LOOK!
Find the power tools mentioned in this feature on the *iCreate* CD

Turbocharge your Mac

Discover how to make your Mac purr with this power user special to tweaking, honing and fine-tuning your system. Rob Buckley is your guide to the remarkable innards of OS X

Remember the first time you saw a Mac running OS X? Wasn't it beautiful? Wasn't it amazing to finally lay eyes on a computer that didn't make you want to head for the toilet with a nauseous feeling in your stomach?

Most people would be more than content to have a computer that didn't make them think they'd been wearing beer-goggles when they'd bought it. But what few Mac owners realise is that what they picked up at the local electronics boutique pretty much has the equivalent of a doctorate in astrophysics, an in-depth knowledge of the works of Shakespeare and a fascination with pre-Columbian art. It's no exaggeration to say that there are many layers to a Mac.

Beneath all the eye candy, there is a whole lot going on in OS X. At its heart is something called Unix, which is a computer expert's system of choice. Worked on over decades, Unix is very powerful and has thousands of tools and applications written for it. Even more importantly, it's very customisable and the power user can usually just edit a text file or add a few items to a command to make Unix do exactly what he or she wants it to do. But there is a catch. Quite a big one. Unix is also very complicated. For one thing, by default, there's not really any graphical way of using it – you just have to memorise whole series of arcane commands, just as if you were learning a foreign language. And instead of there being a logical, consistent way of doing things, almost every element of Unix has its own way of going about its business that's very much dependent on the original programmers that designed it 30 years ago.

So, when Apple decided to build OS X using a Unix core, Steve Jobs and his associates knew that the computer for "the rest of us" couldn't really expose "the rest of us" to the full-on Unix experience without "the rest of us" running away in horror. So they hid it, simply letting the eye candy do all the heavy lifting for us. Unix is still very much a part of your Mac, it's just that you don't have to worry about it.

There's always a trade-off, however, between simplicity and power in the computer world. Sure, the OS X interface with its collection of utilities ▶

and system preferences can configure parts of the Unix core. But it can't configure all of them, since there would be literally hundreds of different choices to pick from and no novice user would be able to use a Mac. For power users, though, this can be frustrating since they often want to be able to change the way a Mac runs in a way that Steve and Co either didn't think was necessary, would be suitable for the general Mac-using population, or that they just didn't think of. Fortunately, there are still ways to get to the Unix underneath to make your Mac go faster, remove bits you don't want, and ensures it runs in the way you want it to. And that's what we're going to show you in this article.

Plagued by daemons

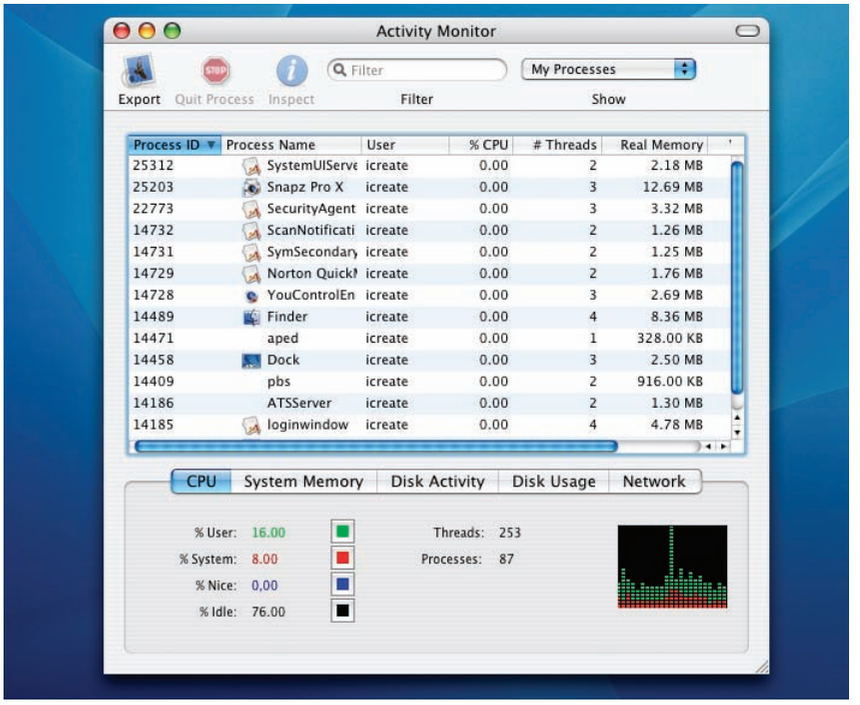
There are, essentially, four big bottlenecks in your Mac's performance: its processor, its hard drive, its memory and its interface. The processor (or processors in some Macs) makes all the calculations necessary for programs to do their jobs, so the faster the processor, the faster the programs go. There's not much you can do to make your processor go faster: while you can upgrade some Macs with better processors, most of the time you have to stick with what you've got. But there is some room for performance improvements here.

The processor doesn't do everything at once. In conjunction with a part of OS X called the 'kernel', it schedules tasks, determining how many times per second it will perform calculations for each program. So, there are two ways that you can go about cranking more performance out of your processor: run fewer programs or give a higher priority to the programs you care about.

Running fewer programs may seem an obvious way of speeding things up to most people; after all, having full-screen movies playing at the same time as you're running Virtual PC, Photoshop and GarageBand generally brings most Macs to their knees, and once you've tried something like that once, you rarely try it again. But there are a whole load of programs that you may not realise are programs at all, since they don't show up in the Dock. These include the various 'daemons' that manage things like Printer Sharing, Personal Web Sharing, Windows Sharing and so on. Some of these sit in memory the whole time, while others wait until a 'master daemon' (on earlier versions of OS X, it was the 'inetd' daemon but Jaguar and Panther use the 'xinetd' daemon) spots a request for your printer, files, et al and launches the appropriate daemon. Even when these daemons aren't actually doing anything, the processor still donates some of its time to them so they can work out what to do.

Similarly, the menu bar icons for things like AirPort signal strength, sound, iChat, Bluetooth, and battery charge all take up a little bit of extra processing power.

And that's just on a standard OS X installation. Once you start installing other programs, the daemons start to proliferate, particularly if you've installed System Preference panels, do things at specific times, or intercede for specific actions: the Retrospect back up software has a daemon to tell it when to launch; Norton Utilities and Norton Anti-Virus have a massed army of daemons to scan for disks being launched, files being used, files being thrown out and so on; and Stuffit Deluxe has daemons for its Archive Via Rename tool, its 'magic menu' and its Archive Assistant.



Activity Monitor is a useful little program that sits in your Utilities folder and allows you to monitor which programs are running and how much processing power they're hogging

> "Few Mac owners realise that what they've picked up at the electronics boutique has the equivalent of a doctorate in astrophysics and a fascination with pre-Columbian art"

Re-nice and easy

So if your Mac starts to feel a bit sluggish compared to when you first had it, you might want to start looking around to see what programs you typically have running and whether you need them. One of the best ways to do that is to use the Activity Monitor application in your Utilities folder, and set it to display 'My processes'. If there are many more processes than you thought there would be, look into ways of disabling them, by uninstalling software you don't need, or removing programs from your Startup Items (in the Accounts System Preference pane) or from your StartupItems folder (in both your own Library folder and the main Library folder).

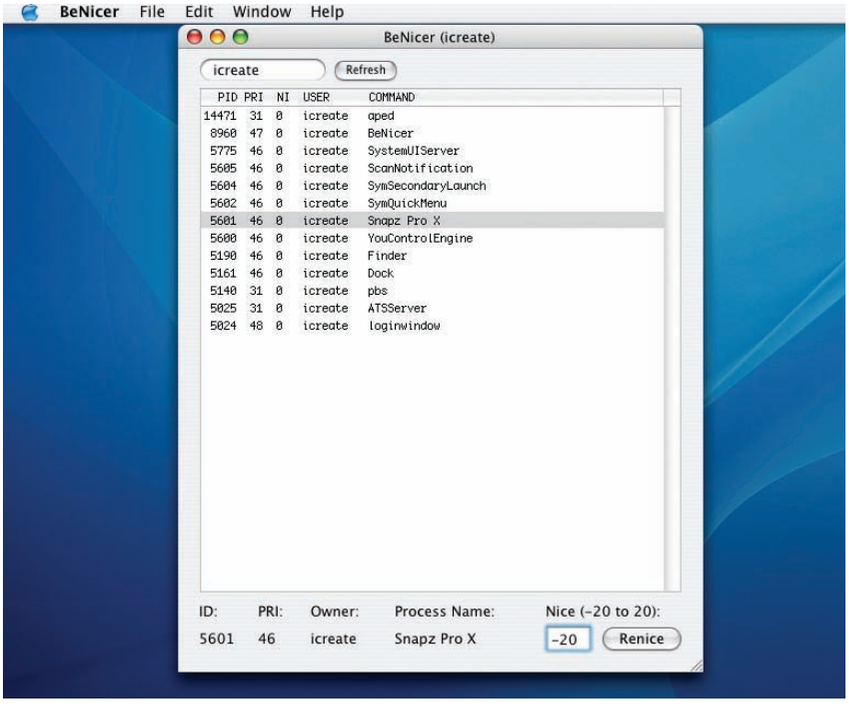
The other way to speed up your programs is to reschedule them. There may be some programs that you leave running in the background that you'd rather didn't get as much processor time as they receive, and there are others that you might want to give some more to. To change the priorities, you're going to have to 're-nice' these programs.

Amazing iLife hacks

One of the great things about OS X is that even when programmers hide things from you, it can often be possible to find them and get them working. iDVD is only supposed to work with Macs that have SuperDrives. However, by placing a couple of files (downloadable from **http://tinyurl.com/2fzx2**) in your home folder, you can make iLife work with any DVD burner – a feature the iLife programmers turned off but left in the final version of iDVD.

Similarly, iPhoto 1.1 had both crop and enhance tools, and Apple's own developer tools let you enable them (see how at **www.powerpage.org/story.lasso?newsID=9372**).

The iLife installer prevents iLife from being installed on certain systems, including OS X Server, even though they work fine on them. But if you use Disk Utility to make a read-writable image of the iLife DVD and mount it, you can type the following command: 'cd /Volumes/Install\ DVD; find . -name InstallationCheck -print -exec rm -rf {} \;' and iLife will install on your system just fine.



Be nice to me: BeNicer enables you to 're-nice' a program and prevent it from stealing all your computing power when you want to do something else, like run Photoshop for example

'Re-nicing' is a reference to the Unix command 'renice', which allows you to change the priorities of your programs. Unix grew up in a time when all computers were multi-user systems and you had to pay for the time you used on a machine, so being able to dictate which programs got priority was important. It's less important on a single-user Mac but the Unix legacy is still there for power users.

But most Mac users don't want to have to learn Unix – probably rightly so – and there's a small cottage industry growing up in developer circles of creating applications that provide simple, graphical ways of performing Unix commands. Re-nicing – a single Unix command – already has at least four programs to manage it from a graphical interface, including BeNicer, Speed Freak, Prio and Renicer, and that's pretty much a norm for some of the more useful Unix commands. Run any of these, pick which applications you want to give greater priority to and you should notice a speed improvement.

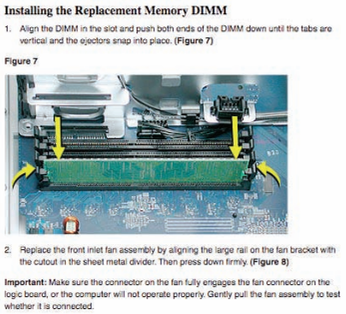
The memory game

Surprisingly though, despite all those Intel adverts, it's not the processor that's the biggest bottleneck on most Macs. Memory and hard drive can exert more of an influence, and fortunately they're far easier to affect on a continuing basis than the processor.

If you bought a Mac and haven't added any memory, get some more added now. It's quite easy to do yourself if you follow the instructions, but there are plenty of Mac shops willing to add memory for a price. The base memory on most Macs is still lamentably low and the bare minimum should be at least 256Mb, but 512Mb to 1Gb of RAM are far more palatable.

If you don't have enough memory, OS X will start trying to juggle your applications around using some sophisticated compression techniques and eventually 'virtual memory'. Virtual memory is a euphemism for great big files, known as a swap files, on your hard drive into which OS X will dump memory content when it doesn't have enough space to keep it in its memory. When it needs it back, it'll read the data from the swap files into the main memory again. The thing is, not only is your hard drive a lot slower than your memory, you're also using your hard drive for storing files; if you're constantly reading and writing files and constantly using the swap files, your Mac's going to get very slow as the hard drive reader jumps around reading and writing information from all the different locations on your hard drive.

You can use a utility like Xupport to move your swap file to a different location and manage your memory better



It's pretty easy to add more memory to your Mac yourself, especially with all the online help available

Improve your memory

So what can you do – other than add more memory – to solve this problem? If you have a lot (at least 1Gb of memory, but preferably 2Gb), you could think about turning off the swap files altogether, but since your Mac will freeze completely the very second you run out of free memory, that's not a good idea for most people. (For advanced users only: it is possible by editing the file /etc/rc to remove the commands that activate the swap files.) The other option is to move the swap file. If you've partitioned your hard drive (that is, split it up using Disk Utility so that it appears in the Finder as two or more separate disks), you could reconfigure the swap file to appear on a partition other than the one you have OS X installed upon. This will avoid any performance drops caused by disk fragmentation. However, the performance gains from this are slight since the swap file is on the same hard drive and so the hard drive reader will still have to jump between the different areas to read and write to the swap file.

A better option is to move the swap file to a different hard drive, since your Mac will be able to read and write to both drives simultaneously. It's possible to do this using standard Unix commands and a text editor, but a number of applications exist to do the hard work for you, such as Xupport, Swap Cop and Swap Relocator.

You'll get similar performance gains for similar reasons if you move your 'Users' directory to ▶

"Surprisingly, despite all those Intel adverts, it's not the processor that's the biggest bottleneck on most Macs – memory and hard drive can exert more of an influence"

Unix jargon explained

Command line: What you get when you run Terminal. It allows you to enter text commands to instruct your Mac.

Cron: A daemon that executes commands at specific times.

Daemon: An application that sits and monitors operations behind the scenes – few daemons appear in the Dock.

Permissions: Each file and folder in OS X has permissions that determine who can read, write or run files.

Pipe: To pass the result of one command to another. So, you could list all the files in a folder using 'ls' and then break the result up into pages using 'more' by typing 'ls | more'

Process: While a program or command is running, it is called a process.

Root: Root is the main directory of a hard drive or the super-user. The super-user can change any program or file, and even delete the entire system!

Shell: A shell determines what commands work, their syntax and their standard behaviour. The default Panther shell is called 'Bash', while Jaguar and earlier used 'tcsh'. Accessing a Mac remotely is done using 'ssh' – the secure shell.

Switch: An additional parameter added after a command to change its function. For example, the 'grep' text search command is case-sensitive; 'grep -I' is not.

iPod hacks

The iPod may be the coolest music player around, but what many people forget is that it's really just a hard drive with built-in music software and headphones. And that means that you can use hard drive tools and Unix commands with your iPod.

Apple appears not to let you copy files from an iPod to your Mac. But if you open a Terminal window, type 'cd', drag your iPod's icon onto the window and press return, you can then type 'cp -R iPod_Control/Music/* ~/', and see a collection of folders containing all your iPod's music files appear in your home directory.

Similarly, the iPod uses some other files on its hard drive to determine its behaviour. If you repeat the above procedure on a 3G iPod but instead of the second command, type "touch iPod_Control/Device/_show_voltage", when you restart it, you'll see that the battery gauge has been replaced by a numerical voltage readout. Type 'rm iPod_Control/Device/_show_voltage' for the second line before rebooting to change back.

a separate drive. But unlike moving your swap file, no one has written a utility to automate the process, so you'll need to use the Terminal.

Terminal velocity

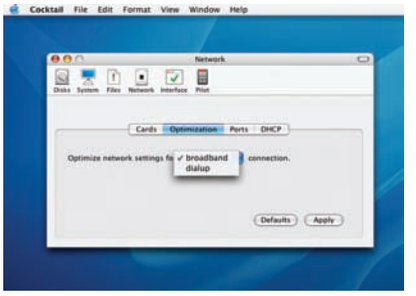
Terminal is a direct conduit to Unix and is quite frightening. Do not be frightened, however. With a few rules to guide you, the Terminal is not that scary. And it has lots of advantages once you get used to it. For one thing it's free, and is available on every OS X Mac, including those without an Internet connection. You can use it to log into other Macs that have Remote Login enabled in their 'Sharing' System Preference pane and you'll be able to use the exact same commands on those Macs. It can be a lot quicker than working your way through menus, tabs and other graphical delights. And if no one's written an application yet to tweak your Mac the way you want it, you can almost always use Terminal to achieve the result you want.

Rule number one of Terminal: all commands are case sensitive, so typing 'cd' is not the same as typing 'CD' – one will change your directory, the other won't. So be sure to follow any instructions exactly when dealing with Terminal.

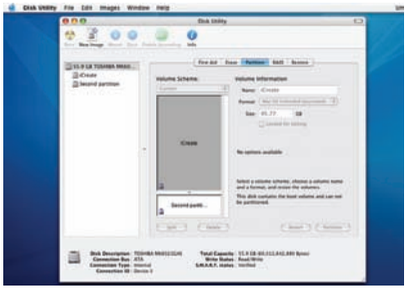
Rule number two: if in doubt, check the manual first. In typical Unix fashion, typing something obvious like 'help' (or 'HELP') won't work. To get help, you need to type 'man', which is short for manual. And you can only get help about specific commands, not about using Unix in general. So don't bother typing 'man' by itself. If you're not sure what command you want, type 'apropos' and a subject to get a list of possible commands.

Rule number three: follow the instructions exactly. For example, 'rm' – short for 'remove' – is the Unix delete command and will delete lists of files. So if you type 'rm cow1 cow2', the files 'cow1' and 'cow2' will disappear immediately (not into the Trash – there is no Trash in Unix, just oblivion) from your current directory (type 'pwd' to work out what that is). If there's an error, it will keep going. Now, you can also use things called 'wildcards' to save on the typing: typing 'rm cow*' will delete any file beginning with 'cow', including 'cow1' and 'cow2'. So what happens if you accidentally slip in a space and type 'rm cow *'? You guessed it: first, Unix will try to delete the file 'cow'. Even if you don't have one, it will carry on with the next item in the list, which is '*'. Since every filename consists of one or more characters, Unix will delete every file in your current directory (except for files beginning with a full stop). So make sure you've typed the command exactly.

Rule number four is to ignore the single quote marks we've used here for the beginning

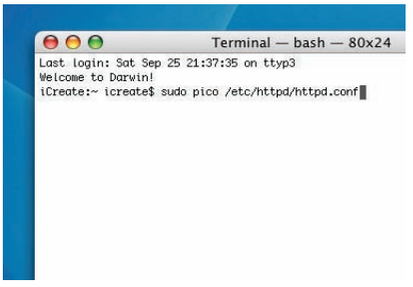


Cocktail party: third party utilities like Cocktail give you a graphic user interface that can be used to tweak Unix settings



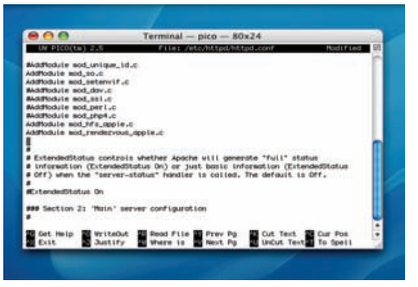
Partitioning your hard drive could help things run more smoothly on your Mac

Step-by-step | Terminal | Enabling PHP



1. Open the Apache configuration file
Panther comes equipped with the powerful Web scripting language, PHP. You just need to edit a file from an administrator account to enable it. First type 'sudo pico /etc/httpd/httpd.conf' in Terminal.

2. Remove the commenting
Use the various commands located at the bottom of the window ('^' refers to the Control key) to find the php4_module line and remove the '#' at the beginning/



3. Too many comments
Scroll down again to find the mod_php4 line and remove the '#' at its beginning. Save the file. Now you need to type 'sudo pico /Library/WebServer/Documents/info.php'.

> “The interface is actually a lot more customisable than you might think from the few options that are found in the Appearance Preference pane”

and end of each Terminal command. With that in mind, you can safely use Terminal to move your Users directory to another disk. Type the following, substituting your username for ‘username’ and the name of the other disk for ‘OtherDisk’:

```
sudo ditto -rsrcFork /Users/
Volumes/OtherDisk/Users
sudo niutil -createprop / /users/username
home /Volumes/OtherDisk/Users/username
```

The ‘sudo’ command lets you run any other command as if you are the super-user; ‘ditto’ duplicates a file; and ‘niutil’ accesses your Mac’s NetInfo database to make changes. These two commands have copied your whole ‘Users’ folder over to the new drive and then instructed the Mac to modify the NetInfo database to look in the new location for your home directory. (Repeat the second command, similarly modified for each user, if there’s more than one of you using the Mac.)

Don’t go on until you’re sure the new user directories are working. You should log out/back in to check this. Once you’re sure everything is okay, type the following to remove the old Users directory and put an alias to the new one in its place:

```
sudo rm -dr /Users
sudo ln -s /Volumes/OtherDisk/Users /Users
```

Interface hacks

The last and most overlooked aspect of performance-boosting is interface customisation. Yes, you might be able to shave a couple of seconds off your Mac’s swap file performance using Terminal hacks and third-party apps. But if the user interface hides things from you or isn’t arranged in the way you like to work, those couple of seconds will be nothing compared to the time lost each day due to bad usability. It’s why Mac users always turn out to be more productive than Windows users in surveys of working practices: it’s just easier to get things done on a Mac.

The interface is actually a lot more customisable than you might think from the few options that are found in the Appearance preference pane. Apple’s programmers have often taken executive decisions to hide certain things from us, because the average user won’t need them and would probably be confused by them. One of the default settings of the Finder, for instance, is to hide a whole slew of Unix directories and any files beginning with a full stop. It’s probably a good thing it does, because those directories would confuse the hell out of most people and files beginning with a full stop in Unix typically are quite important or store configuration data: indeed, OS X installs a directory called .DS_Store into any directory the Finder has looked at to maintain preferences on how the directory should be displayed in the Finder – the average user would probably delete any .DS_Store directory they found and then be mystified as to why the folder no longer displays correctly. But frequently the programmers decide to leave the features in and just remove the parts of the interface that would let us activate those features.

4. Check it works
Enter this text – ‘<?php phpinfo(); ?>’ – and save the file. Stop and start Web Sharing in the system Sharing preference pane then open **http://localhost/info.php** in a Web browser.

Make my list the plist

Now, virtually every OS X application has a preference file that it stores in the Preferences folders of your Library folders. Their filenames end in ‘.plist’, short for property list, so they’re easy to spot. Inside the files are lists of settings and their values. Often, you activate hidden features by altering the preference file. For example, if you alter Safari’s plist to include a string property called ‘IncludeDebugMenu’ with value 1, the next time Safari opens it will have a Debug menu that lets you do all sorts of useful things. To do this, you can use a text editor, such as TextEdit or BBEdit or, if you have the Apple developer tools installed, you can double-click on the plist file and it will open in an application called Property List Editor especially designed for editing plist files. Terminal addicts can use the command ‘defaults write com.apple.Safari IncludeDebugMenu 1’. Tools such as OnyX, Cocktail, Tinkertool and Xupport that provide these features are simply changing the property list files for particular applications through a nice, user-friendly interface.

Plist files, though, are OS X-specific. Most Unix preference files have their own formats, from the relatively easy ‘/etc/ftpusers’ file used by the FTP Access daemon that Apple appropriated from standard Unix, to the utterly baffling preference files of Sendmail (so baffling, it ▶

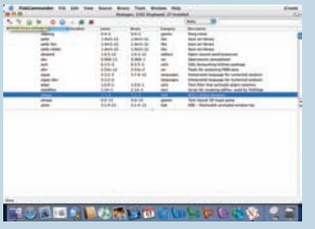
SIX OF THE BEST: OS X POWER TOOLS

Take full control of your system with these great power user apps

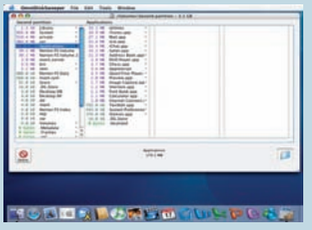
Monolingual
URL: http://monolingual.sourceforge.net/
Licence: Freeware
Free up hundreds of megabytes, if not gigabytes, of hard drive space by removing all those foreign language files most OS X applications contain.

Xupport
URL: www.computer-support.ch/Xupport/info.html
Licence: Shareware
Xupport can enable all those features of OS X that Apple has hidden away, and a whole lot more.

BrickHouse
URL: http://personalpages.tds.net/~brian_hill/brickhouse.html
Licence: Freeware
The built-in Firewall and Internet Sharing of OS X are great: BrickHouse allows you to activate them on pre-Panther versions.



Fink
URL: http://fink.sourceforge.net/
Licence: Freeware
Download and install thousands of Unix applications, including database programs, window managers and games without having to write a single line of code.



OmniDiskSweeper
URL: www.omnigroup.com/applications/omnidisksweeper/
Licence: Shareware
OmniDiskSweeper scans all your directories, even the invisible ones, to let you know where the big files are hiding.

Cronnix
URL: www.koch-schmidt.de/cronnix/description.html
Licence: Donationware
Useful for getting the system’s automatic maintenance scripts to run during the day rather than at night.

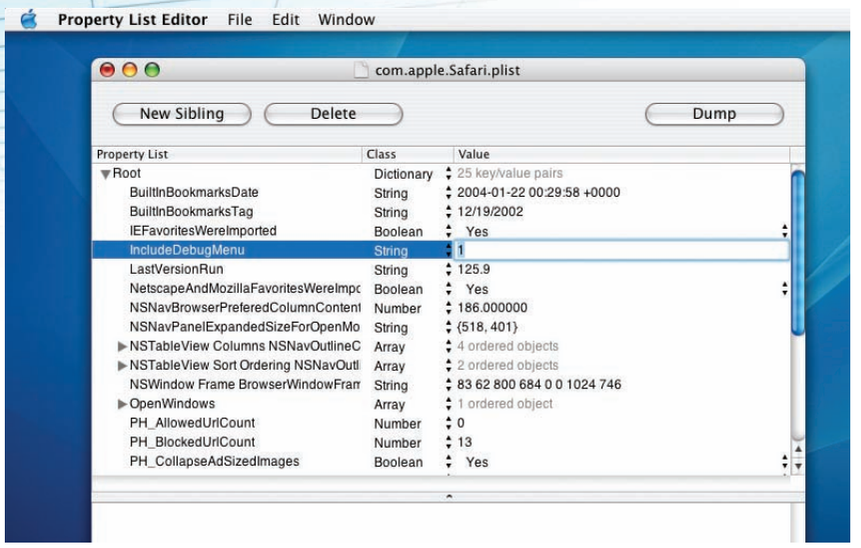
Web resources

Top destinations for more Mac tweakery pokery

Mac OS X Hints
www.macosxhints.com
If anyone hardcore has come up with a way of doing something in OS X, it'll be here, from full-scale shell scripting down to hints on using Address Book.

Apple Developer Connection
http://developer.apple.com/internet/opensource/
Apple has its own tutorials on how to compile and install lots of useful open source tools, including MySQL, PHP, Perl and PostgreSQL, on OS X.

O'Reilly Mac Center
http://mac.oreilly.com/
Not just an online book catalogue, the O'Reilly Mac Center also includes news and articles from experts about how to get the most from your Mac.



Hack into Safari and add a special Debug menu to your list of options by using Property List Editor

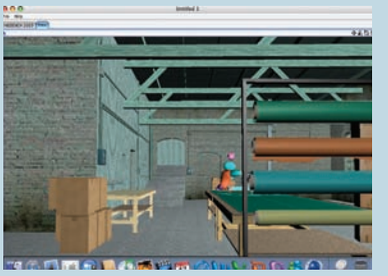
actually has two preference files: one you write yourself, and the one Sendmail actually uses which it generates from your file using a separate program).

Programs such as Broadband Optimizer and Cocktail are able to use the Terminal command 'sysctl', which accesses the deeper Unix system preferences (type 'sysctl –a' to see them all). They can use this capability to tailor OS X's networking capabilities to a far greater degree than the standard Network system preference pane can, potentially speeding up OS X's networking by a considerable amount. But unless you understand how to configure your TCP packets' memory space, stick with the graphical interface rather than try to use 'sysctl' yourself.

Look to the future now

Given the vastness of Unix, not just in terms of what's installed by default in OS X, but also the tools in the Apple developer toolkit and the thousands of pieces of free software you can download from the Internet, it would be impossible to list all the ways you can tweak and tinker with your Mac to get it running exactly the way you want it to. OS X doesn't stand still either. There were no hidden hacks to change the way Exposé worked until it arrived on the scene with Panther, which also updated a considerable portion of the Unix core.

OS X 10.4 should be no different, with Unix enhancements promised aplenty. Since Apple is taking a more leisurely pace getting Tiger out the door than it did with 10.2 or 10.3, there should be more features and more consideration of what aspects of the Mac a user should be able to configure. There will also probably be more utilities to replace the command-line tools power users have had to fall back on, just as Activity Monitor replaced 'top' and 'df' and Network Utility replaced 'ifconfig', 'ping', 'traceroute' and 'whois'. But unless Apple loses the plot and destroys the simplicity of the Mac interface with the multitude of incomprehensible options that plague the Windows and Linux interfaces, there will always be a place for customisation utilities and Terminal hacks in OS X. ■



Benchmarking your system

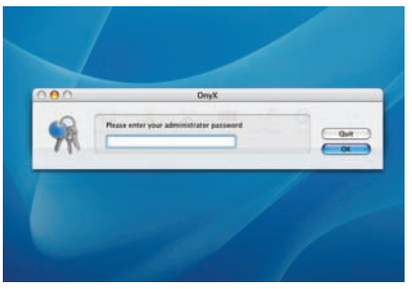
Sometimes your Mac feels slow for some reason. You think back to using that old LCII at school 10 years ago, and somehow it seemed faster. But how can you be sure? Benchmarking is a way to measure the speed of your Mac so that you can compare it with other systems.

There are a variety of different benchmarking packages available, including Norton Utilities and the free Cinebench (**http://maxon.net/pages/download/cinebench.html**), but there's a great deal of controversy about them. How can you be sure the test is written equally well on each platform? Many benchmarking packages are optimised for Windows and Intel processors, and their Mac equivalents are not very well written so the tests run more slowly and give worse results for Macs.

It can also be hard to accurately compare Macs with older Motorola processors and OS 9 (or even OS 6, 7 or 8) and those with Power PC chips and OS X. This is even the case when comparing G4s and G5s, since software needs to be rewritten to really take advantage of the G5. So take the results with a hefty pinch of salt when comparing radically different systems.

What benchmarking really can do is show you whether any of your system optimisations have sped up your system. Run Cinebench before and after your optimisations to be sure what you're doing is helping, rather than hindering.

Step-by-step | OnyX | Access hidden features of OS X



1. Authorise OnyX
OnyX (**www.titanium.free.fr/english.html**) is a useful application for activating hidden features of OS X and doing maintenance. When you run it, you'll need to be logged in as an administrator.

2. Pick a task
Once everything's been authorised, you'll need to pick a task. The 'Appearance' button will let you change some of the default OS X behaviours to speed up the system.

3. Change the defaults
As well as changing the positions of arrows on your scrollbars, the Finder tab can disable zooming and animation of windows and application launches, speeding them up considerably.