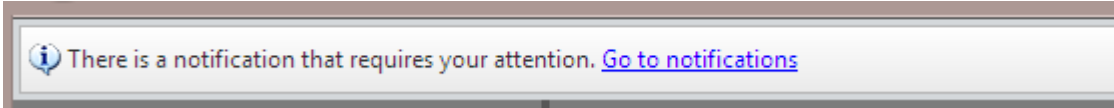


Message bar items



Message Bar items are messages with an accompanying hyperlink that, when clicked, navigates the user to a location in the navigation tree.

Adding Message Bar items

Add a message bar item with `NavInsertItem`.

```
void AddMessageBarItem(LPCWSTR text, LPCWSTR link)
{
    // First get the root of the MESSAGEBAR domain
    NAVSETITEM messageBarRootData;
    memset(&messageBarRootData, 0, sizeof(NAVSETITEM));
    if( !NavGetItem(_T("MESSAGEBAR"), &messageBarRootData) )
    {
        return; // Failed to find MESSAGEBAR root
    }
    NavHandle messageBarRoot = messageBarRootData.hItem;

    // Now add a simple message bar item
    NAVSETITEM newItem;
    memset(&newItem, 0, sizeof(newItem));
    newItem.flags = NI_FLAG_MESSAGEBAR_ICONWARNING; // The icon we want to show.
    NI_FLAG_MESSAGEBAR_ICONINFO, NI_FLAG_MESSAGEBAR_ICONWARNING and
    NI_FLAG_MESSAGEBAR_ICONSHIELD
    newItem.EventMask = NI_EVENT_MOUSE_LCLICK; // Need click events to navigate
    somewhere when the link is clicked
    newItem.strLabel = L"This is a message bar message";
    newItem.strDestination = L"NAVIGATIONTREE\\ControlPanel\\Events"; // Override
    description field as the FQN or alias to navigate to when the link is clicked
    newItem.lParam = 0; // Custom context data
    newItem.pInterface = &notificationInterface; // Interface to handle messages
    from the message bar
    NavInsertItem(messageBarRoot, _T("MyComponentWarningEvents"), &newItem);
}
```

Items added in the manner shown above display the `strLabel` text as the message, followed by a link that says "Go There". If you click the link, the WMC navigates to the location specified in `strDestination`. You can provide a custom formatted message and link by specifying the `NI_FLAG_MESSAGEBAR_CUSTOMMESSAGE` flag . See [Custom link text](#).

The flags field of the `NAVSETITEM` structure dictates how the message bar item will be handled.

Flag	Description
NI_FLAG_MESSAGEBAR_ICONWARNING NI_FLAG_MESSAGEBAR_ICONINFO NI_FLAG_MESSAGEBAR_ICONSHIELD	The type of icon to show. Mutually exclusive.
NI_FLAG_MESSAGEBAR_NOTIFY	Notify the item of the click as opposed to navigating to the location. Calls the <code>NavItemNotify</code> handler of the notification interface supplied in <code>NavInsertItem</code> , passing in the <code>strDestination</code> as the FQL. Without this flag the default behaviour is to go to a navigation location.
NI_FLAG_MESSAGEBAR_CUSTOMMESSAGE	Specify to supply message text with an embedded link with customised text. See below for details.

Custom link text

By default a message bar item contains user-supplied text followed by a hyperlink with the text ***Go There***. You can supply a custom message that allows you to set the hyperlink text as well as place it anywhere within the message. To do so set the flag `NI_FLAG_MESSAGEBAR_CUSTOMMESSAGE` in flags field of the `NAVSETITEM` structure and provide a message in `strLabel`, enclosing the hyperlink portion in `<a>` tags. e.g.

```
void AddMessageBarItem(LPCWSTR text, LPCWSTR link)
{
    // First get the root of the MESSAGEBAR domain
    NAVSETITEM messageBarRootData;
    memset(&messageBarRootData, 0, sizeof(NAVSETITEM));
    if( !NavGetItem(_T("MESSAGEBAR"), &messageBarRootData) )
    {
        return;           // Failed to find MESSAGEBAR root
    }
    NavHandle messageBarRoot = messageBarRootData.hItem;

    NAVSETITEM newItem;
    memset(&newItem, 0, sizeof(newItem));
    newItem.flags = NI_FLAG_MESSAGEBAR_CUSTOMMESSAGE;
    newItem.EventMask = NI_EVENT_MOUSE_LCLICK;

    newItem.strLabel = L"This is the text <a>with an embedded link</a> to navigate to";
    newItem.strDestination = _T("{Notifications}");    // where to go to
    newItem.lParam = (LPARAM)this;
    newItem.pInterface = &notificationInterface;
    NavInsertItem(hMessageBar, _T("NotificationWarning"), &newItem);
    NavHandle messageBarItem = newItem.hItem;
}
```