mail message and calendar entry are read-mode documents.
- Edit mode -- Contains a rich text area and an optional set of fixed fields. A spreadsheet is an edit-mode document.

All documents have a Context menu, which is displayed when a user right-clicks. The commands available on the menu must be specific to the mode the document is in.

#### **Related concepts**

Chapter 2, “User interface layout,” on page 5

---

## **Toolbars**

The IBM Workplace rich client does not display a global toolbar under the main menu bar. However, individual applications can provide a toolbar within specific tabbed windows. For example, IBM Workplace Documents allows users to edit a spreadsheet. When editing the spreadsheet, users see one or more toolbars within the spreadsheet tabbed window.

#### **Related concepts**

Chapter 2, “User interface layout,” on page 5

---

## Status bar

The global menu items provided with IBM Workplace rich client include a **View** → **Status bar** menu item that users can toggle to show or hide the status bar. The status bar is displayed by default.

Use the ControlSet extension point to contribute content to the status bar. See the *IBM Workplace Client Technology API Toolkit User's Guide* for more information.

### **Related concepts**

Chapter 2, "User interface layout," on page 5



---

## Chapter 3. Capitalization and punctuation guidelines

Use appropriate punctuation, such as a periods, exclamation points, or question marks at the end of complete sentences. Add a colon at the end of labels for controls in forms and dialog boxes.

For more information, see the Eclipse User Interface guidelines.

The following table describes when to use headline-style capitalization and when to use sentence-style capitalization:

Capitalization style	Guideline	Use for
Headline	Capitalize the first letter of each word, except the following: <ul style="list-style-type: none"><li>Articles such as: a, an, and, the</li><li>Short prepositions that are between words such as: for, in, of, on, and to.</li></ul>	<ul style="list-style-type: none"><li>Command buttons (push buttons)</li><li>Dialog box title bars</li><li>Menu items</li><li>Menu titles</li><li>Section headers (For example, the header to a section on a form. Section headers should also be bold.)</li><li>Tabs</li><li>Title bars</li><li>ToolTips (When the tooltip is for a toolbar item; all other tooltips use sentence-style capitalization.)</li><li>Window titles</li></ul>
Sentence	Capitalize the first letter of the first word and any proper nouns, such as Workplace.	<ul style="list-style-type: none"><li>Check box labels</li><li>Dialog box labels</li><li>Group box or group bar titles</li><li>Radio buttons</li><li>Text field labels</li></ul>

### Related concepts

Chapter 1, "IBM Workplace Client Technology user interface design," on page 1



---

## Chapter 4. Messages

Whenever possible use the standard message dialog boxes provided in the `MessageDialog` class as part of the `org.eclipse.jface.dialogs` package. The error message can be modeless, which means it allows the user to continue to interact with the application or modal, which means it requires that the user respond to the error dialog box before continuing to use the application. The following sections describe the available Eclipse message types.

### Critical

Informs users of a serious problem that prevents them from continuing their work. Critical error messages are always modal.

Example code:

```
MessageDialog.openError  
(parent.getShell(),  
"Error Title",  
"Error Message");
```

Example of a Critical message:



### Information

Provides information about the results of a command. Users can acknowledge only the message. Information messages are often modeless. However, use your judgment on whether the user needs to respond to the error before continuing to interact with the rest of the application.

Example code:

```
MessageDialog.openInformation  
(parent.getShell(),  
"Info TitleApplicationName",  
"Info Message");
```

Example of an Information message:



## Warning

Alerts users to a condition that requires a decision before proceeding. In many cases you must add buttons to support the Yes, No, and Yes, No, Cancel conditions. Warning messages are usually modal; that is, users must respond before continuing to interact with the application.

Example code:

```
MessageDialog.openWarning  
(parent.getShell(),  
"Warning Title",  
"Warning Message");
```

Example of a Warning message:

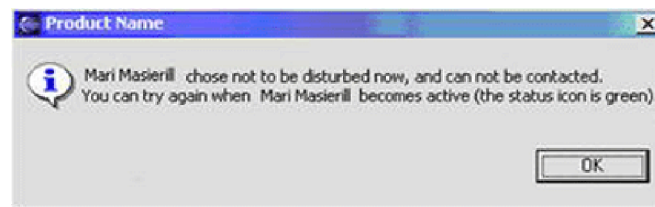


## Custom messages

When a standard message cannot be used, you can create a custom message dialog box. Follow these general guidelines for writing the error message content:

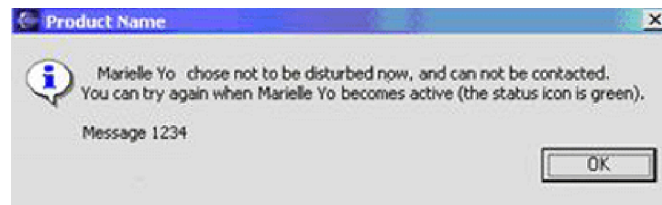
- The message box title is the application name.
- In the first sentence, indicate what went wrong.
- In the second sentence, tell the user what to do about it.

For example:



- Avoid the use of error message numbers, if possible. If you do include a message number, put it at the end of the message.

For example:



- Associate a sound with the error message for accessibility. Use a default system beep.

### Related concepts

Chapter 1, "IBM Workplace Client Technology user interface design," on page 1

---

## Appendix A. Notices

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

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

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

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

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

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

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

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

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

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

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

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

IBM Corporation  
Office 4360  
One Rogers Street  
Cambridge, MA 02142  
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 information 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.

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.

---

## Appendix B. Trademarks

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

IBM  
Lotus  
Workplace  
Workplace Messaging

Windows is a trademark of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.







Part Number: C80P6IE  
Program Number: 5724i2

Printed in USA

G210-2186-00



(1P) P/N: C80P6IE

