Introduction to Free and Open Source Software

Edward M. Corrado

The College of New Jersey

Abstract

The Free Software and Open Source Software movement, with Linux being the most visible example of the numerous "free software" packages, has been gathering increased attention recently. But, what are Linux and the Free Software and Open Source Movements really about? What are the advantages of the Free Software and Open Source philosophy of software creation, and is it truly "free?" These questions, and others, are addressed.

Introduction to Free and Open Source Software

The Free Software and Open Source Software movements have been getting an increased amount of attention over the last five years. Linux is without a doubt the most visible of these projects however there are thousands of other programs that are available as Free Software and Open Source Software. This paper will provide a brief history of the beginnings of the Linux operating system. It will also discuss the Open Source and Free Software movements and the differences between them. While these two terms are often used interchangeably the movements are not identical. Because of this many people (and this paper will) use the acronym F/OSS to describe something the can be described as both Free Software and Open Source Software.

A Brief History of Linux

Unix

Before discussing the history of Linux, we must first look at the history of its fore-runner, Unix. Unix was invented in 1969 by Dennis Ritchie, Ken Thompson, and others at Bell Labs (Ritchie, 1996). In the early days of Unix, AT&T licensed it for a nominal charge to universities and research labs. One reason for this is that they were not allowed to get into the computer business because of their telephone monopoly. One of the licensees of Unix was the University of California, Berkeley. Programmers at Berkeley made numerous improvements and additions to Unix and offered these to others as the Berkeley System Distribution (BSD). The license permitted redistribution with (or without) source code. However, at the time BSD still had some of the AT&T source code in it and when AT&T was broken up in 1984, Unix became a commercial product called

AT&T Unix. This product was expensive and didn't permit redistribution. Eventually all of the original AT&T source code was removed from BSD and there are now a number of commercial and F/OSS versions of BSD available. Three of the most popular F/OSS versions are FreeBSD, OpenBSD, and NetBSD.

GNU Project

In the early 1980's Richard Stallman, then a programmer at Massachusetts Institute of Technology's Artificial Intelligence Lab, was becoming increasingly upset with reluctance of software companies and other programmers to share their source code. He believed that not having access to the source code and not being able to improve it and not being permitted to share those improvements was not only hampering his ability to do his job, but it was also immoral. These factors led Stallman to decide to create his own free operating system that would allow people to view, modify, and redistribute the source code. On September 27, 1983 he announced his project which he named the GNU Project. GNU stands for GNU's Not Unix. In order to ensure that this system would remain free, Stallman created the GNU General Public License (GPL) to distribute the GNU Project under. Stallman created the Free Software Foundation to advance the idea of Free Software and the GNU Project. While there are other Open Source Software licenses, the GPL is still the most popular. This is evident by over 65 percent of the programs indexed on Freshmeat being licensed under the terms of the GPL (O'Hara, 2003). Freshmeat is a web site that maintains an index of Linux, Unix, and other software projects.

Linux

While Stallman and others were working on the GNU project a number of small operating systems were being used to teach computer science students in universities around the world about programing and operating systems. One of these was a small Unix clone called Minix. Minix was inexpensive and source code was available but it was not F/OSS. Linus Torvalds, a Finish computer science student, decided that he would make his own small Unix clone and make it available to others for free. At the time he never thought it would amount to much. In late 1991 Torvalds released the first version of the kernel of his operating system which became known as Linux. The kernel is part of a complete operating system that typically remains in memory and provides fundamental services, including the allocation of memory, scheduling of tasks, and sometimes interfaces with crucial hardware devices such as hard drives (Pountain, 2003). One way to think of the kernel is to look at it as the foundation of a building. Although Torvalds never thought Linux would become widely used when he first started working on it, more and more programmers began to use Linux and contribute improvements back into the project. The Linux kernel, however, would not be very useful without other programs that are able to run on top of it. Luckily for Linux, the Linux kernel was able to run software written for the GNU Project. Since much of the software that comes with a Linux distribution is either part of the GNU Project or is licensed under terms of the GPL many people believe that when referring to the whole operating system, one should use the term GNU/Linux to acknowledge that the efforts of the GNU project in creating the whole operating system.

When GNU/Linux was still new many Unix users and programmers looked down on it as nothing more then a hobbyist's operating system. As it became more mature, gained increased functionality, and became compliant with the POSIX standard it grew in stature. Today it is widely accepted as a commercial grade operating system and is comparable in functionality, stability, and usability with the best operating systems of available under any license.

Distributions

When most people install GNU/Linux, they do not install it from scratch. Most people install what is called a Linux Distribution. A Linux distribution is a collection of programs that includes the Linux kernel, numerous GNU, and other F/OSS programs that are packaged to work together. Typically a Linux distribution is designed to be installed easily. Distributions can either be put together by commercial companies or by groups of volunteers. Some common distributions include Redhat Enterprise Linux, Fedora Core, Mandrake Linux, Novell's SuSE, Debain GNU/Linux, Yellow Dog Linux, Slackware Linux, and many others. There are even some distributions of Linux, including Knoppix and Adios, that operate directly from a CD-ROM and do not need to be installed on a hard drive. These CD-ROM based distributions are an easy way for someone to become familiar with GNU/Linux without having to install it on their computer.

Free Software versus Open Source Software

The term Free Software is typically used to refer to software licensed under the GPL. The term is often used interchangeably with the term Open Source Software. Open Source Software and Free Software are closely related but they are not exactly the same

thing. Free Software is actually a subset of the broader Open Source Software.

The term Open Source is championed by the Open Source Initiative (OSI). In order for a piece of software to be OSI certified as Open Source Software the license terms must meet the terms of the Open Source Definition defined by the OSI. There are ten criteria listed in the Open Source Definition including Open Source Initiative, 2005):

1. Free redistribution
2. Access to the source code
3. The right to make derived works
4. The integrity of the author's source code should be protected
5. There should be no discrimination against persons or groups
6. There should be no discrimination against fields of endeavor (such as limiting the program to educational use)
7. The license must be redistributable
8. The license must not be specific to a product
9. The license must not place restrictions on other software that is distributed with it, and
10. The license must be technology-neutral
11.

For the full Open Source Definition with an explanation of the rationale behind each of these criteria, see Appendix A.

One of the reasons why some people prefer the term Open Source Software to the

term Free Software is that the term “free” has multiple meanings and this can confuse some people. While Free Software is usually free of cost, the term “free” in Free Software refers to free as in free speech instead of free as in free beer. Free Software is the term most associated with the Free Software Foundation and the GPL. The GPL is one of approximately 60 software licenses that are OSI certified. There are four kinds of freedom that the GPL is designed to protect. These freedoms are:

1. The freedom to run any the program for any purpose
2. The freedom to study the program, and to modify it (this requires access to the source code)
3. The freedom to redistribute the program to your neighbor, and
4. The freedom to improve the program and release your improvements to the public (this also requires access to the source code).

The GPL uses something called “copyleft” to ensure that these freedoms are protected not only in the original version of the program, but are also required for all modified versions of the program as well. Copyleft is actually a form of copyright. The program is copyrighted, but is distributed under terms that protect the freedoms of future users. Appendix B contains the full text of the GPL (Free Software Foundation, 1991).

Many of the goals of the Open Source movement and the Free Software movement are the same. For example, the major goal of both movements is to make software free of intellectual property restrictions. However they take very different

approaches to get there. According to Richard Stallman (2002) the Free Software and Open Source movements "disagree on basic principles but agree more or less on the practical recommendations." The Free Software approaches its goals from a more ideological point of view and they base there arguments on moral and ethical reasons. Software should be free so you can help your neighbor. Stallman (2002a) has even argued the proprietary software is harmful to society.

Eric Raymond is the author of "The Cathedral and the Bazaar" - a book describing the F/OSS development process - and one of the founders of the OSI. Raymond (1998) argues that because of the ambiguous nature of the term "free" and because the term would make "a lot of corporate types nervous" argued that the term "Open Source" was more desirable. The Open Source movement takes a more pragmatic approach and bases their arguments on the technical and economic reasons for making source code freely available.

Samples of F/OSS Programs

While GNU/Linux is the most visible F/OSS program there are thousands of other programs available. These programs range from text editors and web servers, to full office suites and web browsers. Some of the most popular programs include the Firefox web browser, the OpenOffice.org office suite, the Apache web server that most of the worlds web sites, and The Gimp graphics editing program. While GNU/Linux may not be able to run your favorite proprietary program, chances are that you can find a similar (or possibly even better) application that will meat your needs.

Advantages of F/OSS over proprietary software

Costs

There are a number of advantages to the use of F/OSS over proprietary software. The first, and most obvious one, is costs. For example, a copy of Microsoft Office 2003 Standard Edition was being sold for the price of $399.99 from CompUSA.com on February, 28, 2005. Compare that to OpenOffice.org's office suite that offers the same types of functionality and available is for free. While every situation is different and should be evaluated separately, many studies and real world experience has also shown that not only are the initial costs cheaper, the total cost of ownership is lower (Margulius, 2003; Sisk, 2003; Total cost of ownership for Linux, 2002).

Source Code and Reliability

The availability of the source code allows you to modify the program or hire a programmer to modify it for you. This is something that closed source programs normally will not allow. Even if you never touch the source code you can still benefit from it being available. Because the source code is available many people around the world can look at it, improve it, and redistribute these improved version to others, including you. This means that bugs can be fixed quicker and makes the programs more reliable. According to the Open Source Initiative (2005a) “Open source promotes reliability and quality by supporting independent peer review and rapid evolution of source code.” A recent study by code-analysis company Coverity supported this claim when it found that the GNU/Linux operating system had less the 20 percent of the amount of bugs then a typical commercial program of similar size (Lemos, 2004).

Security

F/OSS is also by nature more secure. Because programmers can look at the source code they can look for security holes, and then work to fix them. While measuring security is often a difficult task people are no doubt familiar with the problem of viruses and spyware that effects Microsoft Windows and Internet Explorer at a much larger rate then other operating systems and open source web browsers.

Support

Another advantage of F/OSS is support. With proprietary software you are often stuck with whatever level of support the vendor will give you. With F/OSS, because the code is freely available, if you are unhappy with the support you are getting you can always go to another support provider. There are many F/OSS consultants that are willing to provide support for a reasonable price. You are also in a better position to support yourself with F/OSS. This helps to avoid vendor lock-in. If you fell more comfortable with a large company providing support there are a number of such companies that can provide that support include IBM, Novell, Sun Microsystems, and Red Hat. If you want to support yourself there are many mailing lists, newsgroups, and websites that GNU/Linux users can go to if they need help with a problem. There are also local Linux users groups made up of Linux enthusiasts that are more then willing to help someone when they have questions about GNU/Linux.

Conclusions

The GNU/Linux operating system and Free Software and Open Source movements are intrinsically linked to each other. GNU/Linux is a Unix like operating

system that started as a hobbyist's clone of Unix. In a relatively short period of time GNU/Linux has grown up to be a popular operating system for servers and desktops alike. Free Software does not refer just to the price, but instead it refers to the freedom of the computer user to run, modify, improve, and redistribute software without having to pay a fee or ask for permission.

There are literally thousands of Free Software and Open Source programs to choose from. The Free Software and Open Source development model has many advantages to both programmers and businesses. Besides the freedom that using Open Source and Free Software provides other advantages to using it include better security, lower cost, greater flexibility, and greater reliability.

References

Free Software Foundation (1991). GNU General Public License, Version 2. Retrieved February 27, 2005 from http://www.gnu.org/licenses/gpl.txt

Lemos, R. (2004, December 13). Security research suggest Linux has fewer flaws. *CNET News.com*. Retrieved February 21, 2005 from http://news.com.com/Security+research+suggests+Linux+has+fewer+flaws/2100-1002_3-5489804.html

Margulius, D. L. (2003). The real cost of Linux. *Infoworld , 25*(35), 36-43.

O'Hara, K. J. & Kay, S. J. (2003). Open source software and computer science education. *Journal of Computing Sciences in Colleges, 18*(3), 1-7.

Open Source Initiative (2005). The Open Source definition. Retrieved February 27, 2005 from http://opensource.org/advocacy/faq.php

Open Source Initiative (2005a). Open Source Initiative Frequently Asked Questions. Retrieved February 27, 2005 from http://opensource.org/advocacy/faq.php.

Pountain, D. (2003). *The Penguin concise dictionary of computing*. New York, Penguin Books.

Raymond, E. S. (1998). Goodbye, free software; hello open source. Retrieved February 27, 2005 from http://www.catb.org/~esr/open-source.html

Ritchie, D. (1996). The evolution of the Unix time-sharing system. In P. Laplante (Ed.), *Great papers in computer science* (pp.705-717).

Sisk. M. (2003). The Linux revolution. *USBanker, 113*(9), 24.

Stallman, R. (2002). Why free software is better then open source. *Free Software, free society: Selected essays by Richard M. Stallman* (pp. 55-61). Boston: Free Software Foundation.

Stallman, R. (2002a). Why software should be free. *Free Software, free society: Selected essays by Richard M. Stallman* (pp. 45-49). Boston: Free Software Foundation.

Total cost of ownership for Linux in the enterprise (2002, July). Westport, CT.: Robert Frances Group. Retrieved February 24, 2005 from http://www-1.ibm.com/linux/RFG-LinuxTCO-vFINAL-Jul2002.pdf

Appendix A

The Open Source Definition

Version 1.9

The indented, italicized sections below appear as annotations to the Open Source Definition (OSD) and are not a part of the OSD.

Introduction

Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:

1. Free Redistribution

The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

> *Rationale: By constraining the license to require free redistribution, we eliminate the temptation to throw away many long-term gains in order to make a few short-term sales dollars. If we didn't do this, there would be lots of pressure for cooperators to defect.*

2. Source Code

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost–preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

> *Rationale: We require access to un-obfuscated source code because you can't evolve programs without modifying them. Since our purpose is to make evolution easy, we require that modification be made easy.*

3. Derived Works

The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

> *Rationale: The mere ability to read source isn't enough to support independent peer review and rapid evolutionary selection. For rapid evolution to happen, people need to be able to experiment with and redistribute modifications.*

4. Integrity of The Author's Source Code

The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

> *Rationale: Encouraging lots of improvement is a good thing, but users have a right to know who is responsible for the software they are using. Authors and maintainers have reciprocal right to know what they're being asked to support and protect their reputations.*
>
> *Accordingly, an open-source license must guarantee that source be readily available, but may require that it be distributed as pristine base sources plus patches. In this way, "unofficial" changes can be made available but readily distinguished from the base source.*

5. No Discrimination Against Persons or Groups

The license must not discriminate against any person or group of persons.

> *Rationale: In order to get the maximum benefit from the process, the maximum diversity of persons and groups should be equally eligible to contribute to open sources. Therefore we forbid any open-source license from locking anybody out of the process.*

Some countries, including the United States, have export restrictions for certain types of software. An OSD-conformant license may warn licensees of applicable restrictions and remind them that they are obliged to obey the law; however, it may not incorporate such restrictions itself.

6. No Discrimination Against Fields of Endeavor

The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.

Rationale: The major intention of this clause is to prohibit license traps that prevent open source from being used commercially. We want commercial users to join our community, not feel excluded from it.

7. Distribution of License

The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

Rationale: This clause is intended to forbid closing up software by indirect means such as requiring a non-disclosure agreement.

8. License Must Not Be Specific to a Product

The rights attached to the program must not depend on the program's being part of a

particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.

Rationale: This clause forecloses yet another class of license traps.

9. License Must Not Restrict Other Software

The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.

Rationale: Distributors of open-source software have the right to make their own choices about their own software.

Yes, the GPL is conformant with this requirement. Software linked with GPLed libraries only inherits the GPL if it forms a single work, not any software with which they are merely distributed.

10. License Must Be Technology-Neutral

No provision of the license may be predicated on any individual technology or style of interface.

Rationale: This provision is aimed specifically at licenses which require an

explicit gesture of assent in order to establish a contract between licensor and licensee. Provisions mandating so-called "click-wrap" may conflict with important methods of software distribution such as FTP download, CD-ROM anthologies, and web mirroring; such provisions may also hinder code re-use. Conformant licenses must allow for the possibility that (a) redistribution of the software will take place over non-Web channels that do not support click-wrapping of the download, and that (b) the covered code (or re-used portions of covered code) may run in a non-GUI environment that cannot support popup dialogues.

Origins: Bruce Perens wrote the first draft of this document as "The Debian Free Software Guidelines", and refined it using the comments of the Debian developers in a month-long e-mail conference in June, 1997. He removed the Debian-specific references from the document to create the "Open Source Definition."

Copyright © 2005 by the Open Source Initiative

Appendix B

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

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

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to

certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at

your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

> a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
>
> b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
>
> c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable

sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

> a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
>
> b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete

> machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
>
> c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify,

sublicense or distribute the Program 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.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the

Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the

General Public 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.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES 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. THE

ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and

each file should have at least the "copyright" line and a pointer to where the full notice is found.

> <one line to give the program's name and a brief idea of what it does.>
> Copyright (C) <year> <name of author>
> This program is free software; you can redistribute it and/or modify
> it under the terms of the GNU General Public License as published by
> the Free Software Foundation; either version 2 of the License, or
> (at your option) any later version.
> This program is distributed in the hope that it will be useful, but
> WITHOUT ANY WARRANTY; without even the implied warranty of
> MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
> See the GNU General Public License for more details.
> You should have received a copy of the GNU General Public License
> along with this program; if not, write to the Free Software Foundation,
> Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
> Also add information on how to contact you by electronic and paper mail.
> If the program is interactive, make it output a short notice like this
> when it starts in an interactive mode:
> Gnomovision version 69, Copyright (C) year name of author
> Gnomovision comes with ABSOLUTELY NO WARRANTY; for details
> type `show w'.
> This is free software, and you are welcome to redistribute it under certain

conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

> Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.
>
> <signature of Ty Coon>, 1 April 1989
>
> Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.