

Election of a free software license

A practical case

Sergio Arroutbi Braojos

2012/11/26

1 Introduction

As a web developer, someone could think that development being performed has not to be licensed. **Code used is not normally compiled**, as multiple language as, in example, "php", are interpreted by web servers.

Despite this, there are **additional tools** used (scripts to retrieve data from other webs, graphics generation, etc.) that are not part of the web page itself, but help on generating data that will be inserted on it, as, for example, the database.

This document studies **a particular web project**, called "buscobici" (bike-search in Spanish language). There is a web page, buscobici.com, that allows searching for a particular bike, based mainly on custom searches on a Database that contain links to bike bid offers in different stores (similar, for example, to cars.com).

But the web page is not the only software itself to be considered. There is also a git repository, containing a bunch of tools to implement data retrieving, adaptation and population, statistics generation, and the database itself. This repository is under github.com, in particular, under the path:

<https://github.com/sarroutbi/buscobici>

So, in fact, **the code is public accesible**. However, this does not mean that the project is free software, as there is no license information, so, **by default, applying specific national laws of the author's country, Spain in this case, all rights are reserved**.

This document will study this software, how it is implemented, if modifications to other free or non free software have been done, and if so, if redistribution is being performed.

Moreover this, and taking into consideration previous analysis, this document will consider the reasons why this software should be licensed under a free software license, which steps to perform, and the license that could be assigned, in case an election of the license can be done.

2 Acronyms

CC - Creative Commons
FLOSS - Free Libre Open Source Software
GNU - GNU is Not Unix
GPL - General Public License
ISC - Internet Systems Consortium
ODC - Open Data Commons
ODC-DbCL - ODC Database Content License
ODC-ODbL - ODC Open Database License
PHP - PHP: Hypertext Preprocessor
PSF - Python Software Foundation

3 Analysis of implemented source code

The first thing that should be done is the **analysis of the code** that has been implemented, programming languages used, license affecting those programs, and modifications being implemented to programs that are copyrighted by others.

In this case of study, an analysis on the Git repository will be performed. After clonning repository, by means of using "sloccount" applications, it is discovered the existance of next **programming languages Lines of Code under the project**:

bash: 3135 (87.13%)
php: 330 (9.17%)
python: 78 (2.17%)
sql: 55 (1.53%)

"generated using David A. Wheeler's 'SLOCCount'."

So, project, mainly, is composed of "**bash**" scripts, very few lines of code in "**php**" programming language, some very basic "**python**" tools and "sql" code for Database Creation.

The tool used, "sloccount" does not consider some "gnuplot" existing script as another different tool, but, in fact, there are scripts on the project using this tool, so we must also consider this tool, especially when it contains the prefix "gnu".

Taking into account previous considerations, **license of each of the tools and programming languages should be identified**, to consider later license assumptions:

Bash Version:4.2 License: **GPLv3** (or later)

Php Version:5.3 License: **PHP3.01 License**

Python Version:5.2 License: **PSF 2**

GnuPlot Version:5.2 License: **Gnuplot** own license

All tools and program languages are under terms of free software licenses. It should also be highlighted that, despite name of applications, not all applications are what seem to be. **In this case, we find that "Gnuplot" application is not GPL, as the application has nothing to do with the GNU Project.**

"gnuplot" gnu prefix was acquired to substitute the "new" prefix of "new-plot", another program of same characteristics it is based on. So it is clearly not a good idea to make assumptions regarding licenses based on other ideas different from reading a specific license, as the name of the project.

Despite previous considerations, no C/C++ programs exist, so it can be assumed that there are **no programs linking against libraries under GPL license.**

It should also be identified if there is a planification on using new programming languages or existing software, and if so, if modifications will be performed. It is not the case for this particular project. In terms of application, an analysis of what is being done with each program

must be performed, to identify which software freedoms we are entitled, but also our obligations, if any:

Use: The project is using the code. No restrictions to use in any of the licenses, as they are all free software licenses.

Inspection: The project needs no inspection of the tools used source code. However, source code has been obtained by easy means, in particular, via Debian APT repositories, to inspect the licenses.

Modification: The project is not modifying any third party software, so no considerations must be performed on modification.

Redistribution: The project is not redistributing any third party software. All the code existing on the git repository is self - development code, so no considerations must be performed regarding redistribution.

To summarize, our project just uses free software scripting languages and programming languages, so no considerations have to be considered on copyleft clauses. Once considered previous statements, the fact that no modification or redistribution on the original copyleft used source code is done, **it can be ensured that an election of any free software license can be performed.**

It is somehow a similar case to what a Blender user could produce (animations, 3d model, scripts, etc.) with a GPL licensed tool. Blender project asserts to a question about selling Blender creation [1]:

”Anything you create with Blender - whether it’s graphics, movies, scripts, exported 3d files or the .blend files themselves - is your sole property, and can be licensed or sold under any conditions you prefer.”

4 Requirements to accomplish

In this point of the analysis, one thing is obvious. The usage of third party free software being performed on the project allows us to license our code under any software license, privative or free.

First of all, a consideration on which freedoms must be ensured has to be

taken into account. Licensing under a free software license means:

1 - Freedom to use: No limitations on who can use the code and the purpose of usage of the code. In this case, we are allowing to use the code for every use inside legal possibilities, even to create a similar web page that could take traffic from ours.

2 - Freedom to inspect: No limitations on access to the source code have to be ensured for the project. The code will be public accesible and with the selected license available. Future versions will be public as well.

3 - Freedom to modify: No limitations to modifications and use of the modifications. The code can be modified. Later, considerations on what the licensees are entitled to on modifications will be analysed.

4 - Freedom to redistribute source, and redistribute changes: Selection of a free software license entitles the users to redistribute the code. An analysis of the terms to the redistribution will be analyzed.

So, until here, the project fits into a free software license. All the considerations taken for software can be extended to non-software, in terms of usage, inspection, modification and redistribution.

However, the project author have to bear in mind on different aspects that will decide on the election of both the source code license and the non source code license. They are enumerated below [2]:

1 - Do I want to allow privatization of derivative works, and if so, do I want the new privative derivative work to be splitted, keeping out my authorship?

Yes, privatization of derivative works is allowed for any purpose. Freedom to do so as well. However, if derivative works are done, control is lost on the derivative work. Authorship of the new code can be maintained, as this is not a special quality project with great consideration from the community, so no special considerations on authorship change have to be taking into account. For this reason, copyleft licenses will not be considered.

2 - Do I want developers return their modifications to the community, or me as original author, in particular?

There is no need of it. It is very likely this project software not to be modified, and if so, modifications will not suppose great profits to the community,

due to the fact that the code is specialized to do a particular action (crawling from some stores). There is no necessity to license under copyleft license due to this reason.

3 - Do I want to allow licensees to merge or link their program with mine?

This project ensures total freedom to licensees to merge against this program, and freedom to do whatever needed under the license needed for the resulting code. The only limitation, keeping the selected license copyright note.

4 - Do I want widespread coverage, avoid any restriction and/or try to establish a standard?

No need to a widespread coverage. As it was stated before, this project is for particular crawling, not pretending to establish standards, avoid restrictions or reaching to a huge amount of users. Copyleft license is not necessary for this reason neither.

5 - Should my program run with one in particular? Have it any restrictions?

As stated before, no restrictions. Total freedom to choose whatever license needed.

6 - Is there risk that someone requiring a patent license over program?

Program does not use special algorithms, neither advanced programming libraries. It just connect to webs, operates on strings, parses results to create crawling resultant files, generate database files with previous resultant crawling files and generate statistics according to the data.

No special protection against software patents should be considered.

7 - What about documentation and non-software files?

Documentation and non-software files will ensure users freedom as well. By the moment, no documentation has been generated for the project, and for this reason, no considerations will be taken on documentation itself.

The only non software component public available is the Database, as database backups are generated and submitted to a Git repository. This Database can be generated with project source, so it makes no sense to provide it with different permissions compared to the source code itself. For

this reason, **the Database will be licensed under same terms as the source code itself.**

In another vein, statistics information generated with Gnuplot is under test. No trustable statistics have been generated, neither public availability is provided, so no considerations will be assumed for statistics as well. Same issues arise in terms of statistics compared to the database, so same considerations around database will be taken.

However, if necessary for the future, documentation will be licensed in the same terms as before. The documentation will have freedom to be used, inspected, modified and distributed with no restrictions.

5 License selection

Once requirements have been analysed, a selection of the license have to be carried out in order to accomplish them.

5.1 Software code license

Copyleft licenses have been discarded, due to the fact that the code can be used to make privative derivative works. Among permissive licenses, **there is no necessity to:**

- **Keep away authorship of derivative works**, as some permissive license obligates.
- **Insert patent clauses.**

Taking into account previous considerations, permissive licenses are the only option. Moreover this, Academic Licenses are the ones that best fit into the requirements. In particular, ISC license is the easiest license ever seen, because of its simplicity.

This **simplicity**, together with the fact that the license accomplishes requirements, adding the fact that **user's freedom is considered above all**, even for making derivative works, makes this license a perfect option for this particular project.

5.2 Non-software license

On the other hand, a consideration on Database licensing will be considered. Open Data Project [3] is an organisation that provides legal tools to enforce Open Data availability. They also clarify why Databases does not fit well into CC licenses [4]:

”Different types of subject matter (e.g. code, content or data) necessitate differences in licensing. Licenses designed for one type of subject matter as CC licenses were designed for content, and FLOSS licenses for code are not always best suited to licensing another type of subject matter.”

Taking into account previous assertion, there is a whole bunch of licenses defined in the Open Data Commons that helps on selection of a specific DataBase license. This document does not pretend to study all Open Data licenses, but they will be enumerated below:

1. Public Domain Dedication and License (**PDDL**) - Public Domain for data/databases
2. Attribution License (**ODC-By**) - Attribution for data/databases
3. Open Database License (**ODC-ODbL**) - Attribution Share-Alike for data/databases

Among these kind of licenses, the only necessary restriction wished is attribution, so ODC-By license is the one that fits better to our purpose.

For future licensing on documentation, CC-By license will be selected, as this will mean an heterogenous freedom for source code, database and documentation.

6 Practical implementation

Previous sections allowed clarification on which licenses have been selected. ISC license has been selected for Source Code licensing, while ODC-By license has been selected for Database. Next sections cover the practical application of this licenses, and means for the application of the license to the different source code artifacts.

6.1 Software code license application

As mentioned before, ISC license is one of the simplest licenses nowadays. It is difficult to provide free software freedoms in a shorter way. License can be obtained on ISC web page [5]. The simplicity of the license allow us to include it on any single file, for the license to be taken in case of individual file retrieve by anyone. Taking into account the source code files, there is basically two type of files:

- **Scripts:** Python, gnuplot and bash scripts. All this kind of source files will contain the license in the header of the file, with each line preceeded of the comment sign (#) common to this files. Resulting header will be next:

```
#
# Copyright 2012-2013 Sergio Arroutbi Braojos <sarroutbi@gmail.com>
#
# Permission to use, copy, modify, and/or distribute this software
# for any purpose with or without fee is hereby granted, provided that
# the above copyright notice and this permission notice appear in all copies.
#
# THE SOFTWARE IS PROVIDED AS IS AND THE AUTHOR
# DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS
# SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
# MERCHANTABILITY AND FITNESS.
# IN NO EVENT SHALL THE AUTHOR BE LIABLE
# FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL
# DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
# LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF
# CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION,
# ARISING OUT OF OR IN CONNECTION WITH THE USE OR
# PERFORMANCE OF THIS SOFTWARE.
#
```

- **PHP Files:** Apart from scripts, license should be applied to PHP files. PHP files have a different syntax, and resulting header of PHP files will be as follows:

```
/**
 * Copyright 2012-2013 Sergio Arroutbi Braojos <sarroutbi@gmail.com>
 *
```

```

* Permission to use, copy, modify, and/or distribute this software
* for any purpose with or without fee is hereby granted, provided that
* the above copyright notice and this permission notice appear in all copies.
*
* THE SOFTWARE IS PROVIDED AS IS AND THE AUTHOR
* DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS
* SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
* MERCHANTABILITY AND FITNESS.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE
* FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL
* DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
* LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF
* CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION,
* ARISING OUT OF OR IN CONNECTION WITH THE USE OR
* PERFORMANCE OF THIS SOFTWARE.
**/

```

Moreover this, a file named "COPYRIGHT" will be included in the root directory of the repository GIT, to clarify in a different file also the information having to do with the code. The COPYRIGHT notice will contain plain text with the information showed above, with no further comments characters (#, *) in the beginning of each line.

6.2 Non-software license application

Not everything in a project is software and source code. Apart from that, other artifacts as Databases appear, and need a license too in order to clarify Copyright and permissions associated to the database.

As previously stated, license selected will be ODC-by. This license allows all freedoms, with the only restriction of attribution of creator of the database. In order to license a database, ODC asserts on ODC-by license [6]:

"Insert prominently in all relevant locations a statement such as (replacing DATA(BASE)-NAME with the name of your data/database):
This DATA(BASE)-NAME is made available under the Open Data Commons Attribution License: <http://opendatacommons.org/licenses/by/version>"

So, this is what will be done in terms of copyright for the database. In particular, this note will be inserted on the last database updated. It must

be highlighted that this kind of license is for the database itself. Apart from this, a license will be inserted to license the contents [6]:

”Use your own license for the Contents. You are welcome to apply your own specific license to the contents of the database. To do this just add a the second sentence with information about the license you wish to use.”

To do so, next statement can be copied from other license in ODC, in particular, from ODC-ODbL license, to reference ODC-DbCL:

”Any rights in individual contents of the database are licensed under the Database Contents License: <http://opendatacommons.org/licenses/dbcl/1.0/>”

So, to summarize, from now on database backups will contain a header specifying the license of the database. Considering that backups are postgresql format files, with comments preceded by characters ”–”, the header will be as follows:

```
–
– This ”bikesearch” database is made available
– under the Open Data Commons Attribution License:
– http://opendatacommons.org/licenses/by/1.0
–
– Any rights in individual contents of the database
– are licensed under the Database Contents License:
– http://opendatacommons.org/licenses/dbcl/1.0
–
```

Apart from that, COPYRIGHT.database and COPYRIGHT.contents files will be inserted in database backups directory to contain a full license of the database and its contents. Once all previous insertions have been performed, and code and license files are submitted to repository, the project can be considered as Free Software project.

7 Conclusions

Licensing code is a not obvious decision. Deep considerations have to be taken into account:

- Do I really want my code to be free software? And the other (often for-

gotten) stuff, as, in example, databases, images, statistics, documentation?

If so:

- Do I want my code under a permissive or under a reciprocal Free Software License, taking into consideration requirements for my particular project?

Why? Why not?

- Am I using, modifying, redistributing third party Free Software? If so:

- Under which license is copyrighted this third party code?

This document tried to investigate this and other **aspects having to do with the election of a license**, in particular, for a Free Software license.

It should also be foregrounded that, **the earlier these questions are faced, the better**. Leaving licensing for later moment could imply, as in this project, having to change a lot of files once they have been already submitted.

In particular, it can be verified in GIT log, in particular for (commit hash = c4efaf92a5c559571c21e77ec9732f5d382e385f) that for this project license header had to be applied to nearly 70 files, apart from committing the code of the licenses themselves.

If considerations on free software license code would have been adopted earlier, this job would have been easier.

8 References

1. <http://www.blender.org/education-help/faq/gpl-for-artists/>
2. <http://docencia.etsit.urjc.es/moodle/mod/resource/view.php?id=4191>
3. <http://opendatacommons.org>
4. <http://opendatacommons.org/faq/licenses/#why-not-use-a-creative-commons-or-freeopen-source-software-license-for-databases>
5. <http://www.isc.org/software/license>
6. <http://opendatacommons.org/licenses/by/>