Publish multiple Markdown files to HTML in Windows

THURSDAY, APRIL 19TH, 2012

I wrote this script as a means of setting up a dead-simple "knowledge base" in HTML format.

The idea is to write documentation as a collection of plain-text files in Markdown[1] format and have a no-fuss way to publish them as HTML, re-publishing changes as necessary.

In order for this script to work, you need to be on Windows, and you need to install a program called `pandoc`[2].

How to use it:

1. Save a copy of this script file in any folder containing a bunch of Markdown-formatted text files. Include a `stylesheet.css` file in this folder as well if you want the HTML files to have CSS styling.

2. Run the script (double-click it)—it will silently create updated HTML files for every text file in the folder. Only text files whose HTML counterparts are out of date or nonexistent will be processed.

You can either copy and paste the code below into Notepad and save it as a `.vbs` file, or you can download the latest version[3] in a zip file. The code in the download will be more extensively commented, and may also contain enhancements developed since this post was written.

Here's the basic code (provided under the terms of the Artistic License 2.0—http://www.perlfoundation.org/artistic_license_2_0[4]):

```
1 Set objShell = CreateObject("WScript.Shell")
2 Set objFSO = CreateObject("Scripting.FileSystemObject")
3
```

1. http://daringfireball.net/projects/markdown/
2. http://johnmacfarlane.net/pandoc/installing.html
3. https://docs.google.com/open?id=0B9SDJ22NRBkrcEgtcWsyMi1pTFU
4. http://www.perlfoundation.org/artistic_license_2_0

```
strThisFolder = objFSO.GetParentFolderName(Wscript.ScriptFullName)
Set objStartFolder = objFSO.GetFolder(strThisFolder)
strConverterCommand = "pandoc -f markdown -t html -c stylesheet.css -o "

Set objFilesToUpdate = CreateObject("Scripting.Dictionary")

Set colFiles = objStartFolder.Files
For Each objFile in colFiles
   If objFSO.GetExtensionName(objFile.Name) = "txt" Then

      ' Check if HTML version of this text file exists in this folder
      strHTMLName = strThisFolder & "\" & Replace(objFile.Name, ".txt", ".html")
      If objFSO.FileExists(strHTMLName) Then

         ' If it exists, compare the timestamps
         Set objHTMLFile = objFSO.GetFile(strHTMLName)
         If objFile.DateLastModified > objHTMLFile.DateLastModified Then
            'If the text file is newer, add this text file to the list
            objFilesToUpdate.Add objFile.Name, strHTMLName
         End if

      Else
         ' If the file does not exist yet, add this text file to the list
         objFilesToUpdate.Add objFile.Name, strHTMLName
      End if
   End if
Next

' Update all the text files in the list.
colFilesToUpdate = objFilesToUpdate.Keys
For Each strSourceFile in colFilesToUpdate

   objShell.Run strConverterCommand & objFilesToUpdate.Item(strSourceFile) & " " & strSourceFile, 3,
         True
Next
```

Possible Future Improvements:

- The script isn't very helpful about telling you how long the process is going to take. I looked at several options for providing a progress bar or some kind of status output, but ultimately VBScript is just really sucky at this.
- Pandoc is a very powerful converter. One could easily tweak the script to add options for producing LaTeX or even PDF files.