



AMIGA CD32 Games Compilation Toolkit 1.1

By Lennon Lives

Manual

Disclaimer and Terms of Use

The *AMIGA CD32 Games Compilation Toolkit* is freeware and is to remain FREE. The author of this toolkit can be held responsible neither for the contents of the compilations created with it by other users nor for any damage that it may cause to your hardware.

Compilations created with this toolkit may not be sold or used in any other possible commercial way.

If you disagree with these terms of use you must not use this toolkit.

Play fair. Since I offer this package completely free of charge, I'm asking you to credit me for my hard work in your compilations. A fair deal, isn't it?

I hope you enjoy using this piece of software just as much as I did creating it! Go create some amazing compilations!

Setup

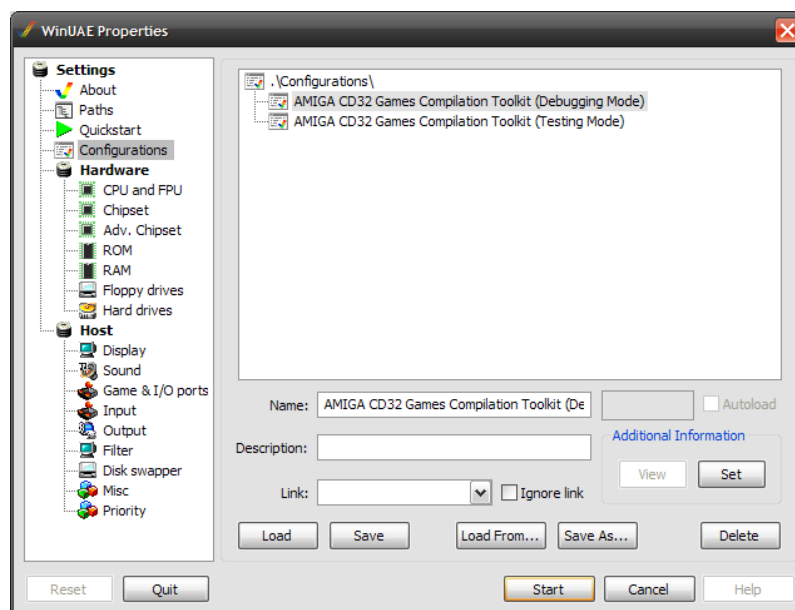
Extract the contents of “ExtractMe.zip” to the *WinUAE* folder on your harddisk, e.g. “C:/Program Files/WinUAE/”. You might be asked if you would like to overwrite “configuration.cache”. Please do so. If you did things correctly, you will now find a new folder named “AMIGA CD32 Games Compilation Toolkit” inside the *WinUAE* folder.

If you want to build your own bootable *AMIGA CD32* ISOs with the toolkit, you will have to download the *AMIGA CD32 Developer Kit* from the following link:

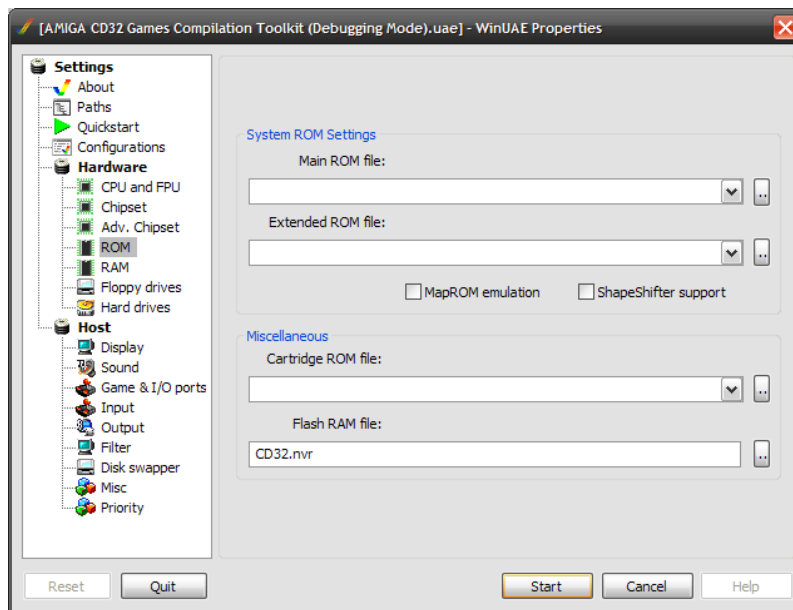
http://static.cd32-allianz.de/downloads/software/CD32_DeveloperKit.7z

Open “CD32_DeveloperKit.7z” with a suitable archiver (*WinRAR* is recommended) and go to “ISO9660Tools_V1.04/ISOCD/”. From there you will need to copy “CDTV.TM” to the “S” folder in the root of the toolkit folder as well as “ISOCD” to the “C” folder in the same location. Please note that these files are not critical for the toolkit to function, but add a very useful feature. The toolkit will automatically check for them and use them if available.

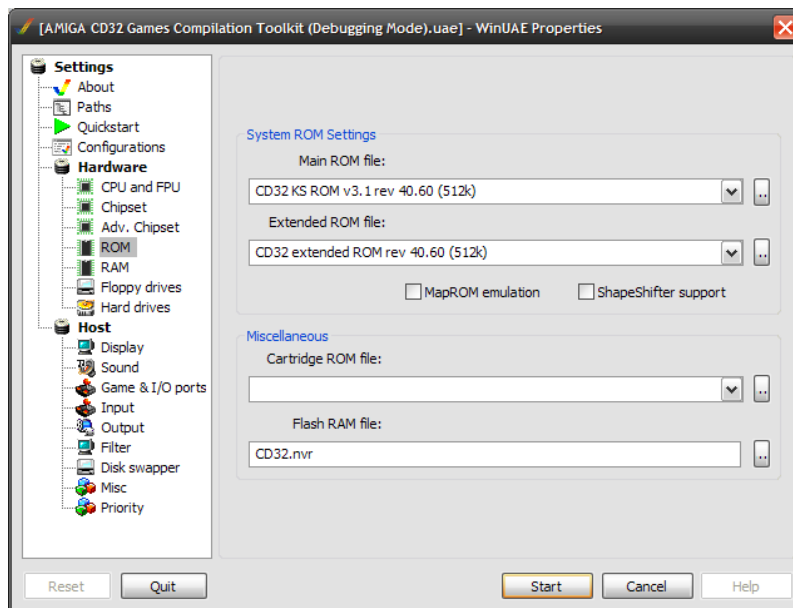
The toolkit comes with predefined *WinUAE* configuration files that you will have to make sure point to the correct ROM files needed for proper *AMIGA CD32* emulation. To do this, run *WinUAE* and click on its “Configurations” tab.



You will find two configuration files specific to the toolkit. Choose “AMIGA CD32 Games Compilation Toolkit (Debugging Mode)”, click on “Load” and then on the “ROM” tab.

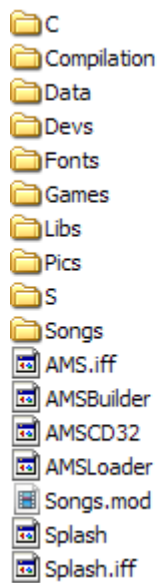


As you will notice, no ROM files have been picked yet which is what you will have to do next. Select the correct main and extended ROM files so the “ROM” tab will look like this afterwards:



Now go back to the “Configurations” tab and click on “Save”. Repeat this procedure for the “AMIGA CD32 Games Compilation Toolkit (Testing Mode)” configuration file and you’re done with setting up *WinUAE*!

The next thing to do is enter the toolkit folder and add the missing Kickstart files, which are still under copyright protection and therefore can’t be included with this toolkit. You will find the contents of the folder to look like this:

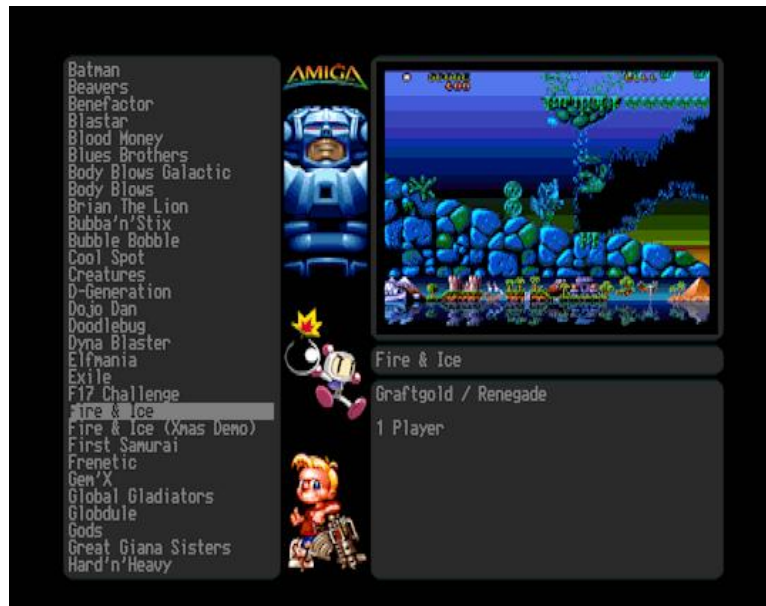


The “Kickstarts” folder is located inside the “Devs” folder. Copy your legally owned *AMIGA 500* Kickstart files there (usually named “kick34005.a500” and “kick34005.a500.rtb”). *WHDLoad* will need them to run quite a few games. Including *AMIGA 1200* Kickstart files as well doesn’t really make sense because of the limited RAM of the *AMIGA CD32* so it is recommended to leave them out. Do not change any of the files that come with the toolkit. Every file provided is critical to the proper functionality of the toolkit!

You are now done setting up the *AMIGA CD32 Games Compilation Toolkit*!

The Advanced Menu System

The *Advanced Menu System* (short: *AMS*) is the heart of this toolkit. It is based on MagerValp's fantastic *Arcade Game Selector*, which has aged quite a bit over the past years, unfortunately. In my opinion, the changes made to the original source code are extensive enough to justify picking a different name for it, hence *AGS* becomes *AMS*. Starting with version 1.1 of the toolkit, *AMS* comes in two flavours: *AMSCD32* specifically for the *AMIGA CD32* (included with this toolkit) and *AMSHD* optimized for HD-equipped AGA *AMIGAs*.



AMSCD32 is a lot faster and easier to handle than *AGS* and includes the following features:

- optimized for the *AMIGA CD32*
- game list allows for an unlimited number of entries
- fast scrolling by 1 step using the Y axis of the digital pad
- ultra-fast scrolling by 5 steps using shoulder buttons
- freely assignable data directories
- support for RAM disk caching to improve overall seek times considerably
- support for custom fonts
- improved 256 colour palette to make your screenshots look a little nicer
- *AMSCD32* automatically restarts after quitting a game
(*WHDLoad*-installed games can be quit using "Play/Pause" on the *AMIGA CD32* joypad.)

When adding games to your compilation or configuring it, you will have to pay attention to how the toolkit processes its data, which will be explained in the next chapters.

Adding a Splash Screen

The toolkit offers the possibility of adding a splash screen displayed during the booting process. To successfully add a splash screen to your compilation, both “Splash” and “Splash.iff”, of which the latter has to be a 640x480x4 bits picture file, must be located in the root of the toolkit folder. In case any of these two files are missing, the toolkit’s “Startup-Sequence” will automatically detect this and won’t display the splash screen. It isn’t recommended to use a splash screen since all it really does is look pretty and delay the booting process.

Adding Songs

AMS uses *HippoPlayer* to play MOD files. You can add your own MOD files by placing them inside the “Songs” folder and editing “Songs.mod” located in the root of the toolkit folder. It should look similar to this:

HiPPrg

SYS:Songs/MyFavouriteSong.mod

SYS:Songs/MyGirlfriendsShittyFavouriteSong.mod

MOD and MED files are supported right out of the box. You can add support for additional file types by copying the respective *HippoPlayer* libraries to the “Libs” folder located in the root of the toolkit folder.

Adding Custom Fonts

AMS uses *Nopaz* to support custom fonts. Simply put a font’s folder and its .font file into the “Fonts” folder in the root of the toolkit folder and rename its FONT file to “Menu.font”. The toolkit already comes with *ClearVT*. Please note that the font to be added must be a standard *Workbench* font and has to support a font size of 8.

Adding Games

Three folders located inside the toolkit folder are critical to adding games to your compilation: “Data”, “Games” and “Pics”.

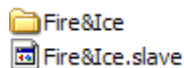
Games, obviously, are stored inside the “Games” folder. If you add *WHDLoad*-installed games, it is recommended to store the slave in the root of the “Games” folder and the associated game data in a separate data folder, which contains ONLY the game data stripped off anything not critical to the execution of the game to save as much memory for the game itself as possible.

Adding a game entry to *AMS* requires you to create three files. As an example, let's create an entry for a *WHDLoad*-installed version of Graftgold's *Fire & Ice*. We will need to create the following files which all have to be named the same except for their file extensions:

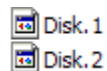
"Fire & Ice.start"

This is the script file that will execute your game. It contains all commands necessary to make your game work. Please note that *AMS* will automatically enter the "Games" folder prior to executing your script file so you can skip this step.

Let's assume we have already copied "Fire&Ice.slave" to the "Games" folder as well as a folder called "Fire&Ice" containing ONLY the game files. Your "Games" folder might then look like this



with the "Fire&Ice" folder containing only this:



To execute the game using *WHDLoad* (of which the *AMIGA CD32* version is already included with this toolkit), "Fire&Ice.start" would only have to contain one simple line:

```
C:WHDLoad Fire&Ice.slave DATA=Fire&Ice PRELOAD
```

However, some games might require a little more memory. Especially *WHDLoad* benefits from each byte of additional RAM it can get, so you want to try rebooting from the optional RAD disk by adding one simple line prior to the commands executing your game. Your script file would then look like this:

```
C:Execute C:MakeRAD  
C:WHDLoad Fire&Ice.slave DATA=Fire&Ice PRELOAD
```

With many games, using a RAD disk won't make a noticeable difference. Since launching a game without rebooting your *AMIGA CD32* takes less time (reloading *AMS* after quitting a game does, too) it is recommended to always try this first. Do some experimenting. You'll get the hang of it eventually.

"Fire & Ice.txt"

This is a simple text file providing some information about your game displayed in *AMS*. It could look like this:

Fire & Ice
Graftgold / Renegade

1 Player

Use an *AMIGA* text file editor for editing.

"Fire & Ice.iff"

This is a 320x128x8 bits screenshot of your game. Be sure to use the same palette as is used in *"AMS.iff"*, *AMS's* background file located in the root of the toolkit folder.

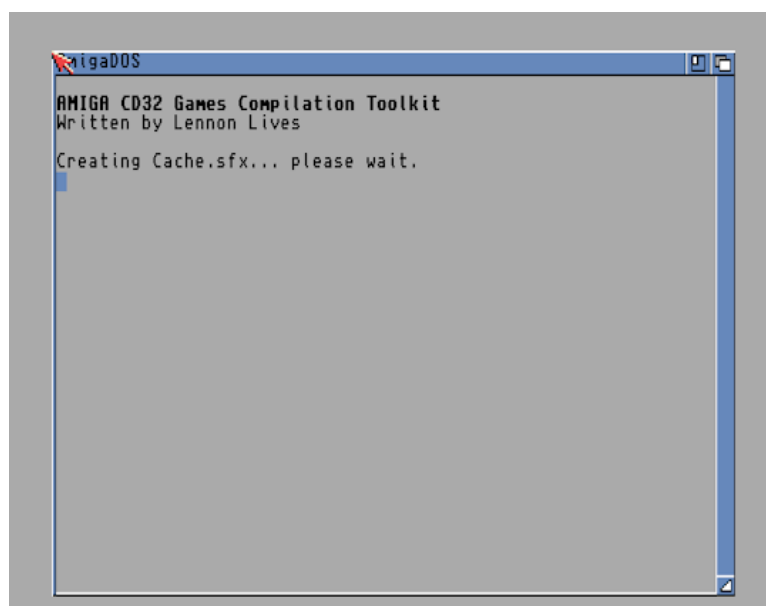
Now that you have prepared all necessary files, place *"Fire & Ice.start"* and *"Fire & Ice.txt"* inside the toolkit's *"Data"* folder and *"Fire & Ice.iff"* inside its *"Pics"* folder.

The toolkit already comes with predefined *AMS* data files. You might want to study them to get an idea of how to create effective script files for your games. Please note that you're not limited to *WHDLoad*-installed games. You can configure your script files to execute just anything you like!

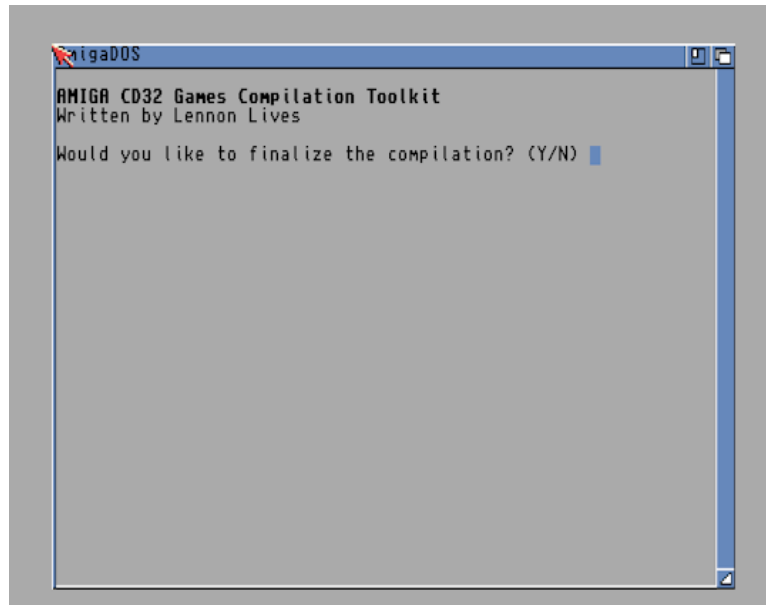
You are now ready to do some debugging!

Debugging Your Compilation and Building an ISO

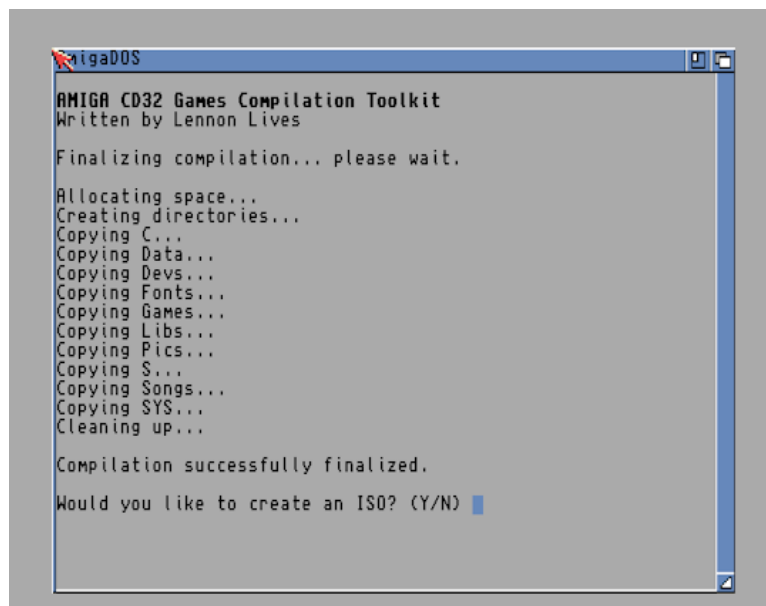
Bring up *WinUAE*, go to the *"Configurations"* tab and double-click on *"AMIGA CD32 Games Compilation Toolkit (Debugging Mode)"* to run the toolkit in debugging mode. The first thing you will notice is that the toolkit will automatically create a file called *"Cache.sfx"*, which is an archive containing critical *AMS* data stored in RAM for the purpose of caching and speeding up things considerably.



As soon as the toolkit is done creating “Cache.sfx” you will be asked if you’d like to finalize the compilation. Finalizing means that all data necessary to create a stand-alone games compilation (including the games and data files you added) will be copied to the “Compilation” folder and cleaned up.



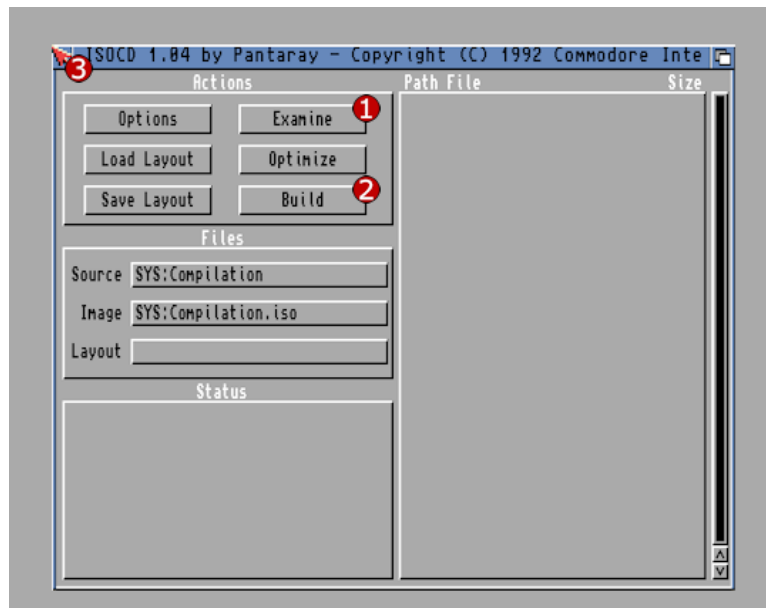
Answering no will take you straight to *AMS*, answering yes will start the finalizing process. Please note that your compilation data will stay untouched! No need to backup anything!



As soon as the compilation has been successfully finalized, you will be asked if you would like to create an ISO (only if “CDTV.TM” and “ISOCD” have been correctly installed as indicated in the chapter *Setup*). Again, answering no will take you to *AMS*. Prior to creating an ISO, your finalized compilation can be thoroughly tested using the “AMIGA CD32 Games Compilation Toolkit (Testing

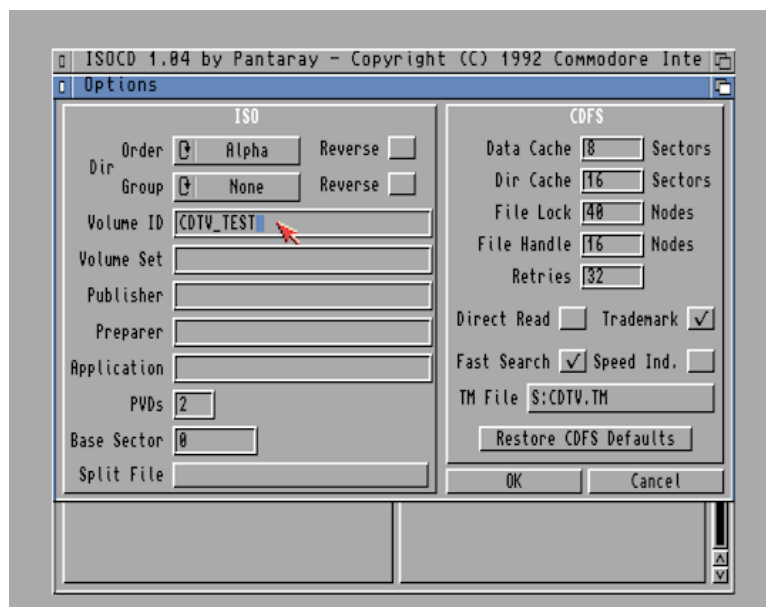
Mode)” configuration file in the *WinUAE* “Configurations” tab, which will give you an idea of how fast your compilation will perform on a real AMIGA CD32.

Answering yes will bring up *ISOCD*, preconfigured to create a bootable ISO for your AMIGA CD32:



All you need to do now is click on “Examine”, then on “Build” and then close *ISOCD* as indicated by the numbers in the screenshot.

If you want to label your compilation individually, click on “Options” prior to clicking on “Build” and change the “Volume ID” accordingly in the upcoming window.



Clicking on “OK” will take you back to *ISOCD*’s main window. Continue with step 2 then.

After closing *ISOCD* you will have successfully created an ultra-fast games compilation for your AMIGA CD32! “Compilation.iso” will be saved to the toolkit’s root folder, waiting for you to burn it.

Please note that this ISO will not contain any toolkit files unnecessary for it to function, e.g. "ISOCD" and "CDTV.TM" will be removed automatically.

Third Party Software

The *AMIGA CD32 Games Compilation Toolkit* includes the following third party software, which is available for free on Aminet (<http://aminet.net/>):

HippoPlayer by Hippopotamus Design.

JST by Jean-Francois Fabre.

Nopaz by Leo L. Schwab.

UnZipSFX by Info-ZIP.

WHDLoadCD32 by Bert Jahn.

Zip by Mark Adler, Richard B. Wales, Jean-Loup Gailly, Onno van der Linden, John Bush and Paul Kienitz.

All third party programs included are properties of their respective owners.

If you feel offended by the inclusion of your particular software, please contact me. It will be removed from the package immediately.

Thanks

If there's one person to be thanked in particular, it's MagerValp. With the *Arcade Game Selector* he created an amazing piece of software that inspired me to make even better. Thanks dude, you're the man!

Eternal thanks to Bernd Schmidt and Toni Wilen for the marvel that is *WinUAE*.

A firm handshake to Bernd Jahn for creating *WHDLoad* – without your hard work, this toolkit would have never come to life.

I'd also like to thank all those incredibly helpful guys at the *English Amiga Board*. Without you, my *AMIGA CD32* would be no more! Many thanks for countless interesting discussions.

Last but not least, thanks to my wonderful missus. She's the real deal and oh-so-patient. I know she hates me for loving my *AMIGAs*, but I think she's finally about to get the bigger picture. Recently, I caught her playing *Soccer Kid* on a test build of the toolkit, stating: "This is great. You know that?" Well, didn't we?