Drafting the Gift Domain

version: 21 March 2004

For more information, or for the latest version of this work, go to:
http://www.greglondon.com

This document was created using OpenOffice version 1.1.0
(which exports directly to PDF)
http://www.openoffice.org

running RedHat Linux
http://www.redhat.com

on a Penguin x86 PC.
http://www.penguincomputing.com/

Table of Contents

1 An Introduction

If you obtained this document through the Internet, chances are good that Open Source software helped get it to you. The Internet contains millions of PC's running Linux, Apache, CGI, and Perl, which are all Open Source programs that serve web pages to the world. Open Source software, at its core, is the Gift of millions of programmers driven by their personal wish to make the world a better place.

People are now taking the Open Source concept and applying it to music, photos, video, text, and other non-software works, and calling it Open Content. This document is Open Content.

How does Open Source and its younger sibling Open Content work? How do you take the ideas of Open Content and apply it to a new project idea? How can you use Open Content to make the world a better place? These are the types of questions I hope to answer in this document.

1.1 Design versus Management

I'm not going to go into the nitty-gritty of Open Content management. Questions about version control, change requests, contributor communications, and similar management-type questions can find answers in "The Mythical Man-Month" by Frederick P. Brooks and "The Cathedral and the Bazaar" by Eric S. Raymond.

http://en.wikipedia.org/wiki/Mythical_Man_Month

http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/

These two works contain valuable lessons on the management of Software and Open Source projects. They should also be read by anyone wishing to create and manage an Open Content project that is NOT software based. Many software lessons transfer directly to works in other mediums.

Instead of Open Content 'management', this document attempts to model and explain Open Content 'design'. "How does Open Content work?" is different than "How do I manage an Open Content project on a day to day basis?"

1.2 Open Content Design

Understanding the principles of Open Content design will allow you to create a new Open Content project that has a chance of succeeding. If you want to create a new Open Content project around music or video, this is a good place to start.

Understanding Open Content design should also allow you to separate Open Content projects from Closed (Proprietary) projects. Some people want to drape themselves in the language of Open Content but behave like a Proprietary project.

This brings us to the first problem in understanding Open Content design: language. There are a lot of things written about Open Content, and a lot of people use different language to describe the same things. Worse, some people use the same language to describe different things. Open Content is also called "Free" and "Public" depending on who you ask.

So, how do you separate what people SAY versus what they DO? How do you separate rhetoric from fact? Fortunately, all Open Content projects can be boiled down to how it relates to the LAW. While law isn't perfect or totally unambiguous or future-proof against change, it at least gives us an objective point to measure what is "Open" and what is "Closed". Law is never written in the passive voice; it is always written in the active voice. Law does not speak about what something IS except by what DOES.

Most Open Content projects can be measured by how it relates to Copyright (and sometimes Patent) law. Copyright and Patent law are part of a larger system of laws called "Intellectual Property Laws".

Which brings me to another problem of language: rhetoric. A LOT of what's written about Open Content (both for and against it) is laden with rhetoric. Open Content proponents have called the Closed/Proprietary model evil, outdated, and unreliable. While Proprietary players have called Open Content viral, a cancer, and unreliable.

Despite the rhetoric some Open Content proponents say that Open Content is about breaking down the old Intellectual Property laws, all current successful Open Content projects work WITHIN current Intellectual Property law, not OUTSIDE it.

2 Intellectual Property Law

If you live in the United States, the majority of Intellectual Property law is based in the Constitution, Section 8, which states that "Congress shall have the power ... To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries; ..."

2.1 Writings and Discoveries

A key phrase is "Writings and Discoveries". The constitution does not say "Ideas and Knowledge" or "Facts and Information".

A Writing is an original work of authorship in fixed form. It is an artistic expression someone created and wrote down, recorded, or put into some fixed media. Writings include literary, dramatic, musical, and artistic works.

A Discovery is a "new", "useful", "non obvious" Invention (a Device or a Method of Manufacture). A Discovery includes a specific mousetrap design and a particular method for manufacturing steel.

Writings and Discoveries do not include ideas, knowledge, facts, and information. Sherlock Holmes-The Red Headed League is a Writing. The idea of mystery novel is not. Wagner's "Flight of the Valkyries" is a Writing. A new, original tune stuck in your head but not written down or recorded in some fixed form is not. A chemical formula that relieves pain is a Discovery. A chemical formula with no known use is not. A new method for forging stronger steel is a Discovery. A list of different types of steel and their strengths is not.

Congress grants Authors certain exclusive rights to their Writings through Copyright Law.

Congress grants Inventors certain exclusive rights to their Inventions through Patent Law.

2.2 Copyright Law

Copyright grants the Author of a specific expression, for a fixed duration, the exclusive right to Copy, Distribute, and create Derived works of that expression. Some types of works also grant the Author the exclusive right to Perform and Display the work publicly.

The rights granted exclusively to the Author does NOT include the right of Fair Use of the work. Fair Use allows anyone to copy, distribute, and create derived works of the original work under certain, limited, circumstances. Using a VCR to record a television program is a famous example of Fair Use copying.

Copyright does not grant the Author exclusive rights on how the work may be read (for a book), or played within the home (for music or video), or discussed, analyzed, criticized, or understood.

An Author holding Copyright to a story about a boy meeting a girl, falling in love, and living happily every after, does not have exclusive rights to all stories of that nature, only the Author's specific story, the Author's specific expression in fixed form.

The Author is granted these exclusive rights for the life of the Author plus 70 years after his or her death for non-commercial Authors. If it is a work-made-for-hire, these rights are granted to the Employer (not the writer/artist/musician/employee) for 90 years from first Publication or 120 years from date of Creation, whichever occurs first. (Historical note: The first U.S. copyright law, written in 1790, gave Authors their exclusive rights for 28 years.)

While the Author has been granted these exclusive rights, he or she may control who else may exercise these rights through a License.

After the exclusive rights granted to the Author expire, those rights go into the Public Domain, and anyone may Copy, Distribute, create Derived works, Perform, and Display the work. The author's heirs no longer have control over the work once it passes into Public Domain.

2.2.1 Securing Copyright for the Author

An Author is granted Copyright privileges to his or her work as soon as the work is in fixed form, written on paper, recorded on audio tape, or recorded on video tape, etc. The Author does not have to register the work with the Copyright Office for these rights. The Author does not have to place a Copyright Notice on the work to gain these rights. The exclusive rights of Copyright are granted to the Author immediately after the work is in fixed form.

Copyright Registration is only required if the Author wishes to bring a copyright infringement suit against someone. The Registration process is straightforward, requiring the Author to fill out a one-page form, include one or two copies of the work, include a small registration fee (~ $30), and send it all into the Copyright Office.

Copyright Notice is not required to secure Copyright for the Author. However a lack of Copyright Notice means that a Copyright Infringer can claim innocent infringement because they didn't know the work was protected. A Copyright notice contains three simple parts:

1. The © symbol or the word Copyright (The letter 'C' in parenthesis should NOT be used)
2. The year of first publication
3. The name of the Copyright holder

For example:

Copyright 2002 John Smith
© 2003 John Doe

2.2.1.1 All Rights Reserved

The phrase "All Rights Reserved" was a required part of a Copyright Notice for members of the 1911 Buenos Aires Convention on Literary and Artistic Copyrights treaty. Aurhor's could not secure their rights in other countries unless "All Rights Reserved" was part of the Copyright Notice. This treaty has been replaced by more recent treaties, and the "All Rights Reserved" phrase is no longer a requirement to secure an Author's rights.

See section 3.10 of this document:
http://www.greglondon.com/dtgd/carroll_copyright_faq.txt

2.2.2 International Copyright

Copyright is an established international concept. The Berne convention (Paris, 1971) is an international copyright treaty signed by 96 countries, including the United States. Each country's legal system must provide protection to the author with regard to the rights of the treaty. Some of this document may be relevant to the United States only. Even though all these countries signed the treaty, each country can enact its own laws.

2.3 Patent Law

Patent law grants the Inventor of a specific Invention, for a fixed duration, the exclusive right to Manufacture, Use, and Sell an Invention.

To be granted a Patent, the Invention is supposed to be Novel, Useful, and Not of an Obvious Nature.

The Inventor is granted these exclusive rights for 20 years from the date they first filed for a patent. Patent rights are not automatic like Copyright. An Inventor has to file for a patent on their Invention, within a year from when the Invention is known, used, or offered for sale, and the Patent Office may grant or reject the patent application.

When the Inventor has been granted these exclusive rights, they may control who may exercise these rights through a License.

After the exclusive rights granted to the Inventor expire, those rights go into the Public Domain, and anyone may Manufacture, Use, and Sell the invention.

2.3.1 Applying for a Patent

The costs associated with a patent are exorbitant when compared to copyright registration. The basic filing fee for a utility patent is $740. A utility patent issue fee is $1200. Patent maintenance fees need to be paid to maintain the patent rights, and their schedule is: $880 due at 3.5 years, $2020 due at 7.5 years, $3100 due at 11.5 years. On top of this, throw in the inventor's attorney's fees, in the neighborhood of $200 an hour. And after the application is filed, there's a waiting period for it to get processed, and there is no guarantee that the application will be approved.

Applying for a patent requires that the Inventor completely describe the invention such that someone working in the same field could recreate the invention. Until the patent is granted to the Inventor, the application remains confidential. If the patent is granted to the Inventor, the application, and the complete description of the Invention, become Public Record.

In exchange for being granted Patent protection on a new Invention, the Inventor must completely disclose the workings of the Invention. There are no secret Patents. (There are, however, Trade Secrets.)

2.3.2 International Patent

A US patent does not provide protection outside the country.

2.3.3 Bad Patents

"It was never the object of patent laws to grant a monopoly for every trifling device, every shadow of a shade of an idea, which would naturally and spontaneously occur to any skilled mechanic or operator in the ordinary progress of manufactures. Such an indiscriminate creation of exclusive privileges tends rather to obstruct than to stimulate invention. It creates a class of speculative schemers who make it their business to watch the advancing wave of improvement, and gather its foam in the form of patented monopolies, which enable them to lay a heavy tax on the industry of the country, without contributing anything to the real advancement of the arts. It embarrasses the honest pursuit of business with fears and apprehensions of unknown liability lawsuits and vexatious accounting for profits made in good faith."

--U.S. Supreme Court, Atlantic Works vs. Brady, 1882

Sometimes, bad patents happen, even today.

2.4 Enforcing Copyright and Patent rights

Copyright, Patent, Trade Secret, and Trademark law are all Civil (tort) laws rather than Criminal laws.

A Criminal case requires proof "beyond a reasonable doubt" to convict, which is described as "99% certainty". A Civil case requires a "preponderance" or evidence, which is described as "greater than 50% certainty".

Punishment in a Criminal case can result in prison time and fines paid to the government. Civil cases can result in a defendant paying the plaintiff Compensatory Damages. Compensatory Damages should equal the Plaintiff's losses that were a result of the defendant's actions.

In some cases, a defendant in a Civil case may have to pay Punitive damages. Punitive damages can exceed whatever losses the Plaintiff may have suffered as a result of the Defendant's actions. Punitive damages are meant to punish a guilty defendant, deter the defendant from doing it again, and deter others from doing the same.

For Punitive damages to be awarded, the defendant's actions must be egregarious (outrageously bad), and the defendant must have intended to cause harm, or have shown a willful disregard for the rights of others, or exhibited gross negligence (a conscious indifference to the rights of others).

When a Copyright or Patent holder believes someone has Infringed on their rights, their first recourse is to sue the alleged infringer for damages.

Title 17 of the United States Code, Section 506, lists a number of criminal offenses, including infringment for commercial advantage or personal financial gain, fraudulent copyright notices, and fraudulent removal of copyright notices.

http://www.bitlaw.com/source/17usc/506.html

2.5 A Right Becomes Property

Copyright and Patent law convert Rights into Property. By granting monopolies to Authors and Inventors, an Author owns the rights of Copyright and an Inventor owns the rights of Patent.

Authors and Inventors can create Writings and Inventions and, by way of their monopoly, can sell them back to the public at a price that will pay for the cost of Writing and Inventing plus give them some level of profit.

Artistic and Scientific research then become subject to the laws of supply and demand. Good Writings and Inventions are rewarded with commercial success. The public gladly pays for a Writing or Invention that they like. Bad Writings and Inventions are weeded out by commercial failure. The public won't pay for a bad idea. The rights to the work eventually return to the Public Domain. And new Writers and Inventors can build upon these Public Domain works to create the next generation of Writings and Discoveries.

The Public surrenders Copyright and Patent Rights for a limited time on the notion that it will improve the Arts and Sciences overall for the Public as a whole, which is summarized in the Constitution:

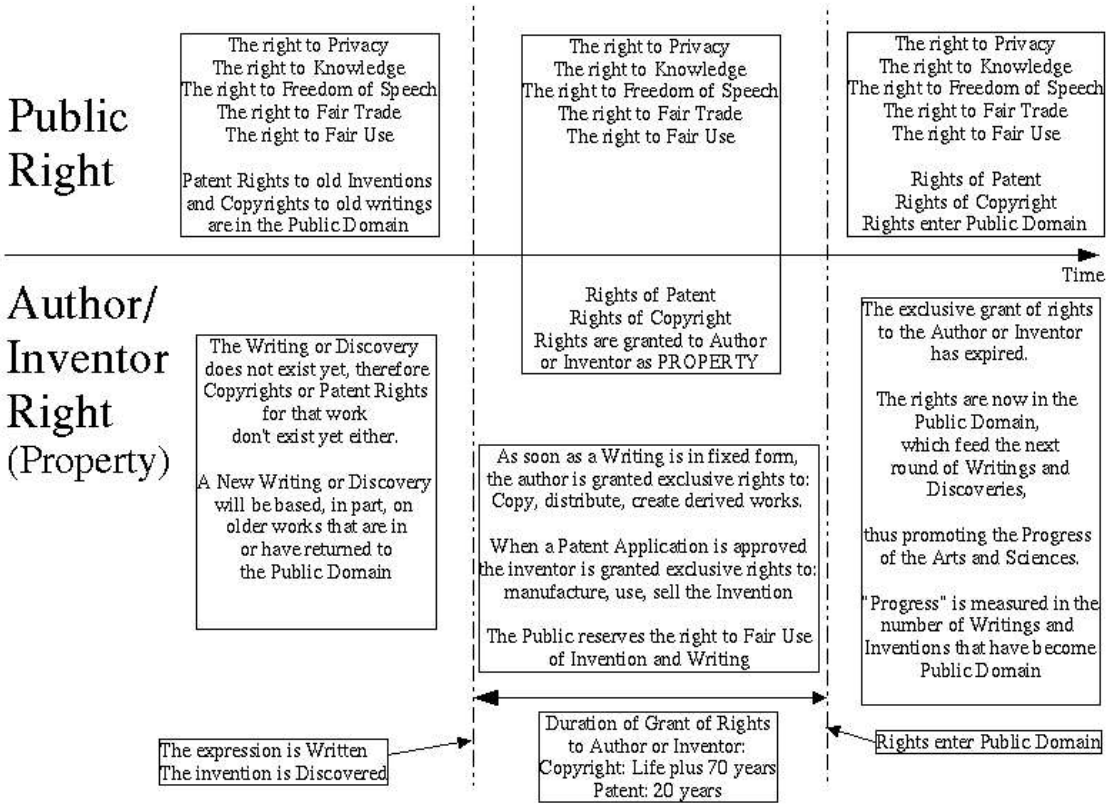
"Congress shall have the power ... To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries; ..."

Despite any rhetoric you might hear to to contrary, I would describe the basic idea behind Copyrights and Patents as brilliant. The government spends very little tax dollars to maintain the Copyright Office and Patent Office compared to the huge amounts that private Inventors and Authors gamble on their research becoming something good enough to be profitable. If an invention or writing turns out to be of no value, the individual researcher loses their money rather than public tax dollars going down the drain. Private Risk generates Public Good.

This assumes that the balance between what the Public surrenders in a temporary monopoly to the Inventors and Authors is offset by the overall Public Good generated by that surrender of rights.

The debate for most is not around the concept of Intellectual Works as PROPERTY, the debate is around maintaining the BALANCE of Intellectual Property (monopoly) versus Public Good.

The diagram below shows the time line of Intellectual Property from before the work exists, to the period the Rights are treated as Property, to the point the Rights enter the Public Domain.



"Rights as Property"

2.6 Licenses

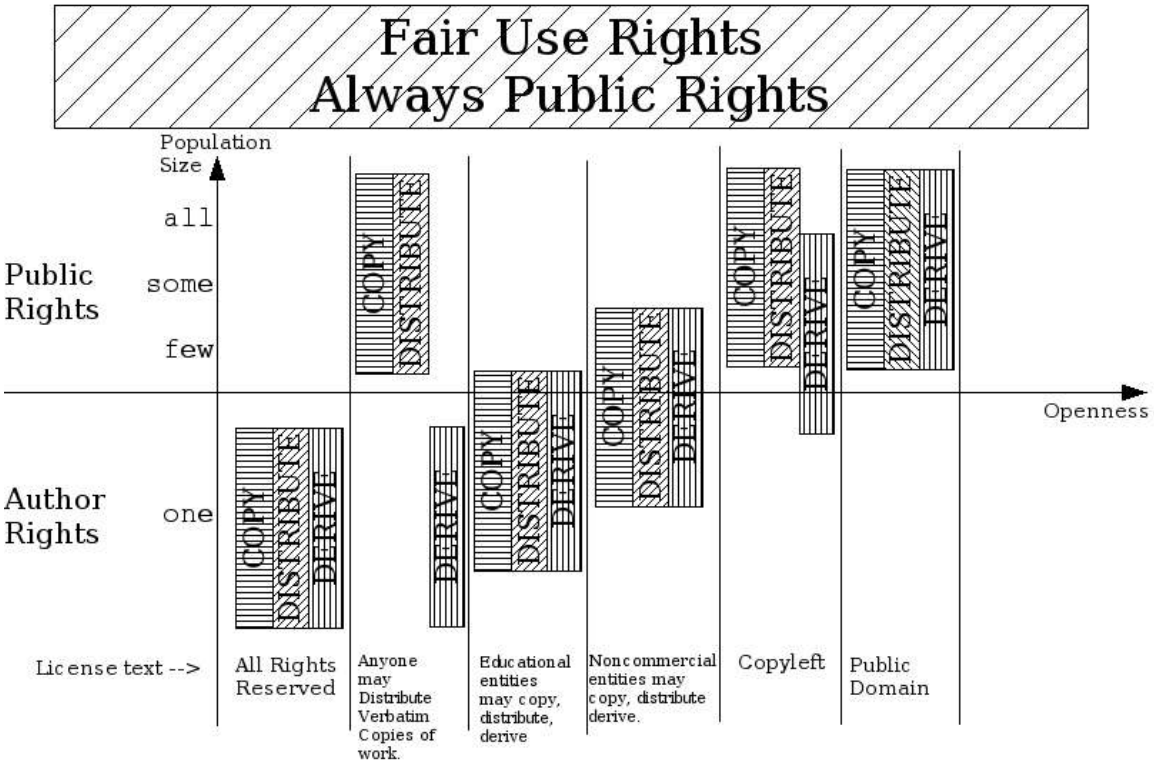
During the time that Authors and Inventors have exclusive Copyrights or Patent Rights to their works, they may grant some, none, or all of those rights to some, none, or all members of the Public. This grant is called a "License". An Author and Inventor may only License those rights which have been exclusively granted to them by Copyright and Patent law. Any other rights regarding the work, including Fair Use Rights, are Public, and cannot be controlled or licensed by the Author or Inventor.

One common license is "All Rights Reserved". This license indicates that the Author or Inventor is retaining all rights for themselves. It doesn't mean ALL RIGHTS, as in every right you can think of; it means "all rights granted to the Author or Inventor by Copyright or Patent law". If it is a Writing, no one but the author may Copy, Distribute, or Create Derived Versions of that Writing. If it is an Invention, no one but the Inventor may Use, Manufacture, or Sell that Invention.

An Author may license their work "All Rights Reserved" and then make money on their Writing by selling copies of it to the Public. This Author may also make money by selling the right to Create a Derived Version of the Writing, for example, by selling the right to convert their book into a movie to a studio.

"Open" or "Public" licenses, as used by Open Source and similar projects, instead of retaining all the rights for the original Author, grant most of the rights to a large part of the Public.

The diagram below shows some fairly common licenses, indicating visually for each license which rights are being licensed to how many people. Only Copyright licenses are shown, since Patent Licenses are usually not part of an Open Content project.



"Basic Licenses"

2.7 Copyright, Functionality and Software

Until relatively recently, the notion of Writing and Inventions were two completely separate worlds. A Patent is granted for an Invention based on its "usefulness", on it providing some new "functionality" that never existed before. A Copyright is granted on the "Creative Expression" of the author.

You can copyright a particular mystery novel you've written, but you cannot copyright the idea of "mystery novel" and monopolize the entire genre. This would give an Author far too much monopoly over the Art of Writing.

You can patent the functionality of an Invention, but only if the Invention is Novel, Useful, and Not of an Obvious Nature.

You cannot Patent a scientific truth or a mathematical expression of it. Newton's second law of motion, that the acceleration of an object is a function of the force applied to it and the mass of the object, cannot be patented. Nor could anyone patent the mathematical expression of that truth:

$$\mathcal{F} = m * \mathcal{A}$$

For most of US history, Writings (Huckleberry Finn), Inventions (the telephone), and scientific truths (E=mc**2) were distinctly separate ideas.

This distinction collapsed with the advent of computers.

Computer Software is a Writing that a computer can 'read' and 'execute' to perform some Function. Because software can provide functionality when it is executed by computer hardware, some argue that software should be eligible for Patent protection.

Software Patents are a relatively new twist to IP law. For quite a few years, the Patent Office would not grant Software Patents. The first software patent was only granted in 1981. That is barely 20 years ago on the scale of approximately 200 years of Copyright and Patent law.

This is a profound collapsing of concepts. Previously, you could never Patent a Writing, and you could never Copyright an Invention. The Copyright and Patent rights granted and the duration of those rights were balanced for a world were Writings and Inventions were wholly separate distinctions. With the advent of software, a Writing can now be Copyright AND Patented.

2.7.1 Software is a Writing

Software is a Writing. By itself, software is no different than a recipe book that says "if you mix this and that, and then cook it for so long, you'll get a nice batch of cookies". A software "recipe" might say something like "if you add all these deposits and then subtract all these expenses, you'll get a balance sheet for your checking account".

2.7.2 Transmitting Software is Speech

When Grandma sends you her recipe for those awesome-choco-chunk-chip cookies she makes, that's speech of the kind protected by the First Amendment. When someone contributes their software to an Open Source project and posts it on a website that transmits it to the planet, that's Free Speech.

2.7.3 Executing Software is Functional

Someday we might have kitchens that take "recipe cards" and do all the work to execute the recipe and make you dinner. Computers are equivalent to this futuristic kitchen. They take a Writing in the form of software and provide some Functionality based on that software.

Executing some software could be illegal because of the behavior it generates. You might write a piece of software that would allow you to gain access into a bank mainframe computer. The program by itself should be legal, protected by the Right to Knowledge. Transmitting the program to others should also be legal, protected by the Right to Free Speech. But if you execute the program, gain unauthorized access to a bank's computer, and add a few zeroes onto your account balance, then that should be illegal.

(Note the use of the word "should" in the above paragraph. It says "should" as in "that's the way I think the law should be written". Whether or not it is written that way is a different matter altogether. Politicians write all sorts of laws that end up getting struck down later as unconstitutional. In the mean time, you could end up spending some Hard-Time, so check with a lawyer for your specific situation. I am not a lawyer. This is not legal advice. This is not a substitute for legal advice.)

2.7.4 Software and Recipes

A recipe is actually a very good analogy for software because they both have similar idiosyncrasies.

You cannot Copyright a recipe for chocolate chip cookies. A recipe by itself is considered "functional" or "utilitarian" and not an "artistic expression".

Though you can't Copyright a recipe, you can Copyright a whole bunch of recipes that also include an Author's creative expression added into it. Most cookbooks are one-part recipe and two-parts talking about the recipes.

Software relates to Copyright similar to the way recipes do. You cannot Copyright the functionality expressed within a program. If you wrote the first program that added up income and expenses, you would not be able to Copyright the functionality of all bookkeeping programs, you would only have a Copyright on your specific expression of a program.

Software is part functionality and part expression. There are many ways to get the average of a list of numbers in software, even though functionally all you're doing is adding up the numbers and dividing by the quantity of numbers. You can code this in different looping structures, for-loops, while-loops. You can use an array of the numbers and index into the array. You could have a linked list of numbers. You could use gotos. You could unroll the loop and hardcode the equation. The expression can receive Copyright protection, while the functionality cannot.

2.7.5 Software Patents and Open Source

Whether or not Software should be Patentable, Software Patents are a legal fact today. They are also a concern for Open Source projects.

If someone writes a bookkeeping program, licenses it "All Rights Reserved", and sells copies of it to the Public, that is not a problem for an Open Source project. An Open Source project could create the same functionality of that bookkeeping program using the expression of programmers willing to license their work "Open Source".

Software Patents are problematic for Open Source projects. If a Software Patent is awarded for some functionality within a program, that functionality is the exclusive monopoly of the Patent holder. An Open Source project cannot recreate their own expression that implements the same functionality without the Patent-holder's license.

This creates a huge incentive for software companies to push Software Patent Applications, hoping to be granted a Patent in an effort to exclude all other Software vendors on that functionality. The problem then becomes that the focus is on achieving a monopoly of any kind, whether or not the functionality adds enough value to the Public Good to justify a 20 year monopoly.

And some companies have recognized Software Patents as a means to lock out Open Source projects from the competition.

2.7.6 The Halloween Documents

The Halloween Documents were some internal memos in Microsoft Corporation that were leaked and published on the web on 1 Nov 1998 (the day after Halloween, hence the name).

You can get the complete story here:
http://www.opensource.org/halloween/
http://www.opensource.org/halloween/faq.php

The short of it is that in the memo, Microsoft identified Linux as a major threat to Microsoft's dominance of the software industry, It then went on to propose a strategy to counter Linux's success:

"De-commoditize protocols & applications"

A "commodity" is basically something that is Public Domain. A mousetrap is a commodity. Anyone can build a mousetrap and sell them because there are hundreds of patented mousetrap designs that have become Public Domain. That means everyone who makes and sells mousetraps has pressure to lower prices, improve quality, and develop a name brand of consumer loyalty if they want to survive.

"Protocols" refer to the way data is exchanged, the way computers and programs talk to one another. The World Wide Web is based on a commodity protocol that anyone can use in their web browser.

"Applications" refer to Word Processors, Web Browsers, Email Tools, and any other program you might use on your computer.

The strategy proposed in the Microsoft Memo basically says that Microsoft can impede Linux's success by using changing protocols and applications so that they can be patented, trade secrets, or otherwise lock Linux out of the functionality.

The memo says nothing about these changes being made to the protocols or applications for the Public Good. They are not new innovations truly deserving patent protection because they are a radical improvement in what is currently available to the public. They simply want to lock Linux out of the market.

This is the downside to the market-driven research mode created by Copyright and Patent law. Patent law especially creates such a severe monopoly that researchers have incentive to apply for patents that don't really deserve to be granted, but sometimes do.

Sometimes, bad patents happen. It doesn't reflect a problem with the idea of Intellectual works as Property, it reflects the pressure from profit driven Authors and Inventors pushing Copyright and Patent law beyond the Limit of Public Good.

2.7.7 Bad Software Patents

Some bad software patents have been awarded. A patent is supposed to be Novel, Useful, and Not of an Obvious Nature. And a lot of Software Patents were Novel and Useful, but glaringly Obvious.

This has put a really bad taste in the mouths of a lot of people working in the software and computer field. Some people argue that Software Patents should not be granted at all. And some say that patents should at least be excluded from software that runs on a general purpose computer.

http://www.gnu.org/philosophy/patent-reform-is-not-enough.html

2.8 The Limit of Pubic Good

There is a limit to the Public Good from Copyright and Patent law.

The right to Copy, Distribute, and create Derived works of a specific expression is a minor right on the scale of all human rights. Granting an exclusive monopoly of these rights to an Author for roughly a century is still minor in the grand scheme of things. Copyright cannot suppress ideas or dissent or knowledge or religious beliefs. Copyright grants an Author an exclusive monopoly on the Author's particular expression of an idea, but it cannot suppress the idea itself.

The right to Use, Manufacture, or Sell a specific invention has somewhat more of an impact than Copyrights do. A Patent grants the Inventor an exclusive monopoly on the Functionality that the Invention is. A Patented invention has a monopoly on its functionality and someone can avoid the patent restrictions only if they can create that same functionality through a completely new Invention. A Patent grants a much more powerful monopoly than Copyright does, therefore a patent must be reviewed before it it granted, and the patent rights are granted to the Inventor for a much shorter time of 20 years.

Copyright is a relatively minor set of rights for the Public to surrender, and so the duration can be longer than Patent durations.

Patent Rights are a more significant set of rights for the Public to Surrender, and so the duration of Patent Rights is much shorter than Copyrights.

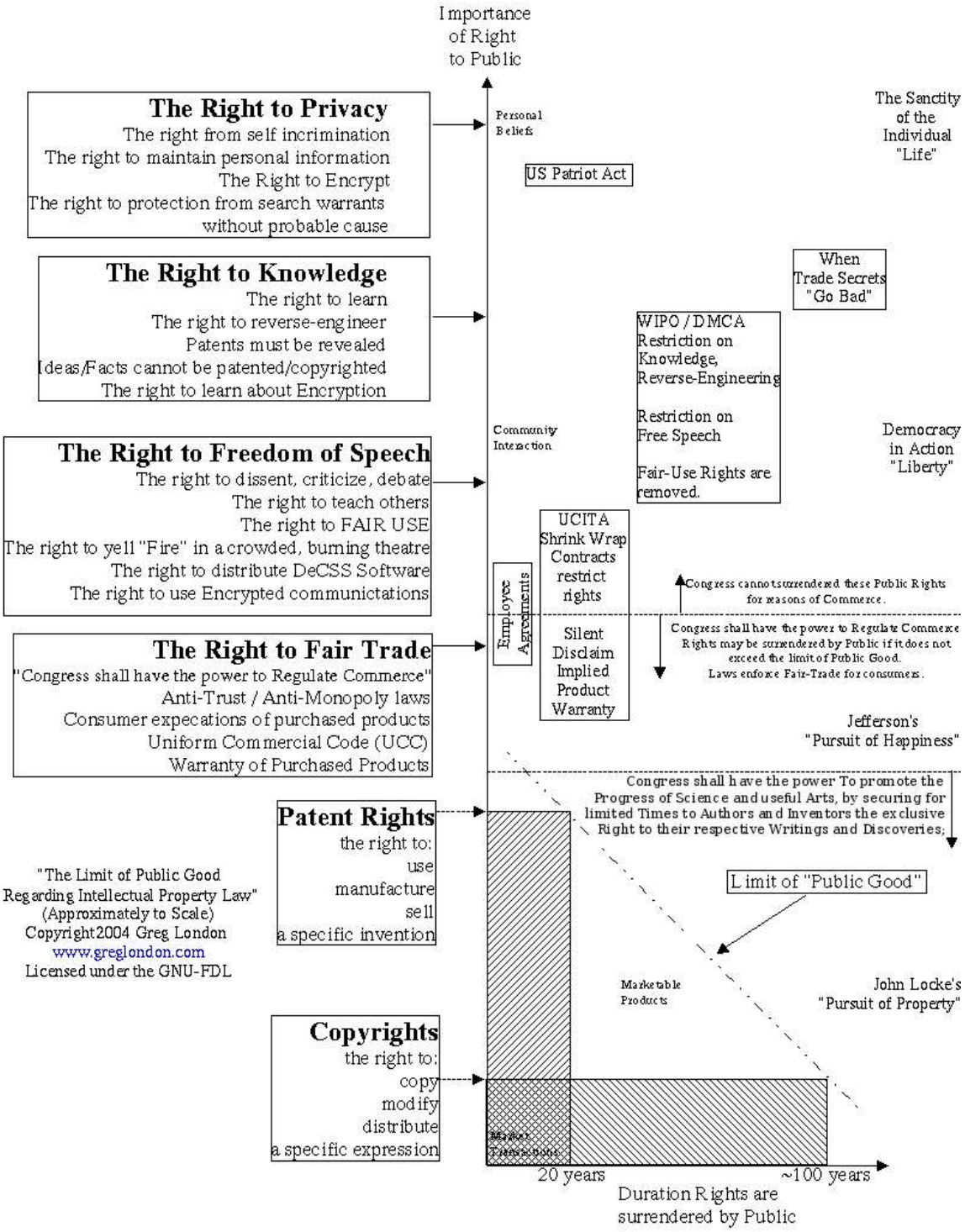
This "Limit of Public Good" can be represented graphically as shown below. The horizontal axis indicates the duration the Rights are granted exclusively to the Author or Inventor. The vertical axis indicates how "important" the Right is to the Public.

The area boxed in by "Patent Rights" vertically and "20 years" horizontally indicates the "cost" suffered by the Public in surrendering the Rights of Patent exclusively to the Inventor. This cost should roughly match the "benefit" the Public receives by creating market-driven research and Inventions that eventually return to the Public Domain.

Were Patents to grant the Inventor more important rights than it does, or were Patents to grant the Rights it does for a longer duration, the cost to the Public would be greater than the benefit to the Public. This indicates the "Limit of Public Good".

The same notion works for Copyright. The area boxed in by "Copyrights" vertically and "100 years" horizontally represents the cost suffered by the Public in surrendering Copyrights exclusively to the Author.

The "Limit of Public Good Diagram" shows Rights beyond Patent and Copyright. Fair Trade, Free Speech, Right to Knowledge, Right to Privacy are shown to put Patent and Copyright into perspective.


Importance of Right to Public
The Right to Privacy
The right from self incrimination
The right to maintain personal information
The Right to Encrypt
The right to protection from search warrants without probable cause
Personal Beliefs
US Patriot Act
The Sanctity of the Individual "Life"
The Right to Knowledge
The right to learn
The right to reverse-engineer
Patents must be revealed
Ideas/Facts cannot be patented/copyrighted
The right to learn about Encryption
When Trade Secrets "Go Bad"
WIPO / DMCA Restriction on Knowledge, Reverse-Engineering
Restriction on Free Speech
Fair-Use Rights are removed.
Community Interaction
Democracy in Action "Liberty"
The Right to Freedom of Speech
The right to dissent, criticize, debate
The right to teach others
The right to FAIR USE
The right to yell "Fire" in a crowded, burning theatre
The right to distribute DeCSS Software
The right to use Encrypted communictations
UCITA Shrink Wrap Contracts restrict rights
Employee Agreements
Silent Disclaim Implied Product Warranty
Congress cannot surrendered these Public Rights for reasons of Commerce.
Congress shall have the power to Regulate Commerce
Rights may be surrendered by Public if it does not exceed the limit of Public Good.
Laws enforce Fair-Trade for consumers.
The Right to Fair Trade
"Congress shall have the power to Regulate Commerce"
Anti-Trust / Anti-Monopoly laws
Consumer expecations of purchased products
Uniform Commercial Code (UCC)
Warranty of Purchased Products
Jefferson's "Pursuit of Happiness"
Congress shall have the power To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries;
Patent Rights
the right to:
use
manufacture
sell
a specific invention
Limit of "Public Good"
"The Limit of Public Good Regarding Intellectual Property Law" (Approximately to Scale)
Copyright 2004 Greg London
www.greglondon.com
Licensed under the GNU-FDL
Marketable Products
John Locke's "Pursuit of Property"
Copyrights
the right to:
copy
modify
distribute
a specific expression
Marketable Transactions
20 years
~100 years
Duration Rights are surrendered by Public

2.8.1 Beyond the Public Good

There is a lot of rhetoric around IP law. And some of it mistakenly blames any problems on the idea of treating Intellectual Works as PROPERTY. But the the problem is not Intellectual Property.
The problem is maintaining the BALANCE of Intellectual Property (monopoly) versus the Public Good it creates.

The above diagram shows a number of boxes that exceed the limit of Public Good.

2.8.1.1 Employee Agreements

If you create Intellectual Property works for your employer, you probably had to sign an Employee Agreement signing the Rights to your works over to your Employer. Some Employee agreements have gone further and require employees to agree not to work for a competitor for a period of time if they leave the company. Some Employee agreements claim ownership on anything you create that may be of interest to your employer, even if you did all the work for creating it on your own time and your own equipment.

If you intend to Contribute to an Open Content project of some kind, make sure that any Employee Agreement you signed to get your job doesn't say that your contributions legally belong to your employer. Your Employee Agreement may make it impossible (or very nearly difficult) for you to Contribute to an Open Content project.

2.8.1.2 WIPO/DMCA

The World Intellectual Property Organization (WIPO) refers to a copyright treaty that the United States entered in 1996. The Treaty demands the signatory countries enact laws to enforce treaty requirements. In 1998, the US Congress passed the Digital Millenium Copyright Act (DMCA) as a law to enforce the treaty requirements.

The DMCA makes anti-circumvention a criminal act, under the banner of stopping Copy Infringement. However, the DMCA grants rather broad and vague powers to Copyright holders far above and beyond the Limit of Public Good marked out by the previous 200 years of Copyright law.

The DMCA makes it a criminal activity for anyone to Reverse Engineer any Encryption feature in any company product, even if the product is a Consumer Product available to the Public, without that company's permission. This effectively gives Patent-Like powers to the company by simply inserting any encryption algorithm into their product. If the company withholds permission to Reverse Engineer the encryption, no one may Manufacture, Use, or Sell that encryption algorithm, even though it is not Patented or worthy of Patent protection.

The DMCA intrudes into the rights of Free Speech, scientific research, Fair Use, Fair Trade, Fair Competition, and reverse engineering. DMCA-based lawsuits by Copyright holders have been frequent and some people have been imprisoned for bad reasons.

Imprisoned? Yes. The DMCA grants exclusive rights to Free Speech, scientific research, Fair Use, Fair Trade, Fair Competition, and reverse engineering to a company or individual, without copyright registration or patent application, but with Criminal prosecution to enforce those exclusive rights. The DMCA extends Intellectual Property law far beyond any Limit of Public Good.

http://www.eff.org/IP/DMCA/unintended_consequences.php

2.8.1.3 Free Dimitry

Dimitry Sklyarov was a Russian citizen who worked for a Russian company called Elcomsoft. Russian law required that all software permit the purchaser to make at least one copy of the work. Adobe had an eBook format which prevented ALL copying. Elcomsoft wrote a program that took an Adobe eBook file and converted it to an Adobe PDF format. The PDF format allowed copies to be made and also allowed other uses that some would consider Fair Use.

Elcomsoft's program fell under the DMCA's "anti-circumvention" clause, which made the program illegal by DMCA standards.

Dimitry had the misfortune of taking a business trip to Las Vegas, Nevada, on 16 July, 2001. Adobe complained to the government about Elcomsoft's program, at which point the FBI arrested Dimitry. He was released on $50,000 bail and restricted to stay within California. Adobe soon called for charges to be dropped, but the Federal government pressed onward. In December 2001, Dimitry was permitted to return home to Russia. Charges were then brought against Elcomsoft.

Note a major difference between Civil and Criminal law here. In a Civil case, if Adobe wants to stop pursuing the case, they drop the lawsuit. In a Criminal case, the government pursues whatever case it deems appropriate, whether the "Victim" wants it to do so or not. Even though Adobe wanted the government to drop the case, it was the government's decision to prosecute or not. Also note that Dimitry was released on BAIL because he was brought up on criminal charges. Pushing Intellectual Property law into the realm of Criminal Law is a MAJOR shift.

On 17 December 2002, a jury acquitted Elcomsoft on all counts. Twelve normal citizens decided the DMCA anti-circumvention law had gone too far. And the jury exercised it's power to acquit Elcomsoft. Other jury's may not be so enlightened though. The DMCA still exists and is very problematic.

http://www.eff.org/IP/DMCA/unintended_consequences.php

2.8.1.4 Trade Secrets

A Trade Secret is any information, expression, or invention that a business chooses to keep secret rather than reveal with Copyright Registration or Patent Application. The design of an Invention must be totally revealed to receive a Patent. In exchange for revealing this knowledge, an Invention that is Novel, Useful, and Not of an Obvious Nature can be the exclusive Patent property of the inventor for 20 years.

If the device would not qualify for a Patent, or if the company wishes to keep the design of the device secret, or if the company hopes to maintain the secret longer than the 20 years provided by Patent protection, a company may choose to keep a device as a Trade Secret.

A benign example of a Trade Secret includes the formula for Coca-Cola, which was sold for the first time in 1894, and is still a Trade Secret over a century later. Trade Secrets are more often, but less famously, used for internal information, such as customer lists, product road maps, etc.

Trade Secret laws provide some protection to a company's Trade Secrets as long as the company guards those secrets. Employee's are not allowed to take company Trade Secrets to new employers when they change jobs. Trade Secret laws, however, provide no protection against Reverse Engineering.

A chemist could buy a bunch of Coca-Cola, subject it to chemical analysis, and if that chemist could determine the formula, the chemist could Manufacture, Use, and Sell the exact same formula. The Coca-Cola company would have no legal protection to prevent this from happening.

(Trademark Law would prevent the chemist from using the Coca-Cola trademark, so the chemist would have to come up with a different name that would not be confused with Coca-Cola. Given the taste similarities between Pepsi and Coke, I think it is safe to say that Coca-Cola maintains its position in the marketplace by "brand identification". The Trade-Secret-Formula is a good James Bond story for the kiddies.)

Patents protect from Reverse-Engineering by granting a monopoly to the Inventor in exchange for publicly revealing how the Invention works. Everyone learns how the Invention works, making Reverse Engineering unnecessary, in exchange for giving the Inventor exclusive rights to the Invention.

Claiming a consumer product, such as the Coca-Cola formula, as a Trade Secret is rare. When it happens, the company is attempting to maintain a monopoly greater than the power of Patent and Copyright for an indefinite length of time. It is a company's prerogative whether to disclose information as a Copyright/Patent or to keep that information as a Trade Secret.

However, keeping information about a consumer product a Trade Secret cannot prohibit Reverse Engineering (or Free Speech, or Fair Use) else a Trade Secret is suddenly far more powerful than any Copyright or Patent, and it lasts forever. See the "Limit of Public Good" diagram.

2.8.1.5 DeCSS

Another example of Intellectual Property law extending beyond the Limit of Public Good is in the DeCSS case.

"Content Scrambling System" (CSS) is an encryption scheme used inside a DVD player. CSS is not patented, instead it was held as a Trade Secret. CSS prevented anyone with a legally owned DVD disc from playing that disc on anything but industry-provided DVD players. Since CSS was not patented, the right to Manufacture DVD-player-like functionality was Public Domain. A number of people, most famously a Norwegian teenager named Jon Johansen, Reverse-Engineered CSS and wrote a program to perform the same DVD-player-like functionality using their Linux machine as a player.

If a device is not Patented, anyone in the Public may Manufacture, Use, or Sell that device.

Jon Johansen was indicted, 9 January 2002, under Norwegian Criminal Code 145(2), simply for trying to access the data on his own DVD. This indictment was at the request of the US DVD Copy Control Association (DVD-CCA) and the Norwegian Motion Picture Association (MAP), allies of the US Motion Picture Association of America (MPAA). He could face two years in prison if convicted.

http://www.eff.org/IP/Video/DeCSS_prosecutions/Johansen_DeCSS_case/

The DVD Copy Control Association (DVD CCA) also sued Andrew Bunner of California, together with hundreds of others, in 1999 for posting, or linking to, the Reverse-Engineered DeCSS code on the grounds that it was a Trade Secret. A preliminary injunction restraining publication of the program was put in place in 1999 by a California court. More than 4 years later, an appeals Court overturned the injunction, saying DeCSS no longer qualified as Trade Secret and that the injunction violated Bunner's constitutional free-speech rights.

http://www.eff.org/IP/Video/DVDCCA_case/20040227_eff_pr.php

DeCSS is defended on the argument that it prevents Copy Infringement

However, DeCSS also prevents legal owners of a DVD disc from viewing the disc on their Linux machine. This is effectively a PATENT on DVD players, without the pesky paperwork involved in filing a Patent application, making the invention public, or a 20 year expiration date. Being prohibited from creating the functionality of DeCSS, prohibits the Public from creating the functionality of a DVD player. BUT A DVD PLAYER IS NOT PATENTED. So members of the Public should have the right to "Manufacture" a DVD player. This is what Jon did.

Suing hundreds of people on the grounds that a Reverse-Engineered technology should remain the company's exclusive property extends Trade Secret law beyond Patent powers and tramples on Free Speech and Right to Knowledge.

If CSS had been sufficiently Novel, Useful, and Not of an Obvious Nature, then it could have been Patented. The company would have been required to make the invention Public, in exchange for receiving a monopoly on Manufacturing, Using, and Selling that Invention. Instead, companies took the gamble to treat CSS as a Trade Secret and lost any exclusive rights to the technology when it was Reverse Engineered.

Unfortunately, a lot of people have been taken to court for the wrong reasons. And it's left a bad taste in a lot of people's mouths over Intellectual Property laws.

2.8.1.6 USA Patriot Act

The USA Patriot Act eliminates a number of restrictions that were foisted on the government by that pesky Bill of Rights.
http://www.eff.org/Privacy/Surveillance/Terrorism/PATRIOT/
It's not an IP related law, but it does have effects on computer usage.

2.8.2 Serving the Public Good

Beyond all the ranting and raving (from both sides), basic Copyright and basic Patent law are really not an issue. These two core components of IP law are balanced sufficiently to serve the Public Good.

What is at issue is when other laws, such as the DMCA, are used by Copyright holders to secure more rights than they are due. When Trade Secret law is misapplied to create a monopoly of a spectrum of rights that includes Patent rights, Fair Use, Free Speech, and Reverse-Engineering, then the Public suffers far more by the loss of these rights than by any benefit that might come of it.

The justification behind Intellectual Property law is "To promote the Progress of Science and useful Arts". IP law can only be defended if it serves Progress. I would read "Progress" to mean "Public Good" not "Profit for Copyright Holders". i.e. more profit does not necessarily mean more Progress.

When IP law, such as the DMCA, is enacted that raises corporate profit above Public Good and Public Rights, then "Progress" is cast aside in favor of "Profit".

When IP law such as Trade Secret law is misused to create a complete monopoly of all rights around a technology, then Public Good is destroyed.

2.9 Respecting Authors and Inventors

Authors and Inventors deserve respect from the Public for their contributions to Progress and the Public Good. If the laws are balanced, then Authors and Inventors are risking private money, private effort, to create something that they hope the Public will want to buy. This is the point of the market-driven research model that current Copyright and Patent law describes. Private risks create Private rewards and Public Progress.

Copyright Infringement is indefensible.

Ranting and raving stuff like "information wants to be free", "it doesn't hurt anyone", "they're charging too much for a legal copy", is a rather naïve attempt to ignore how successful basic Copyright and Patent law has been.

The founding fathers knew that Writings have essentially zero cost to copy and distribute. They also knew that Writings take work to create. So they put Copyright (and Patent) law into the Constitution to encourage market-driven research. Private risks create works that can be sold on the market. Good works are rewarded by lots of sales. Bad works are ignored.

Basic Copyright and Patent Law is a brilliant design.

If a Balance is to be struck between Private Risks and Public Good, then Authors and Inventors must have their Rights respected by the Public. If you like a song that is for sale, then you should PAY for it not STEAL it.

If an Author wants to give their work away for free, that's their decision. But if an Author is taking private risks to create a Writing and relying on Copyrights to recoup their risks, then people need to respect that.

The issues with Intellectual Property law are with the laws that go above and beyond the Limit of Public Good. Copyright should be respected, but enforcing Copyrights should not resort to making Fair Use, Free Speech, Reverse-Engineering, and Scientific Research a Criminal Offense punishable by jail time. Trade Secret law should not be able to prevent Reverse Engineering on a publicly sold product.

2.9.1 From Napster to iTunes

Napster was a file sharing program written by Shawn Fanning in 1999. The program allowed users to share music files on their harddrives, and some users began making illegal copies. A few months after it went online, the recording industry filed a lawsuit against Napster for Copyright Infringment.

The lawsuits against Napster included one from Lars Ulrich of the band Metallica, who slapped Napster with 60,000 copyright violations and a list of 300,000 users who had illegal copies of his band's music. Another lawsuit from Dr. Dre listed 200,000 users with illegal copies of his music.

The lawsuits generated a lot of publicity for Napster. Registered users on Napster peaked at 80 million. The number of users at any given moment peaked at 13 million.

At one point, things got really weird and Napster briefly offered the record companies ONE BILLION DOLLARS (spoken with a Dr. Evil voice) to drop their suit and let them legally stay in business. The record companies didn't believe the offer.

By 2001 Napster was ordered to shut down. And by 2002, it had declared bankruptcy.

In 2003, slightly calmer heads prevailed, and Steve Jobs, CEO of Apple Computers, announced Apple would be providing an online music service called iTunes that would allow people to download music for 99 cents a song. With major record label support, iTunes currently has half a million tracks to download.

The continuing success of iTunes seems to support that the Napster period was freak orgy of copyright infringement. Now that the record companies have caught up with some 20 year old technology called "the internet", iTunes appears to be offering service and product that match what people want and are willing to pay for.

Basic Copyright law appears to be balanced after all.

2.10 IP Law Wrap Up

That is Intellectual Property law in a nutshell. There is a lot of rhetoric around IP law, so I hope that I managed to introduce the law and explain what the rhetoric is all about. If you are going to pursue any kind of Open Content project, you need to know the law, but you'll also need to be able to have a good "smoke detector". There are some problems with the law, too, and I hope I managed to point out the problems with as little rhetoric as possible.

3 Warranties

Open Content projects are also affected by Warranty Laws.

Warranty laws are State laws, rather than Federal, and they vary from state to state.

Warranties can be expressed or implied. An expressed warranty is when the seller expresses a promise such as "warrantied against defects in materials and workmanship for one year". Implied warranties are assumed to come with every product, even if the seller expresses nothing to the buyer. Two implied warranties exist in every State:

1) Merchantability
2) Fitness for a Particular Purpose

Merchantability means the product will do what a person would reasonably expect of it. A hammer is expected to be able to pound nails without breaking.

Fitness for a Particular Purpose means that you tell the seller how you intend to use the product and the seller says the product is "fit" to do that. Even if you intend to do something unusual with the product, if the seller says it will do it, then the product has an implied warranty to do that.

The only way to avoid these implied warranties is to disclaim all warranties. Products sold "As Is" are disclaiming all warranties, including implied warranties of merchantability and fitness for a particular purpose. Some States do not allow "As Is" sales.

Open Source projects Disclaim all product warranties. At the heart of any Open Source project are individual contributers who give their code to the public in the hopes that it may serve the Public Good. These individual contributers are not providing products, nor do they have a budget and a slew of employees needed to test their code and provide warranties of its workability.

Here is a snippet of the warranty disclaimer from the GNU-GPL:

THE COPYRIGHT HOLDERS ... PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

There are usually other disclaimers as well. Check with a lawyer for your situation.

DISCLAIMER OF WARRANTY: I AM NOT A LAWYER. THIS IS NOT LEGAL ADVICE. THIS IS NOT A SUBSTITUTE FOR LEGAL ADVICE.

4 Open Content Projects

Open Content projects are defined, in part, by how their Writings are licensed. Unfortunately, there is no singular license that defines "Open".

4.1 Open Source

For software projects, the term "Open Source" is used. "Open" refers to the relative "openness" that the Copyrights are licensed to the Public. "Source" refers to "source" code, as in whatever Human-Generated-Writings were used to create the program. If a program is compiled into an executable form, it is still a Writing, but much more difficult for humans to read or modify. Therefore the original Source Code is extremely important.

The term "Open Source" has no legal definition, and is not Trademark protected, so anyone could conceivably claim Open Source on their software. To get a consensus of what is and is not Open Source, the Open Source Initiative (OSI) drafted an Open Source Definition (OSD). They then review licenses to see if they fit the definition. If the license meets their definition, OSI will "approve" the license.

Licensing your software with an OSI-approved license is not enough, though. The program being licensed must include all Source Code (all Human-Generated-Works) used to create the work.

If a distribution includes all Human-Generated-Works used to create it, and the distribution is licensed with an OSI-approved license, then the distribution may use the "OSI Certified" Trademark.

http://www.opensource.org/

OSI Approved licenses include:

1. GNU-GPL (General Public License)
2. GNU-LGPL (Library General Public License)
3. BSD License (Berkeley Software Distribution)
4. MIT License (Massachusetts Institute of Technology)
5. Mozilla License
6. Artistic License
7. Intel Open Source License
8. Academic Free License

As the different names show, some use the term "Free", some use "Open" and some use "Public" to indicate the same licensing attitude.

The term "free" is meant to indicate freedom not price. The phrase usually used to clarify is "Free Speech, not free beer." Anyone can take "Free Software" and sell it for money.

Also note that a "Public" license is not the same as "Public Domain".

4.2 OSI-Open Source Definition

The Open Source Initiative (OSI) Open Source Definition (OSD) is available here:

http://www.opensource.org/docs/definition.php

The OSD lists what is required of a License for it to be OSI Approved. To apply the OSI Certified Trademark, a Distribution must use an OSI Approved license AND it must include all Human-Generated-Works.

The OSD requires a license to meet the following requirements:

1) MUST be free to Copy and Distribute Original Work.
2) Source Code MUST be included in Distributions.
3) Derived works MUST be allowed. License must ALLOW (NOT REQUIRE) derived works to be licensed under same license as original.
4) License MAY require Derived works to differentiate themselves from Author's original source code.
5) License MUST not discriminate against persons or groups.
6) License MUST not discriminate against fields of endeavor.
7) Original License MUST be applied to all Original Distributions without requiring relicensing. No further restrictions can be applied.
8) License must not be specific to a product, requiring product to be distributed as a whole
9) License must not place restrictions on works it can be distributed with. (Distributions of unrelated works bundled together are "aggregates" not Derived works)
10) License must be technology neutral. (CD-ROM requirements may soon be as outdated as 5.25" floppies).

Item (1) refers to Copyright law's right to Copy and Distribute the Original Work.

Item (7) refers to Verbatim Copy/Distributions, preventing anyone from re-Distributing the original work under a new license or with additional restrictions. When you Copy-Distribute the Original Work, the Original-License must go with it.

Item (2) requires that all Human-Created-Writings involved in the work must be included with any Distributions of the Original Work. This prevents someone from Distributing a work that is licensed "free to create derived works" but is too difficult because it's only machine generated files.

Item (3) refers to Copyright law's right to Create Derived Works. Derived works must be allowed. However the OSD does NOT place any requirements on how that Derived Work is Licensed. It could be "All rights reserved". It could be the SAME license. It could be a DIFFERENT license.

Item (4) allows the license to require that derivatives somehow indicate what the original work looked like. This is optional. A license may require "Integrity" or not.

I will simplify this to "Author Attribution", meaning that any derived work must indicate that it is Derived from the Original Work. This can easily be accomplished by leaving all original Copyright Notices in tact and appending a new Copyright Notice above the Original in any Derived Work.

Copyright 2004 John Doe Junior

Copyright 2001 John Doe Senior

This is not exactly what Item (4) allows, but it maintains the notion of Author Integrity in a much more flexible manner.

Items (5, 6, 8, 9, 10) all require that the License NOT discriminate in one way or another. Education-Only, Non-Commercial-Only, This-Distribution-Only, This-CD-ROM-Only, etc. are all restrictions that disqualify a License from meeting the OSI-OSD. The license must be "Public". This could be reduced to the license always granting its right to "anyone".

4.3 Simplified OSI-Certification

OSI-Certification could be simplified to the following three requirements:

(A) All Human Generated Works must be included in the Original Work that is to be Licensed.

(B) The Author must License the Original Work to allow Anyone to Copy and Distribute the Original Work under the Original

License.

(C) The Author must License the Original Work to allow Anyone to Create Derived Works, maintaining all previous Copyright Notices, and license it:

[pick one:]

(i) anyway the Deriving Author wishes.

(iI) the same as the Original License and include all Human-Generated-Works in any Distributions.

This effectively reduces the licenses to two options:

(C)(i) Open Content-Public Domain

(C)(ii) Copyleft

4.3.1 Open Content-Public Domain

If the Original Work is licensed with option (C)(i), the work is effectively licensed as Open Content-Public Domain.

When the Copyrights to a work expires, the work enters the Public Domain. A work in the Public Domain can be freely Copied and Distributed and remains Public Domain. An Author can take a work in the Public Domain, Create a Derived Work based on that Public Domain work, and in doing so create a NEW work that is the Copyright Property of the Author to do with as the Author wishes.

Hamlet by William Shakespeare is a work in the Public Domain. Anyone can take Hamlet and create their own particular Derived Work of the original. The Derived Work is the Property of the new Author, but Shakespeare's version is still Public Domain that anyone can Copy and Distribute and Create another Derived Work from.

I call this "Open Content–Public Domain" (OCPD) because there are some differences between OCPD and the legal definition of Public Domain. Public Domain is the state when the Author's rights have expired and the rights become Public. OCPD is the state where the Author's rights have not expired, but the Author has explicitly Licensed all rights away.

For example, one difference is any Copyright Notices should not be removed in a Work that is OCPD. In a truly Public Domain Work, any old Copyright Notices can be removed.

For the most part, OCPD and Public Domain are pretty much the same.

Because Open Content-Public Domain is a mouthful, I may sometimes shorten it to Public Domain or PD. If I say Public Domain in reference to a license, I mean Open Content Public Domain, not Public Domain where the exclusive rights have expired and are now Public.

4.3.2 Copyleft

If the Original Work is licensed with option (C)(i), the work is licensed "Copyleft".

Copyleft is a license that requires all Derived Works to be licensed the same way as the Original. Since this generally applies to licenses that grant anyone the right to Copy, Distribute, or Create Derived Works, then all Derived Works must grant anyone the right to Copy, Distribute, or Create Derived Works of the Derived Work too.

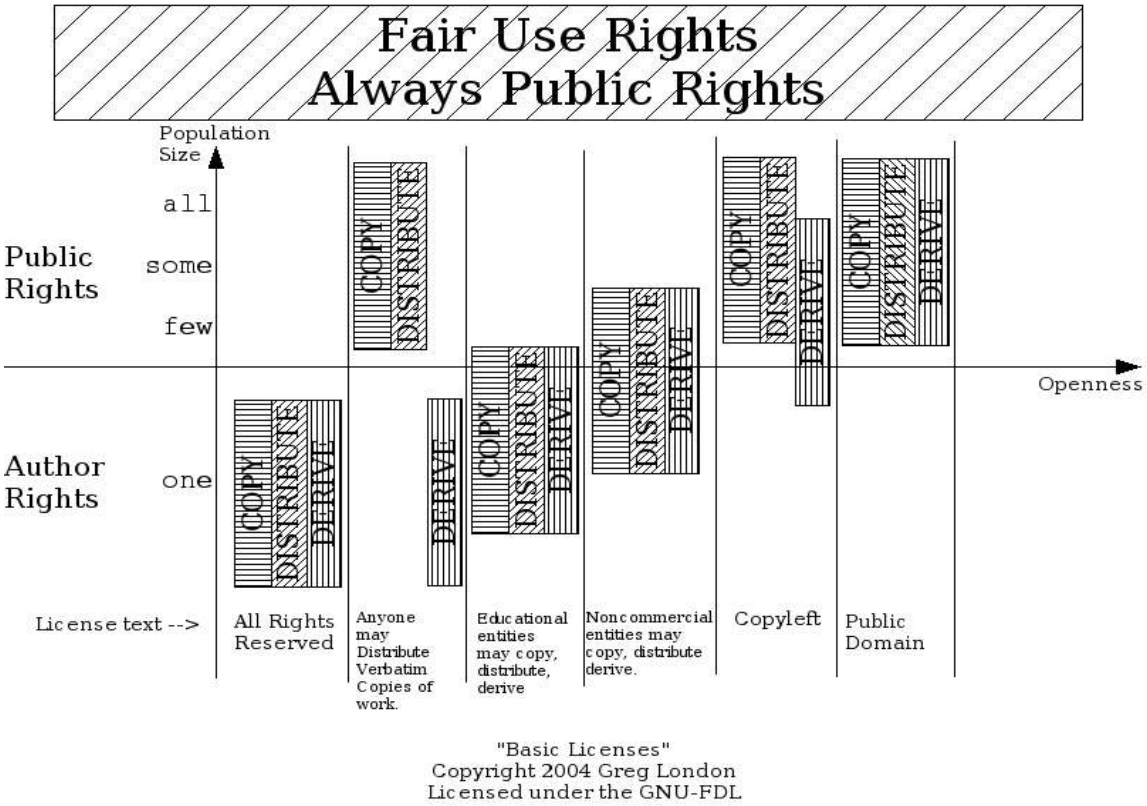
If an Author licenses a Work Copyleft, then the Author essentially has licensed away all rights except the right for anyone (except the Author himself) to create a Proprietary Version of the Work. It is like Public Domain in that anyone can Copy, Distribute, and Create Derived Works, but it is NOT like Public Domain in that Derived Works cannot become the Property of the Deriving Author and instead must remain in the Public Domain.

Some people call Copyleft "viral" because a Copyleft license spreads to all derived works like a virus. In February 2001, Microsoft CEO Steve Ballmer called Linux, which uses a Copyleft license, a "cancer". Using such negative labels may be an attempt to cast Copyleft as somehow "evil", but the fact is that Copyleft operates INSIDE Copyright Law. The Difference is that Authors will use a Copyleft license to give most of the rights to their work to the Public.

As a licensing concept, Copyleft was created by Richard Stallman in 1984 with the GNU-General Public License. About that time, Don Hopkins coined the phrase most often associated with the Copyleft concept:

> "Copyleft -- All Rights Reversed." in opposition to the standard "Copyright -- All Rights Reserved" license.

The diagram below shows how "All Rights Reserved" compares with Copyleft and Public Domain. Note that Education Only and NonCommercial only are NOT "Open Content".



"Basic Licenses"

4.3.2.1 Sorting OSI Approved Licenses

All licenses that meet the Open Source Definition can be sorted into either Open Content Public Domain or Copyleft. Here are SOME of the licenses.

Copyleft:
- GNU General Public License
- GNU Lesser General Public License
- GNU Free Documentation License
- Mozilla Public License
- Open Content License
- Free Art License

Open Content Public Domain:
- BSD License
- MIT License
- X11 License
- Q Public License
- Artistic License

So, if there are only two basic categories, why are there over 50 licenses on OSI's approved license list?

Most licenses call out the name of the organization that submitted it for OSI Approval. Although two licenses may be equivalent in what rights are Licensed to the public, one license may be contain several references to "Corporation X" and another license may reference the "Corporation Y". Most licenses are not written in template form with a blank for the Author's name. Anyone who wants their name in the license drafts a new license and submits it to OSI for approval. This increases the number of OSI-Approved licenses. It also supports the notion that Open Content Authors want attribution for their work.

There are other reasons that there are so many OSI Approved licenses. For the Copyleft licenses, there are differences in what Derived works will be licensed Copyleft and what Derived works may be taken private. When software is Compiled or Linked, it is legally considered a Derived Work. The GPL takes that position. However, the LGPL considers Compiled software to be Derived Works that must remain Copyleft, but software that is simply Linked together can be taken private. So the LGPL is slightly more Public in its licensing.

The GPL licenses contain wording that prevent any Derived Works from being Distributed that contain Patented Software. Some licenses allow Patented Software to be contained in Derived Works.

There are also different disclaimers in the licenses. Some simply disclaim All Warranties, Expressed or Implied. Some spell out a long list of disclaimers.

I also believe that part of the cause of so many licenses is that Open Content is a relatively new concept. It was introduced in 1984 by Richard Stallman under name "Free Software" and "Copyleft". The license was called "General Public License". Mr. Stallman licensed his Emacs text editor under GPL. Linus Torvalds wrote the first Linux kernel in 1991 and licensed it GNU-GPL. Red Hat Software was created as a company in 1995, distributing Linux. The term "Open Source" wasn't coined until 1998. Red Hat went IPO in 1999.

People haven't even been able to agree on a NAME for what is called "Free", "Open", or "Public" depending on who you ask. So it's not surprising that people haven't been able to agree on a license. And as people work out what it is they're trying to accomplish, the licenses get fine-tuned, rewritten, and new licenses get approved.

The "Open Source Initiative" has placed all licenses under the banner of "Open Source" and made it an official moniker, even if the wording of the approved licenses prefer to use "Free" or "Public". Free and Public both have common meanings that are different from what is meant by the Open Source community. Open Source is an invented term and has the advantage of not carrying any historic meanings to confuse people.

Some people still like to rant and rave, however. Some insist on using the term "Freedom", others insist on calling it "Viral". But in actuality, it all falls under the basic Rights given to Authors under Copyright Law.

4.4 Why Has Open Source Worked?

Conventional Copyright and Patent law create a market-driven research model. Individuals are given certain Exclusive Rights to their works for a limited time. An Author or Inventor can use that monopoly of rights to sell the work to the Public and turn a profit. Ideally, the better the work, the more profit. This model works because the market-driven research model is a draw to people who wish to compete in this arena. And it also acts as a filter, weeding out those who don't make salable works from those who do. If the laws are balanced properly between Public Good and Private Incentive, then this model becomes "long term viable", and everyone benefits.

Simply creating a market-driven research model does not mean that everyone will be drawn to start doing market-driven research. Only some are drawn into this model and not others. The people who are drawn to this model are drawn first by its competitiveness and second by the Public Good it creates. Not everyone fits that definition.

The kind of people drawn to an Open Source project are generally drawn first by the Community and Public Good it creates and secondarily by any personal reward that might come of it.

The sourceforge.net website states that it is "providing free hosting to tens of thousands of projects. The mission of SourceForge.net is to enrich the Open Source community". The gnu.org website includes in its mission "to defend the rights of Free Software users". The opensource.org website says it is "for the good of the community". The openoffice.org website's mission is "To create, as a community, the leading international office suite".

The market-driven individuals and companies have a completely different vocabulary. Poking around the microsoft.com website did not immediately reveal any "mission statement". The page for Windows XP exclaimed "Do amazing things with Windows XP", followed by a laundry list of features. This exclamation is directed at "you", the consumer, not the community or world as a whole. This reflects the market drive to create competitive products that can sell.

This isn't to say that the market based model is not as good as the community based model of Open Content. This simply shows how each model affects the very language used by the participants drawn to it.

Two common terms in the Open Content lexicon are "community" and "contribute". If I had to boil down the mission statement of Open Source it would be "To make the world a better place".

This mission or model to contribute to the world draws a different kind of person to it than the market driven model. It draws people who primarily wish to contribute to a community, to make the world a better place, and secondarily reap the personal rewards that come from doing that.

Open Content is a Gift Economy.

5 Gift Economy

Open Source projects, at their heart, create Gift Economies. Here is one definition of a Gift Economy:

http://www.wikipedia.org/wiki/Gift_economy

"a system in which participants give away things of value to the shared benefit of the community."

My definition of a Gift Economy is a system of individual contributors motivated primarily by a desire to contribute to the Public Good and motivated secondarily by any personal reward that may come of it.

This personal motivation draws such individuals together where they can make contributions to a community.

Lewis Hyde wrote a book about Gift Economies called "The Gift: Imagination and the Erotic Life of Property" which is highly recommended reading. If you want to understand the drive behind Open Content projects, you need to understand Gift Economies.

Gift Economies have some rules that apply to Open Source projects.

5.1 Contributions must be "No Strings Attached"

Anyone who contributes to a Gift Economy must do so without attaching any strings to the Gift. The only "requirement" is that the contribution remain in the Gift Economy, to prevent it from collapsing.

Other than the implied restriction that the gift remain in the Gift Economy, the contribution must be made with no strings attached. The beneficiary must be allowed to modify the gift before reinjecting it in the Gift Economy, to contribute something of equal or greater value.

Without the ability to add to a gift, the Gift Economy would simply circulate the original gifts of the original contributors without modification. The total benefit of the Gift Economy becomes limited by what any single Contributor can create, rather than the net effect of the entire Gift Economy working together.

Beneficiaries can take a gift and do whatever they wish to it, by any means they see fit, as long as they contribute something of equal or greater value to the Gift Economy.

5.2 The Gift Must Keep Moving

When Gift Economies rely on physical objects, those objects must continue to move through the group. If a gift stays too long with any one individual, they will get a visit from someone in the Gift Community telling them it's time to give the gift back to the group.

The term "Indian Giver" refers to white-man's encounter with a Native American Gift Economy they did not understand. A Native American would give a gift to an early settler, who would then hold it as their private property. After a time, a Native American would visit the settler and inform him that the settler was expected to move the gift along to someone else or contribute a new gift of equal or greater value.

(It's interesting that the term Indian Giver, created by Laissez-Faire capitalists that would make Adam Smith proud, is defined as "Someone who gives something and then wants it back." An "Indian Giver" actually refers to "Someone who gave a gift intended to remain in a Gift Economy." From the point of view of an American Indian operating in a Gift-Economy, the person who takes the Gift and holds it private might get the label "White-Man Taker" or "Capitalist Privateer". But history is written by the victors, so "Indian Giver" was written into the dictionary.)

The gift is never "consumed" by the receiver. It is held for a while and then passed to the next person. The gift can be transformed. A person can receive one gift and give a different gift of equal or greater value.

5.3 A Gift Economy Generates Community

Gift Economies generate community. A barn raising is a good example. People from a community come together to help one individual at a barn raising. In the long run, any individual gain of receiving an entire barn is balanced by their contribution to help a number of individuals in their barn raisings. Though it balances out from an "energy in equals energy out" standpoint, a beneficial side effect is that bonds between the individuals are created and strengthened.

Market economies do not generate community. Market transactions balance out in an "energy in equals energy out" standpoint. But this balance is achieved in a single transaction, a single sale.

Any single Gift transaction is inherently lopsided. Someone contributes selflessly to someone else. And it may be a long time (or never) before the recipient contributes something back to the giver. This demonstration of selflessness builds a relationship, a community, between giver and receiver.

5.4 Gift Economies Attract People Drawn to Contribute

Gift economies attract people drawn to contribute to a community. The "reward" from the contribution is a function of the "value" of the gift and the number of people who receive it.

This would seem to align with OSI's requirement that a license does not discriminate against persons, groups, fields of endeavor, products, technology, or what the work is distributed with. The bigger the group that can receive the work as a gift, the bigger the reward for the contributers on the project.

5.5 A Gift Economy Requires Vision to Start

The first contributer in a Gift Economy needs to have faith. They are operating in a Market Economy, and contributing a Gift doesn't make sense in a Market Economy. The first person to receive the gift could keep the gift and sell it rather than passing it on. So, the first contributers need faith that enough people will see the Gift Economy, sustain it, and contribute to it.

A vision of a particular Gift Economy helps. A carpenter is not likely to contribute to a project that is ill-formed, with no design, and no direction. A bag of nails and a pile of lumber is not enough to be successful. The carpenter's thinking is that any contribution he makes has a chance of being lost if the project collapses. On the other hand, if the initial contributers have a clear vision of a barn, that same carpenter could see the likelyhood of success, and could see that any contribution he made has a chance of actually contributing something even larger to the community.

Open Projects need a blueprint, a mission, a goal of some kind.

A likely contributer can weigh the individual work involved in making a contribution and compare it to the total "payoff" of the project as a whole. A carpenter might contribute 40 hours to a barn raising that benefits the recipient with the equivalent of thousands of man-hours. A carpenter might be less inspired to contribute 10 hours of work to a project that creates a 20 man-hour benefit for the recipient.

6 A Cathedral, a Commons, and Bazaar landscape in between

In 1997, Eric S. Raymond published a document by the name "The Cathedral and the Bazaar". In it, Mr. Raymond discusses two fundamentally different software development styles: the "cathedral" model of most of the commercial world versus the "bazaar" model of the Linux world. You can read it here:

http://www.catb.org/~esr/writings/cathedral-bazaar/

Much of "The Cathedral and the Bazaar" discusses how to successfully manage an Open Content project in much the same way that Frederick P. Brooks discussed managing a successful software project in "The Mythical Man-Month". Pointy-haired types should check it out:

http://en.wikipedia.org/wiki/Mythical_Man_Month

The Cathedral and the Bazaar is a useful metaphor. I am going to extend it to include the idea of a "Commons", and then show how all three relate to each other.

6.1 The Commons

Rather than thinking of Open Content as a Bazaar, I think it can be more accurately thought of as an Intellectual Commons. A "commons" is any resource that is shared by a community. A pasture could be a commons where everyone in a community could bring their cattle to graze. The oceans could be a commons where anyone can catch fish.

While physical commons are subject to the "Tragedy of the Commons", an Intellectual Commons does not suffer any physical version of this shortcoming. Intellectual ideas and knowledge can not be overgrazed or overfished. You can put works into an Intellectual Commons and everyone can take them out without losing them from the Commons.

The largest Intellectual Commons would be the Public Domain. Everyone can feed off of ideas in the Public Domain, and the rights to all the works in the Public Domain are Public. Anyone can Copy, Distribute, and Create Derived Works. Feeding on the commons is free.

6.2 The Cathedral

The "Cathedral" is Eric S. Raymond's metaphor for the market-driven research model that is encapsulated in Basic Copyright and Patent law. While Mr. Raymond focused on the Cathedral metaphor based on its design approaches, I'm more interested in comparing how rights are used in a Cathedral model versus a Commons.

Someone pursuing the market-driven research model creates a new Writing or Patented Invention. They have exclusive rights to that work. The population size is one. The Copyright or Patent Rights freely accessible by the Public are zero. The cost for accessing those rights are relatively high compared with the gift economy.

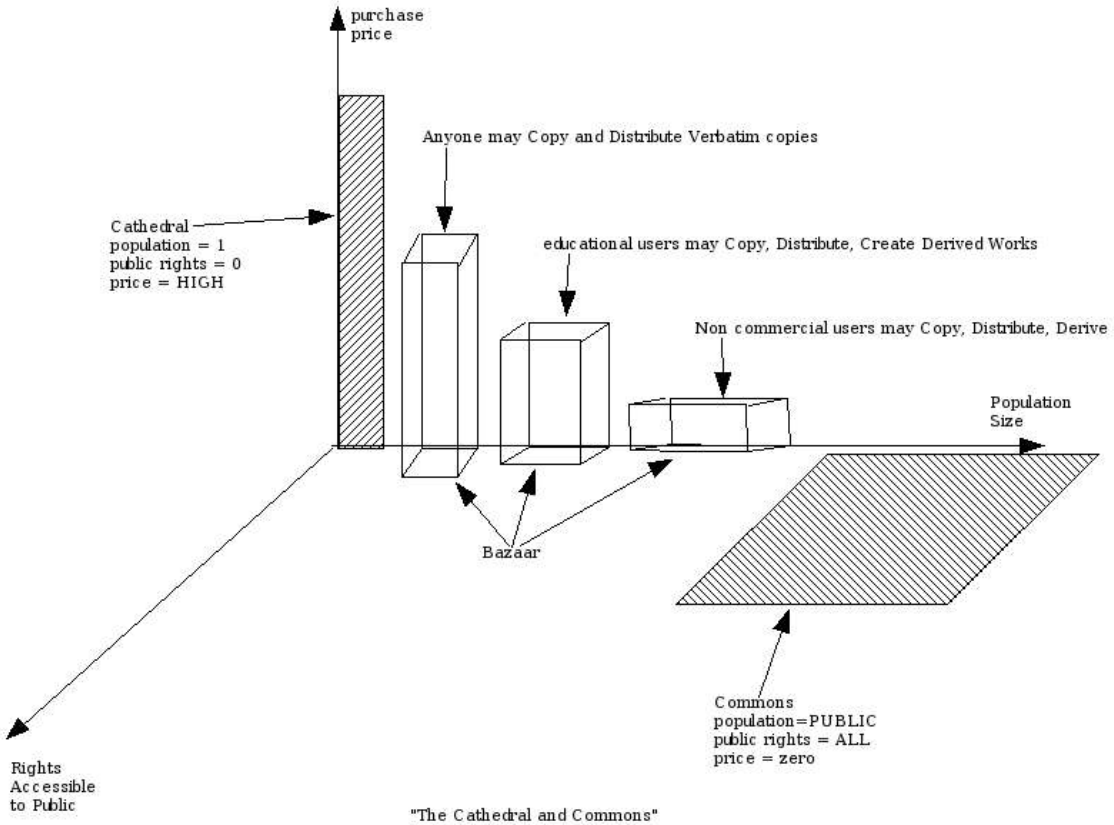
6.3 Bazaar Landscape

The diagram below shows how the Cathedral license and Commons license compare on Public Rights, Population size, and relative Purchase Price. If you squint hard, you can see a Cathedral on the left and a Commons Pasture on the right. (The objects have been spread out on the X axis for clarification.)

It also shows some various other licenses in between, which I label Bazaar. This is in contrast to Mr. Raymond's metaphor. A license that does not allow Derivatives prevents any outside contribution to the work and therefore prevents the work from generating a Gift Community.

Education-Only and Non-Commercial-Only licenses needlessly restrict the community size and the number of beneficiaries. Red Hat has added software to Linux to improve it, and those improvements are Open and remain part of the Gift Economy. Commercial use does not necessarily take the work out of the Gift Economy, and it can actually add and contribute to it as well.

Education-Only has been tried in software licenses before, but they have shown themselves to be cumbersome, open to interpretation of what exactly is "education", and not conducive to generating a Gift Economy large enough to build mammoth, barn-like works.


purchase
price
Anyone may Copy and Distribute Verbatim copies
Cathedral
population = 1
public rights = 0
price = HIGH
educational users may Copy, Distribute, Create Derived Works
Non commercial users may Copy, Distribute, Derive
Population
Size
Bazaar
Commons
population=PUBLIC
public rights = ALL
price = zero
Rights
Accessible
to Public
"The Cathedral and Commons"
Copyright 2004 Greg London
Licensed under the GNU-FDL

Looking at the above diagram, the stark contrast between Open Content and Closed Content become quite clear. While some use a lot of rhetoric to label themselves "Open", I believe the above diagram helps identify where their actions fall in the diagram and therefore what label they deserve.

As an aside, I am struck by how the shape of the Cathedral is so different from the shape of the Commons. And yet, both model self-sustaining economies.

6.3.1 Yin and Yang

The diagram reminds me of the concept of Yin and Yang, which represent two opposing elements of the universe. They are not two poles like good and evil, however. Yin is associated with femininity and community. Yang is associated with masculinity and competition.

Commons => Yin (black) => Community
Cathedral => Yang (white) => Competition

The philosophy of Yin and Yang say that it is impossible to have pure Yin or pure Yang. A Community is made of Individuals who Compete. Individuals are part of a Community. The goal is to achieve a balance between Yin and Yang. The ideal balance between Yin and Yang can be expressed visually as shown below. Yin is represented by Black. Yang is represented by White. Note that the large white community has a bit of competition in it, and notice the large Competitive group contains a bit of Community.

7 From Open Source to Open Content

Extending "Open Source" to "Open Content" is rather straightforward. From a Copyright point of view, certain non-software works give additional rights to the Author. These are the right to Publicly Perform the Work and the right to Publicly Display the Work. An Open Content project should treat these rights like the right to "Copy".

Also, the term "Source" simply refers to any Human-Generated-Work that was involved in creating the work being distributed. Human-Generated-Works would extend the license from Source-Code to any works that were created by a human used in creating the final work.

This requirement would prohibit someone from using 3-D animation software to create a video and then distribute that video without any of the intermediate files used to create it, such as the character models, the props, landscapes, etc. Without these intermediate files, it will be impossible for others to modify or extend the original work.

7.1 Open Content Licensing Requirements

The basic requirements to create an Open Content project of any Medium (software, music, film, text) is a simple extension of the Open Source requirements. (A) affects what is in the distribution. (B) and (C) affect what is in the license for the distribution.

(A) All Human Generated Works must be included in the Original Work that is to be Licensed.

(B) The Author must License the Original Work to allow Anyone to Publicly Perform, Publicly Display, Copy and Distribute the Original Work under the Original License.

(C) The Author must License the Original Work to allow Anyone to Create Derived Works, maintaining all previous Copyright Notices, and license it:

[pick one:]
(i) anyway the Deriving Author wishes.
(iI) the same as the Original License and include all Human-Generated-Works in any Distributions.

This does not deal with Warranty Disclaimers or Patent concerns. But it gives a basic skeleton upon which to build an Open Content project and license.

7.2 Open Content Project Requirements

An Open Content project should be designed to create a "Community" that people can "Contribute" to, with some tangible goal to "make the world a better place" in some way.

All three components are needed. Community means individuals team together, building on each other's works. The right to Create Derived Works is paramount. Without community, you have many individual contributers whose works never combine, never extend, never develop beyond what the first Author created.

From a barn-building perspective, what you don't want is to have individual contributers show up, some build rafters, some build wall frames, and some build doors, but none of the pieces are ever put together. A barn is far more useful than a bunch of building materials laying around on common ground. That's the difference between community and a bunch of individuals.

The project should create opportunities for people to Contribute. This will help to generate the project community, and it will attract the kind of people primarily motivated to contribute to a community. You do not want your project to attract people looking primarily for a place to compete. Someone who is motivated primarily for their personal gain, and secondarily for the good of the community, will always be in conflict with the intent of the project.

The project should have some tangible goal to make the world a better place in some way. An Open Content license by itself is not a project; it's raw materials. A blueprint is needed to map out what you want the project to create, who it will benefit, and how.

7.3 Open Content Model Example

A model example of a well designed Open Content project that is NOT software would be Wikipedia.

http://www.wikipedia.org

Wikipedia is an Open Content Encyclopedia that started in January 2001. By 2004, it had almost a quarter-million encyclopedia entries that contributers have generated, updated, modified, and improved. All contributed works are licensed under the GNU-Free Documentation License (GNU-FDL), which is a Copyleft license. FDL keeps all contributions in the Gift Economy.

An interesting technological twist about Wikipedia is that anyone can make contributions to the project immediately with their web browser. The program that allows this is called Wiki. Open Software projects usually have centralized structures with someone in charge of determining when it's time to make a release. Wikis release every change immediately. For the online encyclopedia content, it appears to be extremely successful.

Wikipedia is long term viable.

It has community which allow contributors to update, improve, and fix other contributor's entries. And it has a specific goal that is big enough that any contributor will feel their effort is well worth the massive payoff of a complete, online, free, encyclopedia.

Wikipedia is effectively enacting the original promise of the internet: all the information of the world available at your fingertips.

That is a wicked cool mission.

7.4 Creative Commons

The creativecommons.org website introduces itself by saying it is devoted to expanding the range of creative work available for others to build upon and share. It's mascot is a cartoon of a cow eating in a "Commons" pasture. It's "About Us" page is titled "Some Rights Reserved". It's mission statement page says it took inspiration from the GNU-GPL. It also mentions "Open Licenses", Freedom, and Public Domain.

Creative Commons offers a range of licenses for Content other than software. It's home page lists musicians, photographers, illustrators, writers, bloggers, filmmakers, educators, scholars as the type of people for whom their licenses were created.

The CreativeCommons-ShareAlike (CC-SA) license is an OpenContent-Copyleft license for non-software Writings. The CreativeCommons-PublicDomain (CC-PD) license is an OpenContent-PublicDomain license for non-software Writings.

In addition to the Gift Economy licenses like CC-SA and CC-PD, Creative Commons also offers Market-Economy licenses. One Market-Economy license is a NonCommercial (CC-NC) license. This grants any NonCommercial entity the right to Copy, Distribute, and Create Derived Works.

An excerpt from the Creative Commons FAQ:

> If I choose the noncommercial license option, can I still make money from my licensed works?
>
> Absolutely. The "noncommercial use" condition applies only to others who use your work, not to you (the copyright holder). When other people use or trade or copy your work, they cannot do so for "monetary compensation or financial gain," unless they get your permission.
>
> One of our central goals is to encourage people to experiment with new ways to promote and market their work. In fact, we designed the noncommercial license option to be a tool to help people make money from their work, by allowing them to maximize the distribution of their works while keeping control of the commercial aspects of their copyright.

According to CC, the point of licensing your writing CC-NC is to "promote and market" and "maximize ... distribution" of your work so that someone will come along and pay you for a Commercial license to your work.

CC-NonCommercial (CC-NC) is a Market Economy license. Someone would use CC-NC in an attempt to get free advertising so they can sell their work. An Author has every right to use the CC-NC license, however, if the intent is to contribute your work to generate a Gift Economy, then CC-NC won't do it.

Wikipedia generated a quarter million encyclopedia entries in its first 3 years with GNU-FDL. The only Creative Commons license that has a chance of creating that powerful of a Gift Economy is CC-BY-SA.

7.5 Opsound

According to its own website at http://www.opsound.org

> Opsound is a record label using an open source, copyleft model, an experiment in practical gift economics, a laboratory for new ways of releasing music.

Opsound requires that all music contributed to its site be licensed CC-BY-SA (Creative Commons-Attribution-Share Alike) or CC-PD (Creative Commons-Public Domain).

Opsound has some of the components to be long term viable. It has the right license. And it appears to understand Gift Economics. What is not so clear on the website is a clear explanation of how the project of Open Content music will make the world a better place. Briefly cruising the website did not reveal any Specific Grand-Scale Goal. Music takes a lot of work to create. Musicians may be more inclined to donate if the end result far outweighed the work they put into their individual contributions. I think it could benefit from a blueprint for a really big barn.

7.6 Magnatune

Magnatune releases music under the CC-BY-SA-NC license. It sells music and merchandise, and it sells relicensing for customers who want to use music for commercial uses. Musicians are offered non-exclusive contracts for their works, which means the musicians can sell their music to someone else if they want.

The Non-Commercial license places it in the Market Economy. And Magnatune appears to be successfully operating in its market driven economy. It appears to be a niche market now, though it might grow over time.

Magnatune operates in a market space slightly better than Print-On-Demand books. It's better because musicians currently don't have to pay Magnatune a huge up front cost setup fee like POD printers would charge. So there isn't an incentive to "sign" anyone who has the money.

When operating in a Market Economy, the rule of thumb is "money should not flow FROM the Author." When an Author pays to get their work published, beware of a scam.

8 Copyleft or Public Domain?

The choices of Open Content Licenses are either Public Domain or Copyleft. The Author has the right to choose whichever license they wish, but which one "should" the author choose? A lot of rhetoric has been generated around this question. The argument around this question is part of the reason there are 50 OSI-Approved licenses.

If the choice is regarding an Open Content project, then the choice should be whichever license creates a self-sustaining gift economy.

8.1 Proprietary Competition and Public Domain

Say a **Public Domain licensed** project took 10 man-months of effort to create. A Proprietary company can take this work for zero cost, add 4 man-months of effort to it, and create a Derived work that has 14 months of effort in it. The Public Domain license allows the company to license this new product "All Rights Reserved". That company ends up with a product worth 14 months of effort and only 4 months of effort invested into it.

If an Public Domain project does not have proprietary competition, this model will mean that they probably will at some point in time.

This Proprietary group competes with the Open Content project for customers/beneficiaries. If the Public Domain project loses a large enough portion of their customers/beneficiaries, then the PD project is in danger of collapsing. The total benefit of the project, in the eyes of the contributers, may fall below their individual contributions. At which point, the Gift Economy collapses.

Imagine a group of carpenters each put a month of work into building a barn. If no one wants the barn because someone is selling a better barn for a small amount of money (by taking the gift barn, expanding it, and licensing it "All rights reserved"), then the Gift Economy collapses because all the carpenters will be thinking "Why bother? No one wants what we're giving away."

The Public Domain group can add another 4 months of effort to their project and attempt to create an Open version of the work that has 14 months worth of effort into it, comparable to the Proprietary version. But the Open project ends up with a 14 month project that cost 14 months of effort, while the Proprietary company has a 14 month project that cost 4 months of effort. The Open project will always be behind the Proprietary company, and will have to expend work to "catch up" at every improvement.

Worse yet, if the Proprietary company adds 4 months of effort that are restricted in some way, the Open project can never catch up, even if it wanted to contribute the extra effort. Perhaps the company adds a new feature that is Patented. Or perhaps the new feature involves

Encryption. The DMCA makes it a criminal activity for anyone to reverse engineer any Encryption algorithm without the Company's consent. Therefore, the company just patented their 4 months of effort without a pesky Patent Application, and with no annoying requirement that the function be Novel, Useful, and Not of an Obvious Nature. No one may recreate that same encryption functionality, so the PD project is locked out.

So now the Proprietary company has a product worth 14 months of work that they only put 4 months of effort into, and they've managed to Lock out the Public Domain project from EVER catching up.

8.2 Proprietary Competition and Copyleft

Say a **Copyleft licensed** project took 10 man-months of effort to create. A Proprietary company can take this Writing for zero cost, add 4 man-months of effort to it, and create a Writing that has 14 months of effort in it. The Copyleft license REQUIRES the company to license this new Writing under the same Copyleft license. Both the Copyleft project and the Proprietary company end up with a Writing that has 14 weeks of effort in it.

The Copyleft license maintains the Gift Economy because the Writing, and all its Derivatives, must remain Open.

Although both sides put in different amounts of effort, both sides end up with a Writing of the same value. The gift effort of 10 man-months made by the Open Content project is not made in vain, because the Open Project is not Excluded from the 4 man-months of effort made by the Proprietary company.

The question "Why Bother?" does not occur for the people who contributed to the Open Content Copyleft project.

This assumes that the Open Content project is not Excluded from the Derived work in some way. A software Patent, some DMCA maneuvering, and similar restrictions, would leave the Open Content project at a disadvantage similar to the Public Domain licensed project.

8.3 Copyleft and Public Domain versus Competition

The table below summarizes how an Open Content project using a Copyleft or Public Domain license fairs against a Proprietary competitor. If the Competitor Derives the Open Writing by adding restricted works (Software patent, DMCA, etc), then the timeline stops at step 2. If the added works are not restrictive, then the timeline can proceed to step 3.

Open Content License->	**Copyleft**				**Public Domain**			
	Open Project		Proprietary Competitor		Open Project		Proprietary Competitor	
effort put In versus total value gotten Out	In	Out	In	Out	In	Out	In	Out
1:Authors Create Work	+10	10	GIFT 10	10	+10	10	GIFT 10	10
2:Competitor Derives work (stop here on Patent, DMCA, etc)	10	10	+4	14	10	10	+4	14
3:Authors "Catch Up" (if not restricted by competitor)	10	GIFT 14	4	14	+4 14	14	4	14

8.4 Best Case – Worst Case Scenarios

The worst-case scenario for an Open Content project would be for the Competitor to exclude the Open Content project from adding similar functional value. The score ends up at step 2, Open=10/10 and Competitor=4/14. Review the Halloween Documents for tips on how to exclude and "De-commoditize" an Open Content project. Note that Copyleft and Public Domain make no diffference in this case.

The best-case scenario for an Open Content project would be using a Copyleft license, and for that license to only allow Derivatives that do not Restrict the Derived Writing. The score ends up at step 3, Open=10/14 and Competitor=4/14.

The point is to choose a license that will allow the Open Content project to be long term viable. This means the project needs to survive in a field of Proprietary Competition.

As it happens, the GNU-GPL is a Copyleft license that does not allow derivatives that add restrictions such as software patents. The GNU-GPL, in its various forms, has been around since 1984 and is the base license for the Linux operating system, which has been built by millions of contributors, generating millions of lines of code, to create an operating system that runs on millions of PCs. The GNU-FDL is Copyleft and is used by Wikipedia, generating a quarter million Open encyclopedia entries in 3 years.

8.5 Choosing Public Domain Licenses

The above argument does not take away an Author's right to place their work under a Public Domain license. If a project is sufficiently short in duration, a Public Domain license might work. If the project can be completed before any competitor arrives, the team will not experience a loss in "customers/beneficiaries" and therefore will not suffer the "Why Bother?" syndrome that could cause the number of contributors to dwindle.

Another reason given to use a Public Domain license on an Open Content project is because the Contributors view the "beneficiaries" of the project to include Proprietary companies. If this is the case, the "Why Bother?" syndrome will probably NOT be a problem when the Proprietary company takes the code private. I'm not sure this is long term viable though, since the commercial advantage will be a draw for ANY company.

Perl is a "small" project when compared to the Linux Operating System and all its Applications. Perl is also, for the most part, short-term in scope. Perl uses a type of Public Domain license, and regards Proprietary companies as its beneficiaries. The attitude in the Perl community seems to be that Proprietary competition can be overcome by perseverance. Perl also qualifies as "successful", having earned the label "the glue that holds the internet together". Most websites rely on perl code to work. So, Public Domain licenses have been used successfully in Open Content projects.

Public Domain licensed projects do not exclude a Copyleft project from using them for longer term projects. Public Domain licenses allow anyone to take a version of the Writing private and license it "All Rights Reserved". But it also allows someone to take a version of the Writing and license it Copyleft. A long term Copyleft project may want to consider taking a Public Domain work semi-private and licensing it Copyleft before adding effort into the work to extend it.

8.6 Self Sustaining Gift Economy

A self-sustaining Gift Economy must be designed to survive alongside Proprietary competition. Over time, the incentive to take an Open Project work private, extend it, and sell it "All Rights Reserved", is too great for companies to ignore, so eventually, someone will take advantage of it if they can. For this reason, a self-sustaining Gift Economy must use Copyleft. Small projects and short projects can use Public Domain, but self-sustaining projects need the protection of Copyleft. Long term projects must use Copyleft or they risk Proprietary Competition killing off the Gift Economy.

Self-Sustaining means that the project creates a Gift Economy that inspires others to contribute to it that create works that benefit a community that inspires more people to contribute to it that eventually creates works that clearly make the world a better place.

Market-Economies can always give people incentive to do work by offering them money.

Gift Economies can only exist, in the long term, on inspiration.

The best way for a Gift Economy to achieve self-sustaining capacity is for each stage of contribution to create something that inspires the next wave of contributers to join the project. At any stage, the contributers are creating something that people can see forwards the Public Good. New contributers are inspired to join the project in numbers that are greater than or equal to the number of old contributers who left the project.

The project is a chain reaction that maintains a critical mass of contributers indefinitely. Copyleft protects the Gift Economy from being collapsed by Proprietary Competition. The Project ongoingly contributes to the Public Good and inspires the next round of contributors.

9 Universal Open Content Licenses

Given that Open Content licenses boil down to Open Content-Copyleft and Open Content-Public Domain, it would be an interesting exercise for someone to take on creating a Universal Open Content License that would work for ANY type of Writing, regardless of medium.

Below are plain-language descriptions of what a Universal Open Content license might look like.

9.1 Universal Open Content-Public Domain

(1) Author grants Anyone the right to Publicly Perform, Publicly Display, Copy and Distribute the Original Work under the Original License.

(2) Author grants Anyone the right to Create Patent-Free Derived Works, maintaining all previous Copyright Notices, and license it anyway the Deriving Author wishes.

9.2 Universal Open Content-Copyleft

(1) Author grants Anyone the right to Publicly Perform, Publicly Display, Copy and Distribute the Original Work under the Original License.

(2) Author grants Anyone the right to Create Patent-Free Derived Works, maintaining all previous Copyright Notices, and license it the same as the Original License and include all Human-Generated-Works in any Distributions.

9.3 Disclaimer

(3) AUTHOR PROVIDES THE WRITING "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

9.4 Yardsticks

I do not recommend anyone use the above licenses. But they at least serve as yardsticks against which you can measure other licenses that claim to be "Open".

However, a single Universal license would have the advantage of being clear and consistent for everyone, would not have license incompatibility problems, and would simplify license selection. I don't think a menu of 50 licenses is a good thing, in the long run.

Also notice that it is impossible to create a license that requires the Original Author to include all Human Generated Works that were used to create the Writing being licensed. It is left to the Contributors to include the Human Generated Works in the Original Work being licensed.

This highlights the issue that OSI addressed with people using Open Source licenses but not including all the Source Code. OSI has Approved certain licenses, but a Work is not "Certified" by OSI unless the Source Code is included.

The Universal Open Content license would have the same issue. People could use the License but not include all the Human Generated Works they used to create the Original Work. However, since Open Content is about Contribution, the people in a Gift Economy should be able to Self-Check themselves to inform everyone that Contributions are must meet a certain minimum level to be useful.

If a project started by centralizing video clips generated using 3D animation software that were created by people who wish to license then as Open Content-Copyleft. It may start out with people contributing nothing but the final footage. However, as the project grew, it would likely morph into people submitting CGI models for "Actors" and "Props" that people could simply use in a movie.

I don't know if an Open Content CGI project exists, but it would be interesting to see it develop.

10 The Gift Domain

In researching and writing this document, it occurred to me that a Universal Open Content license wouldn't have to be the end of it.

Copyright and Patent law were created by Congress "To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries;"

Congress currently has secured Authors a set of rights that encourage a Market Economy.

However, Congress could secure a second set of rights that encourages, fosters, and inspires a Gift Economy of Authors contributing their Writings to a greater community.

I call it the "Gift Domain", since it is like "Public Domain" except it is a Gift Economy.

Authors will be able to contribute their works to the Gift Domain if they so choose. Authors who wish to create a Gift Economy will know that the Gift Domain will protect their contributions. The Gift Domain will allow a self-sustaining Gift Economy to survive alongside, and as a complement to, a Market Economy.

10.1 Drafting the Gift Domain:

Current Copyright law is based within section 8 of the Constitution to secure certain rights for Authors for a limited time to promote the useful Arts. A plain language version of Copyright law might look like this:

> Authors are secured the exclusive right to Copy, Distribute, Create Derived Works, Publicly Perform, and Publicly Display their Writings for a limited duration.

Copyright law could be written within the Constitutional boundaries that would keep current Copyright rights and ADD a second subset of rights that would create a Gift Domain.

> Authors who Irrevocably Commit their Writings into the Gift Domain are secured the exclusive right to Attribution and to Create Derived Works that can be taken out of the Gift Domain.

The Original Author has the exclusive right to Create Derived Works that can be taken out of the Gift Domain. This means that the Public has the right to Create Derived Works, but those works must remain in the Gift Domain.

This leaves the Author with the same rights he would have after licensing his writing Copyleft, except it's not a license granting rights back to the public. Instead, it changes the law to create a Gift Domain that secures a smaller set of rights to Authors.

If an Author commits their work to the Gift Domain, the author is not granted the exclusive right to Copy, Distribute, Publicly Perform, Publicly Display the work. Therefore no license is needed to grant those rights back to the Public.

When an Author creates a Writing, they are automatically granted full Copyright protection to Copy, Distribute, and Create Derived Works. However, after creating the Writing, the Author can commit their Writing to the Gift Domain, permanently un-securing some rights to the Public.

Creating the Gift Domain does not change current Copyright protection. Authors will still default to being secured the right to Copy, Distribute, and Create Derived Versions of their Writings. The Gift Domain simply creates a legal structure for those Authors who wish to contribute their works to the community.

10.2 The End of Software Patents

Software Patents extend far too powerful of a monopoly to the Author/Inventor compared to the Public Good created by allowing purely Software-based patents to be granted.

Pure-Software patents, as a whole, have not demonstrated that the burden suffered by the Public Good in surrendering Patent Rights and Copyrights to an Author/Inventor for a piece of software has been balanced by an equitable contribution to the Progress of Science and useful Arts around software. There is no patented pure-software that has improved anyone's quality of life in any significant way.

10.2.1 Programmer Burden

Copyright grants the Author the exclusive right to Copy, Distribute, and Create Derived Works. Patents grant the Inventor the Exclusive right to Manufacture, Use, and Sell an Invention. The Author of a piece of Patented software gets both Copyright and Patent rights. These rights line up one-for-one.

	Patent	**Copyright**	**Patented Software**
duplicate an instance	Manufacture	Copy	Manufacture and Copy
transfer an instance	Sell	Distribute	Sell and Distribute
extend creation	Use	Create Derived Works	Use and Create Derived Works

To print a copy of someone's book, you need the right to Copy a Writing. To assemble an instance of someone's Invention, you need the right to Manufacture it.

To transfer someone's book from one person to another, you need the right to Distribute. To transfer someone's Invention, you need the right to Sell.

To write a sequel to someone's novel, you need the right to Create a Derived Work. To improve the function of someone's Invention, you need the right to Use it in your invention.

The table below shows how much burden is carried by the creator of a physical invention, a physical book, and a piece of software.

	Patent (Telephone)	**Copyright (Mark Twain)**	**Patented Software One-Click Shopping**
cost to create (invent/write)	high	high	high
cost to duplicate (manufacture/copy)	high	low	zero
cost to ship (sell/distribute)	low	low	zero
extend creation (use/derive)	high	medium	medium
total effort by author/inventor	3 * high	2 * high	1 * high

Comparing the total cost taken on by the Author or Inventor, physical inventions are far more expensive than software, and physical writings (such as books) are somewhat more expensive.

But computers and a global internet make the cost to Copy, Distribute, Manufacture, and Sell software approach zero. A software developer carries a much smaller burden than Inventor of a physical device or Author of a physical Writing. Yet patented software is granted more rights than either.

Additionally, when a consumer purchases a physical Invention or physical Writing, the consumer usually pays for a fully functioning, stand-alone device. For Patented software, the **consumer** has to bear the burden of purchasing a computer system (PC, monitor, printer, modem, peripherals) and paying for an internet connection. When the consumer buys some Patented software to run on their PC, the Author/Inventor carries no burden for the cost of the physical machine to run it on.

The Author/Inventor of Patented software bears far less burden than the Inventor of a physical device or the Author of a physical Writing. Yet patented software is granted more rights than either Inventor or Author.

The Open Source movement didn't even exist in 1980. But Open Source programmers have since contributed software that runs on tens of millions of computers. This huge explosion is a direct result of the cost to Copy and Distribute software falling near zero. A single programmer can write code and distribute it within a Gift Economy for no cost, save for a broadband subscription.

Proprietary Vendors may then argue that the cost to Invent software is far greater than the cost to Invent a physical machine or the cost to Write any other type of writing. But again, Open Source software has shown that this cost is apparently small enough that people are willing to contribute their code to the world to create Linux and other massive Public Projects.

10.2.2 Public Burden

The Public bears a burden by surrendering Copyright and Patent Rights to Authors and Inventors. However this burden is fairly balanced by the Public Good generated by market-driven research promoting the progress of Science and useful Arts.

Software Patents have a much higher burden than either a Patent on a physical device or a Copyright on a writing alone.

The workings of an Invention must be completely revealed to the Public in order for the Inventor to be granted a Patent. In exchange for an exclusive monopoly to Manufacture the Invention, the Public demands that the Science and Knowledge contained in the Invention be made Public. This allows other inventors to learn about the invention and to extend the invention or to create a new invention.

Software, however, has no distinction between revealing the Invention and Manufacturing the Invention. The software is the Invention, and it is a description of the Invention. Software by itself is a Writing. And as it happens, it is the clearest Writing that describes the Invention. The software by itself does not provide the functionality. The software must be run on a computer to function.

The easiest way for the Public to learn and understand how the patented software works is to distribute the software itself or software that describes the same functionality. But since Software Patents consider this infringing on the Author/Inventor's right to Manufacture, Use, and Sell the invention, it can't be done.

There is no way to distinguish between software as an Patented Invention and Software as a Writing or Free Speech. The current state is to have Patent Rights trump the Right to publicly disseminate how the invention works, the right to Free Speech. Therefore, the Public bears more of a burden in surrendering Patent rights to software.

10.3 The End of Software Patents

"Congress shall have the power ... To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries; ..."

Congress is not granted the power to create Patent Rights simply to create profit for the Inventors. Any monopoly will be profitable for the monopoly holder. Instead, Congress shall have the power to legislate Patent Rights solely to promote the Progress of Science and useful Arts. This also requires that the Public Burden suffered by surrendering the rights be balanced by the Public Good generated by the rights.

Software Patents are more of a Public Burden than Patents on a physical Invention. Yet the Author/Inventor carries less burden to create the patented software. The balance has tipped far too much in the Patent holder's favor.

The Public Burden can be lifted by excluding pure software from being granted Patent protection. The Progress of the Science and software engineering can instead be promoted by Congress by legislating the Gift Domain.

A self-sustaining Gift-Economy will respond far better to what its beneficiaries desire. Excluding software from Patent protection will mean that Science will be uninhibited to create software, and computers and the internet will support the free distribution of this Science so that useful programs are created.

The Market Economy has shown it uses Software Patents first and foremost to achieve a monopoly, to "de-commoditize protocols and applications", with little more than lip service towards the Progress of Science or the Public Good.

When legislation is passed creating the Gift Domain, it should also prohibit patents from being granted for pure software writings. Any currently existing software Patents should be honored.

10.4 The Gift Domain and Existing Patents

If existing patents will be honored after the Gift Domain is created, how does the Gift Domain relate to Software Patents?

Software will be treated as a Writing only, and not as an Invention. If the holder of a Grandfathered Software Patent uses the works in a Gift Domain to Create a Derived Work that implements the software patent, that work must be irrevocably committed to the Gift Domain. This means the only rights secured for the Patent Holder are Attribution and Creating a New Derived Work that can be taken out of the Gift Domain.

By using the works within the Gift Domain to implement a Software Patent, the Author/Inventor irrevocably unsecures the right to Manufacture, Use, and Sell their Invention with regard to the works in the Gift Domain.

If the software patent is written using no works from the Gift Domain, the Inventor is still secured the right to Manufacture, Use, and Sell the Invention with respect to those instances.

Patents cannot be used by players in the Market Economy to use the contributions within the Gift Domain and exclude and collapse the Gift Economy.

11 Summary

You should now be able to clearly distinguish what is and is not an "Open Content" project. You should be able to create the framework of a self-sustaining Gift Economy that supports an Open Content project of any medium, be it software or music or photos or video or whatever. And I hope you'll be able to generate the Inspiration around your project that reflects your desire to make the world a better place.

If you are looking to contribute some Writing to the world, there are existing licenses that will put your work into a Gift-Economy. A good "Open Content" license for software is the GNU-GPL. For text, use the GNU-FDL with no invariant sections. (This work is licensed under the GNU-FDL, for example.) For other works, take a look at the CreativeCommons-Attribution-ShareAlike license (CC-BY-SA) without any other additional restrictions. Do NOT apply NonCommercial, EducationOnly, or NoDerivatives restrictions to your work.

There are also existing Open projects that would gladly accept your help. You might want to Google a bit and see if there is a barn raising already going on that you would like to contribute to. But in the end it's up to you to decide if, how, and when you will contribute your time and energy.

Copyright and Patent law are not in opposition to "Open Content". All Open Content projects currently thrive inside of Copyright law. However, creating a Gift Domain would legally recognize the contribution of Open projects, while at the same time protect those contributions inside of a Gift Economy that can exist alongside, and as a complement to, a Market Economy.

The Gift Domain would also solve the problem of Software Patents that has been an issue since 1981. A legally sanctioned Gift Domain would promote the Progress of Science and useful Arts in the area of Software, while removing the excess burden carried by the Public due to software Patents.

The widespread adoption of Copyleft licenses in the various Open Content projects indicates that people want a Gift Domain. Creating a Gift Domain would solve the Software Patent problem. And it falls within the powers granted to Congress by the Constitution.

12 GNU Free Documentation License

Version 1.2, November 2002

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 noncommercially. Secondarily, 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.

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 for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

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.

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 1.2 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.