Cygnus Support
814 University Ave
Palo Alto, CA 94301
+1 415 322 3811
FAX 415 326 1669

June 17, 1990

Simson Garfinkel
52½ Pleasant Street,
Cambridge, MA, 02139

Dear Simson:

Here's a sample of our current sales literature. I hope you find it helpful. Here comes the canned letter:

We're glad to hear of your interest in commercially supported free software. We've expanded our offerings for smaller sites. Here's our latest price sheet, company description, and product literature.

You probably already know some advantages of free software: full source code, quickly evolving products, no royalties. You may also know some of the drawbacks: easy to modify, quick to evolve, hard to support. Cygnus Support handles the hard part, leaving you with the advantages.

We believe that free software can compete head-to-head with commercial software and win. The brochure explains why, I won't bother you with it here.

Cygnus Support combines two hard-to-find things: full source and full support. If your site uses proprietary software because it's supported, we'll provide you with *better* software with equally good support. If your site already enjoys GNU software or other free software, Cygnus Support can support it, freeing your programmers to do real work.

Cygnus Support has been supporting free software for major companies and laboratories since 1989. Our initial support product, *Leveraged Support*, provides larger sites with significant cost savings while giving them access to our world-class free software developers. We have recently expanded our support offerings with *Core Support* to provide smaller sites with a package that yields similar benefits on a smaller scale.

Please call us at +1 415 322 3811 to order Cygnus Support.

Sincerely,

David Wallace
Director of Support
DW:mt

encl: Price sheet
Support description
Company overview

Company Overview

Cygnus Support
814 University Avenue
Palo Alto, CA
94301
(415) 322-3811
FAX: (415) 326-1669

March 9, 1990

1 Introduction

Cygnus Support was founded in 1989 to provide commercial support for free software. Inspired by the success of the Free Software Foundation's GNU project, we based our business on the principle that good software should be shared. To that end, we chose to provide the best software available on the UNIX platform, and to offer the one thing it lacked: commercial support.

We combine the excellent, quickly evolving GNU software with our stable, well-tested releases; guaranteed response via phone, fax, or electronic mail; improved documentation and training; and special help for critical problems. This gives our users the best of both worlds: full source code and full commercial support.

Cygnus Support has signed contracts with NASA and Sun, and we expect to sign contracts with several other companies, including Xerox and GE, during our first year of operation. We are the first company to provide commercial support for free software, and we strive to be the best.

2 Cygnus Support

GNU software has many technical advantages over its proprietary counterparts, like faster execution, fewer limitations, higher portability, and better documentation. It has these advantages because it is free software: capable, interested users have made the improvements that they themselves wanted to use.

In order to deliver these advantages to a large number of users in the commercial market, we needed to make this software "commercial quality" and deliver it with commercial support. We run our own QA on the software we support. We synchronize our release cycles with customer needs. We track bugs. And, as committed users of this software, we make improvements that we see as important.

Our support service ensures that customers' traditional needs are met without sacrificing the benefits of free software. With Cygnus Support, no one need choose inferior software simply because it is "commercial" software.

3 The Founders

Michael Tiemann began working on free software in 1987. He wrote the code for GNU C's function inlining. He wrote a portable instruction scheduler which boosted GNU C's performance by 30% on the SPARC. He is the author of GNU C++, the first available native code C++ compiler. Mr. Tiemann has ported the GNU compiler to the SPARC, Motorola 88000, and National 32032 architectures, as well as adding support for Sun's FPA board on Sun 3s. He ported the GNU debugger to the SPARC and Intel 80386 architectures, extended the debugger and linker to handle C++ features, and ported the linker to SPARC.

John Gilmore was the fifth employee at Sun Microsystems. He participated in design reviews of the instruction sets for the Motorola 68010 and 68020, and Sun SPARC, and worked on the first bringups of UNIX on 68010 and 68020 processors. More than half of his 33 years have been spent programming and designing computers.

For the past eight years Mr. Gilmore has written, collected, maintained, and distributed free software, in conjunction with the Usenet, the Sun User Group, the Free Software Foundation, and many individuals around the world. He has been regression-testing each GNU C compiler release against a large test suite since the first release. He wrote the GNU versions of the "tar" and "uucp" programs.

David Wallace has been involved in computing and Artificial Intelligence for over a decade. He has developed and supported large systems at laboratories around the world, such as Stanford CSLI and Xerox PARC in Palo Alto, the Centre Mondial Informatique in Paris, the MIT and Atari AI labs in Cambridge, and at MCC in Texas and in California.

As with Mr Gilmore, Mr Wallace's involvement in free software predates the GNU project. He was an implementor and maintainer of MACLISP and the original Emacs, which were widely used around the arpanet in the late 70s and 80s.

4 The GNU Project

The GNU project was started in 1983 by Richard Stallman to promote free software. Its aim is to write the best software in the world and release it under a license which encourages software sharing. GNU software is "copy protected" in the sense that the GNU General Public License protects everyone's right to copy it!

The project has already produced a text editor, GNU Emacs; a debugger, GDB; a compiler, GCC; a shell, Bash; and many other utilities. These tools are faster, more powerful, run on more platforms, and come with full source code. GNU Emacs now runs on virtually every machine running UNIX. Major vendors like DEC, SGI, NeXT, and IBM ship it with their machines. Some vendors also ship GDB and GCC as their primary debugger and compiler.

As the GNU project continues to develop powerful tools, Cygnus Support continues to provide innovative support

5 Summary

When choosing a software development platform, one must evaluate both its present capabilities and its future potential. The GNU tools offer not only the best development platform now, but also the most promising evolution path as well. With Cygnus Support, your leading edge designers will have the commercial support they need to use these tools to their fullest advantage.

Cygnus Support Overview

Cygnus Support
814 University Avenue
Palo Alto, CA
94301
(415) 322-3811
FAX: (415) 326-1669

February 27, 1990

Cygnus Support provides commercial support for free software. We give supported access to the best software in the Unix environment. And we provide the best support because we employ the developers.

Free software is faster, more powerful, and more portable than its proprietary counterparts. It evolves faster because users who want to make improvements are free to do so. Cygnus Support tracks these improvements and integrates them into tested, stable versions ready for commercial use. And we back this software with comprehensive support: handing your problems 45 hours/week by phone and 24 hours/day by electronic mail, with a guaranteed response within 4 business hours.

Our initial product is support for C and C++ program development for $100,000 per year. This includes installation, documentation, on-line bug fixing, and new releases. For the price of a full-time employee, you get the resources of an entire support team, freeing your employees to concentrate on your core business.

1 Free software

Although it is not a traditional option, the choice of free software is straightforward when one considers the following:

- Many of the best Unix tools are already free software.

[0] Unix is a trademark of AT&T.
SUN-3, SUN-4, and SPARC are trademarks of Sun Microsystems, Inc.

- Free software tracks and drives industry standards.
- Free software is widely used in the research community.
- Cygnus Support is intimately familiar with this software as we have extensive experience using, debugging, and in many cases *implementing* it.
- As all the source code is freely available, users and developers who wish to extend the software may do so. Free software is truly open software.
- Free software and Cygnus Support free the user to fit the software to his or her needs.

What do we mean by "best Unix tools?" Free software, especially software we support from the GNU project, tends to be faster, more powerful, and more portable than its commercial counterparts.

For instance, the GNU C compiler gcc runs on 14 different platforms; the code it generates is usually 30% faster than that produced by the vendor's compiler. It was the first Draft ANSI-compliant C compiler. It is the only Unix C compiler which lets the user debug optimized code. Any implementation can cross-compile for any other.

The Emacs editor is considered by many *the* indispensible tool of the Unix environment. This extensible editor has been programmed to be the interface for innumerable tools.

The g++ compiler was the first available native-code C++ compiler. It now implements all the features of C++ 2.0, *including* multiple inheritance. It runs on any machine that gcc suppports, and does the same optimizations. It also provides additional C++ specific optimizations.

Although each of the Cygnus Support packages addresses a specific facet of the Unix environment, they all share some characteristic advantages:

- Platform independence. All our packages run on all the platforms we support. Users may move back and forth across them at will.
- Adherence to changing standards. We are committed to the official standards processes. Past experience with POSIX and the ANSI C effort demonstrate our ability to track emerging standards as they evolve.
- Early access to standards, or "try before you buy." Experience with g++ has demonstrated that this open environment is the best in which to experiment with new extensions, if desired. Switches let *you* choose which facilities and extensions to use – or none!
- Customizability. Since we provide source with all our supported software, users are free to customize the software as they see fit.

These features are now available to those who need the reliability of supported software, through a support contract from Cygnus Support.

2 Our support commitment

By support we mean all of the following:

- new releases of system software, in both binary and source form
- phone and fax numbers and a network mailbox for bug reports and immediate assistance
- information about reported bugs and patches, and the patches themselves
- new documentation, as appropriate
- an initial installation and easy-to-install releases

We issue our own releases in order to QA each new version, and to guarantee that reported bugs do not re-appear in later versions. We offer binary as well as source for each release, since some sites want to do their own customization, while others want a version they can just load from the tape and run.

We can phase our delivery to suit customer requirements. Some sites wish always to have the most current version, as soon as it becomes available. Others wish to have wide spacing between releases, and just want work-arounds between releases. We easily accomodate both.

We offer different levels of bug assistance depending on the need and severity of the situation. Specifically, we provide a network mailbox that anyone at the site may use (we will call back, if requested), and a phone number for immediate help for use by specific site contacts. It has been our experience that most users would rather send a report by electronic mail, since most of the necessary information comes from the computer anyway. However when a user is blocked and cannot proceed, help will be only a phone call away.

We do our best to provide the customer with a work-around while we write and test a bug fix. Once the patch has been written, we can make it available either as a patch or as part of a new binary, depending on the customer's needs.

We will bring our bug-tracking system on-line in March 1990. We will provide our customers with access to this system so that they may follow the status of reported bugs.

Support also means better documentation. Some "bugs" are actually correct or ill-specified behavior which was not adequately documented. We augment the documentation or the code to reduce situations which cause problems. For instance, if error messages are obscure, or if people often get confused about the `#include` search paths, we will certainly address the problem.

We will perform the initial installation of our packages at the user site. At that time we can familiarize ourselves with the user's installation (such as path assignments and such). Subsequent distributions can then be pre-configured for the site, so that they may be extracted from tape and run, without the usual, complicated installation procedures.

3 Supported Software

This section describes the software we currently support, as well as software we plan to support in the future.

The initial machines we support are the Sun-3 and Sun-4 (SPARC).

3.1 Cygnus Developer Support

Our first product is a development package, featuring:

- the compilers gcc and g++, and the assembler gas
- the parser utilities bison and flex (for bison we provide a small parser driver so that its output can be included freely into other programs)
- binary utilities, such as the GNU binutils (e.g. ld and ar), gdb, and protoize
- the libg++ library, an upward-compatible version of the C++ runtime
- the standalone documentation programs info and texinfo.
- the GNU Emacs editor

Together, this provides a powerful standalone package for C and C++ development, which runs uniformly across a variety of platforms; there is no need to customize your code for multiple compilers. These compilers are also the only ones which offer simultaneous use of debugging and optimization (ie you may specify -g -O).

When they become available and reasonably stable, we plan to add the GNU version of libc (the C runtime library) and the freed version of libm, the Berkeley math library.

We will also refine the documentation for the binutils, but this need not be done immediately due to our projected initial customer base; we anticipate completion of this by the end of April, 1990.

3.2 Cygnus Shell Support

This package is analogous to the code development package, but for more general tasks. We plan to begin support for this package in the third quarter of 1990. Based on availability, reliability, and user feedback, we have selected the following programs for this package:

- bash, the GNU replacement for sh
- Make (GNU version)
- the GNU versions of sed, tar, egrep, vi, diff, gprof and more

- `gawk`, `perl`, and common applications written in them
- the small shell utilities, like `cat` and `mv`
- the standalone documentation programs `info` and `texinfo`.
- the GNU Emacs editor

Unlike most vendor-supplied versions, these utilities provide extended functionality (such as filename completion and context-sensitive accelerators); Posix compliance; and consistency along all our supported platforms (in other words, a common look-and-feel). Many are significantly faster and have fewer arbitrary limitations than the UNIX versions.

4 Summary

Free software will increasingly lead the industry through the 1990's. Your leading edge designers should be using these powerful tools. With Cygnus Support they *can* be, instead of wasting their time on support.

Glossary

Several terms used in this document:

`ar` A program used to create libraries of compiled functions. One of the `binutils`.

AutoCAD A Computer Aided Design program popular in the PC world.

`awk` A programming language based on pattern matching and strings. See `perl`.

`bash` A free replacement for the Bourne shell (`UNIX` command processor), with command completion, in-line editing, and many other features.

`binutils` A collection of utilities for manipulating compiled code and function libraries.

`bison` A fast parser generator. Replaces `yacc`.

C The system programming language of `UNIX`.

C++ An object-oriented programming language designed at Bell Labs to serve as a successor to C.

`cfront` A pre-processor for the C++ language sold by AT&T. `cfront` translates a C++ program into C, rather than machine code, making debugging difficult.

Compiler A program which translates a program written in a programming language such as C into something the machine can understand. The program the user types is often referred to as the *source* and the result of compilation is often referred to as the *object*.

`diff` A program for comparing differences between two (or three) files.

`egrep` A program that searches files for a text pattern (AKA a *regular expression*).

Emacs An extremely powerful text editor.

`flex` A fast lexical analyzer generator. Replaces `lex`.

The Free Software Foundation An organization devoted to advancing the cause of free software, primarily through the implementation of GNU. Several of the programs we support were written under the aegis of the FSF.

`g++` The GNU C++ compiler. Unlike `cfront`, `g++` is a native-code compiler which can also generate special information for debuggers.

`gas` The GNU Assembler. Turns low-level *source* code into code a machine can execute.

`gawk` The GNU interpreter for the `awk` language.

`gcc` The GNU C compiler.

`gdb` The GNU Debugger. Used to find errors in programs.

GNU Gnu's Not Unix.

`gperf` A perfect hash function generator.

`gprof` A profiler for binary UNIX applications. Used for finding the places where a program spends most of its time, in order to speed the program up.

`info` A hypertext documentation browsing tool. Runs either as part of Emacs or as a standalone program.

Interpreter A program which executes each statement of a program written in a programming language. Contrast this with a compiler, which does the translation all at once, for later execution.

`ISV` Independent Software Vendor (ie a third-party software developer).

`ld` The UNIX linker. Takes a variety of object files (*cf* Compiler) and links them together into a complete program.

`less` Less is `more`.

`lex` A programming language for building pattern matchers based on deterministic finite automata.

`libC` The AT&T C++ library. Provides an object-oriented input/output interface, and not a whole lot more.

`libc` The UNIX C library. Provides standard UNIX services, such as formatted input/output, string searching, data conversion, sorting routines, and an interface to the UNIX operating system.

`libg++` The GNU C++ library. A large set of C++ classes, such as arbitrary precision integers, complex numbers, stacks, heaps, vectors, queues, random numbers, strings, fixed-point numbers, self-balancing tree structures, and a `libC`-compatile input/output facility.

`libm` A library of functions for mathematics.

`make` A program for maintaining dependencies among files, in order to simplify the maintenance of a large program.

`more` A program for browsing files.

OEM Original Equipment Manufacturer. Someone who buys a machine from a vendor and then packages it under a private. Compare with VAR.

perl Another programming language similar in intent to **awk**, but with greater power.

protoize A program for helping convert K&R-style (archaic) programs into ANSI (modern) C.

Posix The specification for a portable operating system decended from **UNIX**.

sed An editor designed to be run in batch mode (ie not interactively).

tar A program for building *file archives*, files which bundle other files or directories together.

texinfo A programming language for building documentation. **texinfo** documentations can be processed by TEX, producing formatted hardcopy output, or by **info**, which provides an on-line hypertext document.

TEX A free typesetting system which is used to produce, among other things, this document.

VAR Value Added Reseller. Someone who adds some features to a software package and re-sells it. The software equivalent of OEM.

vi A primitive screen editor.

yacc A programming language for building parsers based on **LALR(1)** grammars

Cygnus Support
814 University Ave
Palo Alto, CA 94301
+1 415 322 3811
FAX 415 326 1669

Cygnus Support Price List

Support Products

June 1990

Cygnus Support currently offers two support packages at Core and Leveraged support levels.

Cygnus Developer Support

The Cygnus Developer Support includes the following items:

- The GNU C and C++ compilers `gcc` and `g++`.
- The GNU debugger `gdb`.
- The GNU assembler `gas`.
- The GNU C++ runtime library `libg++`.
- The GNU LALR(1) fast parser generator `bison`.
- The programs `gprof`, `ld`, `size`, `nm`, `strip`, `ar`, and `ranlib`.
- The GNU Emacs editor.
- The GNU documentation browser `info`.
- The GNU documentation converters `texinfo` and `texi2roff`.
- The free patch installer `patch`.

Cygnus Shell Support

The Cygnus Shell Support package contains the following items:

- `bash`, the GNU replacement for `sh` and `ksh`.
- The GNU version of `make`.
- `gawk`, `perl`, and common applications written in them.
- The small shell utilities, like `cat` and `mv`.
- The GNU versions of `sed`, `tar`, `egrep`, `vi`, `diff`, `gprof` and `more`.
- The GNU Emacs editor.
- The GNU documentation browser `info`.
- The GNU documentation converters `texinfo` and `texi2roff`.
- The free patch installer `patch`.

The following is a succinct summary of the differences between the various levels of support:

Service	Core Support	Leveraged Support	*à la carte*
Vintage releases	included	included	$1000
Minor releases	$1500	included	$2500
Response time	24 hours	4 hours	N/A
Network patches	free	free	$5/minute
Network access	free	free	$5/minute
Quarterly Newsletter	included	included	$20
Installation	$5000	included	N/A
Validation	$10K	included	N/A
Ports	*à la carte*	one included	$50K and up
Customization	*à la carte*	included	negotiable
Support of Add'l Tools	negotiable	available	N/A
Consulting	negotiable	available	negotiable

Prices for our support packages (applicable to a single site):

Package	Core	Leveraged
Developer's Kit	$25K	$100K
Shell Support	$25K	$100K

A discount is available when purchasing both packages, and when purchasing packages for multiple sites.

Support is available on the following platforms:

Vendor	Model(s)	Operating system
Sun Microsystems	SUN-3, Sparc	SunOS 4.0.3 or later.
Silicon Graphics	IRIS-4D	IRIX SVR3.2
Digital Equipment	VAX	Ultrix and BSD 4.x

Cross-compilation is supported from any of these platforms, targeting machines based on the 680x0, SPARC, Vax, 80386, and 80486. Contact Cygnus if you want support for other host and/or target configurations.

Cygnus Support brings together the benefits of superior software, technical expertise, on-line documentation and commercial support. Our support package gives your advanced developers coherent, consistent sets of tools that they can use to meet your business objectives.

Call Cygnus Support today at +1 415 322 3811 and find out how we can put the power of free software into your hands.

Cygnus Support
814 University Ave
Palo Alto, CA 94301
+1 415 322 3811
FAX 415 326 1669

Company Overview

Cygnus Support

June 1990

Cygnus Support was founded in 1989 to provide commercial support for free software. Its founders — Michael Tiemann, David Wallace, and John Gilmore — began using free software because of its outstanding features and superior quality, but found that maintaining it diverted too much time from their real work. Other users duplicated this support effort, or chose commercial software which was not as powerful, efficient, or useful. Cygnus Support was founded to solve this dilemma by providing centralized, cost-effective support. Using Cygnus Support, users who did their own support can recover valuable staff time. Users of proprietary software now have the choice of using *supported* free software.

Free software is distributed as source code, and users can copy it, distribute it, and modify it as desired. This gives free software extremely wide distribution and use compared to commercial software, creating a large support market. It also causes free software to evolve more quickly than commercial software, since more people work on it simultaneously, adding the features they like and fixing the bugs they find. Centralized support is necessary in such an environment to merge each user's changes and provide quality assurance.

Our products combine the excellent, quickly evolving GNU software with stable, well-tested releases; guaranteed response via phone, fax, or electronic mail; improved documentation and training; and special help for critical problems. This gives our users the best of both worlds: full source code and full commercial quality software.

As a result of our technical experience and unique software product, Cygnus Support has won and continues to win major contracts with leading companies and research labs.

Our products include a C compiler, a C++ compiler, a source-level debugger, a portable assembler, object-code manipulating programs, subroutine libraries, an extensible self-documenting text editor and user interface (Emacs), a variety of text searching and processing tools. and a command interpreter.

Our C compiler, gcc, compiles code more over a dozen different instruction sets. It was the first compiler to implement the ANSI C standard, and is the only compiler that permits the debugging of optimized programs. It runs quickly, and produces code that also runs quickly.

Our C++ compiler, g++, was the first true compiler for C++, producing assembler code rather than translating C++ into C. It implements all the features of AT&T C++ 2.0. It runs on any machine that gcc supports, and does the same optimizations. It also provides additional C++ specific optimizations.

Key Personnel

Michael Tiemann, President, has been writing free software since 1987. He wrote the code for GNU C's function inlining. He wrote a portable instruction scheduler which boosted GNU C's performance by 30% on the SPARC. He is the author of GNU C++, the first available native code C++ compiler. Mr. Tiemann has ported the GNU compiler to the SPARC, Motorola 88000, and National 32032 architectures, as well as adding support for Sun's FPA board on Sun 3s. He ported the GNU debugger to the SPARC and Intel 80386 architectures, extended the debugger and linker to handle C++ features, and ported the linker to SPARC.

He developed the initial version of GNU C++ as core technology for a project at MCC. He has since developed free software at INRIA (French National Institute of Computer Science, Paris), Stanford University, and Sun Microsystems. He frequently gives talks about free software, and has been instrumental in helping to establish commercial acceptance of free software.

John Gilmore, Generalist, was the fifth employee at Sun Microsystems. He participated in design reviews of the instruction sets for the Motorola 68010 and 68020, and Sun SPARC, and worked on the first bringups of UNIX on 68010 and 68020 processors. He produced twenty stable releases of bootstrap firmware and diagnostics for the Sun-1, Sun-2, and Sun-3 over four years. More than half of his 34 years have been spent programming and designing computers.

For the past eight years Mr. Gilmore has written, collected, maintained, and distributed free software, in conjunction with the Usenet, the Sun User Group, the Free Software Foundation, and many individuals around the world. He has been regression-testing each GNU C compiler release against a large test suite since the first release. He wrote the GNU versions of the "tar" and "uucp" programs.

David Wallace, Director of Support, has been involved in computing and Artificial Intelligence for over a decade. He has developed and supported large systems at laboratories around the world, such as Stanford CSLI and Xerox PARC in Palo Alto, the Centre Mondial Informatique in Paris, the MIT and Atari AI labs in Cambridge, and most recently MCC in Texas and in California. At MCC he was not only the primary implementor of the CYC inference system, but for the two years before joining Cygnus he also produced and supported quarterly CYC releases.

As with Mr Gilmore, Mr Wallace's involvement in free software predates the GNU project. At MIT he worked extensively as maintainer and as an implementor of new features for MACLISP and the original Emacs, which were widely used around the ARPAnet in the late 70s and 80s. After leaving MIT for MCC he continued to contribute and maintain a number of widely-used and free utilities for the AI and symbolic computing community.

Cygnus Support
814 University Ave
Palo Alto, CA 94301
+1 415 322 3811
FAX 415 326 1669

Cygnus Support Overview

Support Products

June 1990

Cygnus Support provides commercial support for free software. We give users supported access to the best software in the Unix environment. We also give them access to the services of some of the most qualified support people: the authors of that software.

Our supported free software is faster, more powerful, and more portable than its proprietary counterparts. It evolves faster because users who want to make improvements are free to do so. Cygnus Support tracks these improvements and integrates them into tested, stable versions ready for commercial use. And we back this software with comprehensive support: handling your problems 45 hours/week by phone and 24 hours/day by electronic mail, with guaranteed response time.

Our initial products support tools for C and C++ program development, and support the shell and text-handling utilities. Please see our price list for details and current prices.

This document explains how hiring Cygnus Support benefits advanced development groups by allowing them to *use* state-of-the-art tools instead of maintaining them.

1 Free software

Free software is becoming the solution of choice in the Unix environment, as:

- Many of the best Unix tools are free software.
- Free software tracks and drives industry standards.
- Free software is widely used by the technical leaders in the research community.
- Free software is truly open software. All the source code is freely available. Users and developers are free to extend the software or to have others do so for them.

But better software is not enough; with Cygnus Support as your partner, you will have the software and the support you need to better meet your business objectives. Cygnus is intimately familiar with this software: its personnel have extensive experience using, debugging, and in many cases *implementing* it. Hence we can deliver timely and cost-effective solutions that meet your needs.

Free software's widespread availability has made it pre-eminent in two ways. First, it, tends to be faster, more powerful, and more portable than its proprietary counterparts. Second, new features are constantly being implemented and refined; not only is the code highly customizable, but Cygnus Support may already have the custom features you require.

For instance, the GNU C compiler gcc runs on 14 different platforms. As a cross compiler it can generate code for any of them and for several additional platforms as well. The code it generates is usually 30% faster than that produced by the vendor's compiler. It was the first Draft ANSI-compliant C compiler. It is the only Unix C compiler which lets the user debug optimized code.

The Emacs editor is considered by many *the* indispensable tool of the Unix environment. This extensible text editor has been programmed to be the interface for innumerable tools.

The g++ compiler was the first available native-code C++ compiler. It now implements all the features of AT&T C++ 2.0. It runs on any machine that gcc supports, and does the same optimizations. It also provides additional C++ specific optimizations.

Although each of the Cygnus Support packages addresses a specific facet of the Unix environment, all share some characteristic advantages:

- Platform independence. All packages run on all supported platforms, providing a consistent base in a heterogeneous computing environment. Investments in software are protected as hardware improves and as new vendors offer new machines.
- Adherence to changing standards. We are committed to the official standards processes. Past experience with Posix and the ANSI C effort demonstrates our ability to track emerging standards as they evolve.
- Early access to standards. Experience with g++ has demonstrated that this open environment is the best in which to experiment with new extensions. Switches let the user select which facilities and extensions to use. All extensions may be disabled for maximum compatibility.
- Customizability. Since we provide source code with all supported software, users are free to customize the software as they see fit.

These features will provide you with a coherent, powerful and reliable set of tools to maximize your staff's programming productivity.

2 Our support

By support we mean all of the following:

- new releases of system software in both binary and source form
- phone and fax access and a network mailbox for bug reports and immediate assistance
- information about reported bugs and patches, and the patches themselves
- new documentation (including full documentation sources) as appropriate
- easy-to-install releases

We issue our own releases in order to assure the quality of each new version and to guarantee that reported bugs do not re-appear in later versions. We offer binary as well as source for each release since a particular site may wish to perform its own customization while another may want a version that can simply be loaded and run.

We adjust delivery to suit customer requirements. Some sites require only the most current version as soon as it becomes available. Others prefer to wait a period of time between releases, using temporary work-around patches in the meantime. We can easily accommodate both. Our *vintage releases* provide the stability and reliability needed by our customers who update less frequently, while our *minor releases* make up-to-the-minute features available in a tested, though not fully proven, release.

We offer different levels of bug assistance depending on the need and severity of the situation. Specifically, we provide a network mailbox that anyone at the site may use. We will call back if requested. We also provide a phone number that can be used by specific personnel when immediate assistance is required. It has been our experience that most users would rather send a report by electronic mail. However when a user is blocked and cannot proceed, help is only a phone call away.

We do our best to provide our customers with a work-around while we write and test a bug fix. Once the patch has been written, we make it available either as a patch or as part of a new binary depending on the customer's needs.

We provide good documentation and keep it up to date. When a "bug" turns out to be the result of ill-specified behavior, or even correct behavior which was not adequately documented, we fix the documentation or the code, whichever is appropriate. For instance, if an error message is obscure, or if people often get confused about a particular feature, we improve the program's response to the situation.

3 Supported Software

This section describes the software we currently support, as well as software we plan to support in the future. We offer two packages geared towards advanced users and developers. Both packages run on all supported platforms.

The initial machines we support are the Sun-3 and Sun-4 (SPARC) from Sun Microsystems, the IRIS 4D from Silicon Graphics, and the VAX from Digital Equipment Corp (running Unix). Cross-compilation is supported from any of these platforms, targeting machines based on the 680x0, SPARC, Vax, 80386, and 80486.

As the range of supported platforms is increasing, please contact us if you have further hardware requirements.

3.1 Cygnus Developer Support

The Cygnus Developer Support includes the following items:

- The GNU C and C++ compilers `gcc` and `g++`.
- The GNU debugger `gdb`.
- The GNU assembler `gas`.
- The GNU C++ runtime library `libg++`.
- The GNU LALR(1) fast parser generator `bison`.
- The programs `gprof`, `ld`, `size`, `nm`, `strip`, `ar`, and `ranlib`.
- The GNU Emacs editor.
- The GNU documentation browser `info`.
- The GNU documentation converters `texinfo` and `texi2roff`.
- The free patch installer `patch`.

Together, these constitute a powerful standalone package for C and C++ development. Because it runs uniformly across a variety of platforms, there is no need to customize your code for multiple compilers. These compilers are also the only ones which offer simultaneous use of debugging and optimization (i.e., one may specify `-g -O`).

When they become available and reasonably stable, we plan to add the GNU version of `libc` (the C runtime library) and the free version of `libm`, the Berkeley math library.

3.2 Cygnus Shell Support

The Cygnus Shell Support package contains the following items:

- `bash`, the GNU replacement for `sh` and `ksh`.
- The GNU version of `make`.
- `gawk`, `perl`, and common applications written in them.
- The small shell utilities, like `cat` and `mv`.
- The GNU versions of `sed`, `tar`, `egrep`, `vi`, `diff`, `gprof` and `more`.
- The GNU Emacs editor.
- The GNU documentation browser `info`.
- The GNU documentation converters `texinfo` and `texi2roff`.
- The free patch installer `patch`.

This package is analogous to the code development package but is designed for more general use.

Unlike most vendor-supplied versions, these utilities provide extended functionality (such as filename completion and context-sensitive accelerators), Posix compliance, and consistency along all our supported platforms (in other words, a common look-and-feel). Many are significantly faster and have fewer arbitrary limitations than the UNIX versions.

4 Support Levels

Cygnus Support offers two levels of support: the Core Support level for small development teams who need basic, dependable support services on existing platforms, and the Leveraged Support level for larger sites and for development teams who want to integrate our software with their new environments or platforms. We also offer support *à la carte,* so users can tailor our support to fit their exact needs.

4.1 Core Support

The Cygnus Core Support level is for small development teams using industry-standard platforms. Core Support provides the advantages of free software while retaining full commercial support availability. We provide free updates with each new vintage release, 24 hour response time in acknowledging receipt of customer bug reports, free patches available over the network, and a discount on the price of our minor release tapes. We also provide a quarterly newsletter that reports the progress of our software and the new problems we undertake.

Our supported free software comes not only with full sources to the programs, but full sources to their documentation. From these, users can generate as many copies of the documentation as they need. A hypertext interface (available within Emacs or with the stand-alone `info` program) makes on-line browsing of these documents a snap.

4.2 Leveraged Support

The Cygnus Leveraged Support level provides the advantages of standard supported free software in your custom environment. We also upgrade our service to provide higher speed response to your calls.

Core Support is fine for small organizations using off-the-shelf computers. But if your business is building new computing environments, or managing massive or widespread development projects, integrating our excellent compilers, debuggers, and other tools involves more than simple bug fixes. With Leveraged Support, we'll do the development and integration needed to seamlessly fit our high performance tools into your environment.

If your company pursues many different projects, various project teams could individually buy Core Support. In some cases, it is better for the entire site to buy Leveraged Support. Besides the faster response time and the improved coordination of your development environments, Leveraged Support lets you specify major improvements to be made in our tools, which will fit them more closely to your needs. Because our improvements are also free software, you can distribute them widely within your organization, or to your customers, without extra cost. Cygnus Leveraged Support thus becomes part of your development team, handling the details of the tools that we know best, leaving you free to concentrate on your project's ultimate objective.

Our Leveraged Support serves all people within the site with the highest level of reliability, providing users with a solid foundation upon which to build. To deliver at this level, we offer these additional services:

Fast Response Our Core Support response time of 24 hours is cut to 4 hours, resulting in very rapid response. We send patches and bug fixes on tape or floppy (at no extra charge) by next-day air service when requested.

Documentation We provide printed documentation, in addition to its online source form, as an easy reference when online use is inconvenient.

Installation At the customer's request we perform the initial installation of our packages at your site. Subsequent distributions can then be pre-configured for the site, so that they may be extracted from tape and run without the usual, more complicated, installation procedures. Our personnel can help you lay out shared source and binary trees for maximum consistency across platforms on your network.

Validation When it comes time for a new release (either a vintage release or a minor release), we test against a large internal test suite. As a Leveraged Support customer, we invite you to supply us with your own internal test suites to ensure that your people will be able to run without interruption when upgrading to the new release. We keep all such test cases confidential.

On-line Services We provide access to our networked machines so that you can access our on-line bug tracking system for the latest information about your bug reports and the bug reports of others. Access via internet or high-speed modem gives you up-to-the minute status of all the supported software.

Training Special discounts are available for our training seminars.

Development Cygnus Support will provide you with custom development work on a resource-available basis. We will commit to quarterly work plans delivering new features and improvements of specific interest to you. This is the perfect opportunity to shape the evolution of free software to serve your exact needs.

Porting We will port our software from machines we currently support to others using the same processors as our resources become available. As vendors alter their own software, it becomes more difficult to simply recompile a program to run on many machines. We will identify these portability problems and evolve our software to cope with them. Our goal is to make all your machines more useful.

Retargeting We will retarget our compilers and development tools to generate code for your new processors or new object file formats. Since this is a major undertaking, support for retargeting must be negotiated as part of the Leveraged Support contract.

Future Packages We offer you special support for some software not yet "officially" supported. In so doing, we help meet our Leveraged Support customer's needs while stepping up the quality and availability of additional free software. Once this software is made available as a regularly supported product, we offer you discounts for its support.

For sites that need this high-powered support, we provide a strong complement to your internal support organization. We work with your organization to fix problems and provide updates according to your preferred update policy. With Cygnus Support, your site will see consistent, up-to-date free software that provides the full benefits that free software has to offer.

4.3 *A La Carte* Support

For Core Support users who wish only a few of the services we offer to Leveraged Support customers, we offer an *à la carte* plan:

Minor Releases We make a variety of releases of our software. Most such releases are minor releases, which may include bug fixes and new features, but which have not been burned-in with extensive use in the field. Occasionally we will designate a minor release as a vintage release, after it has proven itself to be stable in actual use at customer sites. Core Support customers receive vintage releases for free, and may choose to buy minor releases.

Consulting We provide consulting services to extend our free software to meet specific customer needs. Our consultants are able to deliver results quickly and reliably. All our consulting work comes with free support for 90 days.

Access to the Cygnus Network We provide access to non-Leveraged Support customers via internet or modem line (up to 19.2Kbps). Users can browse or download our bug-tracking information or our software. Users are billed for total connect time.

Training and Seminars We provide training in a number of areas, including C++ programming, Emacs customization, advanced object-oriented programming techniques, and new compiler or debugger features. Our training and consulting services can provide you with the education and the experience you need to get the job done.

Ports and Customization The staff of Cygnus Support has already performed a number of ports of the software we now support. We are always ready to do more. We will port any or all of the tools we support to new platforms and we will optimize or customize these tools on platforms to which they have already been ported.

Support of Additional Tools Our Core Support level provides support for the tools that advanced programmers need to do high-powered C and C++ development. There is a great deal of additional free software which addresses many other needs. We can provide the support you need in these additional areas.

Please see our current price sheet for more information.

5 Summary

Cygnus Support brings together the benefits of superior software, technical expertise, on-line documentation and commercial support. Our support package gives your advanced developers coherent, consistent sets of tools that they can use to meet your business objectives.

Call Cygnus Support today at +1 415 322 3811 and find out how we can put the power of free software into your hands.

Glossary

Several terms used in this document:

`ar` A program used to create libraries of compiled functions. One of the `binutils`.

`awk` A programming language based on pattern matching and strings. See `perl`.

`bash` A free replacement for the Bourne shell (`UNIX` command processor), with command completion, in-line editing, and many other features.

`binutils` A collection of utilities for manipulating compiled code and function libraries.

`bison` A fast parser generator. Replaces `yacc`.

C The system programming language of `UNIX`.

C++ An object-oriented programming language designed at Bell Labs to succeed C.

`cfront` A pre-processor for the C++ language sold by AT&T. `cfront` translates a C++ program into C, rather than into machine code, making debugging difficult.

Compiler A program which translates a program written in a programming language such as C into something the machine can understand. The program the user types is often referred to as the *source* and the result of compilation is often referred to as the *object*.

`diff` A program for comparing differences between two (or three) files.

`egrep` A program that searches files for a text pattern (AKA a *regular expression*).

Emacs An extremely powerful text editor.

`flex` A fast lexical analyzer generator. Replaces `lex`.

The Free Software Foundation An organization devoted to advancing the cause of free software, primarily through the implementation of GNU. Several of the programs we support were written under the ægis of the FSF.

`g++` The GNU C++ compiler. Unlike `cfront`, `g++` is a native-code compiler which can also generate special information for debuggers.

`gas` The GNU Assembler. Turns low-level *source* code into code a machine can execute.

`gawk` The GNU interpreter for the `awk` language.

`gcc` The GNU C compiler.

`gdb` The GNU Debugger. Used to find errors in programs.

GNU Gnu's Not Unix. It is, however, a free replacement for Unix, being developed by the Free Software Foundation. The GNU Project is the worldwide collection of people and projects which are collectively building the GNU software system.

`gprof` A profiler for `UNIX` applications. Used for finding the places where a program spends most of its time, in order to speed the program up.

`info` A hypertext documentation browsing tool. Runs either as part of Emacs or as a standalone program.

Interpreter A program which executes each statement of a program written in a programming language. Contrast this with a compiler, which does the translation all at once, for later execution.

`ld` The `UNIX` linker. Takes a variety of object files (*cf* Compiler) and links them together into a complete program.

`less` Less is `more`.

`lex` A programming language for building pattern matchers based on deterministic finite automata.

`libC` The `AT&T` C++ library. Provides an object-oriented input/output interface, and not a whole lot more.

`libc` The `UNIX` C library. Provides standard `UNIX` services, such as formatted input/output, string searching, data conversion, sorting routines, and an interface to the `UNIX` operating system.

`libg++` The `GNU C++` library. A large set of C++ classes, such as arbitrary precision integers, complex numbers, stacks, heaps, vectors, queues, random numbers, strings, fixed-point numbers, self-balancing tree structures, and an input/output facility compatible with `libC` version 1.2.

`libm` A library of functions for mathematics.

`make` A program for maintaining dependencies among files, in order to simplify the maintenance of a large program.

Minor release An intermediate release of Cygnus-supported software. See Vintage release.

`more` A program for browsing files.

`perl` Another programming language similar in intent to `awk`, but with greater power.

`protoize` A program for helping convert K&R-style (archaic) programs into ANSI (modern) C.

Posix The specification for a portable operating system descended from `UNIX`.

`sed` A "stream" editor designed to be run in batch mode (ie not interactively).

`tar` A program for building *file archives*, files which bundle other files or directories together.

`texinfo` A programming language for building documentation. `texinfo` documentations can be processed by TEX, producing formatted hardcopy output, or by `info`, which provides an on-line hypertext document.

TEX A free typesetting system which is used to produce, among other things, this document.

`vi` A primitive screen editor.

Vintage release A version of the Cygnus-supported software which we have determined is particularly stable and reliable.

`yacc` A programming language for building parsers based on `LALR(1)` grammars.

Cygnus Support Price List

Support Products

May 1990

Cygnus Support currently offers two support packages at Core and Extended support levels.

The following table outlines the variety of services we offer, the levels which include them, and prices of our *à la carte* services.

Service	Core Support	Extended Support	*à la carte*
Vintage releases	included	included	$1000
Minor releases	$1500	included	$2500
Response time	24 hours	4 hours	N/A
Network patches	free	free	$5/minute
Network access	free	free	$5/minute
Quarterly Newsletter	included	included	$20
Installation*	$5000	included	$5000
Validation*		included	$10000
Ports		1 included	$50K and up
Customization		included	negotiable
Support of Add'l Tools*		available	negotiable

Services marked '*' are only available to our support customers.

Price of for our current support packages (applicable to a single site) is as follows:

Package	Core	Extended
Developer's Kit	$25000	$100,000
Shell Support	$25000	$100,000

A discount is available when purchasing both packages, and when purchasing packages for multiple sites.

Beta

1 Supported Platforms

Software is supported when running on the following platforms:

Vendor	Model(s)	Operating system
Sun Microsystems	SUN-3, Sparc	SunOS 4.0.3 or later.
Silicon Graphics	IRIS-4D Series	

2 Cygnus Developer's Kit

The Cygnus Developer's Kit includes the following items:

- The GNU Emacs Editor
- The GNU C and C++ Compilers (`gcc` and `g++`)
- The GNU Debugger (`gdb`)
- The GNU Assembler (`gas`)
- The GNU C++ library (`libg++`)
- The GNU Fast LALR(1) Parser Generator (`bison`)
- The programs `gprof`, `ld`, `size`, `nm`, `strip`, `ar`, and `ranlib`.

3 Cygnus Shell Support

The Cygnus Shell Support package contains the following programs:

- The GNU Emacs editor
- `bash`, the GNU replacement for `sh`
- The GNU version of `make`
- `gawk`, `perl`, and common applications written in them
- The small shell utilities, like `cat` and `mv`
- The standalone documentation programs `info`, `texinfo`, and `texi2roff`.
- The GNU versions of `sed`, `tar`, `egrep`, `vi`, `diff`, `gprof` and `more`

Beta

FREE SOFTWARE/PAID SUPPORT

Intellectual property is a confusing issue. While we firmly believe in the right of broad copyright/patent protection, call us pro-choice: We also believe that its use is a matter of individual discretion, and the best business strategy frequently is to give it away. Indeed, sometimes we think the whole issue of protecting software is irrelevant; the item in short supply is knowledge of what software to use, how to install it and how to represent business problems. In the long run, however, much of even this knowledge will be represented as software (and easily, with increasingly automated ways of converting high-level knowledge into code). So the protection issue won't go away; we'll be arguing about the copyrightability of business processes and software content and evaluations instead of look & feel. The purpose is not just to promote the ***development*** *of software, but also its marketing, distribution and use. (See Release 1.0, 89-8.)*

The following is not an attempt to resolve these issues, but a little story about some "unbundling" of product and ancillary values which implicitly argues that intellectual property is irrelevant. Call it an experiment.

Perhaps the strongest proponents of free -- not just "open" -- software are to be found at the Free Software Foundation. The Foundation, founded by the legendary RMS (né Richard M. Stallman) in 1985, is revered among programmers for its GNU Emacs text editor, C compiler and debugger and other tools. They are mostly the product of RMS, with help from a lot of friends, since everyone has access to the source code.

RMS, who lives at the MIT AI Lab but hasn't been on the payroll since 1984, fervently believes that all software should be free -- both free of all but delivery and service charges, and liberated in the sense that source code is available so that others can fix or enhance it. This is at the extreme end of a not totally linear "spectrum" with these salient points along it:

- free. Anyone may copy the code and redistribute it. Its freeness must be preserved: No one may charge for the code itself, at least according to the Free Software Foundation license, although they may charge for delivery, service and support. Source code must be available, and derivative works are encouraged.

- open. The specs of a standard (interface, language, whatever) can be copied, but the owner may charge for the specific implementation, and the code itself may not be copied. (Things will get sticky here with the advent of automatic reverse-engineering.)

- public. You can talk to the interface of the original, but you can't replace it. For example, you could be a front-end to a back-end, but you couldn't copy the back-end.

- semi-open. A semi-standard is freely available for copying or reimplementation with the payment of royalties.

- proprietary. No one may reimplement "it," let alone copy it outright. There's usually disagreement as to how much "it" covers -- look & feel, structure, sequence and organization, interfaces, commands/language, or just the code itself.

The issues of financial and source code freedom are separate but linked. Free software can be a moral issue, but proponents of free software also generally feel that free software and the right to create derivative works are the way to better software. You need the source code if you want to improve on the original. In the technical market served by the Foundation, that makes particular sense; most of the users are eminently capable of building the software themselves, although it's more efficient to reuse and improve the excellent work done by RMS and crew. The Foundation's GNU software includes the Emacs text editor; gdb, a debugger; gcc, a compiler; Bash, a shell; and many other utilities. Many major vendors ship GNU Emacs with their machines, and Motorola, Data General, Berkeley and NeXT ship the Foundation's gdb and gcc as their primary debugger and compiler.

Enforceable freedom

In fact, the only thing the Free Software Foundation won't let you modify is its license agreement, which specifies that anything that reuses that free software must also be free. What precisely that means has been controversial, and is why some people who really like the GNU G++ library and bison parser-generator don't use them. Most software tools produce code that is separate from the original, but object-oriented tools and parser-generators such as bison and yacc produce output that contains large chunks of the original code -- i.e. derivative works. But the Free Software Foundation board is on the verge of deciding that proprietary software can call a free library without either party to this peer-to-peer, cooperative processing transaction becoming contaminated, as long as source code and full "free" rights come with the library. The library stays free and can be copied; the proprietary code can be charged for.

It's ironic how complex intellectual property is, even when it's free. How far and how strictly can the creator control what is done with his creation? What happens when the free and the proprietary get scrambled? Even when software is free, you have to license its use to ensure its freedom.

Copyleft: Freedom for the rich in knowledge

So, the customers love the free software, and many of them love the ability to improve on it. They share their enhancements, fix bugs, and everything's great. But the market is centered on people willing to do all this, people who like to wallow in source code. What happens when you get to a less sophisticated market? How do users find the good stuff without advertising and sales efforts? How do they know what to do with it? The rest of the world wants documentation, fast service, prompt support, law and order...all the things you get when you pay for them.

This all sounded like an opportunity to three fond Emacs users, who have just banded together to form Cygnus Support (cyGNUs, get it?) to provide support to needy GNU software users. So far they have two customers, at NASA Ames and Sun Microsystems, at $100,000 a year each. That's the basic price per site, on the basis of about 100 users; prices are negotiable. (Cygnus is also in negotiations with Citicorp and General Electric.)

Co-founder Gumby Wallace, 26, who helped build and maintain MACLISP and the original Emacs, is straightforwardly eager to make money -- but only in return for a valuable service. He's also a true believer in the practical

415-543-8311

value of free software. "We choose to provide the best software available for UNIX, and add the one thing it lacks -- full support," he says. With Cygnus support, no one need choose inferior software simply because it is "commercial" software, says Wallace. Cygnus adds commercial touches such as QA, bug-tracking, installation and documentation.

The other founders are Michael Tiemann, 26, and John Gilmore, 35. Tiemann has written a lot of free software, including a portable instruction scheduler which improved GNU C's performance by 30 percent on the SPARC. Gilmore was the fifth employee at Sun Microsystems (after Joy, Bechtolsheim, Khosla and McNealy). He is an expert on microprocessor architecture and the sort of close-to-the-hardware issues involved in porting operating systems.

An example? Or an anomaly?

So now we come to the interesting questions. Is the combination of the Free Software Foundation and Cygnus a model of how the world could be? Or is it non-representative? If no one owns the software, who will go out and market it and foster its use? Perhaps people who get paid for doing so directly -- rather than through the sale of software. That would help avoid conflicts of interest: The consultant would recommend the best software, rather than that which he owns. But of course he will actually recommend the software he knows best and is best able to support, enhance, etc.

Traditional thinking -- to which we subscribe -- holds that the software will attract that infrastructure if it gets intellectual property protection; otherwise it will languish unknown because no one has an interest in promoting that particular piece of work.[1] How unbundled can you get? Will people build software without rewards? What's the impact of object-oriented programming on all this, and cooperative applications? Standards?

Could the software industry become one of support and product selection assistance, where the software itself is free? Already, you can buy most management books for a nominal fee (akin to the $150 charge you pay the Free Software Foundation for sending its tapes, with a turnaround time measured in weeks, or $1000 to Cygnus for its Federal Express service), but you may pay thousands of dollars a day for a consultant to implement the same advice or even just deliver it to you in a form tailored to your business. See Software Hygiene, page 11. (In the context of hypertext publishing, the links would be protected, while the underlying text chunks would be free.)

Pay for effort, not for results

Programmers, after all, may be motivated to do the best work they can, and to keep on improving the software, for sheer intellectual satisfaction. The best way to encourage them to do so is to make software and source code -- both the original and the improvements -- freely available. On the other

----

[1] *When you look at a complex business arrangement like the one for Home Lawyer on page 14, among Hyatt, OverDrive, MECA and Egghead, surely it is only the notion of interests in property that can be defined, created, marketed and sold that makes the wheels of commerce turn. (This product is an especially commendable example, since it reduces the need for lawyers.)*

hand, technical writers, support specialists, the people who fill orders are less likely to be so motivated. They need jobs that pay just like everyone else. But their efforts are necessary to get people to use the software -- and so it certainly makes sense to pay *them*. And programmers get paid for programming, even by the Free Software Foundation, with seven staff programmers; they just don't get paid directly for the software they write.

Pro-choice

This all makes sense, but we can't help thinking that allowing software to be protected on the margin will result in more software being built, and more people working to spread its benefits. Property rights in software, as in other goods, will provide incentives for owners and their representatives to exploit intellectual property to the fullest. RMS counters that proprietary software reaches only a limited market, whereas we see that the owners' interest in selling a lot of it counters the restrictions imposed by copyright. Owners take the trouble to publicize the stuff, explain its benefits, even support it for "free" -- or at least absorb the costs in their license fees. Better we should have a competitive market in support, says RMS. We will anyway as vendors unbundle.

The reaction to free software is interesting. There's this notion of contamination that hangs in the air -- both because it's free and an implicit insult to people who charge for their software, and because of its assumed association with virus-ridden software of uncertain provenance. In fact, the GNU products are strictly controlled to ensure their freedom, and a side benefit of the availability of source code is that it's easier to detect viruses. So those of you who counted on the absence of intellectual property protection to reduce bureaucracy, think again. The definition and preservation of the integrity of intellectual property requires the precise definition of that property. That definition is no more overhead than skin is overhead because it merely contains the body's "working" organs.

On the other hand, there's also an interesting reaction to copyright/patents on the part of the software liberators. They fear that protection will freeze the world, covering everything so that no one may write anything because someone else already owns it. We think that misses a number of far-reaching distinctions between original and derivative works, and the possibility of writing something *new*. If a particular piece of work is so good that everyone wants to use it, that's great. If the owner doesn't license it, we would count on the market to come up with a reasonable substitute. (Yet to simplify things, we'd prefer to see a shorter period of protection.)

In the end, why not let people who want to give software away do so, and let the others protect theirs as they will? The only rule is that you can't give something away and then change your mind midstream and start charging people or suing them. Nor can you start giving away for free something that people have started reselling under license from you. Most of these issues can be covered in well-constructed contracts -- as long as it's guaranteed that creators can control or assign the rights to control their creations.

Although we don't agree with all the premises of the Free Software Foundation, we're delighted it exists. For starters, it takes all approaches to make a market. And if the Foundation's approach consistently produces superior software, it will win. That's what happens in a free market.