

How can I check if a file exists?

By Darren Green
Version 2000
Level Intermediate

Checking for the existence of a file before trying to import it can be very useful in ensuring your systems run smoothly. You can use the Scripting [FileSystemObject](#) to check for the presence of a file using ActiveX Script. If the file exists you can obviously import it or initiate an alternative process, such as a sending a warning to someone or just ignore this error all together.

The simplest method is to use an ActiveX Script Task that returns a success (DTSTaskExecResult_Success) or failure (DTSTaskExecResult_Failure) result as appropriate. The following example reads the filename (DataSource) property from a named connection, and returns success or failure based upon the existence of the file:

```
' Pkg 211 (File Exists - 1)
Option Explicit

Function Main()
    Dim oFSO, sFileName

    ' Get the name of the file from the Connection "Text File (Source)"
    sFilename = DTSGlobalVariables.Parent.Connections("Text File (Source)").DataSource

    Set oFSO = CreateObject("Scripting.FileSystemObject")

    ' Check for file and return appropriate result
    If oFSO.FileExists(sFilename) Then
        Main = DTSTaskExecResult_Success
    Else
        Main = DTSTaskExecResult_Failure
    End If

    Set oFSO = Nothing
End Function
```

The problem with using an ActiveX Script Task is that because the task fails, so does the package. This is often inappropriate, for example you may have package scheduled every 15 minutes that imports a file. This file is only produced when certain conditions have been met on an external system, and the absence of the file is by no means cause for concern. This is where you can use the power of an ActiveX Workflow Script. Workflow scripts use the DTSScriptResult constants, one of which is DTSScriptResult_DontExecuteTask. As you'd expect from the name, the task associated with this step does not execute, but importantly it does not raise any errors either.

To use an ActiveX workflow script, select the task that must be prevented from executing when the file is missing. Right-click Workflow Properties, or Workflow followed by Properties as appropriate. On the Options tab select "Use ActiveX Script", and then click Properties. You will now need to code the checking mechanism and return the appropriate result. DTSScriptResult_ExecuteTask allows the task to execute and DTSScriptResult_DontExecuteTask blocks the workflow path and prevents the task from executing.

The example below takes it's filename name from a global variable, purely to demonstrate and alternative mechanism of supplying this information. Reading it from a connection as above would work equally well. The actual check using the FileExists method is the same:

```
' Pkg 211 (File Exists - 2)
Option Explicit
```

```
Function Main()  
    Dim oFSO, sFileName  
  
    ' Get the name of the file from the global variable "ImportFileName"  
    sFilename = DTSGlobalVariables("ImportFileName").Value  
  
    Set oFSO = CreateObject("Scripting.FileSystemObject")  
  
    ' Check for file and return appropriate result  
    If oFSO.FileExists(sFilename) Then  
        Main = DTSScriptResult_ExecuteTask  
    Else  
        Main = DTSScriptResult_DontExecuteTask  
    End If  
  
    Set oFSO = Nothing  
End Function
```