

Remote Filepicker

Table Of Contents

Remote Filepicker	1
1. Introduction	1
2. How it works	1
3. Configuration	2
4. How to use the component	2
5. Technical Support	3
6. Pricing.....	3
7. Screenshots.....	4

1. Introduction

The Remote Filepicker is a .NET user-control which enables users of your application to browse the file-system of a remote computer.

This functionality is commonly required by client-server applications where the client-side has to display the server's file-system.

Typical use case:

- A Program consist of a server-component and a GUI running on the client's workstation
- The client's GUI application has to select a path on the server.

The user interface is similar to what people are used to from the Microsoft Windows dialogs which pop up whenever a user saves or opens a file. The standard windows dialogs are limited to the local computer. The Remote Filepicker overcomes exactly that limitation.

2. How it works

The control communicates with its server-part using TCP/IP with a configurable port. It uses the remoting facilities of Microsoft's .NET framework. It can establish a connection independently from your application OR it can use a remoting connection already established by your application.

The control itself can be used with any .NET 2.0 or higher application. A version which can be used with older .NET 1.1 applications can be made available upon request.

3. Configuration

The following public properties may be set to influence the runtime-behavior of the user control:

<u>Property</u>	<u>Description</u>
BrowserDialogType	Enumeration which determines, whether the user should: <ul style="list-style-type: none">➤ Select an existing file➤ Select an existing folder➤ Enter a new file name (for saving a file)
InitialDirectory	The folder from where the navigation should start. Default: C:\
JailUser	Security: The user can only see that folder and its subfolders. Other locations of the server's hard drive(s) will not be accessible. Neither will be network locations. Note: If the InitialDirectory does not exist the directory above that will be chosen. If that does not exist either the recursion will go down until the drive level. So be careful to set only existing directories if security is a concern.
Text	The text displayed in the dialog's caption. If an evaluation licence for private use is being used, then a copyright notice will be appended.
Settings.*	Server name or IP/Address, Port and service name

4. How to use the component

On the server-side:

- Reference two .NET Assemblies: RemoteFilepickerHelper.dll and RemoteFilepickerControl.dll

- Initialize the Server: `Printcom.RemoteFilePicker.InitServer()`

On the client-side:

- Reference two .NET Assemblies: `RemoteFilepickerHelper.dll` and `RemoteFilepickerControl.dll`
- Add the control to your Visual Studio Toolbox
- Drag & Drop the Filepicker to your Project
- Display the Remote Filepicker by calling `ShowDialog()`
- Read the resulting filename from the `Browser.Filename` property

Note: The remote Filepicker will NOT transfer the file or its content in any way.

5. Technical Support

You may contact filepicker-support@printcom-gmbh.de for technical assistance or feature requests. We will certainly consider your feedback and integrate it into the user control if deemed valuable.

6. Pricing

Please see <http://www.remote-filepicker.com> for information about licensing and pricing.

Note that this component may NOT be used in commercial applications without obtaining a license first. Neglecting will lead you to serious legal trouble.

7. Screenshots

