



AMIGA VARIATIONS



Deluxe Paint was the Amiga's first killer app – former Amiga artists still wonder why there's nothing else like it.

Atari maestro Jay Miner, Commodore released the Amiga in 1985. It eclipsed all previous micros, and the design elegance remains unmatched. Three revolutionary custom chips offered 4096 colours, four-channel sampled sound, plus a 16/32-bit Motorola 68000 CPU, serial, parallel and two mouse ports and an expansion bus called Zorro.

The way it fitted together was more important than raw specs, or the sub-systems. Amiga architecture transcended the sum of the parts, augmented by a ROM-based multitasking device-independent WIMP OS. Commodore squandered profits from millions of Amiga 500 sales trying to buy into the commodity PC market. 32-bit Amiga production shifted to Europe in the 1990s, after a buy-out. When Escom followed Commodore down the pan, Gateway 2000 took a turn; now a fifth Amiga company continues to develop hardware and software, but it's hard to imagine any new Amiga matching the impact of the original A1000.

The subtlety, performance and tight integration of the Amiga system makes it exceptionally hard to emulate. Modern PCs are very complicated, but their parts don't mesh seamlessly like

Simon Goodwin accepts a tough challenge to emulate the ultimate home computer.

Amiga ones, struggling to do the same even at 100 times the clock rate. Amiga emulation was a myth for more than decade – the acronym *UAE* once stood for *Unusable Amiga Emulator* – but now the *U'nix Amiga Emulator* really works. It spawned variants like *XFellow* and *Amithlon*. We'll focus on the main version.

Organisation

UAE comes with well-organised C source, and substantial technical docs. The mantra is 'you'll probably have to experiment a little'; a friend with Amiga experience can help a lot if you're a newcomer.

Compiled *UAE* versions are on the coverdisc but, for best compatibility, rebuild from source. Start with **.configure**, adding **--with-svgalib** for optional full screen display and **--enable-sound** for OSS audio. 'docs/unix/README' notes other options.

A default build took my K6-2/500 eight minutes, making a 1.6MB executable. There is no **make install** but you can run the executable from the directory you configured in, after listing Amiga resources in the `~/uaerc` file, as explored in the 'Amiga fuel' section. The `jit-0.8.21.tar.gz` package gives an **--enable-jit** option for experimental code translation, adding a control panel and boosting CPU emulation at the expense of compatibility. An MMU emulation patch reduces speed but boosts stability, especially when writing new code. PC hosts have more blocking issues than a real Amiga, as PC hardware was never designed for multi-tasking. Linux inherits PC limitations, so an emulated Amiga cannot be as smooth as the real thing – most obviously in mouse movement and screen updates – but may be faster in other respects, especially if you're used to 16-bit Amigas from the 1980s rather than later 32-bit A1200, A3000 or A4000s.

UAE Versions

The current stable release 0.76 of *UAE* emulates 16-bit Amigas reasonably (A1000, A500, A600, A1500 and A2000 without PC

co-processor) but lacks support for the 32-bit AGA chip set and later CPUs. 0.8.x releases are notionally betas, but more capable and quite reliable, so some recent distros include 0.8.x versions; 0.8.21 was the latest when this article was finished in August. *UAE* 0.76 can emulate a 32-bit 68020 processor with optional 68881 floating point unit, and retargeted graphics via *Picasso96* which lets system-friendly Amiga programs drive PC-style frame buffers. Few games support *Picasso96*, but it offers extra resolution and colour depth for later productivity applications.

A 15 or 16-bit screen depth is recommended. Extra bits per pixel benefit *Picasso96*, but hamper original Amiga graphics emulation. An 800 by 600 pixel window works well, as the original Amiga TV output was limited to 724 by 576 plus a border. Hires *Picasso96* requires the Kickstart 3 ROM, introduced in 1992, and 68020 or later 32-bit CPU emulation. These options confound some older Amiga software. Config can use a bewildering variety of switches on the command line, or a `~/uaerc` file, *GTK+* configuration window, or front-ends like *MyMiggy*.

Printer output defaults to *lpr*. *Ghostscript* filters give quality printout via the AmigaOS Postscript driver. BSD-style network sockets can link *UAE* to host networks. Serial port access from the emulator is diverted to `/dev/tty` but shaky, especially on input to the emulator. Mouse tracking is problematic unless you config the Linux pointer to disappear or run an Amiga program called *Mousehack* to tie it to the Amiga one.

Compatibility

UAE runs most of the Amiga software from the 1980s and many later titles, but it's far from perfect even on ostensibly 'fast' PCs. Most classic games and demos give recognisable if not identical results, but I found *Octamed6* sequenced sound was unlistenable at any quality setting on my machine, stuttering even on simple four-channel modules. *Deluxe Paint IV* AGA ran OK, if slowly. *Vista* got stuck while 'Initialising' and didn't render anything. I got the French A1200 demo to play without obvious sound and graphics glitches by reducing the render rate to one frame in four in the default window on my Debian 2.2 Linux box with SoundBlaster Live and Matrox G400. Attempts to render frames faster led to jerky graphics and sound, yet this demo runs smoothly on a basic A1200 with a 14MHz 68020 sharing access to 2MB slow memory.

Jerky sound can be tuned up from the command line. These parameters allow double the buffer size and the lowest common sampling rate you can use without shredding samples. The size and rate parameters are not on the setup tabs but once set from



The new look for AmigaOS on UAE and real Amigas with the latest 3.5 or 3.9 system release, Kickstart 3.1 and 'newicons'.



Some of the open sourcerers who develop UAE.

the command line you can save them with the default config from the GUI. Editing the `~/uaerc` file gives the finest control.

Bottlenecks

UAE is an interpreting emulator; overhead of instruction decoding is high, especially on PCs, architecturally limited compared with *UAE*'s original RISC workstations. A 65,536-entry despatch table chews up data cache, and 70 Pentium instruction dispatch slots are consumed in decoding before starting to interpret each 68K instruction, which can do a lot more per cycle than an x86. *UAE* also vets every memory access, using another cache-churning table of 65536 pointers, to mediate memory-mapped ports.

'Just In Time compilation' of code blocks drastically reduces such overhead. Extra memory holds translated blocks and an index. After each block JIT emulators check to see if the next is pre-translated. If so, it can be run without decoding; otherwise it is translated and buffered. JIT requires at least 6MB for its tables, and prefers 12MB. Some programs benefit from even more.

The 64K word table is used a bit more cleverly by the JIT engine, and 2.4 Linux kernels let the host Memory Management Unit help with the sifting. Another speedup comes from using the CMOV Pentium Pro instruction. Despite these complications part-translated combination code can run programs several times faster than the interpreting code. Register-bound loops may outrun the fastest 68060s on a fast PC.

JIT variations

uae-jit is a pre-beta compiling version of the emulator. It only emulates 32-bit 68Ks and relies on defensive programming to flag the need for cache flushes, upsetting many old Amiga apps. The JIT code only works on x86 hosts, though the technique may be easier to implement on other processors, and cannot handle the PowerPC code in the fastest apps written for dual-processor Amigas. The JIT releases, named after UK football teams, lack AGA chipset emulation and access to host TCP/IP, but can compile FPU code as well as that for 68020, 68030 and 68040 integer units. They even support a second PCI graphics card, dedicated to the emulation, using Amiga Virge 3D drivers and XFree86 v4 tweaks. The JIT patch has excellent docs.

The author now develops a commercial emulator, *Amithlon*. This only runs on Pentium Pro or later PCs, using a custom Linux kernel, emulating AmigaOS but little classic Amiga hardware – just enough to put up a grainy boot menu, after that you are dependent on emulation of third-party Amiga add-ons for peripheral access. However you do get fast nVIDIA and Matrox graphics, plus easy access to PCI and USB devices rarely available on real Amigas.

Amithlon only runs a small minority of Amiga games, but storms through power applications from the mid-Nineties like *ImageFX* and *Audio Evolution* that leverage add-on sound and



Amiga 1500 – the A500 is reborn with revolutionary "two floppy drives as standard" format.

« graphics. *AHI* and *Picasso96* extensions divert pure software requests from the Amiga hardware to PC cards. Licensing arguments have stifled *Amithlon* sales. We've waited a year for a review copy, or even demos.

Amithlon has split the Amiga scene; for some AmigaOS is all that matters now, and a fast CPU emulation on new hardware is more important than compatibility with the old Amiga; others argue it's not an Amiga without the custom chips, or the PowerPC chosen for future Amigas. While *Amithlon* only runs a late subset of Amiga programs it does so fast and expandably, and shows a neat way to embed Linux, reminiscent of EC64 (*LXF15*) although CD rather than floppy-booted.

Amithlon emulates *Intuition*, the Amiga GUI, rather than the Amiga as most people remember it. JIT compilation speeds up the CPU dramatically, though not entirely reliably, but does nothing for the bigger bottleneck, custom chip emulation. I had to drop the redraw rate to one frame in two to run the A500 game *Turrican 3* on *UAE*. Sound and scrolling were still grainy by real Amiga standards, though the game was playable. Graphics diversions like *DGA* and *SVGLib* avoid the substantial overhead of X, but custom chip emulation, for sound as well as graphics, still burns far more *UAE* cycles than CPU when running typical Amiga programs.

The interpreting *UAE* will seem sluggish to anyone who has used an accelerated 32-bit Amiga, but the custom kernel and direct drivers in *Amithlon* mean it feels like a very fast Amiga with 32-bit peripherals.

Amigaesque

There are ways to get an Amiga-like environment on Linux without running a full emulation. *Amiwm* is an X window manager modelled on the Amiga *Workbench*. It can move, position and resize windows and supports multiple screens with backdrops, which can be dragged up and down as on the Amiga. *Worker* and *Gentoo* mimic Amiga *Workbench* alternatives like *SID*, *DirWork* and *Directory Opus*. These two-window file managers are ideal for computer housekeeping if you grew up with the Amiga equivalents, and they are configurable to launch your favourite Linux tools.

Latest UAEs

UAE1.0 hovers on the horizon. Current 0.8.x betas add support for AGA playfields and sprites, though collision detection remains a



This *AmiWM* desktop brings Amiga sliding screens to X.

weakness. Audio is better, though even more CPU intensive, and direct SCSI access allows CD burning from the AmigaOS emulation, via a virtual 'uaescsi.device' that supports direct SCSI commands but not (yet) filesystem access. This should suit CD burners and scanners, but not disks, which are better accessed through the Linux file system.

Experimental snapshot code can save and restore the machine state; this is good for floppy-based games but little use for applications because it can't save or restore the hard disk emulation state. Emulated drives are now called DHn rather than UAEn, matching Commodore defaults, and work more like the real thing. Copper co-processor emulation is more accurate and the 68040 CPU is emulated, though the choice is academic as the instruction set and speed are the same.

UAE is a very accomplished emulator and if you've never used an Amiga you'll find it an impressive system. It won't put my A4000/060 into retirement yet, but the gap is closing. Your coverdisc holds *UAE* versions, patches, demos, games and utilities – why not track down a *Kickstart*, *Amithlon* or *Amiga Forever* CD, and try for yourself?

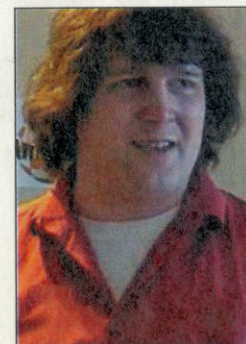
The end is nigh

Next month we get into *MESS*, the *Multi Emulation System*, in the final column of this series, plus a round-up of a dozen or so other emulators that might otherwise slip the net.



Original Amiga *Workbench 1*, with graphical glitches caused by *XFellow* infelicities.

only 320 low-resolution pixels per line and the emulated mouse tagging the Linux one in the wide borders. This format is adequate for old games, and contributes to the speed of *Fellow*, but unless you have a sluggish Intel-based Linux box, *UAE* is likely to be more useful to you than *XFellow*.



Amiga emulation JIT maestro Bernie Meyer.

XFellow 0.03

Ported from DOS

XFellow is a port based on *WinFellow 0.4.2*. *Fellow* was designed for MSDOS so the port only runs on x86-compatibles. *Fellow* was derived from portable *UAE* source, adapted because that was sluggish on PCs because it used C exclusively and did not take shortcuts to the PC hardware. Chunks of *UAE* were re-written in assembler, simplified and bodged for mid-Nineties commodity hardware, yielding an emulator which was less stable and compatible but could run many Amiga programs on old PCs that would only crawl if asked to run them in *UAE*.

XFellow requires *NASM 0.98* for the assembler parts, as well as *GCC* for filesystem support, and *SDL 1.2*. Nine *GTK+* tabs configure aspects of the system. It emulates 68000 to 68030 processors, including cut-down *EC020* and *EC030* versions, at the standard A500 clock rate or half, double or quadruple that.

XFellow can play 8 or 16-bit mono or stereo sound at rates from 15650 Hertz, one sample per TV line, up to the CD rate of 44.1KHz, optionally diverting output to a WAV file. The default writes to 'dev/dsp' via OSS. Two gameports accept Linux key, mouse or joystick input, vital for many Amiga programs. Up to four floppy disk images are configurable. I couldn't get the hardfile and host file system controls to do anything, and some configuration table options are ignored.

XFellow is slow and prone to graphics corruption, hard to exit without closing the launch shell or bombing the emulator. 'unhandled event messages' fill the shell as it runs; it tends to make temporary files with MSDOS paths like C:\ prefixed, and segfaults if the ROM is not found. The X version lacks docs. I did get the *Kickstart* animation and a *Workbench*, but the Amiga screen occupied just the middle quarter of a big window with

Amiga fuel

The Amiga OS is built around a kernel ROM called *Kickstart*, 256KB in the early *Kickstart1* machines. 512KB *Kickstart 2* and 3 versions were more capable and configurable, though most *Kickstart 3* images require a 32-bit processor.

Kickstart, like AmigaOS, is still on sale and you need a copy from an Amiga or the officially-licensed *Amiga Forever* emulation pack to emulate an Amiga legally. *UAE* includes skeletal *Kickstart* emulation, enough to run a few simple demos, but you need the real thing to launch a desktop or shell commands. Once you've got your *Kickstart*, *UAE* requests a bootable disk by showing a picture of a floppy. The next hurdle is that the Amiga disks process a track at a time, rather than dividing tracks into small sectors with gaps. Amigas automatically synchronise to read one track every rotation, whatever the initial position of the disk.

PC disk controllers can't work this way, so you can't insert an Amiga floppy and use it like one in a FAT or HFS format. Amiga hardware can read and write those formats, and any other, but you need a real Amiga.

Floppy tools

UAE, like other emulators, can use images of the contents of a disk held as Linux files. *transdisk* and *trackwizard* read and write such 'ADF' files on real Amigas. You need to compress the result to fit it on a 720KB disk for transfer to Linux, using an AmigaOS port of ZIP, *GZip*, *BZ2* or Amiga standards *LHA* and *LZX*.

CrossDOS, bundled with AmigaOS 2 and 3, and the free *FAT95* or *Messydos* from Aminet, will format, read and write PC-compatible floppies. You can pack files onto those and transfer them either way via *mttools* or the Linux FAT mounter. If your Amiga has an HD floppy there's no need to compress the 900, 120byte ADF file for 1.44MB media. Amiga-format high density floppies hold 1760KB so they will need compression or splitting.

The Amiga's own compressed disk image format, DMS (short for Disk Masher), can run in the emulator, reading and writing ADFs to any emulated floppy drive. The Linux tool *readdisk* moves files from an ADF to your local filesystem, but only understands the old Amiga *Kickstart1* file system, not the later Fast File System the custom systems on commercial releases. To read FFS disks:

```
mount df0.adf /df0 -t affs -o loop
```

This mounts disk image 'df0.adf' at the mount point '/df0', created as a subdirectory by a super-user.

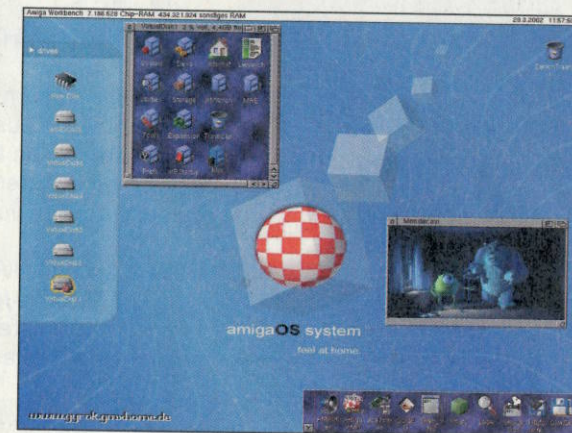
The *adskutil* Perl script gives FTP-style command line access to the contents of ADF files. It supports **LS**, **CD** and **GET** but lacks **PUT**, **DELETE**, **MKDIR** and other commands to change the disk, so it's really only useful for looking inside ADF files without loading the emulator though it is a good guide to Amiga disk internals.

UAE's **-m** option allows native file directories to be accessed from within the emulator:

```
uae -m Work:/home/simon/amigawork
```

Put Aminet 'lha' archives there and unpack them within *UAE* to preserve file attributes. Unpacked files can be accessed more quickly, as there's no need to emulate floppy rotation, but some loaders use those timings and insist on reading an ADF image.

There are two hardware ways round the disk incompatibility. The *Disk2fdi* hack may let you read 880K Amiga floppies from the command line or by patching later *UAE*s. It switches a PC disk controller abruptly between a second drive of any type, with an



arbitrary blank 'scratch' disk in, and a 3.5" disk with an Amiga disk in it. Running the disk controller at double speed, to pick up clock as well as data bits, with luck and a following wind this snarfs a snatch of Amiga data in a format that the PC can unravel later.

Disk2fdi is worth a try, but won't write Amiga disks and can't read some at all. There is a reliable answer for those prepared to buy a special controller. *Catweasel* from Individual Computers lets you read and write Amiga floppies directly in a Linux shell or *UAE*. This ISA card and its open-source drivers can handle old FM and MFM (1771, 1793, etc.) disk formats as well as the peculiar Amiga ones, in single, double or quad (1760K) density; the hardware supports old Commodore and Apple GCR formats too.

Wired links

Another way to transfer files is via the serial ports – an original Amiga can manage rates up to 19,200 baud with a properly wired RS232 cable, and accelerated 32-bit models can drive the motherboard port reliably at up to 57,600 baud. *Term* for AmigaOS and *Minicom* for Linux work well together.

Power Computing and Eyetech sell Ethernet kits with *Samba*, and there are NFS ports too. Since Linux can read and write Amiga hard drives, the quickest way to move a lot of data may be to move a drive or ZIP or similar disk between systems. Linux can mount Amiga hard disks using the **-t affs** switch as long as you have 'Amiga Fast File System' support in your kernel:

```
mount /dev/sda1 -t affs -o rw /amiboot
```

```
uae -m Workbench/amiboot
```

Once you have a bootable system you'll find mounds of free software for your emulated Amiga. Gigabytes of well-organised Amiga files, rarely crippled or undocumented, appear online in the Aminet archive, or on about 50 CDs. Aminet trounces any other platform's single-source site. **LXF**

Links

Amiga accessories: www.eyetech.co.uk
Amiga CDs: www.schatztruhe.de
Amiga Forever: www.cloanto.com/amiga
Amiga hardware: www.powerc.com
Amiga software: www.aminet.net
AmiWM:
www.lysator.liu.se/~marcus/amiwm.html
Catweasel Linux: www.soundtracker.org/raw

Disk2FDI: <http://fast.emuunim.com/disk2fdi>
LXF Emus online:
<http://simon.mooli.org.uk/LXF>
MyMiggy launcher:
<http://mymiggy.sourceforge.net>
UAE home: www.freiburg.linux.de/~uae/
UAE-JIT: <http://byron.csse.monash.edu.au>
XFellow: <http://xfellow.sourceforge.net>

Amithlon boots a tiny Linux kernel and JIT 68K compiler from CD into a PentiumPro, turning it into a fast new-generation Amiga which lacks Classic Amiga compatibility.



Amiga 600 – bring it down from the attic, and relive the glory days of Amiga's (and your) youth.