

VisualHub Automation

Version 1.24, 6/21/07

by Tyler Loch - Fearless Leader, Techspanion LLC

"I always like to start at square one. It's like the Itsy-Bitsy Spider thing."

- Sal Soghoian

What is Automation?

Automation is the art of being lazy. In the age of computers, you may find yourself with a frequently-followed workflow leaving you to point and click like a monkey on the same stuff over and over again. Automating that workflow means the computer will do all that hard work for you, leaving your time freed up for other things, like contributing to funny cat memes on the Internet.

A little inspiration and elbow grease in the beginning can pay off in a big way.

Automation in VisualHub

Starting with VisualHub 1.24, a Tech Preview of the automation support in VisualHub 2.0 was offered. This allows VisualHub users to create their own automated workflows (or "scripts" as the kids are calling them nowadays) that can do anything from convert a video to another format with a single click, or watch a server for a set of files and launch a distributed Xgrid workflow, burning a disc when complete (and more. People can go absolutely nuts with scripts. I should know...).

VisualHub is automatable via the AppleScript language - simple, concise, yet powerful.

By The Way...

This isn't a primer on AppleScripting itself. For that, Apple has a great list of resources here:

<http://www.apple.com/macosx/features/applescript/resources.html>

How to Start

Automation support is available as a Tech Preview, and is not included in the official VisualHub download. You'll have to shoehorn it in.

- 1: Download the Automation Tech Preview: <http://www.techspanion.com/visualhub/automation.scpt.zip>
- 2: Control-click (right-click if you got 'em) on the VisualHub application, and select "Show Package Contents" from the contextual menu.
- 3: Drill down a couple folders. Double-click on "Contents", then "Resources", and finally, "Scripts".
- 4: Drag the "automation.scpt" file you just downloaded into this Scripts folder. You can close the window when done.

If you did it right, then the next time you open VisualHub its title bar will read "VisualHub 1.x Automation Tech Preview".

Starting Your Script

Since VisualHub itself is an AppleScript application, it's wasn't possible to develop a standard AppleScript dictionary (ironic, isn't it?), so a little improvisation was in order. This Primer documents the special pseudo-dictionary created for VisualHub.

To load the VisualHub pseudo-dictionary, start your script with this "Magic Line":

```
tell application "VisualHub" to set VisualHub to load script (scripts path of main bundle & "/automation.scpt" as POSIX file)
```

This creates the special (and aptly-named) `VisualHub` variable that we will call VisualHub with.

Commands to VisualHub can now be made as follows:

```
tell VisualHub
    AddFiles(fileList)
    StartConversion()
    QuitApp()
end tell
```

(Continued on Page 2)

Dictionary Commands and Syntax

CheckStatus() - Returns the number of files complete (granular to partial files) and current status message in an AppleScript list.

Example result:

```
{0.403704, "Less than 1 minute remaining for File 1..."}
```

AddFiles(theFiles) - Adds files to VisualHub's File List. Accepts AppleScript lists only. AppleScript-style colon-paths preferred, POSIX paths also allowed.

Example commands:

```
AddFiles({"Macintosh HD:Users:scripter:Movies:awesomevid.mpg", "Macintosh HD:Users:scripter:Movies:stupidvid.mpg"})
```

```
AddFiles({"/Users/scripter/Movies/awesomevid.mpg"})
```

ClearAll() - Clears VisualHub's File List.

SetSaveLocation(saveas) - Sets the "Save To:" field. AppleScript-style colon-paths preferred, POSIX paths also allowed. Setting the **saveas** argument to **null** clears the field.

Example commands:

```
SetSaveLocation("Macintosh HD:Users:scripter:Movies")
```

```
SetSaveLocation(null)
```

StartConversion() - Just like clicking the Start button.

LoadSettings(theFile) - Loads a .vhub Settings file. Use this for setting any format options, etc. AppleScript-style colon-paths preferred, POSIX paths also allowed.

Example command:

```
LoadSettings("Macintosh HD:Users:scripter:Desktop:mysettings.vhub")
```

QuitApp() - Just like Apple-Q, or File->Quit.

ForceQuit() - Bypasses VisualHub's warnings and kills the program (and its processes) dead.

TurnXgridOn(saveas) - Enables Xgrid encoding. **saveas** is the same behavior as **SetSaveLocation(saveas)**.

TurnXgridOff() - Disables Xgrid encoding.

StitchWhenDone(filename) - Enables Stitching for formats that support it. Set **filename** to a name without a file extension. It will be added for you.

Example command:

```
StitchWhenDone("stitchedname")
```

Common Workflow Examples

Things can get ridiculously powerful really quickly. What follows are some relatively simple generic scripts to get you going.

Convert a Folder's Worth of AVI Files

This script gets a list of .avi videos from the Desktop of the user "scripter", and converts them using the settings saved in a .vhub settings file, then quitting.

```
tell application "VisualHub" to set VisualHub to load script (scripts path of main bundle & "/automation.scpt" as POSIX file)
```

```
tell application "Finder" to set everyavi to (every file of desktop whose name extension is "avi")
```

```
tell VisualHub
```

```
    AddFiles(everyavi)
```

```
    LoadSettings("Macintosh HD:Users:scripter:Desktop:mysettings.vhub")
```

```
    StartConversion()
```

```
    QuitApp()
```

```
end tell
```

(Continued on Page 3)

Create a Hot Folder

This uses AppleScript's Folder Action capability to watch a folder, and execute the script whenever a file is added. Output files are placed when done into the ~/Movies folder.

```
on adding folder items to this_folder after receiving these_items
    tell application "VisualHub" to set VisualHub to load script (scripts path of main bundle & "/automation.scpt" as
        POSIX file)
    tell VisualHub
        AddFiles(these_items)
        LoadSettings("Macintosh HD: Users: scripiter: Desktop: mysettings.vhub")
        SetSaveLocation("Macintosh HD: Users: scripiter: Movies")
        StartConversion()
        QuitApp()
    end tell
end adding folder items to
```

VisualHub Automation in a Bash Script

Using nefarious methods, it's possible to even use VisualHub with a shell script. Keep in mind the console user needs to be logged in via the GUI (not necessarily in the foreground) for this to work.

This script finds a file via "mdfind", and passes that as a bash variable to a specially-crafted osascript command.

```
filetoconvert=`mdfind thisoldfile.avi | head -1`
```

```
osascript -e 'tell application "VisualHub" to set VisualHub to load script (scripts path of main bundle &
"/automation.scpt" as POSIX file)' -e 'tell VisualHub' -e "AddFiles({\"$filetoconvert\"})" -e 'LoadSet-
tings("Macintosh HD:Users:scripiter:Desktop:mysettings.vhub")' -e 'SetSaveLocation("Macintosh
HD:Users:scripiter:Movies")' -e 'StartConversion()' -e 'QuitApp()' -e 'end tell'
```

Discussions and More

Discuss, collaborate, show off and complain on Techspansion's Automation Forum:

<http://www.isquint.org/cgi-bin/ikonboard.cgi?act=SF;f=13>

Fine Print

The VisualHub Automation Tech Preview is belpha (halfway) quality, and is unsupported. It is provided for VisualHub customers to experiment/play with because it's cool.