

## ccd.net, Community Cultural Development, and Open Source

*Bong Ramilo*

In 2003, ccd.net explored the possibility of opening the source code of the web site itself for collaborative development using Open Source methods used to develop software such as the Linux operating system. This possibility was not pursued due to limitations imposed by the license of the original web site code.

ccd.net then decided to support the development of an Open Source web application that could be integrated into the web site. The plan was to recruit developers who will produce an online application collaboratively, using an Open Source approach that produced software such the Linux operating system.

Before this plan went ahead, however, ccd.net recognised that some groundwork was needed in familiarising the ccd.net community with the concepts and practices of Open Source; this would prepare the community for their participation in any Open Source project ccd.net wishes to undertake.

This article is an initial contribution to discussing the relevance of Open Source to ccd.net in particular and to community cultural development (CCD) in general. It is hoped that discussions about ccd.net vis-a-vis Open Source will generate interest and more discussion about adapting Open Source to projects for and beyond the ccd.net web site.

The discussion will first outline Open Source concepts and issues as preparation for discussing implications and possibilities of Open Source to CCD and ccd.net.

### What is Open Source?

The term "Open Source" is not [yet] part of usual CCD lingo. It would be good to define it from the start to avoid any confusion caused by unfamiliar jargon. There are, however, numerous definitions of "Open Source" – do a "define:open source" in Google and you'll get around 25 definitions from the Web.<sup>1</sup>

Here's one to begin with:

Computer programs or operating systems for which the source code is publicly available are referred to as open-source software. Inherent in the open source philosophy is the freedom of a distributed community of programmers to modify and improve the code. The most widely known example of open-source software is the Linux operating system.<sup>2</sup>

The term Open Source thus refers to whether source code – the instructions to the computer written in a programming language – is available or not (open or closed) for public use, modification, and distribution. All computer programs have source code and in the traditional, commercial model of software development (as practiced by Microsoft, for example) source code is not available to users or developers; closed source computer programs come in "binaries" (that is, in a form readable only by the computer).

Open Source software is considered to be generally more reliable and can improve more quickly

---

<sup>1</sup>An official and more technical definition is available from the co-inventors of the term Open Source. Visit <http://www.opensource.org/docs/definition.php> for the Open Source Definition.

<sup>2</sup><http://iet.ucdavis.edu/glossary/#o>.

than proprietary software because its source code is available to all who wish to use and modify it. The Open Source Initiative describes the advantage of free access to source code thus:

The basic idea behind open source is very simple: When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it, people fix bugs. And this can happen at a speed that, if one is used to the slow pace of conventional software development, seems astonishing.<sup>3</sup>

Free access to source code may not be important to most end-users; how many users would like to modify the source code of their operating system or word processor even if they could? These same users, however, still benefit from the work of others who *do* modify the code to improve the software.

The Open Source community is not limited to programmers; it includes testers, documentors, distributors, and most importantly end-users (many of whom may be programmers too) – the feedback they provide substantially informs the process of debugging and enhancing the software. Eric Raymond describes this community-based development process as one where "many eyeballs tame complexity" in code.<sup>4</sup>

Open Source software is thus also referred or compared to Free Software because of the freedom associated with using, modifying, and distributing it. Open Source in fact arose from the Free Software concept and movement, pioneered in large part by Richard Stallman and the Free Software Foundation. The term Open Source was coined in February 1998 by Eric Raymond and other members of the free software movement as part of a "marketing campaign" to promote free software while avoiding the negative stereotypes associated with the term Free Software.<sup>5</sup>

While Open Source software is free in the sense of liberty, much of it does not necessarily belong to the public domain; that is, Open Source software is licensed. Copyright is protected by a variety of Open Source licenses. The Open Source Initiative lists more than 50 Open Source licenses.<sup>6</sup> Perhaps the most prominent and famous of these licenses – and the one under which Linux is licensed – is the Free Software Foundation's GNU General Public License (GNU GPL or "Copyleft").<sup>7</sup>

Much Open Source software is also free "as in free beer."<sup>8</sup> That is, much of it is royalty-free and can be used and redistributed free of charge. Linux, for example, can be downloaded from various sources or redistributed on discs by anyone without paying any money to Linus Torvalds (the originator and trademark-holder of Linux) or to a Linux distribution outfit (e.g. Red Hat).

<sup>3</sup><http://www.opensource.org>.

<sup>4</sup>Eric Raymond, <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s05.html>.

<sup>5</sup>Eric Raymond describes the origins of Open Source at <http://www.catb.org/~esr/faqs/hacker-revenge.html>.

Raymond claims that the term "Open Source" would avoid the confusion caused by the ambiguity of the word "free" in the English language – "one suggesting a price of zero and one related to the idea of liberty" – and counter "the strong association of the term 'free software' with hostility to intellectual property rights, communism, and other ideas" that would alienate many prospective users of open source or free software. The Free Software Foundation maintain that "the Free Software movement and the Open Source movement are today separate movements with different views and goals, although we can and do work together on some practical projects." They also maintain that while they disagree with Open Source on many basic principles, they see the enemy as proprietary (or closed source) software, not open source software. For the Free Software Foundation's views on the difference between free software and open source, see <http://www.gnu.org/philosophy/free-software-for-freedom.html>.

<sup>6</sup>See <http://www.opensource.org/licenses/> for links to the various licenses.

<sup>7</sup>See <http://www.gnu.org/licenses/gpl.html> for the text of the license.

<sup>8</sup>The Free Software Foundation clarifies its use of "free" thus: Free software is a matter of liberty, not price. To understand the concept, you should think of 'free' as in 'free speech,' not as in 'free beer.' " See <http://www.gnu.org/philosophy/free-sw.html>.

## Open Source and Community

Another dimension of Open Source which makes it interesting and relevant to CCD is that Open Source software development, particularly since Linux, has become strongly community-based and community-driven. Ashton Mills describes this community dimension in Linux (and Open Source) thus:

Linux has evolved, and continues to evolve, under the open source philosophy of the free sharing of ideas. If not for this massive, shared, open community environment, Linux – and much of the open source software that runs the Internet – wouldn't be here. It's a testament to this community and its philosophy that the creation of a powerful operating system such as Linux could have come about through the interaction and cooperation of thousands of people worldwide.

Linux is a symbol of what's possible when we work together as a whole, sharing each other's ideas and building upon each other's work. It's built by everybody, for everybody, and it's free.<sup>9</sup>

There are various models through which Linux and other Open Source projects harness the energies of hundreds if not thousands of participants. Raymond describes the model that evolved principally from the Linux experience as the "Bazaar" where the development process is open to all ideas and contributors, with frequent version releases; this model differs from the traditional "Cathedral" model of software development that involved a small group of programmers who tried to perfect each version before (infrequent) releases.<sup>10</sup>

Note too that Open Source communities are virtual communities, involving contributors (usually volunteers) around the world, cooperating through the Internet. Raymond in fact partly attributes Torvald's success in building such a wide developer community to the Internet, without which Linux (and similar projects) would not have progressed as quickly and as effectively.<sup>11</sup>

This dimension of community development should have particular relevance to ccd.net as it (ccd.net) is essentially a place for building virtual community and its cultures.

### Where's the (community) art in all this?

At a ccd.net steering committee teleconference in February 2003, the issue of "where's the art in all this" was raised. Fair enough: Open Source belongs traditionally to the world of software engineering and is (at least for now) alien to CCD.

One response to this question is to point out that not every activity in CCD involves making art (and not everything we do is artful, even if we are trying to make art). CCD involves administrative, organising, promotional, and other work. These days, we use more and more new technologies to do our work (artistic or otherwise); we use software for producing, retrieving, and distributing information; for keeping track of our finances; for communicating with our peers.

The CCD community should thus seriously consider using Open Source tools for doing such varied tasks. Not only are these tools technically superior and financially effective, but the software is produced in a community-oriented processes, some of which are very similar to CCD

<sup>9</sup>Ashton Mills, ed. The Advanced Linux Pocketbook, ACP Tech, Sydney, 2001.

<sup>10</sup>Raymond's "The Cathedral and the Bazaar" discusses these two different models and other aspects of Open Source in detail. This essay and his other writings are available in book form, or online at <http://www.catb.org/~esr/writings/>.

<sup>11</sup>Raymond discusses this point and others in his article "The Social Context of Open Source Software", at <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s11.html>.

processes [more on this later]. Using Open Source would then be a matter of consistency with and loyalty to community development principles as well.

Another possible response is that there is actually art in software engineering. Writing code is like writing poetry because it demands compression, indirection, and other elements of poetics. When writing code collaboratively, community writing or community art methods can come in handy too. Collaboratively developing web sites with "art" (images, sound, etc.) makes the dimension of "art" and community art more obvious in what is basically software development.

Finally, there is the response that Open Source can actually inform artmaking directly. Saul Albert explores this issue in his essay "Open Source Tactics for Collective Arts Practice."<sup>12</sup> Albert examines Raymond's "musings" about Open Source and attempts to apply Open Source concepts and practices to artmaking (collective and otherwise).

His discussion proceeds from Raymond's key concept that Open Source software is developed in the context of a gift culture or gift economy, where programmers create software to give away ("gifts") to increase one's prestige among one's peers.<sup>13</sup> Albert sees collective art practices in a similar context:

Within "Open Source" collective art practices, which function in a gift economy, collective projects reach fruition by distributed effort, and are circulated and promoted and most importantly evaluated within the collective. This independence from traditional forms of validation means that the user/authors involved are not wholly reliant on commissions and residencies for the satisfactory development and success of their projects. These projects can take place within a community that is focused on the artwork.<sup>14</sup>

Albert admits, however, that the relationship of Open Source and collective art have not been examined rigorously in his essay. A more rigorous examination of this relationship is beyond the scope of this article (and this writer); this is, however, perhaps one area of investigation that ccd.net can pursue.

## **Open Source Possibilities for ccd.net and CCD**

In a [small] nutshell then, Open Source software is free (as in liberty) and free (as in free beer); it is also highly reliable and secure, and evolves very quickly. It is based on a world-wide community of programmers, users, and support mechanisms linked largely by the Net. It is a viable and often superior alternative to proprietary software. It can help create art and can be created artfully. It is steadily gaining market share, both on servers and desktops in government and in other sectors.

Hopefully, it will grow steadily in the CCD sector in Australia as well. Here are possibilities for growth:

### *Adoption of Open Source Software*

CCD practitioners and organisations can look at Open Source alternatives to the proprietary software they now use. Here is a short list of operating systems and applications and their possible alternatives:

---

<sup>12</sup><http://twentiethcentury.com/saul/os.htm#1a>.

<sup>13</sup>See Raymond's "The Hacker Milieu as Gift Culture" in "Homesteading the Noosphere" at <http://www.catb.org/~esr/writings/cathedral-bazaar/homesteading/ar01s06.html>

<sup>14</sup><http://twentiethcentury.com/saul/os.htm#1a>.

Microsoft Windows	Linux, e.g. Red Hat ( <a href="http://www.redhat.com">www.redhat.com</a> ), Debian ( <a href="http://www.debian.org">www.debian.org</a> ), Mandrake ( <a href="http://www.mandrakelinux.com">www.mandrakelinux.com</a> ), SUSE ( <a href="http://www.suse.com">www.suse.com</a> ).
Apple MacOS	Linux, e.g. Yellow Dog ( <a href="http://www.yellowdoglinux.com">www.yellowdoglinux.com</a> ), Debian, Mandrake, SUSE
Microsoft Internet Explorer and Netscape Navigator	Mozilla ( <a href="http://www.mozilla.org">www.mozilla.org</a> )
Microsoft Outlook Express, Eudora	Ximian Evolution ( <a href="http://www.ximian.com">www.ximian.com</a> )
Microsoft Office	OpenOffice.org ( <a href="http://www.openoffice.org">www.openoffice.org</a> )
Adobe Photoshop	The GIMP ( <a href="http://www.gimp.org">www.gimp.org</a> )

There are many other software tools that are in use in CCD work, no doubt. Chances are there will be Open Source applications that are available to complement or replace proprietary tools.<sup>15</sup>

There have been concerns about Linux and other Open Source software not being too user-friendly, or that changing software that one is accustomed to will introduce unnecessary disruptions. These are legitimate concerns, and there are many more issues that need to be addressed if Open Source is to be more widely adopted.

Rather than detail and discuss these issues here, it may be better left to [ccd.net](http://ccd.net) perhaps to provide a forum for such discussions. The forum can be online through the Forums on the web site and through an email list. Maybe offline, face-to-face forums such as seminars and workshops will be useful too.

### *Undertake Open Source Software Projects*

This article is partly meant to prepare the groundwork for an Open Source web application development project planned by [ccd.net](http://ccd.net). The basic aim is to use Open Source ways to build an online, web-based application that can be integrated into the existing web site.

Some initial ideas include building an interactive gallery. The specific project will be determined based on more consultation and discussion with the [ccd.net](http://ccd.net) community. Or it may start with an individual programmer or developer's desire to scratch a personal itch in true hacker tradition.<sup>16</sup> The exact nature of the project, and the participants in it, will hopefully be determined through discussions through [ccd.net](http://ccd.net) forums.

Also, developing Open Source projects does not need to be limited to the [ccd.net](http://ccd.net) website or to creating work for the Web. Open Source, after all, was around before the web and the Internet (and was responsible to a large degree for the invention and development of the web and the Internet). Maybe an application or applications can be developed that address CCD workers' specific needs using Open Source methods.

### *Open Source Alliances*

To develop software for [ccd.net](http://ccd.net) and for other CCD purposes will require, however, competencies – there will need to be programmers, testers, documentors, and other workers collaborating on projects. Some of these competencies can be found in the CCD field; some need to be developed. We therefore need to build links with others who are interested in collaborating on Open Source projects.

---

<sup>15</sup>This article was written using OpenOffice.org Writer, running on Yellow Dog Linux, on an old Apple PowerBook, along with lots of coffee.

<sup>16</sup>Eric Raymond observes that "Every good work of software starts by scratching a developer's personal itch." See <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s02.html>.

There are other community organisations and individuals who use and advocate for Open Source; these include Linux Users Groups, community activists, and some NGOs.<sup>17</sup> We need to find these individuals and groups and build relationships that can assist in developing software tools that are commonly needed; more important, however, is building links to share ideas and resources that will advance our common community development goals.<sup>18</sup>

*Where's the art (again)?*

Finally, as CCD is concerned with art, the possible connections between Open Source and CCD (or art making) explored initially by Saul Albert (and possibly others) will need to be explored further. While CCD is not purely "collective art practice" as described by Albert, a lot of CCD is; some Open Source principles and practices may be relevant to CCD, some may not be. More exploration is needed to find the relevant bits.

## **Conclusion**

This article sought to explore Open Source and its possible relevance to community cultural development and to ccd.net. It is hoped that dialog on issues raised or merely hinted at will continue on ccd.net and elsewhere.

---

<sup>17</sup>See <http://www.linux.org.au> for information on Linux groups in Australia, and <http://www.cat.org.au> about community activism and new technologies. [Communityweb@cat.org.au](mailto:Communityweb@cat.org.au) is a mailing list about community and new technologies which includes representatives from Greenpeace, Wilderness Society, Doctors without Frontiers, and others.

<sup>18</sup>Communit (<http://www.communit.info>) will host (at time of this article's writing) the "Community Information and Communications Technology (ICT) conference" in Adelaide in May 2004. This conference will be an excellent opportunity to find potential allies. Communit itself may be a good prospect as long-term ally.