



[All Topics](#), [MFC / C++ >> Dialog and Windows >> General](#)  
<http://www.codeproject.com/dialog/cfolderdialog.asp>

VC7, VC6, XP, W2K,  
Win9X, MFC

Posted 27 Mar 2002

Updated 23 Jul 2003

60,665 views

## CFolderDialog - Selecting Folders

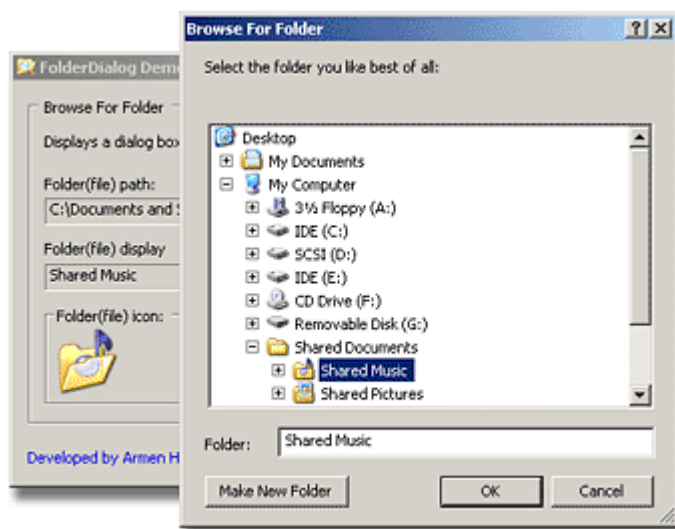
By [Armen Hakobyan](#)

The CFolderDialog class allows you to add a folder-selection dialog box to your applications

43 members have rated this article. Result:

Popularity: [7.79](#). Rating: [4.77](#) out of 5.

- [Download executable - 15.0 Kb](#)
- [Download project - 33.0 Kb](#)
- [Download source only - 4.0 Kb](#)



## Introduction

As I mentioned in my other article "[CIconDialog - Selecting Icons](#)", recently, developing one wizard application I needed one dialog to select an icon from executables and another one for selecting folders on multiple ones, but did not find anything about in MFC. So, [CFolderDialog](#) was written. It wraps the `SHBrowseForFolder` API.

## Sample Usage

The [CFolderDialog](#) is derived from [CCommonDialog](#) and acts like any common dialog. See sample usage:

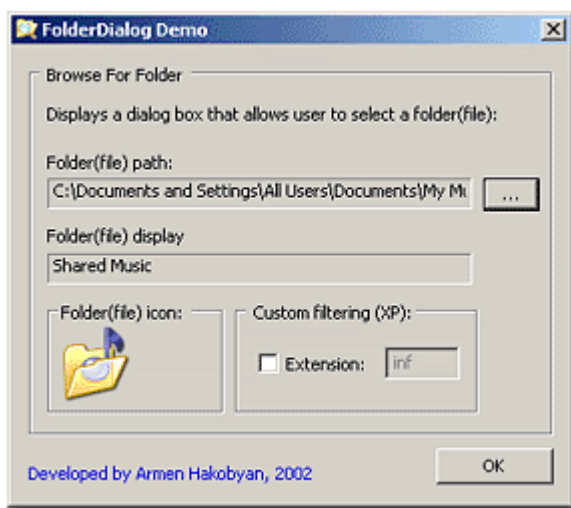
```
//...
#ifndef __FOLDERDLG_H__
#include "FolderDlg.h"
#endif
// ...
void CSomeDialog::OnSomeHandler( void )
{
    CString strFolderPath( _T( "c:\\Winnt" ) ); // Just for sample
```

```

CString strDisplayName( _T( " " ) );

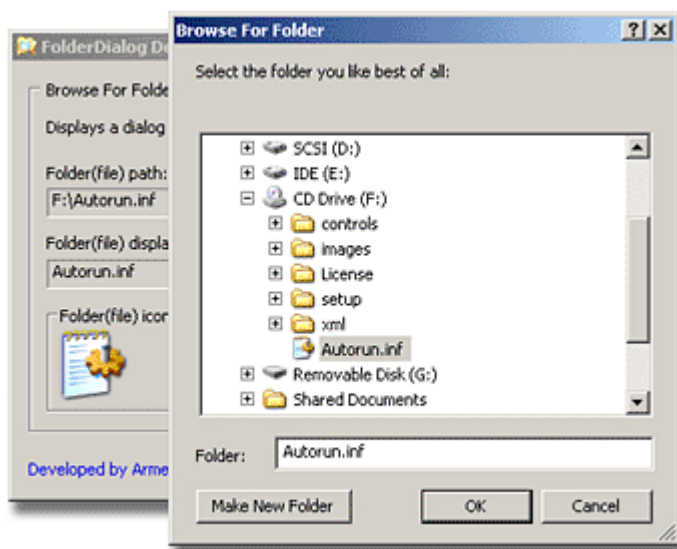
CFolderDialog dlg( _T( "Dialog Title" ), strFolderPath, this );
if( dlg.DoModal() == IDOK )
{
    strFolderPath = dlg.GetFolderPath();
    strDisplayName = dlg.GetFolderDisplayName();
    // Use folder path and display name here ...
}
//

```



See demo project source for more.

## Custom Filtering

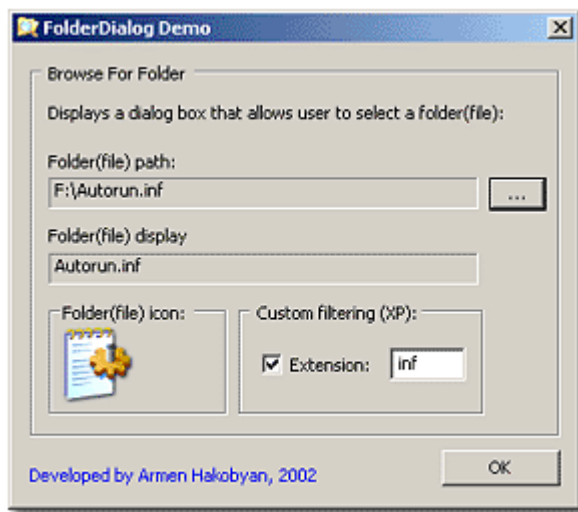


Under Microsoft® Windows® XP or later, you can do custom filtering on the contents of the dialog box. To create a custom filter, follow these steps:

1. Set the `BIF_NEWDIALOGSTYLE` flag in the `uFlags` member of the `CFolderDialog` constructor. Override the `OnIUnknown` virtual member function in the derived class. On `OnIUnknown`, the function's `lpIUnknown` parameter will contain a pointer to an instance of `IUnknown`. Call `QueryInterface` on that `IUnknown` to obtain a pointer to an

`IFolderFilterSite`.

2. Create an object that implements `IFolderFilter` - derive a class from it that implements all basic pure virtual member functions of `IUnknown` and implement `IFolderFilterSite::ShouldShow` and `IFolderFilterSite::GetEnumFlags` functions, that do filtering.
3. Call `IFolderFilterSite::SetFilter`, passing it a pointer to your custom `IFolderFilter` derived class. `IFolderFilterSite::ShouldShow` and `IFolderFilterSite::GetEnumFlags` methods can then be used to include and exclude items from the tree.
4. Once the filter is created, the `IFolderFilterSite` interface is no longer needed. Call `IFolderFilterSite::Release` if you have no further use for it.



I have added a sample custom filtering (look at the picture, the dialog shows only ".inf" files in the tree). Thanks to [Arik Poznanski](#) for his article "[C# does Shell, Part 1](#)". For more information, please see the source code.

## Class Members

### Base Class

| `CCommonDialog`

### Data Members

- | `m_bi` - The Windows `BROWSEINFO` structure. Provides access to basic folder dialog box parameters.
- | `m_szFolPath` - Contains the path of the folder selected with the dialog.
- | `m_szSelPath` - Contains the folder path to be initially selected when the the dialog opens.

### Construction

Constructs a `CFolderDialog` object:

```
CFolderDialog( LPCTSTR lpszTitle = NULL, LPCTSTR lpszSelPath = NULL, CWnd*
pParentWnd = NULL, UINT uFlags = BIF_RETURNONLYFSDIRS )
```

- | `lpszTitle` - Title to display in the top of the dialog.
- | `lpszSelPath` - The folder path to be initially selected when the the dialog opens.
- | `pParentWnd` - A pointer to the file dialog-box object's parent or owner window.

- | `uFlags` - A combination of one or more flags that allow you to customize the dialog box. For more information see `BROWSEINFO` structure in the Platform SDK.

## Operations

- | `DoModal( void )` Displays the browse for folder dialog box and allows the user to make a selection.
- | `SetSelectedFolder( LPCTSTR lpszPath )` - Sets the folder path to be initially selected when the the dialog opens.
- | `GetSelectedFolder( void ) const` - Gets the folder path to be initially selected when the the dialog opens.
- | `GetFolderPath( void )const` - Retrieves the path of the open folder.
- | `GetFolderDisplayName( void )const` - Retrieves the display name of the currently open folder.
- | `GetFolderImageIndex( void )const` - Gets the image associated with the selected folder. The image is specified as an index to the system image list.
- | `GetBI( void )` - Retrieves the `BROWSEINFO` structure of the `CFileDialog` object.
- | `GetBI( void )const` - Retrieves the `BROWSEINFO` structure of the `CFileDialog` object.

## Overridables

- | `OnInitialized( void )` - Called when browse dialog box has finished initializing.
- | `OnSelChanged( LPITEMIDLIST lpItemIDList )` - Called when browse dialog box selection is changed.
- | `OnValidateFailed( LPCTSTR lpszFolderPath )` - Called when the user typed an invalid name into the edit box (if any) of the browse dialog box. Return zero to allow the dialog to be dismissed or nonzero to keep the dialog open.

*Microsoft® Windows® XP or later:*

- | `OnIUnknown( IUnknown* lpIUnknown )` - Provides an `IUnknown` interface to the client for custom filtering of the contents of the dialog box, using `IFolderFilterSite` and `IFolderFilter`.

Functions, that are valid to be called only from **Overridables**:

- | `EnableOK( BOOL bEnable = TRUE )` - Enables or disables the browse dialog box's OK button.
- | `SetSelection( LPITEMIDLIST lpItemIDList )` - Selects the specified folder.
- | `SetSelection( LPCTSTR lpszFolderPath )` - Selects the specified folder.
- | `SetStatusText( LPCTSTR lpszText )` - Sets the dialog box status text.

*Shell version 5.0 or later:*

- | `SetExpanded( LPITEMIDLIST lpItemIDList )` - Specifies a path to expand in the dialog box.
- | `SetExpanded( LPCTSTR lpszFolderPath )` - Specifies a path to expand in the dialog box.
- | `SetOKText( LPCTSTR lpszText )` - Sets the dialog box "OK" button text.

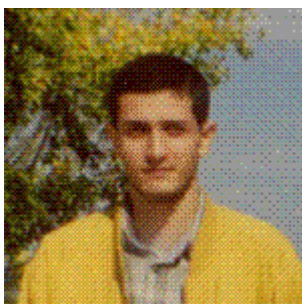
## Notes

This class compiles without any warning at level 4, and fully supports both ANSI and UNICODE.

## Version History

- | 27 Mar 2002
  - Posted the article.
- | 30 Mar 2003
  - Some code changes
  - Added missing in old Platform SDK new flag definitions
  - Added support for both MFC 6.0 and 7.0
  - Added OnIUnknown handler for Microsoft® Windows® XP folder filtration
  - Added SetExpanded, SetOKText and GetSelectedFolder functions
- | 30 May 2003
  - Added OnSelChanged default implementation.
- | 14 Jul 2003
  - Added custom filtering sample for Microsoft® Windows® XP or later
  - Set SetExpanded and SetOKText to noinline (for safe USES\_CONVERSION)

## About Armen Hakobyan



Armen Hakobyan is Security Officer of Information Systems Security Division in the Central Bank of Armenia.

Click [here](#) to view Armen Hakobyan's online profile.



## Discussions and Feedback

 33 comments have been posted for this article. Visit <http://www.codeproject.com/dialog/cfolderdialog.asp> to post and view comments on this article.

[All Topics](#), [MFC / C++](#) >> [Dialog and Windows](#) >> [General](#)  
Updated: 23 Jul 2003

Article content copyright Armen Hakobyan, 2002  
everything else Copyright © [CodeProject](#), 1999-2004.