

APPLESCRIPT BASICS

by Jerry Stratton

Copyright © 2011

<http://www.hoboes.com/NetLife/AppleScript/>

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 published by the Free Software Foundation; with no Invariant Section, with no Front-Cover Text, and with no Back-Cover Texts. A copy of the license is included in the section entitled “GNU Free Documentation License”

Getting started	1
Why AppleScript	1
Script Editor	1
Recording Actions	2
Carefully arrange the desktop	2
Simple Scripts	4
Comments	4
Dictionaries	4
Create a news screen	4
Automatically Reload Web Pages	5
Shell Tell	6
Repetitive Tasks	6
Talking Clock	6
Drag and Drop	8
Other Things You Might Do	9
AppleScript and the Command Line	10
Running AppleScript from the Command Line	10
Calling the Command Line from AppleScript	10
More Information	11
AppleScript: The Definitive Guide	11
AppleScript on Negative Space	11
Introduction to AppleScript Overview	11
Open Source License	12
Gnu Free Documentation License	12

GETTING STARTED

Why AppleScript

AppleScript can automate monotonous, repetitive tasks, or tasks that need to be performed at a specific time. Mac OS X can (using it's built-in scheduler via programs such as Apple's iCal or Embraceware's Awaken) run any script on the minute you need it to run.

AppleScript makes it easy to “talk” to standard Macintosh applications. It can “glue” applications together so that each application does what it does best.

AppleScript is easy to understand; it's almost English:

```
tell application "Safari" to make new document with properties {URL:"http://  
    www.hoboes.com/"}
```

This is a valid AppleScript statement. If you use it, Safari will create a new document, and inside that document it will open the hoboes.com web site.

Script Editor

Every Macintosh comes with a program called “AppleScript Editor” that allows you to create AppleScripts. AppleScript Editor is in the Applications folder, inside of the Utilities folder.

RECORDING ACTIONS

If an application supports it, recording actions allows you to quickly build up a basic AppleScript. For example, if you go into AppleScript Editor and hit the “Record” button before making a new folder, you might end up with something like:

```
tell application "Finder"
  activate
  make new folder at folder "Desktop" of folder "jerry" of folder "Users" of startup
    disk with properties {name:"untitled folder"}
  set name of folder "untitled folder" of folder "Desktop" of folder "jerry" of
    folder "Users" of startup disk to "Quick New Folder"
end tell
```

This tells the program (“application”) called “Finder” to first activate, then make a new folder, and then change the name of that folder to “Quick New Folder”.

Unfortunately, most applications don’t support recording; if they do, however, it is often the easiest way to make a new script.

Carefully arrange the desktop

Let’s say that you want to always start up your computer with three folders open: your Desktop, your Documents, and your Downloads.

Go into Script Editor and get rid of whatever is currently there. Click “Record” and then switch to the Finder.

Hold down the Option key, pull down the Finder menu, and choose “Close All”.

Use Command-N to open a new window, and from that window open your Desktop folder.

Move your Desktop window to the upper left, and then arrange it to take up a bit less than a quarter of the screen.

Do the same for your Documents folder, but put it in the upper right.

Then do it again for your Downloads folder, but put it in the lower left.

Go back to AppleScript Editor and click “Stop”. You should have something like this:

```

tell application "Finder"
  activate
  close every window
  close Finder window 1
  make new Finder window
  set target of Finder window 1 to folder "Desktop" of folder "jerry" of folder
    "Users" of startup disk
  set position of Finder window 1 to {99, 44}
  set bounds of Finder window 1 to {99, 44, 995, 699}
  make new Finder window
  set position of Finder window 1 to {1008, 44}
  set bounds of Finder window 1 to {1008, 44, 1934, 700}
  set target of Finder window 1 to folder "Documents" of folder "jerry" of folder
    "Users" of startup disk
  make new Finder window
  set target of Finder window 1 to folder "Downloads" of folder "jerry" of folder
    "Users" of startup disk
  set position of Finder window 1 to {102, 730}
end tell

```

Next, test it. There's one odd thing in it: after it closes every window, it tries to close each window individually. But it can't do that, because it just closed all of them.

Delete the lines that look like "Close Finder window 1". Your script will look like:

```

tell application "Finder"
  activate
  close every window
  make new Finder window
  set target of Finder window 1 to folder "Desktop" of folder "jerry" of folder
    "Users" of startup disk
  set position of Finder window 1 to {99, 44}
  set bounds of Finder window 1 to {99, 44, 995, 699}
  make new Finder window
  set position of Finder window 1 to {1008, 44}
  set bounds of Finder window 1 to {1008, 44, 1934, 700}
  set target of Finder window 1 to folder "Documents" of folder "jerry" of folder
    "Users" of startup disk
  make new Finder window
  set target of Finder window 1 to folder "Downloads" of folder "jerry" of folder
    "Users" of startup disk
  set position of Finder window 1 to {102, 730}
end tell

```

You can save this as "Arrange Folders" with File Format "Application", and every time you run it it will close all of your Finder windows and open up just the three you want, arranged the way you want them.

SIMPLE SCRIPTS

Comments

When you are writing your scripts, you will want to leave comments for yourself. You can preface any line with two dashes and that line will be ignored by AppleScript. You can write whatever you want on that line.

You can surround any set of lines with an open parentheses and two asterisks (close with two asterisks and a close parentheses) to cause AppleScript to ignore all of those lines.

```
(**
  This is a comment.
  None of this is AppleScript
**)
make new document
--close all of the windows we've opened
close all windows
```

Dictionaries

A useful feature of AppleScript Editor is the ability to look at the “dictionary” of scriptable applications. If you look at the dictionary for Safari, Word, or Finder, you can see what things those programs can do, and what commands it takes to get them to do those things.

Create a news screen

We can do the same with web browser windows that we did with Finder windows. Type the following script into your AppleScript Editor and save it as an application called “Open News Overview”:

```

tell application "Safari"
  activate
  --{xStart, yStart, xEnd, yEnd}

  --open CNN news in upper left
  set CNN to make new document
  set CNNWindow to window 1
  set bounds of CNNWindow to {10, 25, 600, 600}
  set URL of CNN to "http://www.cnn.com/"

  --open Google news in upper right
  set Google to make new document
  set GoogleWindow to window 1
  set bounds of GoogleWindow to {610, 25, 1210, 600}
  set URL of Google to "http://news.google.com/"

  --open Memeorandum in lower left
  set Memeorandum to make new document
  set MemeorandumWindow to window 1
  set bounds of MemeorandumWindow to {10, 625, 600, 1200}
  set URL of Memeorandum to "http://www.memeorandum.com/"

  --open blog in lower right
  set Blog to make new document
  set BlogWindow to window 1
  set bounds of BlogWindow to {610, 625, 1210, 1200}
  set URL of Blog to "http://www.hoboes.com/Mimsy/"
end tell

```

This will open four windows arranged in a four-part square. You can edit the “bounds” of each window as you see fit.

Automatically Reload Web Pages

If you’ve paid attention to the “Save” screen for your scripts, there is an option which by default is unchecked called “Stay Open”. If you check this, your script will not quit when it is done. It will stay open, and can continue working.

AppleScripts work by sending and receiving “messages”. When we “told” Safari to make new documents and open URLs, our script sent the appropriate messages from our script to Safari.

If our script stays open, the system will also send it messages. Whenever an application “quits”, the system sends it a “quit” message. Our script gets that also (it automatically knows what to do when it receives that message: it quits). Another message our script receives is the “idle” message. The script receives the idle message when it isn’t doing anything.

Add the following lines to your “Open News Overview”:

6—Simple Scripts

```
on idle
  tell application "Safari"
    repeat with overviewWindow in every window
      tell document of overviewWindow
        set URL to URL
      end tell
    end repeat
  end tell
  return 60
end idle
```

This is what is called a “handler” in AppleScript. It “handles” messages. When you handle a message, you can send a return message. In this case, we’re telling the system that we’re done with our idle, and we would like to have another idle message in 60 seconds (“return 60”). By default, idle times are in 30-second increments. The first idle message is sent immediately after the script first goes idle in order to set the idle time to a different default.

This just repeats through every window of Safari and tells the document in that window to reload its URL (“set URL to URL”).

Shell Tell

You’ll notice that we have two tells, one inside of the other. First, we tell Safari to do something, and inside of that we are telling one of Safari’s documents to do something.

Repetitive Tasks

One of the things you can use scripts for is to automate boring and repetitive tasks. One of the features that AppleScript has to make it easy to automate repetitive tasks is the “repeat” structure. (Other scripting languages might call it “while”, “for”, or “foreach”, among other names.)

Everything between “repeat” and “end repeat” will be repeated until our instructions are done. We told it to repeat “with overviewWindow in every window”. Within that repeat “block”, the variable “overviewWindow” will start with Safari’s first window the first time through, and then repeat for each of Safari’s windows, until there are no more windows left.

We could have done the same thing by typing those lines between the “repeat” four times, as we did when we opened the windows to load the web pages. But this makes it a lot easier. And it means that if we later add a fifth web page (we’ll need an awfully big monitor) we won’t need to change this part of the script at all.

Talking Clock

One of the features of the standard additions is the current date and the ability to speak out loud.

```
set volume 1
copy the (current date) to theDate
say theDate as string
```

This makes your computer quiet (so you don't bother other people near where you are using this tutorial), and then it copies the current date into a "variable" called "theDate". Finally, it says whatever is in the variable "theDate". The "say" command can only say text. If you try to say something other than text, AppleScript tries to send that thing a message called "say" instead of sending that item to "say". So you need to tell AppleScript to convert that thing to text first. This is why we put the "as string" after "say theDate".

You can look at what it is speaking by adding "get theDate" to the end of your script and then looking at your window titled "the result" (pull down the Windows menu to see this listed).

What it is speaking is a overly complex. It is reading the numbers without recognizing what they are. For example, it is reading 2002 as "twenty zero two" on the computer on which I'm writing this, and it is trying to speak the letters "AM" as a word "am". We'll need to help it along.

```
--be nice to our neighbors
set volume 1

on idle
  --get the current date and time
  copy the (current date) to theDate

  --get the weekday
  copy the weekday of theDate as string to theDay

  --the hours (from 1 to 12)
  copy the hours of theDate to theHour
  if theHour is greater than 12 then
    --13:00 to 23:00
    set theHour to theHour - 12
    set theAMPM to "PM"
  else if theHour is "12" then
    --12:00
    set theAMPM to "PM"
  else if theHour is "0" then
    --00:00
    set theHour to "12"
    set theAMPM to "AM"
  else
    --01:00 to 11:00
    set theAMPM to "AM"
  end if

  --the minutes (including preceding zero)
  copy the minutes of theDate to theMinutes
  if theMinutes is less than 10 then
    set theMinutes to "0" & theMinutes
  end if

  say theDay & ", " & theHour & ":" & theMinutes & " " & theAMPM

  return 60
end idle
```

8—Simple Scripts

The “ampersand” or “&” character glues together two pieces of text. In this case, we’re putting our variable “theHour” after our variable “theDay” and putting a comma and space between them. Then, adding theMinutes with a colon in front of it, and then theAMPM with a space in front of it. So it will end up looking like “Wednesday, 6:21 PM”.

Which then gets spoken.

Make sure you save your talking clock as an “Application” that will “Stay Open”.

Drag and Drop

When you drop a file onto a program, that program gets a message explaining what happened. You can let your AppleScripts accept those messages with the “open” handler. Make a script with nothing but:

```
on open (theItems)
  repeat with anItem in theItems
    tell application "Finder"
      copy the name of item anItem to theName
    end tell
    display dialog theName
  end repeat
end open
```

Save this as an Application called “Uploader”. Then, drop a few files on it. It should name each file in a dialog box.

You could use this feature to “tell” an FTP program such as Fetch or Interarchy to upload files automatically to your web site. When we do our upload, we’ll want to make sure we don’t upload things accidentally that we don’t want. Usually, we just want to upload images and text files (html files are usually text). Change your Uploader script to:

```
property allowedTypes : {"JPEG", "TEXT", "HTML"}

on open (theItems)
  repeat with anItem in theItems
    tell application "Finder"
      copy the name of item anItem to theName
      copy the file type of item anItem to theType
    end tell

    if theType is not in allowedTypes then
      display dialog theType & " (" & theName & ") is not allowed for upload."
    end if
  end repeat
end open
```

We created a property (very much like a variable) that contains our allowed types. If the file is not an allowed type, we display a warning message. This allows you to add that type to the list of allowed types if you wish to.

Finally, we need to tell Fetch to upload the file. You may wish to look at Fetch’s dictionary.

```

property allowedTypes : {"JPEG", "TEXT", "HTML"}

--replace the following with your host's information
property remoteHost : "hostname.example.com"
property remotePass : "password"
property remoteUser : "username"
property basePath : "/path/to/upload/"

on open (theItems)
  repeat with anItem in theItems
    tell application "Finder"
      copy the name of item anItem to theName
      copy the file type of item anItem to theType
    end tell

    if theType is in allowedTypes then
      copy basePath & theName to fullPath
      tell application "Fetch"
        activate
        store anItem host remoteHost path fullPath user remoteUser password remotePass
          with sftp
      end tell
    else
      display dialog theType & " (" & theName & ") is not allowed for upload."
    end if
  end repeat
end open

```

The same script should work with Interarchy.

We're using an "if" structure in this script. The "if" structure performs everything between "if" and "end if" depending on certain conditions. Often, there will be one set of statements before an "else" for if the condition works out, and another set of statements after an "else" for if the condition does not work out.

If "theType" is in the list of "allowedTypes", we work with Fetch. Otherwise, we inform the user that this is not an allowed type.

Other Things You Might Do

Create a simple alarm clock: ask for a message, a time, and then sleep to that time and speak that message.

Publish a word processing document as RTF, HTML, and PDF--then automatically upload it using the other scripts.

Grab and write a random quote from a database of quotes.

Combine multiple PDF files into a single file.

APPLESCRIPT AND THE COMMAND LINE

If you use OS X and you use OS X's command line or cron scripts, you can combine AppleScript with the command line. If you have no idea what that last sentence means, you can safely skip this chapter. This is not a tutorial on using the command line or on using Unix.

Running AppleScript from the Command Line

The easiest way to add an AppleScript to your command line application is through the “open” command. If you type “open ~/Documents/MyAppleScript” on the command line, your AppleScript called “MyAppleScript” in your Documents folder will run. You can use “open” in conjunction with cron, with Perl scripts, or with any of the shell scripting languages. Call it as you would any other command line utility.

You can also call—and create—AppleScripts with the “osa” commands. These are “osascript”, “osacompile”, and “osalang”. In some versions of OS X, you will need to call these by using the full path to the command, that is, “/usr/bin/osascript”. If you simply type “osascript” on the command line, it will give you an obscure error. The “osa” commands cannot be used via cron if you intend to call another application, such as Microsoft Word or FileMaker. Due to the internal workings of Mac OS X's version of Unix, cron does not have permission to call GUI applications (this is a horribly simplified reason, look on the newsgroups if you want more information). The “open” command will work from cron, however.

And, of course, remember that if you plan on calling an application from AppleScript, that application must be able to run. If it is a GUI application, then your GUI must be running: you must be logged into your computer. If you leave OS X turned on but logged out, GUI applications will be unable to open.

AppleScript is great for those scripting tasks for which the Unix “pipeline” metaphor doesn't make sense.

Calling the Command Line from AppleScript

You can also call your command-line scripts and utilities from within your AppleScripts. Use the “do shell script” command for that. For example, if you have a Perl script called “makeHTML” in “/usr/local/bin” you could use

```
do shell script "/usr/local/bin/makeHTML"
```

to run the command.

MORE INFORMATION

AppleScript: The Definitive Guide

<http://www.hoboes.com/Mimsy/hacks/applescript-definitive/>

AppleScript is an extremely useful tool for automating the repetitive things you do in Mac OS X applications, as well as for tying a long workflow together into a double-clickable icon. This Definitive Guide is a great reference for the AppleScript scripting language.

AppleScript on Negative Space

<http://www.hoboes.com/Search/tags/AppleScript/>

I've written several application-specific tutorials and examples on my blog and web site.

Introduction to AppleScript Overview

<http://developer.apple.com/library/mac/>

This is Apple's guide to AppleScript. It's more than a bit technical, but also comprehensive.

“The best book on programming for the layman is *Alice in Wonderland*; but that's because it's the best book on anything for the layman.”

OPEN SOURCE LICENSE

Gnu Free Documentation License

Version 1.3, 3 November 2008

Copyright (C) 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The purpose of this License is to make a manual, textbook, or other functional and useful document “free” in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. Applicability and definitions

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The “Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public,

that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

The “publisher” means any person or entity that distributes copies of the Document to the public.

A section “Entitled XYZ” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as “Acknowledgements”, “Dedications”, “Endorsements”, or “History”.) To “Preserve the Title” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. Verbatim copying

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. Copying in quantity

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. Modifications

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the

Document itself, or if the original publisher of the version it refers to gives permission.

- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. Combining documents

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. Collections of documents

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. Aggregation with independent works

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. Translation

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled “Acknowledgements”, “Dedications”, or “History”, the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. Termination

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of

this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. Future revisions of this license

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. Relicensing

“Massive Multiauthor Collaboration Site” (or “MMC Site”) means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A “Massive Multiauthor Collaboration” (or “MMC”) contained in the site means any set of copyrightable works thus published on the MMC site.

“CC-BY-SA” means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

“Incorporate” means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is “eligible for relicensing” if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.