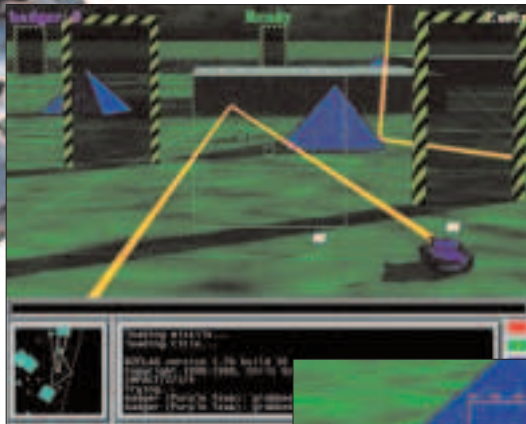


The monthly GNU Column

BRAVE GNU WORLD



Welcome to another issue of Georg CF Greve's Brave GNU World. In LM14 we introduced FreeCIV and asked for other examples of Free games. Here are some of the games that were suggested



BZFlag

BZFlag

It doesn't take a lot of time to explain the concept behind this issue's first game, BZFlag. 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 a LAN or via the Internet.

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

In the beginning, the game was written for HP-UX workstations using an IRIS GL-like graphics library but when the third SGI Indizone Contest came up, Chris rewrote it completely, improving the graphics and adding sound. BZFlag subsequently 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.

Tim Riker has now taken over as the BZFlag maintainer and more than a dozen developers work

on it with him on SourceForge, where it made it into the top ten of the most active projects on multiple occasions.

According to Chris, the engaging gameplay and the balanced dynamics are what makes BZFlag particularly special. The forward-, rotation-, shoot and reload speed of the tanks are tuned so that a duel between experienced opponents can end up as complicated dance where a small tactical mistake decides everything. Comments on the homepage seem to second this: it seems that even people in the

US 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. UDP packets are now transmitted by the server through UDP 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 problems created by networking. It will also 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.

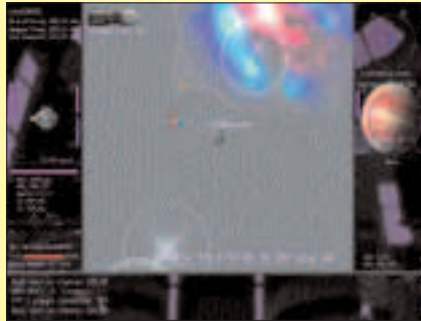
Help is wanted with the port to Macintosh, which isn't progressing as well as Chris would like. The networking protocol is not the only part of BZFlag which could be improved: the graphics, artificial intelligence of computer foes and the physical model are all aspects worthy of attention.

XShipWars

Anyone who played the space fight simulation Elite some years ago will most likely be fascinated by XShipWars. In a similar vein to Elite, the player takes over the role of a space ship captain who must 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 may help other players or attack them.

The game originated in 1995/1996 as the purely text-based ShipWars and only 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



XShipWars

General Public License.

By products of this development are two other projects, which are being used outside XShipWars. The libjsw library is a high-performance joystick library and 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, which are not quite up to modern standards – fast and fluent gameplay were considered to be more important. Also the decisions necessary in a dogfight might be too much for a beginner to handle.

Other than that, XShipWars is very

mature and stable and the balance of action and strategy is something Taura is especially 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 reimplemention based on OpenGL and modern techniques is planned.

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.

Currently, the largest XShipWars universe is Terminus by Stein Vrale. 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.

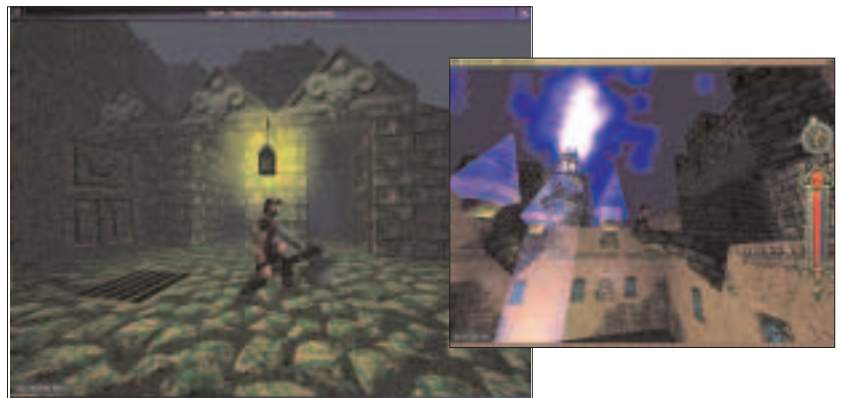
Arkanae

Our next Free game is the role playing adventure, Arkanae. In this game the player becomes a hero who has to save a world terrorised by seven incredibly powerful beings, known as the Arcances. These beings were once trapped by a powerful crystal for the sake of humanity and are now roaming free again.

The quality of the game's 3D graphics may surprise some, as will the 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.

Arkanae is programmed in Java and uses the Opale.Soya Java 3D engine, written specifically for the game, which allows Arkanae to use 3D hardware acceleration. On a side note, this engine is now being used by the Fraunhofer Institute for 3D visualisation in the biomodelling part of the M3 project (Man Model Measurement).

As the game uses Java, the project is fairly platform independent and runs on GNU/Linux, Mac OS X and Windows, although at the same time it does have the



usual Java problems. This means that getting the right installation of Java is sometimes a gamble and 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, neither of which are Free Software.

This means that Arkanae, being licensed under the GNU General Public License as Free Software, is



Arkanae

dependent on a proprietary platform prone to some technical problems.

Even though this is not as critical for a game as certain other applications, it does show that the Free Software community should be more conscious about either furthering the Free Java implementations or doing without Java completely. 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" homepage.

Anyone interested in trying out Arkanae should not let their fun be diminished by this, as Arkanae has quite a bit to offer and is already available in French, English, German and Italian.

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

The authors, Bertrand Lamy, Jean-Baptiste Lamy and Althea Chia, plan to complete Arkanae soon and finish the scenario. Bertrand's future plans include writing another game for which he is currently

preparing a 3D engine in C. Jean-Baptiste discovered Python and wants to realise 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 role playing games should take a look.

TRAMP

The TRAMP (Transparent Remote (file) Access, Multiple Protocol) project should be of interest to many Emacs users. With it, Kai Großjohann 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 enables users to transfer a 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 remotely 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 via 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 so 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

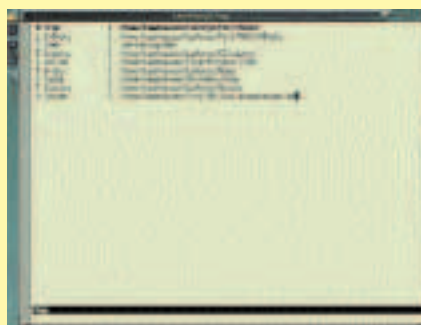
cdargs

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. cdargs also provides an integrated interface to administrate the bookmarks, although they are simply saved as lines in an ASCII file so they can be conveniently edited with any text editor.

Stefan 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 internationalisation, documentation, portability, case-insensitive pattern matching and configurable keymaps, Stefan is receiving help mostly from Claus Brunnema and Tomi Ollila. New releases are to be expected soon.



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

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

Kopi

The Kopi project of Decision Management Systems GbmH (DMS) provides a development environment for database applications under Java, JDBC and JFC/Swing.

As far as Thomas Graf, the compiler and flagship of the project, is aware, 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 and disassembler for the Java Virtual Machine (JVM), a library for the 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 strength 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 (Jikes Automated Compiler Killing Suite) test suite. If you prefer hard numbers, according to the Kopi team, the KJC in client/server mode executes the Jacks test suite (1,845 compilations) in less than 37 seconds on a 1GHz Athlon with Red Hat 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 optimisations. 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, which is why DMS hired Martin Lackner, who developed several extensions of the upcoming Java 1.4 specification in his thesis at the TU Vienna. Mamisoa Rajosvah is also 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.

Enough for today

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.

cdloop

Claus Brunzema also wrote the 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 enables users to adjust the marks and even save them as bookmarks.

cdloop was written in C++ with Guile and Guile-gtk 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.



Most projects never inform me whether a significant increase in interest has occurred; although I did receive a very happy mail telling me the interest in a particular project increased significantly – from a few hundred visitors to a few thousand within a few days.

Although there are no hard numbers, from my experience I am quite certain that a feature in the Brave GNU World will raise the profile of a project and sometimes create 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.

Info

Send ideas, comments and questions to Brave GNU World	column@brave-gnu-world.org
Homepage of the GNU Project	http://www.gnu.org
Homepage of Georg's Brave GNU World "We run GNU" initiative	http://brave-gnu-world.org http://www.gnu.org/brave-gnu-world/rungnu/rungnu.en.html
BZFlag homepage	http://bzflag.org
Arkanae homepage	http://arkanae.tuxfamily.org
Opale.Soya Java 3D engine	http://opale.soya.tuxfamily.org
GNU and Java homepage	http://www.gnu.org/software/java
XShipWars homepage	http://wolfpack.twu.net/ShipWars/XShipWars
libjsw homepage	http://wolfpack.twu.net/libjsw
YIFF homepage	http://wolfpack.twu.net/YIFF
Terminus XShipWars universe	http://xsw.terminator.net
TRAMP homepage	http://tramp.sourceforge.net
Kopi homepage	http://www.dms.at/kopi
cdargs homepage	http://www.skamphausen.de/software/cdargs
cdloop home page	http://www.cbrunzema.de/software.html#cdloop