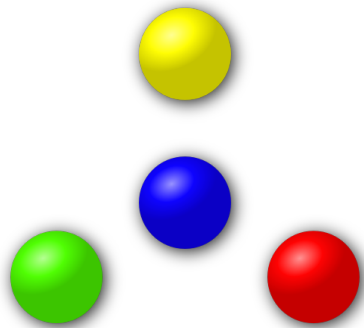


MOVICON

0.2b 13 September 2009



Guillaume Pasquet

This manual documents movicon (version 0.2b, 13 September 2009).

Copyright © 2009 movicon project.

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

Table of Contents

1	Introduction	1
2	Installation	1
3	Quick Tutorial	1
3.1	Movicon’s MVC concept.....	1
3.2	Folders.....	2
3.3	Database setup.....	2
3.4	Configuration	2
3.5	Creating the controller	3
3.6	Making the view	3
3.7	The details page	4
3.8	Making a form	5
4	Standard library	6
4.1	lib-controller.php.....	6
4.1.1	movicon_load_controller	6
4.1.2	movicon_get_segment	6
4.1.3	movicon_get_session	6
4.1.4	movicon_set_session	7
4.1.5	movicon_unset_session.....	7
4.1.6	movicon_get_post.....	7
4.1.7	movicon_get_get	7
4.1.8	movicon_print_r	7
4.2	lib-model.php	7
4.2.1	movicon_load_model.....	7
4.3	lib-view.php.....	7
4.3.1	movicon_load_view	7
4.3.2	movicon_site_uri.....	7
4.3.3	movicon_404	7
4.3.4	movicon_site_redirect	8
4.3.5	movicon_form_open	8
4.3.6	movicon_anchor	8
4.3.7	movicon_redirect	8
4.3.8	movicon_base_uri.....	8
4.3.9	movicon_resource.....	8
4.4	Database drivers	8
4.4.1	Array conditions	8
4.4.2	movicon_db_get	9
4.4.3	movicon_db_insert.....	9
4.4.4	movicon_db_update	10
5	GNU Free Documentation License	10

1 Introduction

Movicon is a very simple implementation of the MVC framework for web applications written in PHP. Unlike other famous implementations, it aims to be simple, straightforward and particularly easy to tweak and deploy.

Movicon offers many helper functions to simplify the task of programming and to easily capitalise your code. The standard library features html and site-wide helper functions, and includes an fully abstracted database module. This module is shipped with a *MySQL* driver.

This document explains how to deploy and use *movicon*. It also describes each function of *movicon*'s standard library.

2 Installation

You can get *movicon* from it's official website; <http://home.gna.org/movicon/>. Official releases are available as compressed packages, and you can also checkout the latest code from the version-controlled repository.

However way you choose, unpack the resulting code into a place where your webserver can access. Once done, browse to the location where you installed *movicon*. If you can see *movicon*'s welcome page, then everything's ready!

3 Quick Tutorial

This short tutorial will briefly describe and explain how to set up a proper *movicon* work environment, how to program a small CD directory, and how to package and release it.

Note that this tutorial assumes that you know how to set up and administrate *MySQL*. The details of how to set up the required tables will not be discussed here.

3.1 *Movicon*'s MVC concept

The *MVC* concept is a way of splitting your code into manageable and maintainable pieces based on their function within the project. Thus three distinct functions are defined:

Model The model is the data-related part of the code. Its job is to read and write and process data at a low-level (cleaning the data up etc.)

View A view is an interface to your program. It can be in any form, the point is that things are shown and things are captured there.

Controller The controller is the intelligent part of the program. It takes input from the view and outputs to it, reads and stores data through the model.

Well, you could think that it's not as impressive as it first sounds, and that — with a bit of method — you could just do it by yourself without framework. Well, you'd be right, and *movicon*'s goal is not to get in your way; it is made to help you structuring your code in the MVC way.

3.2 Folders

Movicon's root folder contains multiple files and folders. Each is important to the framework operation. Note that you can move the folders around provided that you also reflect the changes in the framework's configuration.

'`index.php`'

This is *movicon*'s bootstrap. You can of course modify it as you require, but must leave the several lines present in this file.

'`conf.inc.php`'

This file contains the framework's configuration. The location of the system folders and the modules to load are defined here.

'`404.php`'

When a page that doesn't exist is requested, the system will automatically redirect people to this page. You can make it look nice and personalise it if you like.

'`lib/`'

This is the system's folder. Dragons be there! Trespass only if you know your business. This folder will be described later in this document.

'`controllers/`'

This folder contains the controller. You will write one during this tutorial.

'`models/`'

The models need to be created here.

'`views/`'

The views are here. For web applications, views are typically a lot of HTML with some bits of embedded PHP.

3.3 Database setup

For this tutorial, we will need two simple tables within a database. The first table will contain the CDs. you can call it `cds`.

id	title	artist
1	Out of it	Brad Sucks
2	Between two worlds	Maya Filipic

Then we will need a table for the songs of each CD. Let's call it `songs`. Of course, each song must be linked to an album and the column `cd_id` will contain the album's id.

id	cd_id	track	title
1	1	1	Dropping out of school
2	1	2	Certain death
3	1	3	Fake it
4	2	1	Stories from Emona I
5	2	2	Stories from Emona II

3.4 Configuration

First of all, we need to configure our framework. Go ahead and open the file '`conf.inc.php`'. It contains system definitions in arrays. Read through them to get familiar with *movicon*'s setup.

Now, we will use *MySQL* as our database to hold the CD collection's data. To do this, set to `mysql` the `db-type` option:

```
$cfg['db_type'] = "mysql";
```

The following *db_conn* options are options specific to *MySQL*. Fill these options with the right values so that the framework can successfully establish a connection with your database server.

Finally, we need to tell the framework to load the database connection. At startup, the system looks into the *components* array to see what needs to be loaded. Locate the array, and put the 'database' value in:

```
$cfg['components'] = array('database');
```

Please note that the *components* option needs to be an array. A single string will not be loaded.

3.5 Creating the controller

If you open a web-browser to the location of your 'index.php', the default controller will be called. This is defined in the configuration as the *def_controller*. You can change this value to the name of the controller you like. Alternatively, you can reach your controller directly in the first segment of URI.

The URI segments are bits of text separated by slashes in the URI. For example <http://mydomain/index.php/1/2/3>. You can see the segments 1, 2 and 3. These are defined right after 'index.php'. Segment 1 is the name of the controller, segment 2 is the name of the controller's function to be executed. All other segments are only arguments to the called function.

Create a file called 'cd.php' in the 'controllers/' folder. The first thing we need to do is to create an *index()* function. If no function is called amongst the segments, then the *index()* function will be called by default.

```
1 function index() {
2     $data['cd_table'] = movicon_db_get('cd');
3     movicon_load_view('cd_list_view', $data);
4 }
```

Let's explain this bit of code before going further. The function *movicon_db_get()* fetches a whole table as a jagged array. *movicon_load_view()* loads the specified view for display. We are using the *data[]* array to store the data that we wish to pass on to the view.

3.6 Making the view

If you browse onto your controller (i.e. <http://mydomain/index.php/cd/>), you will see that nothing is displayed yet. We need to make a view for this. Create the file 'cd_view.php' in the 'views/' folder and put the following code inside:

```
1 <h1>My CD collection</h1>
2 <h2>Albums list</h2>
3
4 <?php foreach($data['cd_table'] as $album):?>
5 <h4><?php echo $album['title'];?></h4>
```

```

6 <p><?php echo $album['artist'];?></p>
7 <p><?php movicon_anchor('cd/details/' . $album['id'], 'Details');?></p>
8 <hr/>
9 <?php endforeach;?>

```

As you can see, a view is mostly HTML. The php bits are only there to display the data processed by the controller. This code takes advantage of a foreach loop to process the table's data row by row.

Note that — on line 7 — we have prepared a link to see the album's details. The function `movicon_anchor()` is handy to create internal links, just put the segments as first argument and the link's text as second parameter.

On the *details* function, we want to display track informations for the selected album only. Passing the album id to the function is the ideal way to do this and you can see that we actually pass it in the third segment.

If you try to access the controller, it will now show your album's list.

3.7 The details page

Let's create the details function and view. Edit the controller 'cd.php' and add the following code:

```

1 function details() {
2     $id = movicon_get_segment(3);
3     $data['songs'] = movicon_db_get('songs', array('cd_id' => $id));
4     movicon_load_view('details_view', $data);
5 }

```

On line two, we get the argument passed in the third segment. Then this value is used during the table's query on line three. It is interesting to see how the the conditional queries work in movicon. All conditions are to be stated in an array, the array must be associative in the way 'field' => 'value'. More advanced type of queries are described in this documentation, but this suffices for the moment.

Now to the view, create a file called 'details_view.php' in the 'views/' folder and put the following code into it.

```

1 <h1>My CD collection</h1>
2 <h2>Album details</h2>
3
4 <table>
5     <tr><th>Track number</th><th>Title</th></tr>
5     <?php foreach($data['songs'] as $song):?>
6     <tr>
7         <td><?php echo $song['track'];?></td>
8         <td><?php echo $song['title'];?></td>
9     </tr>
10    <?php endforeach;?>
11 </table>

```

Just as we did for the albums list, we use a foreach loop to display all the songs into a table. Nothing new here.

Now if you click on the *details* link, you will see the list of the songs in the album. We already have a basic CD collection manager, but that's not enough; we can't create new albums yet.

3.8 Making a form

Let's create an input form on the album list so we can easily insert new albums. The form needs to allow input of the album's name, author and the list of tracks. Open up the view 'cd_view.php' first and modify it as follows.

```

1 <h1>My CD collection</h1>
2 <h2>Albums list</h2>
3
4 <?php foreach($data['cd_table'] as $album):?>
5 <h4><?php echo $album['title'];?></h4>
6 <p><?php echo $album['artist'];?></p>
7 <p><?php movicon_anchor('cd/details/' . $album['id'], 'Details');?></p>
8 <hr/>
9 <?php endforeach;?>
10
11 <!-- Beginning of form -->
12 <?php movicon_form_open('cd/add_cd');?>
13 <p>Album name: <input type="text" name="album[title]"/></p>
14 <p>Author: <input type="text" name="album[artist]"/></p>
15 <p>
16     Songs info (input <em>track;title</em>):<br/>
17     <textarea name="songs" rows="10" cols="35"></textarea>
18 </p>
19 </form>

```

The form is quite easy to read through. The helper `movicon_form_open()` opens automatically an HTML form that links to the specified segments.

We are using a `textarea` to enter the songs info. Note that we need the songs info to be entered into the `textarea` as `track;title`. This is a crude way and could probably be improved with some javascript, but it'll work.

Now open the 'cd.php' controller and add the following function.

```

1 function add_cd() {
2     // Adding the album.
3     $album_id = movicon_db_insert('cds', movicon_get_post('album'));
4     // Formatting the songs data.
5     $songs = explode("\n", movicon_get_post('songs'));
6     // Inserting data.
7     foreach($songs as $song) {
8         $song_row = explode(";", $song);
9         movicon_db_insert('songs',
10             array('NULL',
11                 $album_id,
12                 $song_row[0],

```

```
13             $song_row[1]
14             )
15         )
16     }
17     movicon_site_redirect('cd/');
18 }
```

Let's have a closer look at some points here. First — on line 3 — we insert the album's details into the database. This returns us the ID of the inserted row into `$album_id`. Then there's a whole job to break the textarea's input into arrays, based first on the new lines characters, and then on the semicolon character.

The insert query on line 9 is interesting for it does not involve an associative array. This will work, however you must take great care to put the values in the right order, or your program might break.

Finally, since this function does not print anything on screen, we don't want the user to look at a blank page. Rather we use the helper `movicon_site_redirect()` which redirects the user on the specified segments.

Congratulations, you have achieved your first *movicon* application!

4 Standard library

This chapter will list and describe each function that *movicon* makes available to help you in your programming task. The functions of the *standard library* will be grouped by file, so as to make it easier to find them if you need to do some modifications.

4.1 lib-controller.php

'lib-controller.php' contains all the controller-related functions.

4.1.1 movicon_load_controller

This function allows you to load a controller. It is called by the host everytime *movicon* is accessed. The first parameter must contain the controller's name.

If the controller is not present, then the function redirects to a 404 error and returns false. Otherwise, it loads the controller and returns true.

If no function is specified in the segments, then `movicon_load_controller` attempts to call the `index()` function within the controller.

If either the specified function or the `index()` function doesn't exist, then the function redirects to a 404 page and returns false.

4.1.2 movicon_get_segment

This function is made to read URI segments. Segments are numbered from 1, not 0. If the segment you require does not exist, this function returns false. Otherwise, it will return the contents of the segment.

4.1.3 movicon_get_session

This function helps to get a session variable. It takes the variable's name as parameter. If the variable does not exist, then the function will return false.

4.1.4 `movicon_set_session`

This function helps setting session variables. It simply takes two parameters, the variable's name and its value. It doesn't return anything.

4.1.5 `movicon_unset_session`

This function deletes a session variable. It takes the variable's name as first argument and does not return anything.

4.1.6 `movicon_get_post`

This is a helper function to get `$_POST` data. The first parameter is the variable's name. It returns false when the variable doesn't exist.

4.1.7 `movicon_get_get`

This helper fetches the `$_GET` data. Its only parameter is the name of the variable to fetch. If the specified variable does not exist, it returns false.

4.1.8 `movicon_print_r`

This function helps you to dump an array within *pre* tags. It is only useful for debugging while you make your application.

4.2 `lib-model.php`

This file contains all helper functions related to the models.

4.2.1 `movicon_load_model`

As for the controllers, this function loads up the model required as first parameter. If the model doesn't exist, then it returns false.

4.3 `lib-view.php`

This file contains all functions that are useful to work with views.

4.3.1 `movicon_load_view`

This function takes a view name as first argument and optionally some data as second argument.

It attempts to load the required view. If the view doesn't exist, then it shows a 404 error and returns false.

4.3.2 `movicon_site_uri`

This function takes a string representing URI segments as parameter. It returns the full URI with segments.

4.3.3 `movicon_404`

Takes no parameters. It generates an error 404 page.

4.3.4 `movicon_site_redirect`

Takes URI segments as argument. It redirects to the specified URI. If the URI does not exist, this is handled by the controller and shows a 404 error page.

4.3.5 `movicon_form_open`

Takes URI segments as argument. It opens an html form pointing to the specified segments.

4.3.6 `movicon_anchor`

Takes URI segments as first argument and an explicit name as second argument. It generates an html hyperlink to the specified segment and affects the given name to it.

4.3.7 `movicon_redirect`

Takes an URI as argument and redirects there. This doesn't work with segments, just plain URI.

4.3.8 `movicon_base_uri`

Takes a relative URI as argument and returns the absolute URI from *movicon*'s root.

4.3.9 `movicon_resource`

Takes a relative path as argument. It returns the absolute path to resources directory corresponding to the given relative path.

4.4 Database drivers

Movicon provides a fully abstracted database interface. It is designed to handle most common actions without having to type any DB-specific SQL dialect.

The default *movicon* package ships a *null* database driver, which exists only to demonstrate the standard database interface, and a *MySQL* driver, which is fully functional.

If your favourite database driver is missing, you can probably modify the *MySQL* driver to support your database instead. Releasing new drivers to include them in the standard distribution would of course be much appreciated.

4.4.1 Array conditions

An array condition is an array that will be parsed to generate a *WHERE* clause to the query. The array must be associative in the form `'field'=>'value'`. The default comparison is an *equal* — i.e. `array('field' => 'value')` would produce `WHERE 'field'='value'`. You can change this behaviour with value prefixes:

- ! This expresses a difference: `WHERE 'field' != 'value'`
- > or < These are greater than and lesser than comparisons. If they are followed by an `=`, then an equality property is added: `WHERE 'field'>='value'`
- % This generates the *LIKE* operator for *MySQL*. If you want to use two wildcards on the searched text, then you need to put two `%` before the value: `array('field'=>'%value%')` will generate `WHERE 'field' LIKE '%value%'`.

A hash indicates that the value is actually a table's field. It can be just the field name, or the table and field name in the form: '#table.field'.

If your associative array has many elements, then these are automatically linked with an *AND* relation:

```
array('field1'=>'value1', 'field2'=>'!value2')
```

would generate:

```
WHERE 'field1' = 'value1' AND 'field2' != 'value2'
```

If you want to make an *OR* relation, then you may use the | prefix on the field's name as follows:

```
array('field1'=>'value1', '|field2'=>'!value2')
```

would generate:

```
WHERE 'field1' = 'value1' OR 'field2' != 'value2'
```

Finally, you could also want to make a series of sub-comparison enclosed in brackets. For this, you will need to make a jagged array with *OR* or *AND* as key:

```
array('field1' => 'value1',
      'AND' => array('field2' => 'value2',
                    '|field3' => '>value3')
      'OR'  => array('field2' => '!value2',
                    'field4' => '<value4')
    )
```

would generate the following *MySQL* statement:

```
WHERE 'field1' = 'value1'
      AND ('field2' = 'value2'
           OR 'field3' > 'value3')
      OR ('field2' != 'value2'
           AND 'field4' < 'value4')
```

4.4.2 movicon_db_get

This function fetches an entire database table into a jagged array. Its first argument is the name of the table and the second is an array condition.

The first argument may also be a non-associative array, in which case each value is considered as a table's name. This feature is especially useful for joint queries.

If the query fails, this function returns false.

4.4.3 movicon_db_insert

This function takes a table name and an array of values as arguments. The values array can either be associative or not.

The array can be associative as `array('field'=>'value')`, in this case, the value '`value`' will be inserted in the field '`field`'.

Otherwise, each value will be inserted in the same sequence as the fields. This means that if you do not put the right number of values or if your values are not in the right sequence order, then your query might fail, or you might break your table. You are encouraged to use an associative array.

This function will return you the id of the inserted row, 0 if no id is defined in the table, and false if the insert query fails.

4.4.4 `movicon_db_update`

This function takes the table to modify as first argument, an associative array of values as second argument and a condition array as third argument.

The function returns true on success or false on error.

5 GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 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.

0. 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 non-commercially. 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.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C)  year  your name.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
Texts. A copy of the license is included in the section entitled ‘‘GNU
Free Documentation License’’.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with...Texts.” line with this:

```
with the Invariant Sections being list their titles, with
the Front-Cover Texts being list, and with the Back-Cover Texts
being list.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.