

Brave GNU World

Georg C. F. Greve
greve@gnu.org

Dec 06, 2001

Welcome to another issue of Georg's Brave GNU World. Issue #31 (<http://www.gnu.org/brave-gnu-world/issue-31.en.html>) introduced FreeCIV and I asked for other examples of Free games. The feedback was quite good and so some games of different genres will be presented as a result.

1 BZFlag

It doesn't take a lot of time to explain the story of this issue's first game, BZFlag (<http://bzflag.org>). Every player drives a tank through a 3D landscape trying to shoot other players, capture enemy flags and pick up gadgets. This makes it a classic "shoot em up" game for multiple opponents who can play over LAN or internet.

BZFlag originated in the "Program of Computer Graphics" of Cornell University, where Chris Schoenemann discovered it as an effective way of avoiding work on his thesis in 1992. With the participation of other students, BZFlag was evolved quickly and soon became the favorite time to waste time for many students.

In the beginning, the game was written for HP-UX workstations using a iris-gl-like graphics library, but when the third "SGI Indizone Contest" came up, Chris rewrote it completely, improved the graphics and added sound. BZFlag won in its category. After further enhancements, Chris ported it to GNU/Linux and Win32 and released it as Free Software under the GNU General Public License.

By now Tim Riker has taken over as the BZFlag maintainer and more than a dozen developers work on it together with him on SourceForge, where it made it into the top ten of the most active projects multiple times.

According to Chris, especially the very good gameplay and the balanced dynamics are special about BZFlag. The forward-, rotation-, shot and reload speed of the tanks are tuned so that duels between experienced opponents can end up as complicated dances where a small tactical mistake decides everything.

Comments on the home page seem to second this. It seems that even people in the "U.S. Army Combat Maneuver Training Center" like playing BZFlag.

A problematic part of BZFlag is the networking code. The game was originally designed for local networks (LAN), on which it transmitted fast-changing information by multicast or broadcast, while the communication with the server was done by TCP to make sure no information about captured flags and such was lost.

In order to allow internet play, UDP packets were also transmitted through TCP, which created massive timing problems. By now UDP packets are being transmitted through UDP by the server, but it is planned to replace all usage of TCP, as it tends to block the connection in case of transmission problems.

The next release will aim to reduce the latency prob-

lems created by networking, also it will have fewer bugs and be more "cheat-proof." In the long term, an overhaul of the graphics engine and more modes of play appear useful.

Help is wanted with the port to Macintosh, which isn't progressing as well as Chris would like it to. Also the networking protocol is not the only improvable part of BZFlag. Graphics, artificial intelligence of computer foes and the physical model are aspects always worthy of attention.

2 Arkanae

The second game of this issue is a representative of the roleplaying genre. In Arkanae(*Arkanae home page* <http://arkanae.tuxfamily.org>) the player becomes a hero who has to save a world terrorized by seven incredibly powerful beings, the "Arcances." These beings were once trapped by a powerful crystal for the sake of humanity and are now roaming free again.

The game has 3D graphics in a quality that may surprise some and has a widely non-linear plot. The player can chose whether to fight for good or evil and how to find the hidden treasures, solve the quests and fight monsters.

The programming language used for Arkanae is Java and it uses the opale.soya (<http://opale.soya.tuxfamily.org/>) Java 3D Engine that was written specifically for Arkanae and allows using 3D hardware acceleration.

On a side note: this engine is now being used by the Fraunhofer Institute for 3D visualization in the biomodelling part of the M3 project ("Man Model Measurement").

Due to using Java, the project is pretty platform independent and runs on GNU/Linux, Mac OS X and Windows, but at the same time it has the usual Java problems.

This means that getting the right installation of Java is sometimes a gamble and also there is still no Free Java implementation offering the full range of features. Arkanae, for instance, is dependent on the Sun JVM 1.3 or BlackDown, both of which are not Free Software.

This means that Arkanae, being licensed under the GNU General Public License as Free Software, is dependent on a proprietary platform prone to some technical problems.

Even if this is not as critical for a game, it does show that the Free Software community should be more conscious about either furthering the Free Java implementations or doing without Java. As the idea behind Java seems useful, it is probably better to strengthen the Free alternatives. An introduction can be found on the "GNU and Java" home page (<http://www.gnu.org/software/java/>).

But people interested in trying out Arkanae should not let their fun be diminished by this, because Arkanae has quite a bit to offer and is already internationalized for French, English, German and Italian.

The problem child of Arkanae is the Windows platform, because keyboard management and music suffer from lacking thread support.

The authors, Bertrand Lamy, Jean-Baptiste Lamy and Althea Chia plan to complete Arkanae soon and finish the scenario. Future plans of Betrand include writing another game for which he is currently preparing a 3D engine in C. Jean-Baptiste discovered Python and wants to realize some projects in it.

Help with Arkanae would be especially useful in form of more levels. A level editor does exist, although it is rather complicated to use, therefore Bertrand offers an introductory course for would-be level designers.

Enough said, fans of roleplaying games should take a look.

3 XShipWars

Anyone who played the space fight simulation Elite some years ago will most likely be fascinated by XShipWars (<http://wolpack.twu.net/ShipWars/XShipWars/>). Similar to Elite, the player takes over the role of a space ship captain having to survive a big universe with the help of diplomacy, trade and combat.

The game mainly consists of a client and a server. The server handles the universe and allows clients to interact with the universe and other players. The universe itself is largely created by the interaction of the players; players with special permissions can even create new objects and shape the universe. Additionally, special modules exist to give computer-controlled ships artificial intelligence (AI). Depending on their empire, these ships can help other players or attack them.

The game originated in 1995/1996 as a purely text based "ShipWars" and became graphical in 1998. Whilst the text-based version had been written in Fortran, only C/C++ is used today. (X)ShipWars is developed by Wolfpack Entertainment, a loosely-knit group of computer graphics specialists and programmers that have, as they say themselves, "somehow managed to work together on this project for 5 years" and release it under the GNU General Public License.

Byproducts of this development are two other project, that are being used outside XShipWars. The libjsw (<http://wolpack.twu.net/libjsw/>) library is a high-performance joystick library and YIFF (<http://wolpack.twu.net/yiff/>) is an audio-server geared towards games. When these projects were started, neither GTK nor ESD were even being thought of.

Taura Milana, who answered the questionnaire about XShipWars, believes the weakness of XShipWars to be the graphics that are not quite up to modern standards, because fast and fluent gameplay were considered to be more important. Also the decisions neces-

sary in dogfight might be too much for a beginner.

Other than that, XShipWars is very mature and stable and especially the balance between action and strategy is something Taura is very fond of. The current development of XShipWars is more geared towards little gadgets than real functional expansions.

More interesting than the plans for XShipWars are plans for a "ShipWars 3D," which will be the third generation of the game. For this, a complete reimplementa- tion based on OpenGL and modern techniques is planned.

But those who would like to join the fray immediately don't have to wait that long, because even without OpenGL support, XShipWars is a lot of fun.

The currently largest XShipWars universe is "Terminus" by Stein Vrale(<http://xsw.terminator.net>). Thanks to the motivation of the XShipWars community, the Terminus universe evolved into a place where every system, every planet and every ship are unique, there is no repetition. Players have a big universe to explore and in which they can trade and upgrade their ships.

Enough games for this month. I'd like to provide some more options to waste long winter nights with Free Software, so if you know other games or happen to work on them, please let me know. [1]

4 TRAMP

The TRAMP ("Transparent Remote (file) Access, Multiple Protocol") project (<http://tramp.sourceforge.net>) should be of interest to many Emacs users. With it, Kai Grojohann provides a GPL add-on to Emacs that bears certain similarities to Ange-FTP.

In a networked environment, having to edit a file on another machine is a typical necessity. Ange-FTP allows transferring this file through FTP, edit it locally

and transfer it back to its origin.

Instead of relying on FTP, TRAMP supports shell-based transmission protocols like ssh, telnet and rlogin as well as su and sudo. TRAMP is also designed in a way that should make it reasonably easy to add other ways of accessing a machine, should the need arise.

Through the shell connection, TRAMP first checks whether the file exists, what its permissions are and so on. For editing the file, TRAMP offers two possibilities. Files can be modified directly through the shell connection, which is the so-called “inline” method or transferred to the local host, which is similar to the working of Ange-FTP. Kai calls this method “out-of-band.”

Additionally, TRAMP has limited remote-execution capabilities. This allows integration with the “Version Control” (VC) mechanisms, so a file under CVS control can be remote edited through TRAMP and committed to the repository with the usual ‘C-x v v’ key sequence.

Theoretically, it is also possible to use TRAMP for editing files on Unix machines on a Windows machine, but getting the configuration right is a bit tricky; it appears to require a very special version of ssh, for instance. Kai cannot provide detailed instructions for this at the moment, if anyone feels capable of giving him a few more details, please do.

The opposite case, using a Unix-Emacs to edit files stored on a Windows machine, is even more complicated as TRAMP assumes a Unix-like environment on the remote host.

The different Unix-implementations are sometimes quite different, so in order to be as stable as possible, TRAMP always tries multiple ways to assess the file status, for instance. There should be no problem with any of the standard Unix-variants.

TRAMP already has quite a bit of history, in earlier versions, its name was rssh.el and then rcp.el. Right

now Daniel Pittman is working on a reimplementation of TRAMP, which Kai likes a lot. Future plans include also supporting asynchronous/interactive shell-commands and making the setup of connections more intelligent.

Also Kai would like to finally assign the copyright to the Free Software Foundation, so TRAMP can become an official part of the GNU Emacs.

5 Kopi

The Kopi project (<http://www.dms.at/kopi/>) of “Decision Management Systems GbmH” (DMS) provides a development environment for database applications under Java, JDBC and JFC/Swing.

As far as Thomas Graf is aware, the compiler and flagship of the project, KJC, is the only Free Java-compiler entirely written in Java itself to fully support the recent Java specifications.

The Kopi environment also offers an assembler & disassembler for the Java Virtual Machine (JVM), a library for generation and modification of JVM class files and XKJC, a compiler for embedded SQL in Java. The finishing touch is “Visual Kopi,” an application framework providing a special high-level language for creating database applications.

Thomas Graf sees the biggest strengths of Kopi to be the high quality and speed of the KJC, which beats other compilers like that of Sun (Javac) or IBM (Jikes) in the Jacks test suite (“Jacks is an Automated Compiler Killing Suite”). If you prefer hard numbers: according to the Kopi team, the KJC in client/server mode executes the Jacks test suite (1845 compilations) in less than 37 seconds on a 1GHz Athlon with RedHat GNU/Linux 7.1.

Also the KJC provides complete diagnostic messages of different types pointing out dangerous/redundant code, “dirty,” old or unreadable syntax or possible optimizations. This should make it much easier for

any developer to maintain the code.

Keeping Kopi up-to-date with current specifications is one of the main goals for the future, so DMS hired Martin Lackner, for instance, who developed several extensions of the upcoming Java 1.4 specification in his thesis at the TU Vienna. Also Mamisoa Rajosviah is working on the "Developers Guide" for Visual Kopi, which should be finished soon.

Similar to the GNU Compiler Collection (GCC), the Kopi runtime-libraries are released under the GNU Lesser General Public License, while the rest is released under the GNU General Public License.

At the end of this issue I'd like to quickly introduce two small but interesting projects.

6 CDArgs

CDArgs (<http://www.skamphausen.de/software/cdargs/>) is an extension of the 'cd' shell command written by Stefan Kamphausen in C++.

It extends the 'cd' command with bookmarks as well as a directory browser. Also it provides an integrated interface to administrate the bookmarks, although they are simply saved as lines in an ASCII file, so they can be edited conveniently with any text editor.

He got the idea for this project when he saw an article in the German computer magazine iX, in which Michael Schilli implemented similar (although not quite as advanced) features in a Perl script. Since the script was under a non-Free license, Stefan picked up the idea and implemented it anew with some more features under the GNU General Public License.

For further development in terms of internationalization, documentation, portability, case-insensitive pattern matching and configurable keymaps, he is receiving help mostly by Claus Brunzema and Tomi Ollila. New releases are to be expected soon.

7 CDLoop

Claus Brunzema also wrote the CDLoop (<http://www.cbrunzema.de/software.html#cdloop>) project, which is a CD player with special looping capabilities. The normal users of this software would be musicians writing down parts of a CD in notation or playing along with them.

Normally, AB loop markers can only be set, but not modified afterwards, so everything depends on pushing the button at the exact right moment. CDLoop allows adjusting the marks and even saving them as bookmarks.

CDLoop was written in C++ with guile and guile-gtk as graphical toolkit and is available under the GNU General Public License.

Thanks to guile it is possible to program and configure CDLoop in pretty much any imaginable way, so even very unusual applications should not be a problem.

That should be it ...

One question that reaches me occasionally, is: "How much publicity will a Brave GNU World feature create for me?" Unfortunately there is no final answer to this question.

Most projects never inform me whether a significant increase in interest has occurred; although it did happen that I received very happy mail telling me the interest in a project did increase significantly — sometimes from a few hundred visitors to a few thousand within a few days.

But there are no hard numbers I can offer. Also the Brave GNU World is released successively in the different media, sometimes up to 2 months after the first release on the web.

From my experience I am quite certain that a feature in the Brave GNU World will raise the profile of a

project, however, and sometimes it creates connections that would not have been created otherwise. So like every time, I'm asking for ideas, comments, feedback, questions and project descriptions to the usual address. (*column@brave-gnu-world.org*).

8 Info

[1] Home page of the GNU Project:

<http://www.gnu.org/>.

[2] Home page of Georg's Brave GNU World:

<http://brave-gnu-world.org>

[3] "We run GNU" initiative:

<http://www.gnu.org/brave-gnu-world/rungnu/rungnu.en.html>

Please send FSF & GNU inquiries & questions to **gnu@gnu.org**. There are also other ways to contact the FSF.

Please send comments on Georg's Brave GNU World (in English or German) to *column@gnu.org*, send comments on these web pages to *webmasters@www.gnu.org*, send other questions to *gnu@gnu.org*.

©2001 Georg C. F. Greve

Permission is granted to make and distribute verbatim copies of this transcript as long as the copyright and this permission notice appear.

About the Author Georg Greve is the president of FSF-Europe, which founded in 2001, located in Germany.

MOSCOW. Dec 31 (Interfax) — Russian computer software specialist Dmitri Sklyarov charged with violating the Digital Millennium Copyright Act in the United States returned to Moscow on Monday.

According to an Interfax reporter, at the airport he was welcomed by his family and close friends. Sklyarov told Interfax that he would see the New Year in with his "near and dear ones."

Earlier Vladimir Katalov, executive director of the Russian Elcomsoft company for which Sklyarov worked, told Interfax that the programmer plans to hold a news conference for American journalists and meet the leadership of the Electronic Frontier Foundation that supported him ever since his arrest.

Under an understanding Sklyarov pledged not to break American laws even outside the United States until all charges against him are dropped. Lawyers say this will not happen earlier than in year's time.

Katalov said that Sklyarov will continue working for Elcomsoft but will refrain from developing products related to electronic books in one way or other. "At the advice of lawyers the company has imposed a moratorium on the development of such products," he said adding that Sklyarov will have plenty of other work to do because "a professional of such a standard will not remain without a job."

Sklyarov was arrested on July 16 in Las Vegas where he delivered a report at a conference on computer technologies. He made a program allowing to circumvent the security systems imposed by the program of U.S. Adobe Systems eBook Reader and read books in digital format.

Dmitry Sklyarov is a 26-year-old Russian citizen and Ph.D. student who studies cryptography. He is the father of two children (2 1/2 year old son, 3 month old daughter).
