

Microsoft® FrontPage® 2002

How to Send Form Results to a Database and an E-mail Address and Use the File Upload Component Simultaneously

Microsoft Product Support Services White Paper

Written by Jerry Bryant
Published on 7-21-2001

Abstract

This white paper explains how to use Microsoft® FrontPage® 2002 to send form results to a database and an e-mail address, and to use the File Upload component at the same time.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2000 Microsoft Corporation. All rights reserved.

Microsoft and Microsoft FrontPage 2002 are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

CONTENTS

INTRODUCTION	1
SERVER REQUIREMENTS	2
CLIENT REQUIREMENTS	3
Creating the Database Connection	3
Creating a File Upload Folder in your Web	4
Creating the Pages	4
Create the Form Page	5
Create the Custom Confirmation Page	5
Switch Back to Form.htm	5
Create the Process1.asp Page	7
Create the Process2.asp Page	8
Create the Display.asp Page	9
SEE IT ALL WORK	11

INTRODUCTION

Microsoft® FrontPage® 98 introduced a new form handler that allows you to send the results of a form to an e-mail address. This feature has become one of the most widely used features in Microsoft FrontPage.

In Microsoft® FrontPage® 2000, Microsoft added the ability to send form results to a database as easily as to an e-mail address. However, the **Send to Database** feature and the **Send to E-mail** feature cannot be used simultaneously through the Microsoft FrontPage interface.

In Microsoft® FrontPage® 2002, Microsoft added the File Upload form component so that you can easily upload files to a Web site from a Web page. Like the e-mail component, the File Upload component does not work with the Send to Database feature.

Fortunately, it is fairly easy to use all features at the same time with custom code. This white paper describes how to set up this functionality and provides examples that you can use in your own Web site to add this powerful ability. The examples use Active Server Pages (ASP) technology.

Because you cannot use the File Upload form component and Send to Database feature simultaneously, you need to first submit the File Upload component and other form information to a text file and then transfer this information to a Microsoft® Access® database. To accomplish this, you can use a combination of custom code and the Database Results Wizard. Microsoft does not recommend using this method for high-volume Web sites where there may be many users submitting information at the same time.

The examples in this white paper demonstrate how to create pages to collect user information and an image of the user to be displayed with the user's information.

SERVER REQUIREMENTS

For the examples outlined in this white paper to work correctly, the following capabilities must be available on the Web server that hosts your Microsoft FrontPage Web:

- Microsoft Active Server Pages (ASP)
- Microsoft Data Access Components (MDAC)
- Microsoft Simple Mail Transfer Protocol (SMTP) Service
- Microsoft FrontPage 2002 Server Extensions

Most Microsoft Internet Information Services (IIS) Web servers have these capabilities for you to use. You may need to check with your hosting company to verify that these technologies are available to you and are correctly configured for your use.

CLIENT REQUIREMENTS

Creating the Database Connection

A database connection is required in order to write to the database. Because you create database connections on a per-Web basis in Microsoft FrontPage, the first thing that you must do is create a new Microsoft FrontPage Web. To create the Web, follow these steps:

1. On the **File** menu, point to **New**, and then click **Page or Web**.
2. In the **New Page or Web** task pane, click **Web Site Templates**.
3. In the **Web Sites** dialog box, click **One Page Web**.
4. In the **Specify the location of the new Web** box, type the path to the Web, and then click **OK**.

NOTE: *If you do not create the Web as a server-based Web, you can test your site by publishing the Web to a Web server that meets the server requirements that are listed in the Server Requirements section earlier in this document.*

After you create the new Web, create a new database and database connection. The example that follows establishes a database connection to the text file to which the form results are initially written, submits form data to the `_private/form_results.txt` file, and stores the data in the comma-delimited format. To establish a database connection to a text file, create a `Schema.ini` file, and place it in the same directory as the text file. To do this, follow these steps:

1. Start Notepad.
2. Type the following:

```
[form_results.txt]
ColNameHeader=True
Format=CSVDelimited
MaxScanRows=25
CharacterSet=OEM
Col1=NAME Char Width 255
Col2=EMAIL Char Width 255
Col3=COMMENTS Char Width 255
Col4=FILE Char Width 255
```

3. On the **File** menu, click **Save**. Name the file `Schema.ini`, and save it to the `_private` folder in your Web site.
4. Create another file in Notepad. Don't add anything to the file. Save this file as `Form_results.txt` in the `_private` folder of your Web.

-
5. Switch to Microsoft FrontPage and open your Web. On the **Tools** menu, click **Web Settings**, and then click the **Database** tab.
 6. Click **Add**.
 7. In the **Name** box, type *text*.
 8. In the **Type of connection** section, click **File or folder in current Web**, and then click **Browse**.
 9. In the **Files of Type** list, select **Microsoft Text Driver**.
 10. Open the `_private` folder in your Web, and then select the `Form_results.txt` file. Click **OK** three times to close all dialog boxes.

Creating a File Upload Folder in your Web

To create a File Upload folder in your Web, follow these steps:

1. Open your Web.
2. In the **Folder list** pane, right-click the top level folder, point to **New**, and then click **Folder** on the menu that appears. Name the folder *Files*.
3. Right-click the new folder and then click **Properties** on the menu that appears.
4. In the **Properties** dialog box, click to clear the **Allow scripts to be run** check box.
5. Click to select the **Allow anonymous upload to this directory** and the **Allow uploaded files to overwrite existing filenames** check boxes, and then click **OK**.

Creating the Pages

In the following example, you will create five pages. This set of pages works together to collect information about a user and to display an image of the user on a profile page (page 5). The first page is the form page. It contains both the Hypertext Markup Language (HTML) form that collects the information and the File Upload form component. The second page is a custom confirmation page. It uses a META refresh tag to send the user to page 3. Page 3 is an ASP page. It uses custom ASP code to pull the information that was submitted to the text file from the form. It takes this information and transfers it to another form for verification. The fourth page contains a Database Results region. The Database Results region uses an INSERT SQL statement to write the data to a Microsoft Access database. Finally, the fifth page contains another Database Results region that displays this information.

Create the Form Page

The form page is an HTML page that contains the form to be submitted. To create this page, follow these steps:

1. In a new page, point to **Form** on the **Insert** menu, and then click **Textbox**.
2. Double-click the text box. In the **Name** box, type *name*. If you want, add a label. Click **OK**.
3. On the **Insert** menu, point to **Form**, and then click **Textbox**.
4. Double-click the text box. In the Name box, type *email*. If you want, add a label. Click **OK**.
5. Click anywhere on the form. On the **Insert** menu, point to **Form**, and then click **Text Area**.
6. Double-click the text area box. In the **Name** box, type *Comments*, and then click **OK**.
7. Click anywhere on the form. On the **Insert** menu, point to **Form**, and then click **File Upload**.
8. Double-click the File Upload text box. In the **Name** box, type *File*, and then click **OK**.
9. **Save** this page and name it Form.htm.
10. Keep this page open because you will use it again.

Create the Custom Confirmation Page

When the form (Form.htm) is submitted, the custom confirmation page will send the user to the first ASP page.

1. Insert a new page.
2. Switch to HTML view.
3. Directly under the <HTML> tag, type the following:

```
<meta http-equiv="refresh" content="0; url=../process1.asp">
```

4. Save this page as Confirmation.htm and close it.

Switch Back to Form.htm

After you create the confirmation page, finish working on Form.htm.

1. Use the form on this page to create the Microsoft Access database. To do this, follow these steps:
 - a. Right-click anywhere in the form, and then click **Form Properties** on the menu that appears.

-
- b. Click **Send to database**.
 - c. Click **Options**.
 - d. Click **Create Database**.
 - e. When you see a message indicating that the database has been created, click **OK**. In the **Options for Saving Results to Database** dialog box, click **Cancel**.

NOTE: There is a tab for the File Upload component in the **Options for Saving Results to Database** dialog box. Changing the settings and attempting to send the form results directly to the Microsoft Access database and uploading a file at the same time will fail.

- f. In the **Form Properties** dialog box, under **Where to store results**, click **Send to**. Verify that the **File Name** box shows the following folder and file:

_private/form_results.txt

NOTE: In steps a through f, you created a Microsoft Access database based on your form and saved it in a new folder called fpdb. In the database, column headings are Name, Email, Comments, and File.

2. In the **Form Properties** dialog box, click **Options** and then click the **File Results** tab.
3. In the **File Format** list, click **Text database using comma as a separator**. In the **File name** box, change the file extension from .csv to .txt.
4. Click the **E-mail Results** tab.
5. In the **E-mail address to receive results** box, type the e-mail address of the recipient whom you want to receive the form results.
6. Click the **Confirmation Page** tab.
7. In the **URL of confirmation page** section, click **Browse**. Select the Confirmation.htm file, and then click **OK**.
8. Click the **Saved Fields** tab.
9. In the **Form Fields to Save** list, select **B1** and then press DELETE to remove it from the list.
10. Click the **File Upload** tab.
11. Click **Browse**. Select the **Files** folder, and then click **OK**.
12. Click **OK** two times.

13. Save the page.

Create the Process1.asp Page

Process1.asp contains custom code that pulls the last record from the Form_results.txt file and then writes this information into a form with text input boxes. This allows the user to verify his or her information before it is sent to the Microsoft Access database. Hiding the URL to the uploaded file in a hidden field prevents the user from changing it. After the file is uploaded, the user should not need to make changes.

1. Create a new page in Microsoft FrontPage. Switch to HTML view and type the following code directly under the <html> tag:

```
<%
'////////////////////////////////////
'// The first two lines of code are calling the connection
'// object for the text file that is stored in the
'// global.asa file.
'////////////////////////////////////

Set Conn = Server.CreateObject("ADODB.Connection")
Conn.Open Application("text_ConnectionString")
'////////////////////////////////////
'// Create a record set object, execute the
'// database connection and a SQL query.
'////////////////////////////////////

Set RS = Conn.Execute ("SELECT * From form_results.txt")

'////////////////////////////////////
'// Loop through the database and
'// select the last record entered.
'////////////////////////////////////

Dim iCnt
Do Until RS.EOF
iCnt = iCnt + 1
Name = RS("Name")
Email = RS("Email")
Comments = RS("Comments")
File = RS("File")
RS.MoveNext
Loop
'////////////////////////////////////
'// Close the record set and the connection.
'////////////////////////////////////

RS.Close
Conn.Close
%>
```

2. Switch back to normal view.
3. On the **Insert** menu, point to **Form**, and then click **Textbox**.
4. Double-click the text box. In the **Name** box, type *name*. In the **Initial Value** box, type <% = Name %>. Click **OK**.

-
- Repeat steps 3 and 4 for both the e-mail and comments fields. Use the names and initial values listed in the following table:

Name	Initial Value
Email	<% = Email %>
Comments	<% = Comments %>

- Right-click the form and then click **Form Properties** on the menu that appears.
- Click **Send to other**. In the **Send to other** list, click **Custom ISAPI, NSAPI, CGI, or ASP Script**.
- Click **Options**. In the **Action** box, type *Process2.asp* and then click **OK**.
- In the **Form Properties** dialog box, click **Advanced**.
- Click **Add**.
- In the **Name** box, type *File*. In the **Initial Value** box, type <% = File %>. Click **OK**.
- Click **OK** two more times to return to your page.
- Save this page as Process1.asp and then close it.

Create the Process2.asp Page

The Process2.asp page will contain a Database Results region and a custom SQL statement that will insert the information passed from Process1.asp into the Microsoft Access database.

- Create a new page in Microsoft FrontPage.
- On the **Insert** menu, point to **Database**, and then click **Results**. The Database Results Wizard starts.
- In step one of the wizard, click **Use an existing database connection**. In the **Use an existing database connection** list, click **Form**, and then click **Next**.
- In step two of the wizard, click **Custom query**, and then click **Edit**.
- In the **Custom Query** dialog box, type the following:

```
INSERT INTO Results (Name, Email, Comments, File) VALUES ('::Name::',  
'::Email::', '::Comments::', '::File::')
```

Click **OK**. Click **Next**.

- In step three of the wizard, click **More Options**. In the **Message to display if no records are returned** box, type the following:

Your profile has been accepted!

Click **OK**. Click **Next**.

7. In step four, in the **Choose options for the records returned by the query** list, click **List – one field per item**.
8. Click to clear the **Add labels for field values** and **Place horizontal separator between records** check boxes. Click **Next**.
9. In step five, click to clear the **Add search form** check box. Click **Finish**.
10. Save this page as Process2.asp and then close it.

Create the Display.asp Page

The final page of this example displays records from the Microsoft Access database one at a time. This example dynamically displays images from a folder location that is stored in the database as the URL. For additional information about this process, please see the following Microsoft Knowledge Base article:

[Q216285](#) FP: How to Display a Picture Using a Microsoft Access Database

1. Create a new page in Microsoft FrontPage.
2. On the **Insert** menu, point to **Database**, and then click **Results**. The Database Results Wizard starts.
3. In step one, click **Use an existing database connection**. In the **Use an existing database connection** list, click **Form**. Click **Next**.
4. In step two, click **Record Source**. In the **Record Source** list, click **Results**. Click **Next**.
5. In step three, remove any fields that you do not want to be displayed in the database results, and then click **Next**.
6. In step four, in the **Choose formatting options for the records returned by the query** list, click **List – one field per item**. Click **Next**.
7. In step five, click **Split records into groups**. In the **Records per group** box, type **1**. Click **Finish**.
8. Locate the database column value that reads "<<File>>" and delete it.
9. In the table cell from which you just removed the column value, insert any picture from your own images or from clip art.
10. Right-click the image and then click **Picture Properties** on the menu that appears.
11. Click the **General** tab. Under **Picture Source**, click **Parameters**.
12. In the **Path** box, remove the path. Click **Insert Field Value**, and then click **File** on the menu that appears.

-
13. Click **OK** two times to return to your page. You see a small broken image icon in the <<File>> column.
 14. Save the page as Display.asp and then close it.

SEE IT ALL WORK

Now that everything is complete, preview the Form.htm page, and type your information in the form. When you submit the form, the data that you typed is written to the Form_results.txt file, and the e-mail is sent. The custom confirmation page then passes you to the Process1.asp page that displays the information that you submitted in text input boxes. Clicking **Submit** writes the data to the Microsoft Access database. To view the records submitted, in your Web browser, type the address to the Display.asp page.