

ATPM

10.07 / July 2004
Volume 10, Number 7



About This Particular Macintosh: About the **personal** computing experience™

Cover Art

Copyright © 2004 Andrew Kator.
We need new cover art each month. [Write to us!](#)

Editorial Staff

Publisher/Editor-in-Chief [Michael Tsai](#)
Managing Editor [Christopher Turner](#)
Associate Editor/Reviews [Paul Fatula](#)
Copy Editors [Raena Armitage](#)
[Johann Campbell](#)
[Chris Lawson](#)
[Elyn Ritterskamp](#)
[Brooke Smith](#)
Vacant
Web Editor [Lee Bennett](#)
Publicity Manager *Vacant*
Webmaster [Michael Tsai](#)
Beta Testers [The Staff](#)

Contributing Editors

[Eric Blair](#)
[Matt Coates](#)
[Matthew Glidden](#)
[Ted Goranson](#)
[Andrew Kator](#)
[Kirk McElhearn](#)
[Robert Paul Leitao](#)
[Wes Meltzer](#)
[David Ozab](#)
[Sylvester Roque](#)
[Mike Shields](#)
[Gregory Tetrault](#)
[Evan Trent](#)
How To *Vacant*
Interviews *Vacant*
Opinion *Vacant*
Reviews *Vacant*

Artwork & Design

Graphics Director *Vacant*
Layout and Design [Michael Tsai](#)
Cartoonist [Matt Johnson](#)
Graphic Design Consultant [Jamal Ghandour](#)
Blue Apple Icon Designs Mark Robinson
Other Art RD Novo

Emeritus

RD Novo
Robert Madill
Belinda Wagner
Edward Goss
Tom Iovino
Daniel Chvatik
Grant Osborne

Contributors

Lee Bennett
Eric Blair
Matthew Glidden
Ted Goranson
Andrew Kator
Greg Maddux
Matt Johnson
Robert Paul Leitao
Gregory Maddux
Wes Meltzer
Ellyn Ritterskamp
Sylvester Roque
Evan Trent
Christopher Turner
Charles Wu
Macintosh users like you

Subscriptions

Sign up for **free** subscriptions using the [Web form](#) .

Where to Find ATPM

Online and downloadable issues are available at <http://www.atpm.com>.

ATPM is a product of ATPM, Inc.
© 1995–2004, All Rights Reserved
ISSN: 1093-2909

Production Tools

Acrobat
Apache
AppleScript
BBEdit
CVL
CVS
DropDMG
FileMaker Pro
FrameMaker+SGML
GraphicConverter
Mesh
Mailman
Mojo Mail
MySQL
Perl
Photoshop Elements
Python
rsync
Snapz Pro X
ssh
StuffIt
Super Get Info

The Fonts

Cheltenham

Frutiger

Isla Bella

Marydale

Minion

Reprints

Articles and original art cannot be reproduced without the express permission of ATPM, unless otherwise noted. You may, however, print copies of ATPM provided that it is not modified in any way. Authors may be contacted through ATPM's editorial staff, or at their e-mail addresses, when provided.

Legal Stuff

About This Particular Macintosh may be uploaded to any online area or included on a CD-ROM compilation, so long as the file remains intact and unaltered, but all other rights are reserved. All information contained in this issue is correct to the best of our knowledge. The opinions expressed in ATPM are not necessarily those of this particular Macintosh. Product and company names and logos may be registered trademarks of their respective companies. Thank you for reading this far, and we hope that the rest of the magazine is more interesting than this.

Thanks for reading ATPM.



Sponsors

About This Particular Macintosh has been free since 1995, and we intend to keep it that way. Our editors and staff are volunteers with “real” jobs who believe in the Macintosh way of computing. We don’t make a profit, nor do we plan to. As such, we rely on advertisers and readers like you to help us pay for our Web site and other expenses.

We’ve partnered with CafePress.com to bring you high-quality [ATPM merchandise](#). For each item you buy, \$1 goes towards keeping the atpm.com server running. You can also help support ATPM by buying from online retailers using [our links](#). If you’re going to buy from them anyway, why not help us at the same time?

We also accept direct contributions using [PayPal](#) and [Amazon’s Honor System](#). We suggest \$10 for students and \$20 for individuals, but we greatly appreciate contributions of any size.

Finally, we are accepting inquiries from interested sponsors and advertisers. We have a variety of programs available to tailor to your needs. Please contact us at advertise@atpm.com for more information.



Welcome

by Robert Paul Leitao, rleitao@atpm.com

Welcome to the July edition of *About This Particular Macintosh!* This month the iPod takes a letter step back and makes way for the "Hpod," HP's branded iPod product manufactured by Apple but distributed by the PC and printer maker. This month our e-zine takes a step forward and brings you all the latest news for Macintosh users.

Put a Tiger in Your Mac

With apologies to the old Esso ad campaign, June's World Wide Developers Conference was much about the next upgrade to Apple's OS X operating system. Code-named Tiger, the new OS sports several new features including optimizations for the G5 chip and 64-bit applications. Tiger will ship in the early part of 2005, more than a year before the anticipated release of Longhorn, the long-overdue upgrade to Windows XP.

It seems a bit ironic that Apple didn't stream the keynote address live over the Internet or by satellite. It's ironic because other major announcements at the WWDC included the release of new Apple cinema displays. Why would Apple choose not to display the new displays the moment they were announced?

We're in the Army Now

In late June the US Army announced the purchase of 1,566 G5 Macintosh Xserves for use in a super computer cluster to research the aerodynamics of hypersonic flight. As Apple's supercomputer division begins to take off, we expect similar purchases by universities and research facilities in the coming months.

The Apple iPod Invasion

ATPM readers over age fifty may remember the British music invasion of the early 1960s. Forty some years later Apple Computer is returning the favor. In the first week of operation Apple's European iTunes Music Store sold over 800,000 songs in the UK, France, and Germany, with more than 450,000 songs sold in the UK alone.

Splish, Splash, Napster's Taking a Bath

After crossing the Atlantic and opening a digital music store in Britain, Roxio's Napster unit appears to have been overwhelmed by the flood of business to Apple's iTunes Music Store recently introduced in Britain, France, and Germany.

Attempting to carve out business in the wake of Apple's success, Napster has recently announced a new deal with Best Buy that provides the retail giant with an opportunity to acquire up to 6% of Roxio's common stock.

In response to the iTMS introduction in Europe, even OD2, the previous European leader in legal online music distribution, agreed to a buyout and will now be owned by Loudeye. The corporate office will be moved from the UK to Seattle.

AAPL Hits New Highs

The European iTunes Music Store isn't the only new hit for Apple. As June came to a close, Apple's share price was hitting new multi-year highs. AAPL is now trading at its highest levels since the fall of 2000. After reaching a new multi-year high of \$34.18, AAPL ended the month trading at about \$33.00 per share.

The Ultimate Music Machine

Apple and BMW have partnered on a new iPod accessory, a \$149 adapter that allows BMW owners to control their iPod through the music controls on the steering wheel. The adapter cable is installed through the glove box. Apple calls it "The first seamless integration of iPod and automobile." We call it an expensive iPod accessory for luxury car owners.

AirPort Express: Music and Money Over Thin Air

Can Apple products defy gravity? Apparently that's so. The new Apple AirPort Express streams music from iTunes through your home stereo powered speakers. This latest "beyond the box" product from Apple is also designed to help Apple's stock price continue to defy gravity by adding to the high gross margins on products sold.

Signaling a change in product focus, Apple reduced the price on AirPort Extreme base stations and cards.

Audible Comes Through Loud and Clear

Audible.com continues to grow its business and its sales through the iTunes Music Store. The company is now signing contracts with colleges and universities to provide spoken-word content to students on campus. Watch in the coming months as more spoken-word and other non-music products find their way to iPod owners via of the iTunes Music Store.

Summer Fun

Although many Mac watchers were disappointed that speculation about announcements of new iMacs and iPods did not come true at the WWDC, the first full month of summer promises to be a hot one for Apple product users. We'll be back next month with a look at Apple's quarterly financial results. Until then, please enjoy our July issue. This month's issue includes:

The Candy Apple: Take Me Out to the Ball Game

Ellyn Ritterskamp digitizes a big moment.

Bloggable: Apple in the Balance

Wes Meltzer's musings for this month discuss the crossroads Apple seems to be at, specifically in regard to security and vulnerabilities, the priority of music for Apple, and Web browser innovation. There's more in the Mac blogosphere: haxies may not be all that bad after all, what potential Switchers want to hear from other Switchers, and when does the iTMS censorship filter go too far?

Outliners: Three Topics on the Future of Outlining

After an unexpected problem last month, Ted Goranson returns with his *About This Particular Outliner* column. This month, he deals with three subjects related to futures: the potential for outlining when Frontier goes open source, the new territory of Web Outlining, and the state of outlining applications as a basis for the new *ATPO* feature on outliner news.

Networks in Action: Using WEP Security on an AirPort Network

Don't share your network connection with the world. Find out how to enable Wired Equivalent Privacy and keep your network data private.

How To: Summer Photo Fun

Reader Charles Wu shares an idea to make fun—and very personal—photo postcards.

How To: What to Do When You Need a Little More Room

Freeing up space on your hard drive may solve more problems than you realize. Sylvester Roque has some tips to regain some of that space that don't involve simply trashing a bunch of large files that you may or may not want later.

Cartoon: Cortland

Todd and Angie mire within their PC network-based office. Meanwhile Cortland may get the last laugh with his old boss, Chad.

Cartoon: iTrolls

The iTrolls encounter 17-year-old bugs, a lack of coffee, Noah, and WWDC.

Desktop Pictures: Volcano National Park

This month's features part two of Christopher Turner's 2001 trip to Kilauea Volcano National Park on the Big Island of Hawai'i.

Review: BOOQ BP3 System

Christopher Turner reviews BOOQ's BP3 System: backpack, laptop sleeve, and mobile device pouches, all in one tidy sales unit. Holy PowerBook portability, Macman!

Review: iTalk

Between the time he received the iTalk and the time he wrote this review, Lee Bennett had recorded six short interviews, two telephone conversations (over the speakerphone, with the caller's permission!), and a half-hour meeting—all of which were later transcribed. Suffice to say, he *likes* the iTalk!

Review: Parliant PhoneValet

Thanks to this small piece of hardware, Andrew Kator has found a way for a small business to inexpensively acquire a professional phone logging and messaging center.

Review: SymmetryWorks 3.1 and Tessella 1.21

In preparation for next month's column on mindmapping outliners, Ted Goranson reviews two Illustrator plug-ins that create symmetry patterns.

Review: Wicked Cool Shell Scripts (book)

Are the rumors true? A book on shell scripting that claims to cover OS X and actually comes through? Eric Blair checks out *Wicked Cool Shell Scripts* to find the truth.



E-Mail

InfoSelect

Darren Eger's note about Info-Select raises a good point. There is nothing that I know of quite like IS for Mac, though Tinderbox is probably even more powerful since it combines the "go find it" capability of IS with the ability to create (and ignore) a structure if you want. DEVONThink is another product that has some of this flexibility, though I found its interface a little obscure.

Tinderbox, if you try to use it to the limits of its capacity, has a pretty formidable learning curve and is not a poor man's game—it's sale-priced at the moment at "only" \$95, but it is a remarkably powerful application and quite flexible. Ted wrote about it at some length in the series (speaking of which, it's too bad that we missed another installment—that's three passes since the series started [two equipment problem misses and the not-very-substantive April Fool's Day parody] makes me realize how attached to it I've become).

Darren, I'd suggest reading the [back issues](#) of Ted's articles posted on the ATPM Web site and focusing in on Tinderbox and DEVONThink. You can download trial versions of each. I downloaded my trial version of TB about a year ago and it was lamed a little, but still functional enough to give me an idea of what it could do and I bought it. The DEVONThink trial seemed unlimited and does not, I think, have an expiration date. I didn't wind up using it but I think my evaluation copy is still working.

—*Stephen Chakwin*



Darren Eger asks whether anyone knows of an OS X application that has the search abilities of InfoSelect.

My knowledge of InfoSelect is second-hand, but as I understand its real time search capability, the OS X program Hog Bay Notebook has it and seems designed around a method of working like Darren Eger describes.

—*Stephen R. Diamond*

Blender

After wading a little deeper, it seems that the option to render an animation works when using the `-a` instead of the `-f` command. Also, it does help if in Blender you *don't* specify a folder to render to. The rendered file will then either be placed in the root of your hard disk or in a folder at the same place where your `.blend` file is located.

—*Hoxolotl*

Mariner Write

I did a lot of trials with Mellel, Nisus Writer Express, and Mariner Write, and came to the same conclusion Paul did earlier this year. I think NWE 1.x is a fantastic *base* for a word processor, but there were some make-or-break features it was missing. Write reminds me a great deal of older Mac word processors like MacWrite II and WriteNow, and I mean that in a complimentary way. When NWE 2 comes out later this year, though, I'm going to have to revisit the comparison—adding styles, tables, and footnotes brings its basic feature set to a usable level and some of the unique features already in 1.x are, well, unique.

(It's possible that AbiWord 2.2 will be out by then and run correctly on OS X. It may be worth taking a look at, also—you certainly can't beat its price.)

—*Watts Martin*

Connecting

The company I work for just bought a G5 to update the graphics department, which works fantastic. Unfortunately, the IT department will not connect the Mac to the server because they fear A) that the Mac will not be hid behind the firewall and will leave a gaping hole for outside sources to sneak in by, and B) that they will not be able to control what the Mac user has access to.

I have tried to explain the security and accounts features on Panther to the IT department, but they are weary nonetheless.

I am hoping that you can either offer any advice or even any outside literature/Web sites/ anything on the subject of adding a Mac to a PC network with extreme security considerations.

—*neela*

If the LAN you are on has a firewall, what's the problem? If there is either a firewall on the router, or if there is a software firewall of some sort on the LAN, the Mac is no different than a PC. Firewalls don't care what is behind them. They simply block packets and allow traffic through specified ports, mapped to specified IP addresses. That's it. It makes no difference if you are using a Mac, PC, Unix box, or a PDA. It's all just packets.

Now if there is no firewall on the LAN, and your IT people are (unwisely) depending on each and every client to provide its own firewall via the operating system or some layer installed over it...then technically they do have a legitimate concern because they are not familiar enough with a Mac to understand how to turn on and configure the firewall that is provided by the operating system.

OS X has as good a firewall as any other operating system, better than many (certainly more secure than anything Microsoft has ever come up with) and it is very easy to engage and configure. You should be able to demonstrate this to them in a few mouse clicks using the System Preferences. Again, this is totally unnecessary if there is a firewall on your LAN. It would be redundant. Any IT guru worth his salt should understand that.

As for controlling access...are they talking about controlling what the G5 can access on the network server(s)? Or are they talking about limiting access to files/folders on the G5 depending on who is logged in to it? These are different issues.

First of all, if they are concerned about controlling access to the server(s)—they shouldn't be. As with the firewall, the Mac is no different than a PC. Presumably we're talking about a Windows file server. The fact that you are logging on from a Mac is transparent to everybody. It makes no difference whatsoever. Set up an account on the server(s) with your name, password, and privileges. Log on from the Mac. Log on from the PC. No difference. You will have access to those files you are granted access to, and not those which you haven't been granted access to. It doesn't matter what computer you log on from. This is a *server side* issue not a client side issue.

If the concern is limiting access to data on the actual G5 itself, this is what the multi-user architecture of the operating system is designed to do. OS X has a Unix based multi-user system, which allows you to create users and configure their user experience, including which resources their privileges offer them access to. It is analogous to the Windows multi-user concept although it is, frankly, a bit more elegant in its implementation.

In short your IT guys have no reason at all to be concerned. —Evan Trent

Copyright © 2004 the ATPM Staff, editor@atpm.com. We'd love to hear your thoughts about our publication. We always welcome your comments, criticisms, suggestions, and praise. Or, if you have an opinion or announcement about the Macintosh platform in general, that's OK too. Send your e-mail to editor@atpm.com. All mail becomes the property of ATPM.



Take Me Out to the Ball Game

Every now and then it's good to stand back and realize how good we have it. We should do it every day, really, but many of us don't make it that often. I had occasion to reflect on this idea recently and thought you might like coming along for the ride. I don't know that there are any profound revelations here, just me, marveling at how something that seemed miraculous five years ago, seems almost pedestrian today.

The short version of the story is: my mom and I sang not one but two national anthems at a baseball game. Twenty-four hours later, I e-mailed an audio clip of us to some friends and posted it on a Yahoo message board for some other friends and acquaintances. A week later there was a video, which might have been done a lot sooner had there not been a weekend and an out-of-town trip gumming up the works. News travels fast these days, even in the hands of someone like me who doesn't know very much about the nuts and bolts of technology.

The longer version is that over the past few years, I've begun making an effort to tackle things I'm afraid of. I've quit analyzing whether a particular challenge holds a fear of failure or a fear of success. Both can hold us back. The trick is to decide if there is anything to be gained by accomplishing the challenge, and if there is, to just do it. The worst-case scenario, in most cases, is that we try something and fail. Unless we are trying to be air traffic controllers, it's usually not a problem. We may be humiliated but usually not for long. A few exploits hang around to haunt us, but often they are the source of merriment at parties, rather than debilitating shame for years and years. Most of our friends admire us for trying stuff, and the ones who don't, well, maybe we didn't want to be friends with them anyway.

In December, I considered entering a contest where a TV network would make your dream come true. I thought my dream was to be a bullpen catcher for a day at a professional baseball game. That would have been fun, but I would have had to learn how to catch a fastball first, and then convince our local team to let me do it. I was driving to work with a new CD in the car about then and singing along with Clay Aiken's remake of "Bridge Over

Troubled Water.” I realized that my favorite moments at baseball games had little to do with bullpen catchers and everything to do with singing. We sing to start the game, and we sing in the middle of the game. Right then I decided I would see about singing the national anthem at our AAA team game this summer.

I checked the schedule and found out we still have one Canadian team in the league, and that team makes one visit here this year. I called my mom and asked if she would sing with me, and if she would learn “O Canada.” As she is a good singer and accustomed to leading the congregation at church, she was willing. We checked the Web site for audition information and started rehearsing.

Audition day came. I reviewed the worst-case scenario: the worst thing that could happen is that we were terrible, they would say thank you, and that would be that. We would still get to have lunch and sit in the sun. We’d still have our health. Nothing truly awful could happen. So we did it. Our hook about being able to do “O Canada” was useful, and they said they would call us.

They did. We were scheduled for the third game of the four-game series with the Ottawa Lynx, on a Thursday night. On Tuesday night, they called and asked if we could fill in on Wednesday afternoon. We said sure and viewed it as a free dress rehearsal. We did it that day, and my dad videotaped it. We were able to make some tweaks to our presentation based on the video, and the second go-round was even smoother. I guess this is the first example in this story of how technology made something better. Without the video, we would not have realized we needed to smile and look into the crowd more.

I held a digital audio recorder and got the whole thing, even some stuff that the crowd didn’t hear because of the microphone being off while we got ready. I sent the recorder home with my dad, who has a free Hotmail account with a 1 MB e-mail limit. Of course the file was 1.1 MB, so he couldn’t send it from home. He tried to burn it onto a CD but only got a directory file. I sent him back home with instructions on how to use Yahoo mail, as Yahoo had, *just that week*, upped its limit to 10 MB (it must have known we were coming). Once I got it, I posted it to a Yahoo message group with a couple of hundred online acquaintances and a few close friends. I also e-mailed it to half a dozen people. It was astounding to think about how connected we can be now, even miles away from each other. We can replay an event that took place a day before, in exquisite detail.

I had the 8 mm video transferred to a DVD and am in the process of figuring out how to get that online, too. I love learning stuff. I love that we can save these things forever: babies walking, kittens climbing, kids graduating, people singing anthems. Photos are great. Capturing a moment is tricky stuff. This business of saving and disseminating several minutes of a moment is a whole different animal for me. And I like it.

Copyright © 2004 Ellyn Ritterskamp, eritterskamp@atpm.com.



Apple in the Balance

Apple (and, by extension, the Mac) is not really at great risk, this month or any month. Rest easily.

Nevertheless, this past month suggests that Apple is at a crossroads. The iPod is ascendant; the recent security vulnerabilities have been fixed and explained in great detail; the future development of Windows and its components demonstrate that the company has a Robert Frost moment ahead of it. The present fork in the road is no greater a choice than many of the other decisions Apple has made at such points in the past, but that does not diminish the significance of what may lie ahead.

Steve Jobs, Gil Amelio, and John Sculley have, so far, led the company down the road less traveled, and so far, they have been the better for it. Today, they seem poised to take either road, and, in a striking change from the past, the road more often traveled could be the better choice. But that's for you—and, ultimately, I suppose, Steve Jobs—to decide.

In case you either live under a rock or didn't read [last month's column](#) (or both), there were a series of significant potential exploits detailed online in OS X, some of which were subsequently resolved in security updates. If you didn't know, read last month's column, please; for those who know, let's move on to the news.

[John Gruber](#) (apparently next to whom all other Mac pundits pale), provides an excellent analysis of what Apple's present, and future, could have looked like if they and the Mac user community were less vigilant. This is, he says, the "[broken windows](#)" theory, in which Windows machines regularly get infected because both Microsoft and the user community tolerate exploits regularly—and are prepared to clean them out—but not respond proactively. He chalks some of the difference up to market share, perhaps to please people like [Paul Thurrott](#), but argues persuasively that Mac users' response to these exploits—angry writing and other histrionics—demonstrates that we won't tolerate exploits. Sociologists (*inter multa alia*) have long applied the broken windows theory to

urban decay, saying that fixing windows as soon as they're broken demonstrates vigilance and zero tolerance; some Middle Eastern specialists and many Israelis make the same argument for express cleanup and normalization after a suicide bombing. Gruber says:

We all benefit from the fact that the Mac community has zero tolerance for vulnerabilities. Not just zero tolerance for security exploits, but zero tolerance for vulnerabilities. In fact, there is zero tolerance in the Mac community for crapware of any kind.

If some "freeware" software for the Mac surreptitiously installed some sort of adware/spyware/crapware, there'd be reports all over the Mac Web within days. Uninstallation instructions would be posted (and thus made available to all via Google), and the developer who shipped the application would be excoriated.

Zero tolerance, on the part of the user community, is the only policy that can work.

That doesn't mean that everyone agrees, in any sense. For instance, [Charles Miller](#) says that due to Metcalfe's Law, the ability of a trojan to spread rapidly is a function of the square of the size of the network, rather than the size itself. Consequently [it's only logical](#) that Windows' share of viruses far outstrips the Mac's. He does admit that there's no room for complacency, but says that there are plenty of active, nontolerant people who use Windows—they're just far outstripped by casual users who really, really want whatever the malware comes packaged with.

Although that seems reasonable, I don't think Metcalfe's Law sufficiently discredits Gruber's argument; after all, Mac users *do* care, and that's why exploits tend to be stopped in their tracks before anyone acts on them.

Reading so much about comparative security is a bit of a citation to *It's A Wonderful Life*, looking at what could be and then gladly returning to what is. Gruber wrote, the very next day, that Apple did [precisely what they needed to](#): they fixed the vulnerability quickly and documented what they did thoroughly. Zero tolerance on Apple's part.

Everyone's favorite beleaguered computer company is also at a crossroads where the iPod is concerned. Steve Jobs told [The Guardian Unlimited](#) that not only did he admire Sony, probably the well-trodden path, but that he thought [the iPod would have to stand alone](#):

[Neil McIntosh]: When the iPod was launched, you said it might lead people to the Mac platform. Does you still believe that?

[Steve Jobs]: No. We brought the iPod to Windows. That was a big decision. That was basically a decision not to use the iPod to drive people to Macs. We're going to use it as a music device, and we're going to put it on Windows. The majority of iPods we sell are used on Windows.

So Apple now has an entire division devoted to producing something that may not enhance the value of Mac sales. At all. Sounds a lot like Sony, doesn't it? Could that be the direction that Apple is headed? Jack at [As the Apple Turns](#) wonders (facetiously) if maybe Apple making music its top priority could be "[slightly sinister](#)."

What are they going to do with the iPod? Are they going to try to diversify into a multimedia company? Here's your chance, video iPod people! (I really hope not. But still.) Does AirPort Express suggest the new direction?

Speaking of the iPod, here are two more important news tips:

- At [Gizmodo](#), there was an interesting debate between the author of the OGG codec and an engineer who worked on a competing MP3 player as to whether the iPod can play OGG Vorbis files. The question seems to be, does the iPod have the horsepower? The engineer says [no](#); the author says [yes](#).
- As it turns out, not everyone is happy about the iPod either, so maybe Apple should stick to computers. At least one person in New York City, thanks to [Engadget](#) and [Gothamist](#) (not the kind of source you usually see in *Bloggable*), has voiced their displeasure both in person and on the Internet, defacing iPod posters in the Big Apple (heh). Think [Neistat Brothers](#), but no stencils: What does the "i" in "iPod" stand for anyway? "[Isolation](#)"? "[Insurrection](#)"?

Now, what was that I said about Microsoft and Web development earlier? Lately there's been a good deal of chatter about Microsoft and its ability to innovate, or whatever you want to call it, as far as the Web is concerned. In light of Internet Explorer for the Mac's cancellation (among others), it's certainly an interesting question, right?

Internet Explorer's development is stuck until 2006, and Microsoft may have [reached the plateau as far as innovation is concerned](#), according to [John Allsop](#). They have no incentive. But—could Apple innovate?—he asks. After all, Safari's already halfway to Windows, too, what with iTunes. Right? Sounds perfect? Except [it's not](#), according to [Dave Hyatt](#), who should know. Just because WebKit really isn't available on Windows doesn't mean we can't hope that Apple could step into the void where that's concerned.

Dovetailing nicely into this perspective is [Joel Spolsky's](#) lengthy discussion of Microsoft and the potential [death of its Windows APIs](#) in the face of Web software. Although developers may be using .NET, they could also be using Mono, or some other Web development platform, and clients would no longer be tied to using Windows. Again, could that be Apple's opening to innovate, in the browser space rather than in the client space? And does that make us the suckers, buying into a platform whose existence is irrelevant?

John Gruber, again chipping in, suggests that Web software is important particularly because of its platform agnosticism and its less limiting UI standards. He adds that Microsoft misread the competition when they killed off Netscape: [the Web itself was their enemy](#), not Netscape. And the Web is what's hurting Win32, like Joel says.

If you want to talk platform-agnostic, how about Frontier's kernel being [open-sourced](#)? Since it runs on OS9, OS X and Windows, and can be used to drive a database in an easy-to-learn manner, why not? Mac developers, unite!

So Apple has their fork in the road ahead of them, as we've seen. But there's plenty of other interesting news from June:

- [APE isn't all bad!](#) Unsanity's Rosyna and a series of commenters dispute what role haxies play in system instability, starting from the APE framework's technical background and working through potential problems in modules.
- [Apple is a lover](#), or [a gay shopping buddy](#), or [not just the same crap, but different](#). Love letters to Apple, and to the Mac OS. Potential Switchers might eat this stuff up. If they don't, they're probably not potential Switchers.
- And for your fun of the month, does the iTunes Music Store's censorship filter go [a little too far](#)? Not anymore, apparently.

Phew! That's pretty much it, folks. Keep your eyes open, as that fork in the road is coming up soon. Trust in Steve Jobs, and let me just say, I'm glad we're sitting in the passenger seat.

Copyright © 2004 Wes Meltzer, wmeltzer@atpm.com. Wes Meltzer no longer wants a shovel for the snow, but rather an air conditioner for the heat. And skeptical Mac users. He welcomes comments and e-mail as well.



About This Particular Outliner

by Ted Goranson, tgoranson@atpm.com

Three Topics on the Future of Outlining

Frontier Kernel To Be Open Source

In previous columns, we've mentioned [Frontier](#) as somehow on the fringe of the Mac outlining world. It looks as if Frontier, or some important pieces of it, are to become open source, and this could spawn a revolution in outlining on the Mac. When things congeal, we'll report on this. In the meantime, we'll provide some background and explain why Mac outliners might benefit.

Dave Winer did not invent outlining on a computer. I don't know who did, but it grew out of the MIT Lisp world. I encountered it there in the 70s. But Dave certainly was the first to productize it and bring it to the desktop for non-programming use. His ThinkTank and then MORE were killer applications in their own space, leading to widespread use of the outlining paradigm—indeed, even to *ATPO*.

MORE and the DOS-based sister products were sold to Symantec, which killed them. Winer's next company was called UserLand. I was involved from the first release. Userland had a support forum on CompuServe—this was before the Web—and during that early period some history of Frontier emerged.

The Origin of Desktop Scripting

The story goes that Winer was discussing with the young Bill Gates, who had some office applications on the Mac, what to do after creating standalone applications. They talked about a scripting environment that could be used to tie desktop applications together, similar in some respects to long existing Unix capabilities. But this new scripting idea would actually control applications as if through their user interfaces.

Gates took this idea in two directions. He created the application suite which has hardwired links, and he extended Basic into a scripting language. I was still loosely affiliated with MIT when Bill was at Harvard, long before this conversation. I remember him demonstrating an Altair Basic, which Paul Allen had written and ported to Apple and

DOS. They had some naive notions about programming science, notions which—as it turned out—successfully caught fire with the first crop of hobbyist programmers. But all that is another story.

The point is that Gates has always had an unnatural bias in favor of Basic as a solution for any problem as a result of his early success. In fact, it seems as if Bill's only exposure to actual development was with this Basic product. Basic then and now (as the core of Visual Studio .Net) is a toy language as languages go and was an odd choice for a scripting language, but they have made it work. Incidentally, Microsoft has steadfastly refused to port its Basic to the Mac, considering it a major competitive advantage for Windows.

Winer and his programmers took a more sophisticated tack: they used the C and Pascal languages as models, creating something code-named Juicy, which became UserTalk. Programmers, including the amazing Doug Baron (also mentioned below), wrote the Frontier environment for UserTalk, which we'll talk about in a minute. This was code-named Caboodle.

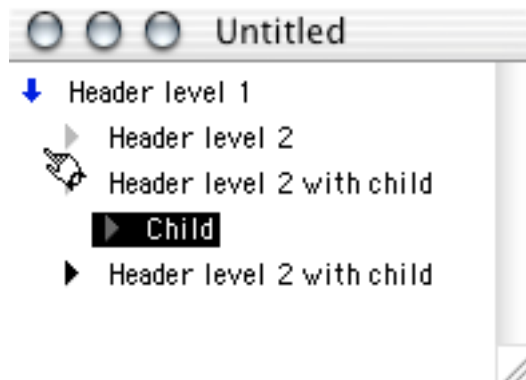
Baron wrote an extensible platform that could reach anywhere on the Mac. Winer—and later many of the Frontier users—enriched the platform through UserTalk routines and scripts.

Early in 1992, UserLand released the first scripting environment for the Mac, really the first complete desktop scripting environment anywhere. It included its own objectbase (now called the object database) which held data, much of the working code as scripts and menu bars (the extensions we mentioned). This code was easily extended and modified within Frontier, using itself. That itself included an outliner used for programming and debugging! The objectbase was also organized as an outline, and one of the document forms was an outliner, all modifiable and controllable. UserLand actually published a formal description of UserTalk in [Backus Naur Form](#), which was a very geeky thing to do.

(As an aside, three years earlier IBM had introduced [Rexx](#) as the scripting language of its OS/2 operating system. Rexx had been developed by IBM in the late 70s and by this time could be considered the first *real* desktop scripting language. Rexx, plus another older language, [Lisp](#), provided the central ideas behind the open source [Python](#), which today is the fastest growing cross-platform scripting language. Incidentally, Python's syntax is based on indentation, very outliner-like.)

With the introduction of Frontier, the sophisticated end of the Mac world went crazy, and the heart of the Mac beat strongly on that CompuServe forum for three or four years through experimentation, innovation, and selfless help. In practice, this was an open-source effort; the underlying philosophy presaged the Web—any action can affect or control any piece of code or data anywhere. For instance, I wrote the glue for [FrameMaker 5](#), effectively turning a text tagging application into (among other things) an XML-like system-wide database (before XML) not unlike Microsoft's Longhorn plans for [Avalon](#), but slow. (FrameMaker is now at 7.x, is used in the current production of ATPM, and has recently been discontinued on the Mac.)

Many coders still wistfully recall how useful the outline paradigm was for coding UserTalk (and of course before that for Lisp). I have in the last month heard from many folks suggesting that similar outline editing be created for Python. The screenshot below is reproduced from our *ATPO* column on [interface design](#).



Frontier's Outliner

With System 7, Apple introduced Apple events which Frontier was designed to employ. Well after Frontier was released, Apple came out with its own scripting language, [AppleScript](#) which has been supported sometimes languorously since. It is an odd mix of natural English and several syntactic conventions. Like Visual Basic, it is easy to learn enough to do simple things and very, very hard to master for complex tasks.

Apple also developed the [Open Scripting Architecture](#), which allowed for many scripting languages they clearly expected to appear. UserTalk quickly became an OSA component, and it was normal for AppleScripts to be mixed with UserTalk and for both to be created, run, and maintained within Frontier.

The future of scripting and outlining never looked brighter. I even factored it into a rather advanced AI research program I was managing.

Then, Winer developed extreme dislike of Apple management, supposedly because Apple refused to license Frontier as an integrated part of the OS. He announced that the product would have to move over to the “more friendly” Windows. But in 1995, he announced Frontier dead (unsupported even for paid users) and gave it away as a free product.

He later got interested in the Web and revived Frontier as the basis for some Web tools for content management, aggregation, and blogging. (ATPM was even published using Frontier for some time during this period.) Most of this was done on Windows and as of today, Windows is the preferred platform.

Those Web applications are banal, not revolutionary like the original Frontier and not of interest to outliners except as described in the topic on Web Outlining that follows. They do however have an enthusiastic, though diminishing user base. Both desktop scripting and the Mac were all but abandoned by Frontier except for the minimum necessary to support the Web products, [Radio](#) and [Manilla](#).

But over time, Frontier found itself competing with all sorts of commercial and open source projects; it never hit a home run and is now a niche product. Making the original Frontier kernel open source (but not Manila or Radio) is an attempt to save the company by reviving all that free development and excitement of the early days.

It might not save UserLand, but this could be a very big deal for the Mac outlining community. Imagine an extensible editor that can be used as a desktop outliner and outlining programming environment. Imagine something that is easily programmable and extensible within itself by the moderately skilled. So far, you have an unholy cross between [Emacs](#) and [jEdit](#), both of which have desktop outliner projects. Imagine also that you can create, debug, store, and reuse OSA scripts natively. Suppose that the contents of those scripts—in addition to being modifiable—could interact with, control, or modify essentially any application or file your Mac can reach.

Suppose the language were as well designed as Python, using indentation creatively. (The already-mentioned Python is one of the shining stars of the scripting world.)

Suppose it were all free.

My own guess is that it is too late for UserTalk to catch up as a major scripting language. And the editor/debugger that was advanced 14 years ago now seems long in the tooth except for the handiness of outlining at many levels; [Eclipse](#) (which doesn't support outlining) has the buzz these days. So what I expect is for an open source developer to use Frontier to create an advanced desktop or Web outliner, possibly collaborative like [SubEthaEdit](#) and cross-platform with a built-in macro and extension language that is also an OSA scripting language

It could be something, something worth *ATPO* tracking as a big deal in outlining and reporting to you. We'll let you know what happens as it happens. There are lots of issues involved here, things that will be in your mind as you decide on your particular outliner—things like how important to you are the Mac interface and operating system goodies; how important is the ability to tweak every molecule (or have dozens of clever geeks tweaking them in different ways for you to select); and what kind of scripting capabilities you find handy.

Web Outlining

Recently, I spent a few hours with Marc Barrot in New York discussing Web outlining. You can see his invitation in his [comment](#) on the March edition of *ATPM*. Marc is at the center of two extremely interesting developments in outlining.

I admit that I do not understand the future of Web outlining very well; I do not think anyone does. My own expertise is with outlining as a single-user tool on a desktop. But many of the tools we examine in *ATPO* deal with the Web as a source for information, either by clipping Web/RSS content or by referencing URLs.

The last *ATPO* [surveyed](#) [NoteTaker](#) and [NoteBook](#), which both can publish versions of themselves to the Web in a form essentially identical to their desktop appearance, in a read-only form as literal notebooks.

Sometime soon, we'll similarly survey the amazing [Tinderbox](#), which by many measures is the most powerful outliner on any desktop. It has rather robust internal scripting to produce blog-like Web sites of notes. It actually stores its files as XML, so you build a site with its own agents and export tools or roll your own using the daunting XSLT language. But Tinderbox Web sites do not have outline controls; all the outlining is erased in the transfer to the Web.

[FreeMind](#) is an open source mindmapping type outliner that publishes an indented folding outlining to the Web. Pretty impressive (and free!), but you have to create in a non-Mac-friendly mindmapping mode.

ActiveRenderer

What if you had a tool that could take your outline from your favorite desktop outliner and display it on the Web as an outline with collapsible triangles and all? That last part is what [activeRenderer](#) does. By an odd coincidence (well, not really), activeRenderer is written in UserTalk and was originally intended for integration with the Frontier-hosted blogging product, [Radio](#). Instead, it is being used to render OPML files to the Web from other outliners (and also from Movable Type). OPML is an XML format, incidentally created by Dave Winer, supported by [FO](#), [iLiner](#), [Hog Bay NoteBook](#), Circus Ponies NoteBook, NoteTaker, [OmniOutliner](#), and [MyMind](#).

Check out Marc's [gallery of rendered Web sites](#).

There are a couple of very nice touches to how this works. They are so elegant, one wonders why we don't see them in some desktop outliners. If you hover over a collapsed arrow, a pop-up appears of the collapsed content (or a comment instead). Marc hastens to credit Michael Poremba with this notion in his earlier [rendering project](#).

▼ **Tuesday, April 27, 2004**

▶ **rssBox and activeRoll wizards**

The next version of activeRenderer will include wizards to help bloggers to their templates using activeRenderer's [activeRoll](#) and [rssBox](#) modules.

Gwénaél, who first came up with this brilliant idea, is hard at work getting the wizards ready. If your French isn't too rusty, you may try out his [demo](#) of the [rssBox](#) and [activeRoll](#) wizards.

We'll be working together next week in Paris, our target is to release activeRenderer by mid June. More on the coming 2.3 version later.

5:27:09 PM.

▼ **Wednesday, March 31, 2004**

▶ **Gallery: Gwénaél Le Dréan**

▼ **Friday, March 26, 2004**

▶ **activeRenderer version 2.2 released at last**

ActiveRenderer Pop-up

If you hover over an uncollapsed arrow, a vertical line appears under to show the scope of the children.

▼ **Tuesday, April 27, 2004**



rssBox and activeRoll wizards

The next version of activeRenderer will include wizards to help bloggers side rolls to their templates using activeRenderer's **activeRoll** and **macros**.

Gwénaél, who first came up with this brilliant idea, is hard at work those wizards ready. If your French isn't too rusty, you may try out prototypes of the **rssBox** and **activeRoll** wizards.

We'll be working together next week in Paris, our target is to release a new version of activeRenderer by mid June. More on the coming 2.3 version.

5:27:09 PM.

▼ **Thursday, April 22, 2004**

▶ **Elections Demo Page Updated**

▼ **Tuesday, April 20, 2004**

ActiveRenderer Expanded

ActiveRenderer has a really great feature: transclusion. Transclusion is an idea and term invented by hypertext legend Ted Nelson, who we introduced in the [very first ATPO](#). Transclusion is the hypertext/Web version of cloning. In outliner cloning you effectively say: make a live updating virtual view of this header and children, where edits to either are reflected in both. In the Finder, these are called aliases.

Transclusion is subtly different; it introduces an intermediate marker, a URL. With cloning you say to show this content. With transclusion, you say: show whatever content someone has associated with this marker. You may not know what it is at all. For instance, you can transclude the latest entry from AT^{PM} publisher Michael Tsai's [blog](#), or the latest weather, or the info from eBay that you are tracking.

Here are a couple of screenshots from Marc's online tutorial for [transclusion](#).

- ▼ **Step 3:** And it's even worse than it appears :-)
 - ⤴ Now click on the 'up arrow' icon that serves as this
 - ▶ **How weird! This outline expands into itself, it literally**
 - Of course, it's far more interesting to include *other* outlines, as
 - they can be expanded inside the current page, providing a way to
 - the linking node.
 - ⤴ There's more: click the 'up arrow' wedge at the left of the
 - expand linked rss feed** display RSS news feeds as outlined in regular outlines.
 - ⤴ Outline inclusion also allows the creation of distributed outlines, which can be
 - see by clicking on this paragraph's wedge.
 - 📄 If the linked content is simple HTML, as opposed to other content types like
 - OPML/RSS/GIF/JPG/PNG/SWF/MOV, the link is rendered as a simple text link
 - in the wedge, and it opens in a new window when clicked.
- ▶ **Step 4:** Creating included content with Radio's outlining
- ▶ **Step 5:** OPML links nuts and bolts.

Ready to Transclude

The outline triangle in this case is an upward pointing arrow over which the user has moved her browser's cursor in preparation for clicking. The next shot shows that arrow open, meaning the content has been retrieved; in this case it happens to be the news items from the activeRenderer site.

▼ **Step 3:** And it's even worse than it appears :-)

⬆️ Now click on the 'up arrow' icon that serves as this

▶ **How weird! This outline expands into itself, it literally**

Of course, it's far more interesting to include *other* items, but they can be expanded inside the current page, preventing the linking node.

👉 There's more: click the 'up arrow' wedge at the left

- ▶ activeRenderer version 2.3 released
- ▶ rssBox and activeRoll wizards
- ▶ Elections Demo Page Updated
- ▶ activeRenderer For Movable Type Progress
- ▶ Gallery: Gwénaél Le Dréan
- ▶ activeRenderer version 2.2 released at last
- ▶ Release delayed
- ▶ Gallery: Gerald Gleason
- ▶ Mailing Lists
- ▶ Documentation on formatting

Transcluded Items

If Marc is sufficiently encouraged, he may open this up to be more easily usable by OPML-producing Mac outliners. [Let him know](#) and we'll see.

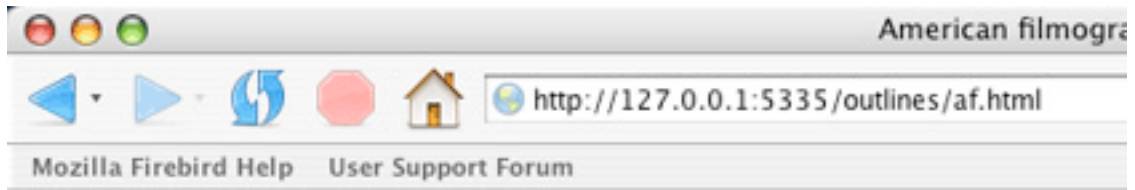
webOutliner

ActiveRenderer is cool, but the thing that brought me to New York to discuss was [webOutliner](#). This is a different beast altogether. The idea behind it is that you create outlines in the Web browser. Just as with desktop outliners, the outline creation and publishing is via the same interface, only in this case, it is a browser and your files are on a Web server somewhere.

That means that, in theory, many people can contribute to the same outline. In a way, this is much like a [Wiki](#), but without all the massively unintuitive editing and user interface compromises of wikis. (As a side comment, Hog Bay Notebook has an interesting merger of a desktop wiki and outlining.)

And, unlike Wikis, this could be multimedia friendly. Multimedia, especially linked multimedia, is a heavyweight job. Creating the killer application in this space is going to require tons and tons of horsepower, way beyond even the next generation of desktops. The way to go is to build it on a Web server and worry about efficient delivery, intimate interaction, and tailorability at the user end. A new outlining paradigm?

Here's what it looks like today. You view an outline just like in activeRenderer:



American filmograp

▼ 1946

▶ Frank Capra

▼ John Ford

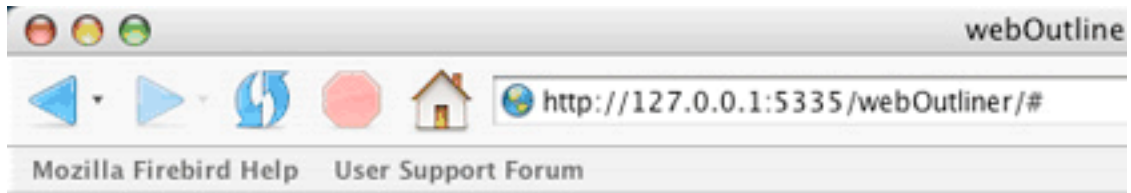
▼ My darling Clementine

Among Ford's finest Westerns and one of the greatest sense of poetic tragedy, My Darling Clementine is perv feeling of loss—of family, place, honor, and self-worth hero, Wyatt Earp (Henry Fonda), can never find the peace church-going pioneer families pay him to establish. Equ reluctant romantic, deeply chivalrous like other Fordian Among the more elusive protagonists of the Western g seems laid back, reserved, or repressed.

His friend and ally, Doc Holliday (Victor Mature), is a d gambler who carries a tragic weight. Earp's sorrow is q introspective. His gentle attempts to achieve balance o Sunday morning yield one of cinema's most sublime m

webOutliner Outline

But the entries aren't just read-only. Clicking an entry once selects it. The toolbars on the left and top appear and support most of the things you would do with a desktop outliner.



webOutliner

New Open Publish Mail Post

address: /af.opml - status: modified



American filmography

- ▼ 1946
 - ▶ Frank Capra
 - ▼ John Ford
 - ▼ My darling Clementine

Among Ford's finest Westerns and one of the greatest sense of poetic tragedy, My Darling Clementine conveys a feeling of loss—of family, place, honor, and self. The hero, Wyatt Earp (Henry Fonda), can never find church-going pioneer families pay him to establish a reluctant romantic, deeply chivalrous like other. Among the more elusive protagonists of the Western, he is laid back, reserved, or repressed.

His friend and ally, Doc Holliday (Victor Mature) is a gambler who carries a tragic weight. Earp's sorrowful and introspective. His gentle attempts to achieve balance will yield one of cinema's most sublime moments.

webOutliner Selection

And double-clicking turns the entry into editable content. Changes are available to everyone.

webOutline

http://127.0.0.1:5335/webOutliner/#

Mozilla Firebird Help User Support Forum

webOutliner

New Open Publish Mail Post

address: /af.opml - status: modified

American filmography

- ▼ 1946
 - ▶ Frank Capra
 - ▼ John Ford
 - ▼ My darling Clementine

Among Ford's finest Westerns and one of the greatest sense of poetic tragedy, My Darling Clementine of loss—of family, place, honor, and self-worth Wyatt Earp (Henry Fonda), can never find the pioneer families pay him to establish. Equally, deeply chivalrous like other Fordian leading men elusive protagonists of the Western genre, Earp reserved, or repressed.

His friend and ally, Doc Holliday (Victor Mature) gambler who carries a tragic weight. Earp's sorrowful introspective. His gentle attempts to achieve better morning yield one of cinema's most sublime moments.

webOutliner Editing

Now is that cool?

The history of how webOutliner came about is pretty interesting in itself. Based on the success of activeRenderer, Marc Canter talked to Marc Barrot about a collaborative Web outlining tool that could form a basis of a next-generation collective multimedia publishing tool. [Marc Canter](#) is the guy who invented MacroMind Director, the thing that spawned the Macromedia empire and eventually [Flash](#).

He's a rare bird, one of the older visionaries who is actually worth listening to. He encouraged Barrot to do webOutliner. Marc Barrot strikes me as a talented engineer, but he had some heavyweight help, recruited by Canter.

Marc says of this:

Doug Baron developed part of the outliner's JavaScript front-end code, quite a feat when one considers he had virtually no JavaScript development experience before. Then again, Doug is probably the only person I know who would be able to develop an outliner in his sleep. Doug also provided some of the Frontier back-end code, drawing on his deep knowledge of an environment he had contributed to create in the first place.

Danny Goodman is the only real DHTML Artist I've ever met. He contributed the amazing outliner drag and drop JavaScript code, probably the most sophisticated piece of JavaScript I've seen and been able to understand so far.

[*Ted's note:* [Danny Goodman](#) is a well known stalwart in the Mac scripting world because of his terrific books.]

I'm currently working with Phil Pearson, a Python guru from New Zealand, to port the webOutliner's back end code to an Apache/Python architecture. This will enable us to run the webOutliner server on Unix-based systems, which provide a more stable and scalable environment than the current Windows/Frontier.

Why is this of interest to *ATPO* readers? Because it may be the harbinger of the long anticipated collaborative multimedia commons. Because it could change the face of desktop outlining applications. Because it revisits something that was at the very soul of what it meant to use a Mac. And because it is cool.

Marc Canter speaks to the future:

With the availability of an open source webOutliner—new kinds of online tools and services are possible which take advantage of the inherent, instinctual nature of outlining. By supporting OPML, Broadband Mechanics is providing the ideal, open tool and codebase for others to leverage and enjoy.

Marc Barrot speaks to the future:

In the past years, I've been most impressed with desktop Web publishing, as demonstrated by Userland's Radio and by online collaborative Web site creations, as demonstrated by the different flavors of Wikis.

So I've decided to add my interest and knowledge of outlines in the mix, and explore two directions:

Desktop outline publishing: given current desktop outliners, including the worst and unfortunately most widely used one: Microsoft Word for Windows, how can we add both a "save as" new format to create a dynamic HTML version of the edited outline, and the background mirroring task in order to copy the HTML version on a publicly accessible site. I am already working with other developers to create those features in a number of outliners, and would definitely welcome new volunteers.

Web outlining: I'm frustrated by Wikis' very limited ways of entering and modifying texts. We (that is Phil Pearson and I) are currently working on building a more reliable and scalable webOutliner backend based on Apache and Python scripts. Once this is done, I'll combine WikiWords and transclusion, add a couple of other wiki-like navigation and indexing features, and we'll come up with a new and I hope more compelling outline based group publishing system.

Eventually, adding to an outlined site contents shouldn't require more than either firing up your favorite outliner and saving for publication, or loading the published page into a Web browser and entering modifications directly. We intend to attain both: power through simplicity.

The State of the Art Snapshot

We (our vast staff) have decided to introduce an *ATPO* news feature that reports progress in the outliner community. We'll continue with our focused features as before—next month on mindmapping—and afterwards let you know about any updates or other noteworthy news in the previous month.

I know the average Mac outliner enthusiast will find it hard to keep track on his own—I can hardly do it via direct contact with many developers. Things are heating up outlinerwise all over—and especially on the Mac—and this experiment could just become overwhelmed. So I ask readers and developers to let me know directly when something interesting happens.

To kick start the reporting feature, we'll list here the applications we will be tracking and briefly note their current status. In the listings, we mention if an application is developed in Cocoa, Carbon, or whatever. This may matter to you since many of the Cocoa applications can use the increasing collection of text editing tools supplied by Apple and third parties; Mike Ferris's [TextExtras](#) has some pseudo-outlining capabilities. Carbon outliners should run under OS 9, which might be important to you.

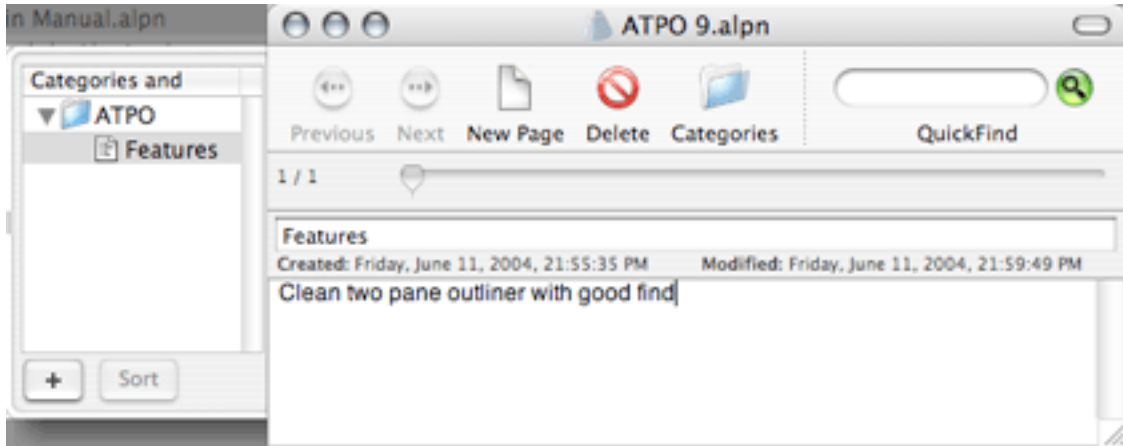
Each listing has a screenshot to give you a flavor of the user interface. The screenshots have a rough listing of features; please don't consider the feature listings authoritative. Mindmapping software gets short shift in this because I only show the outline view.

[Alep](#) (Macchampion)

Environment: Cocoa

Price: \$10

Current version: 2.5.3

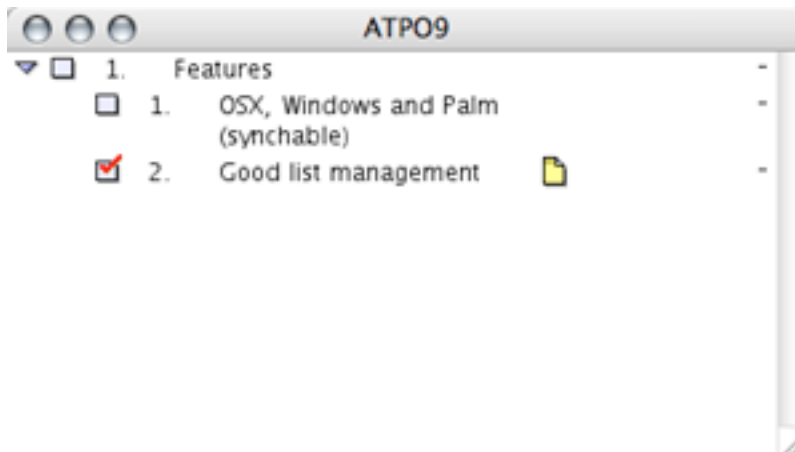


[BrainForest Deluxe](#) (Ultrasoft)

Environment: Carbon

Price: \$25

Current version: 3.0.8

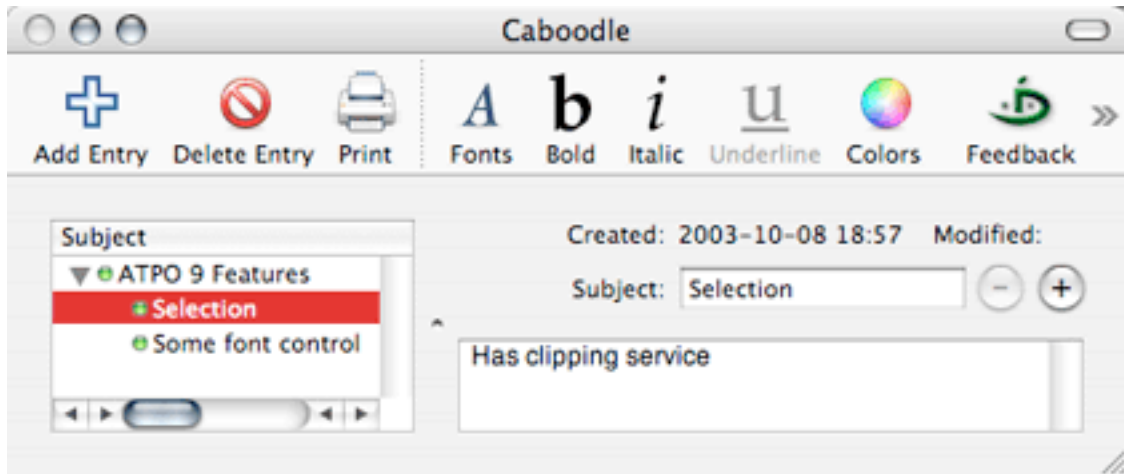


Caboodle (Dejal Systems)

Environment: Carbon

Price: free

Current version: 10.b1

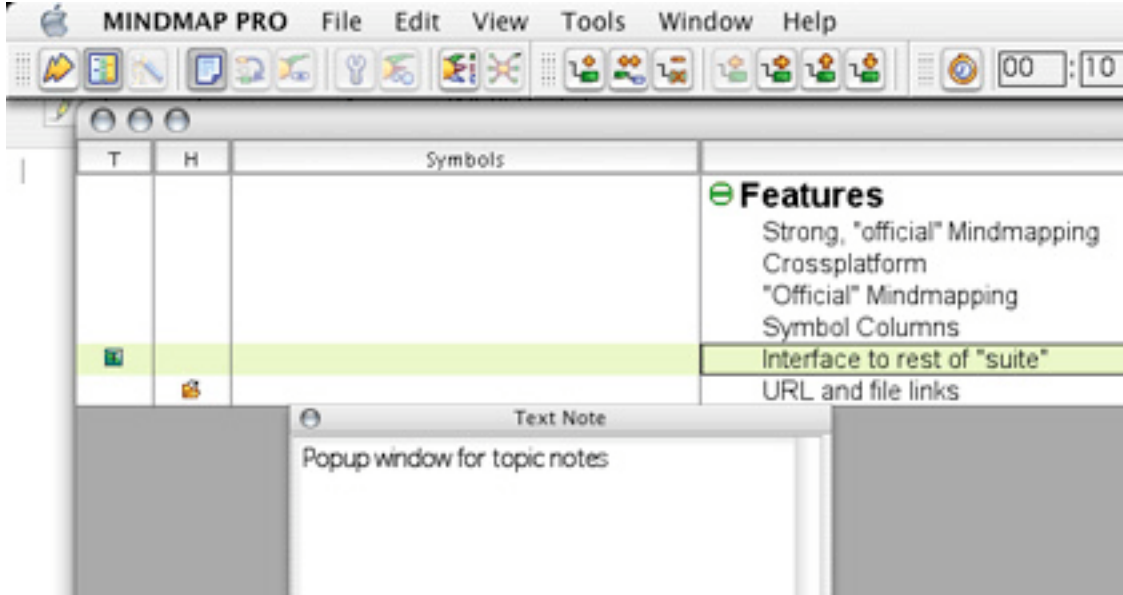


ConceptDraw Mindmap (Computer Systems Odessa)

Environment: Carbon

Price: \$99; Pro version \$149

Current version: 3.0

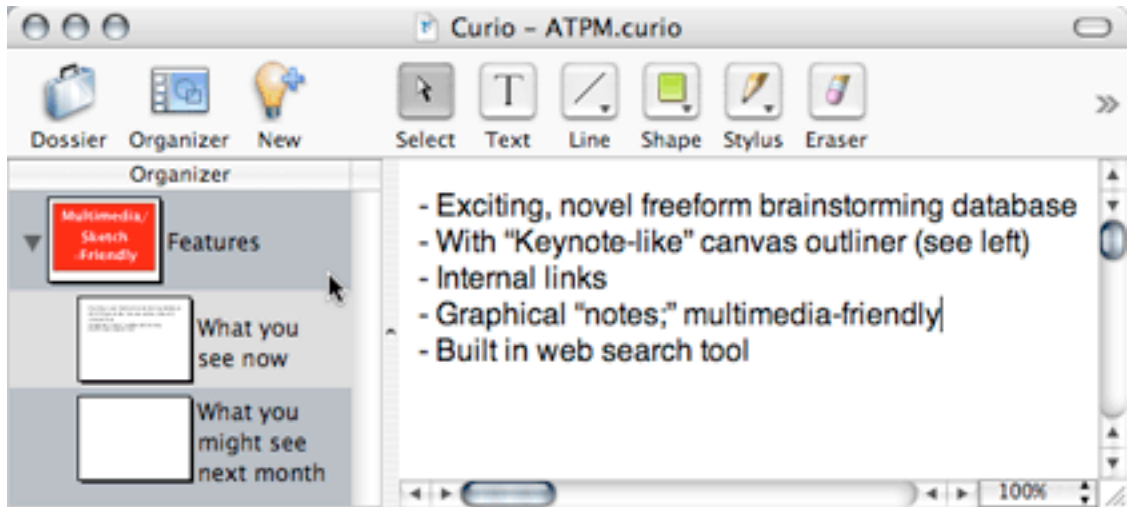


Curio (Zengobi)

Environment: Cocoa

Price: \$99

Current version: 1.2

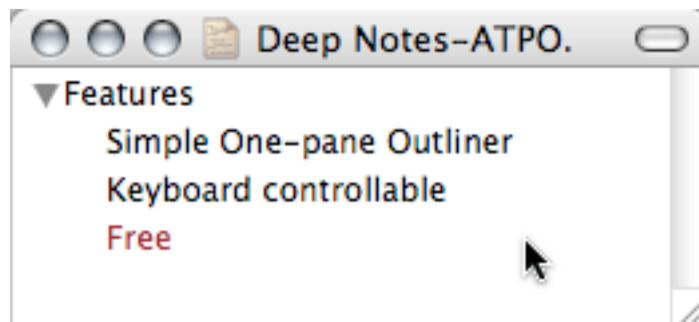


Deep Notes (Amar Sagoo)

Environment: Cocoa

Price: free

Current version: 1.2

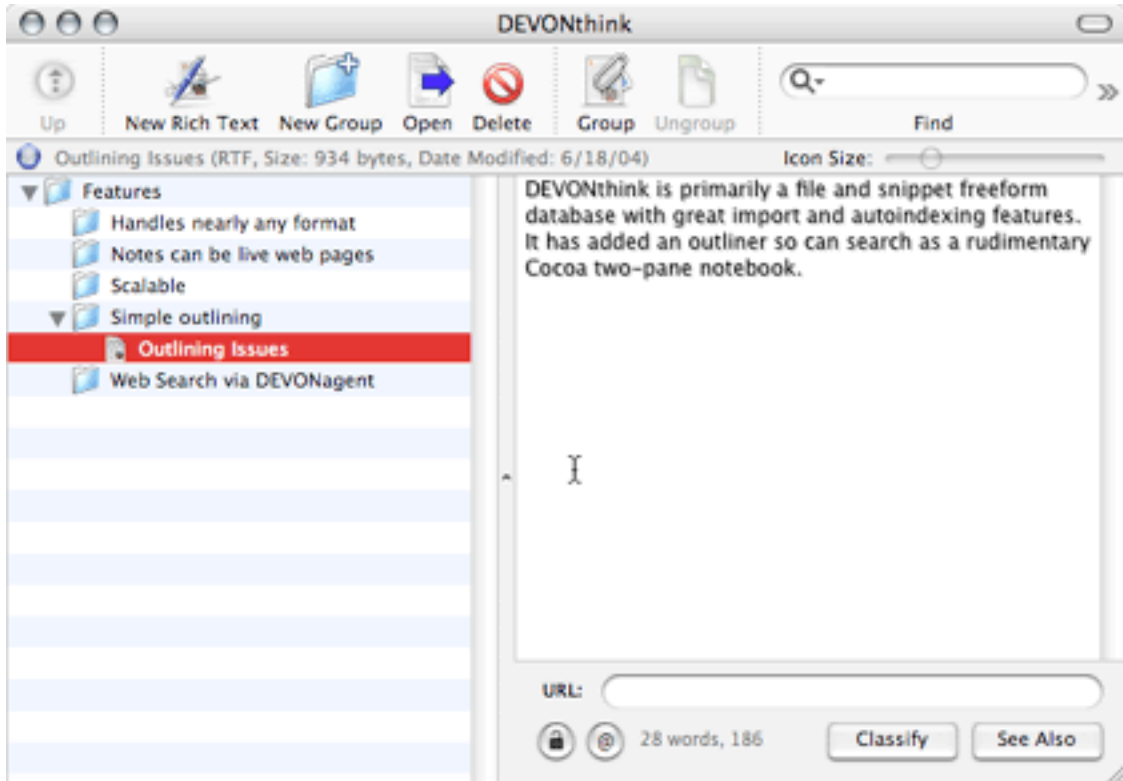


DEVONthink PE (Devon Technologies)

Environment: Cocoa

Price: \$40 (note version \$20, with DEVONagent, \$65)

Current version: 1.8.1

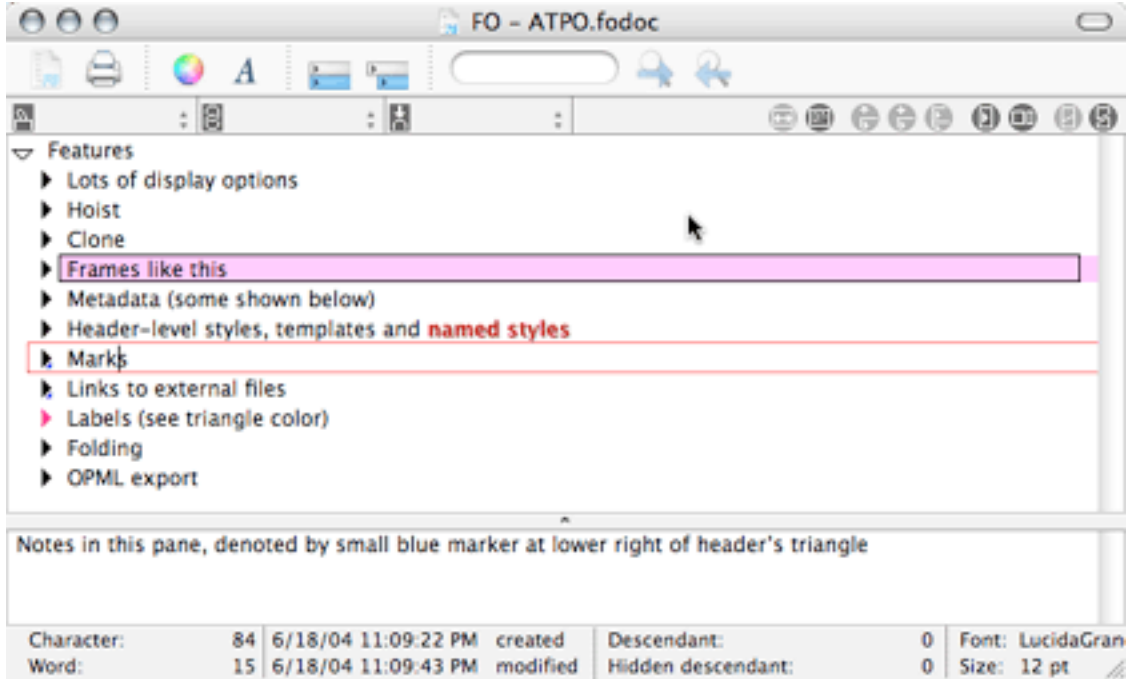


[FO \(Takashi T. Hamada\)](#)

Environment: Cocoa

Price: in beta, price not yet set

Current version: .09u

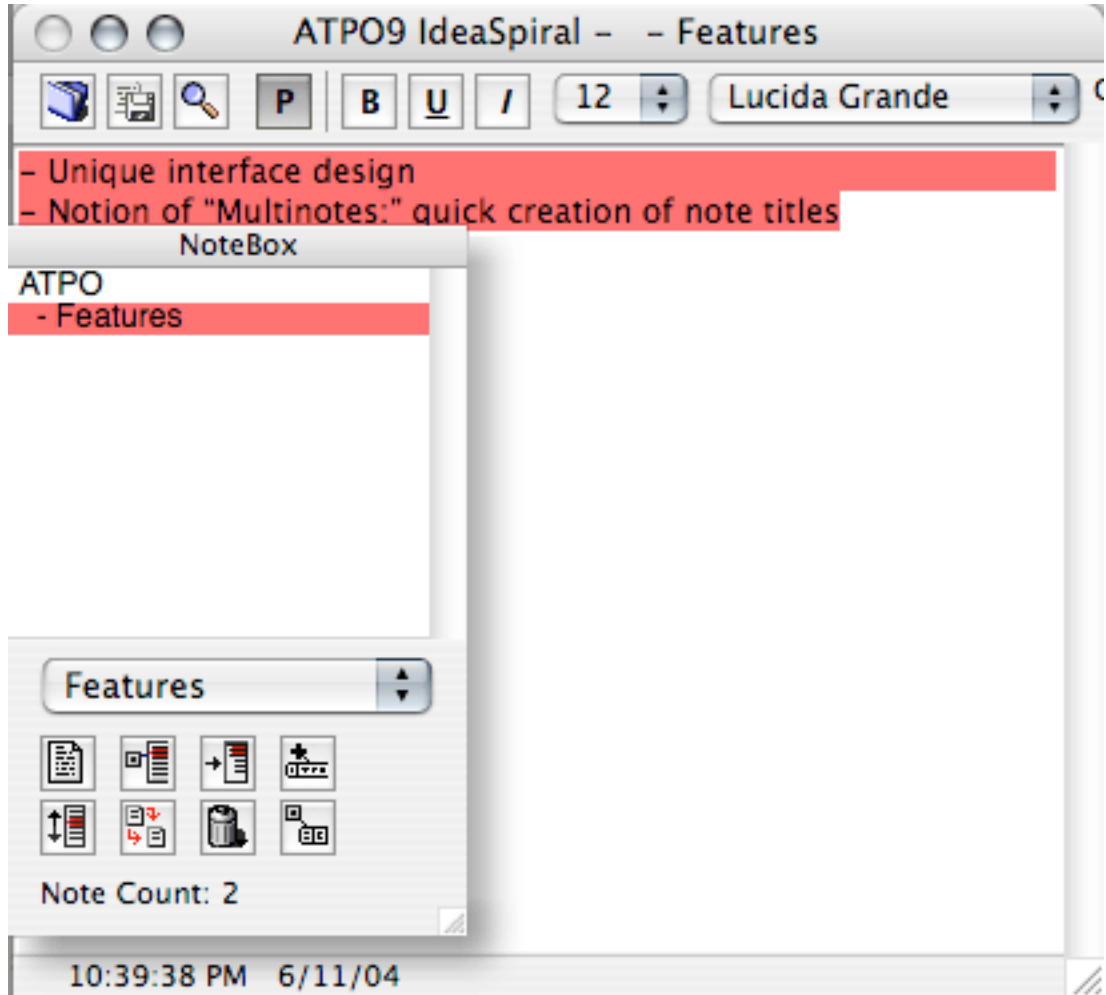


[IdeaSpiral \(midnite.lightman\)](http://midnite.lightman.com)

Environment: Carbon

Price: \$19

Current version: 1.5.4

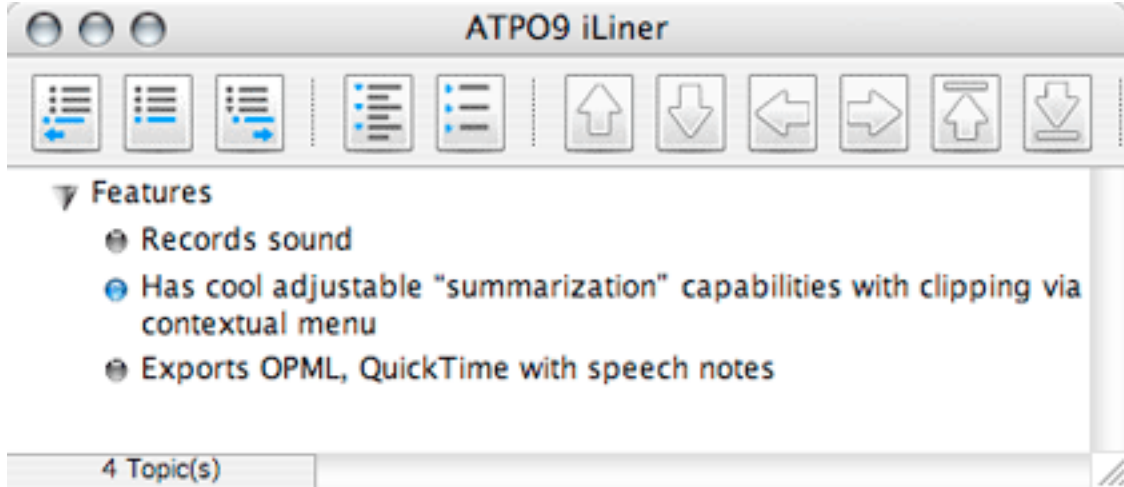


iLiner (Mercury Software)

Environment: Carbon

Price: \$30

Current version: 2.0

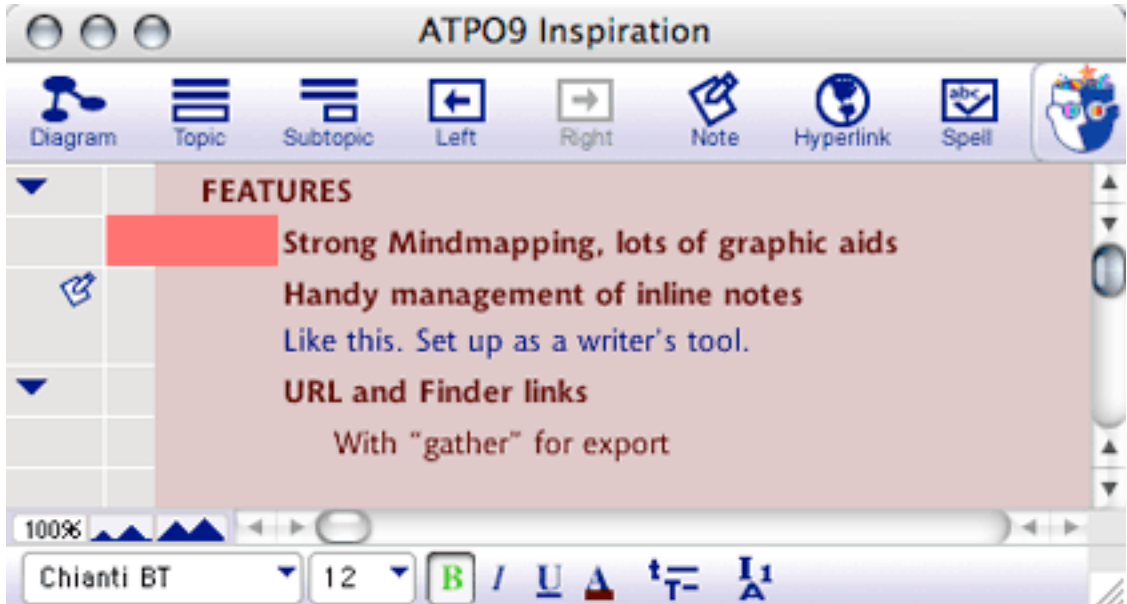


Inspiration (Inspiration Software)

Environment: Carbon

Price: \$70

Current version: 7.5

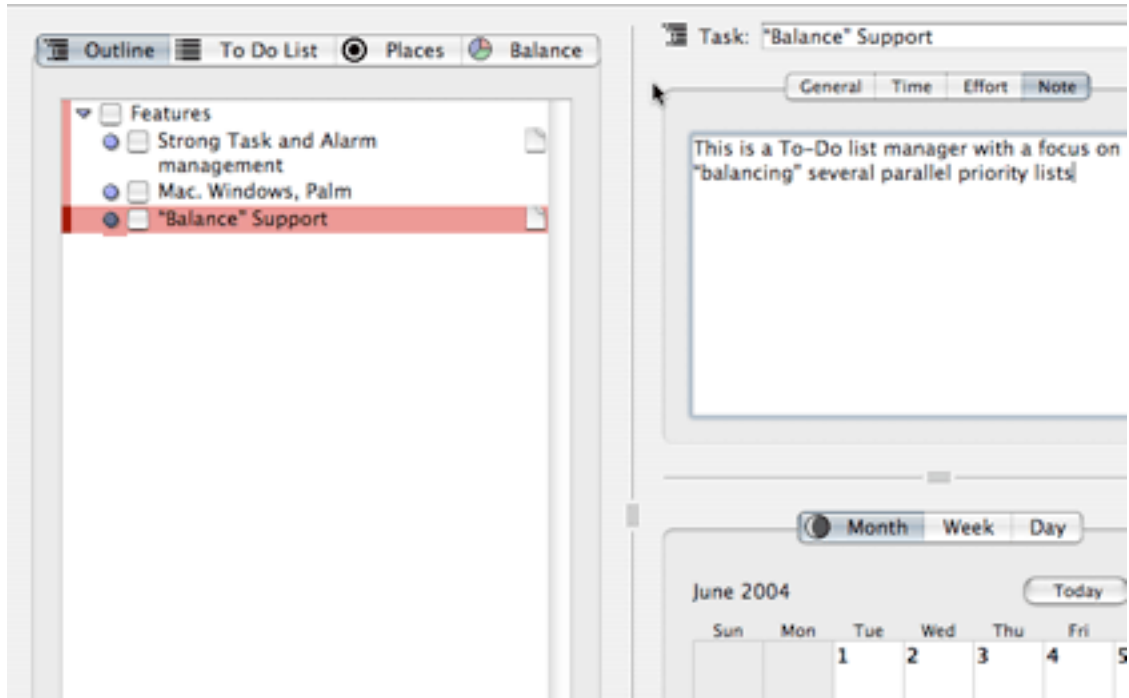


Life Balance (Llamagraphics)

Environment: Carbon

Price: \$65

Current version: 3.2.4

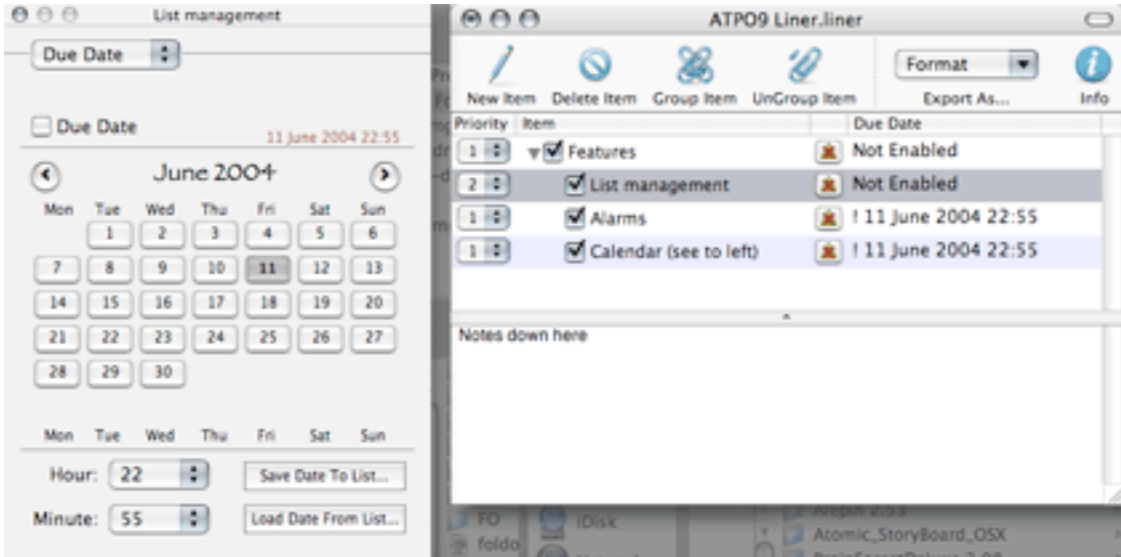


Liner (iMedia)

Environment: Cocoa

Price: \$20

Current version: 2.0

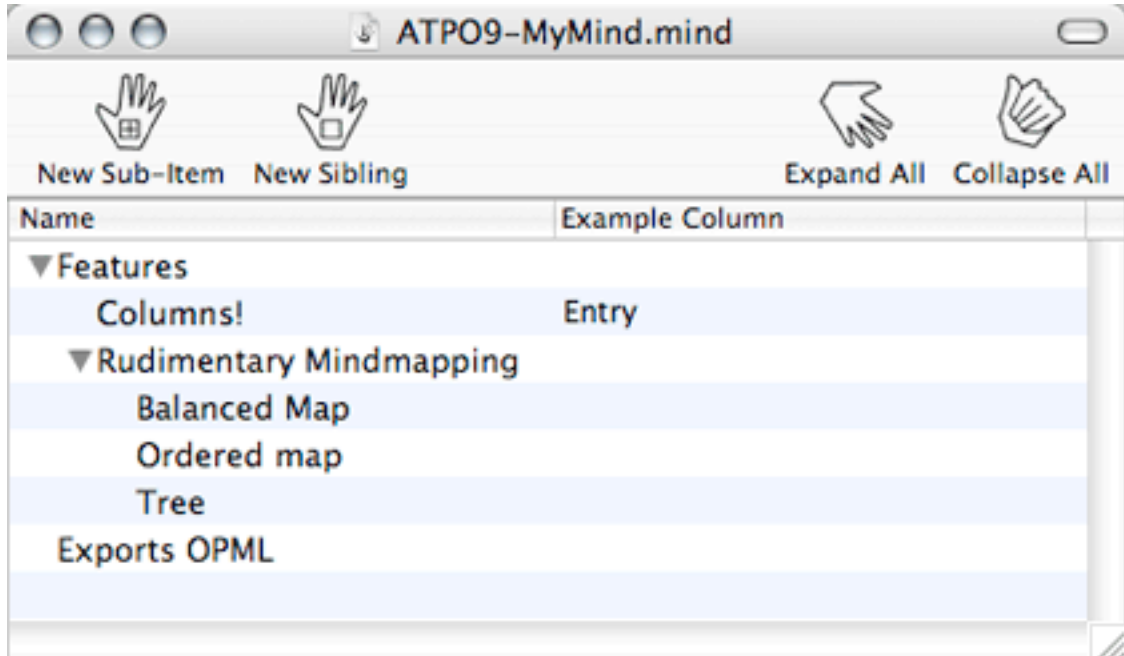


MyMIND (Sebastian Krauss)

Environment: Cocoa

Price: donationware

Current version: 1.0

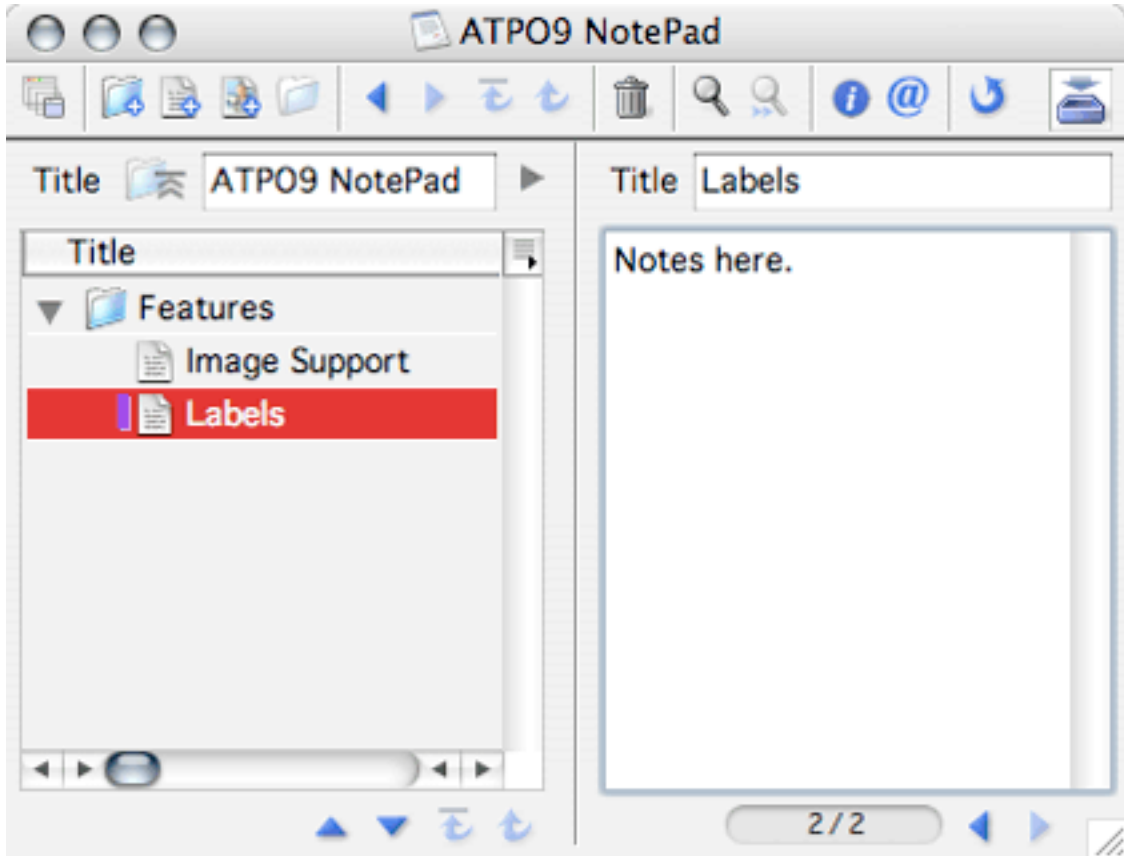


NewNotePadPro (Hisadonia Island)

Environment: Carbon

Price: \$23

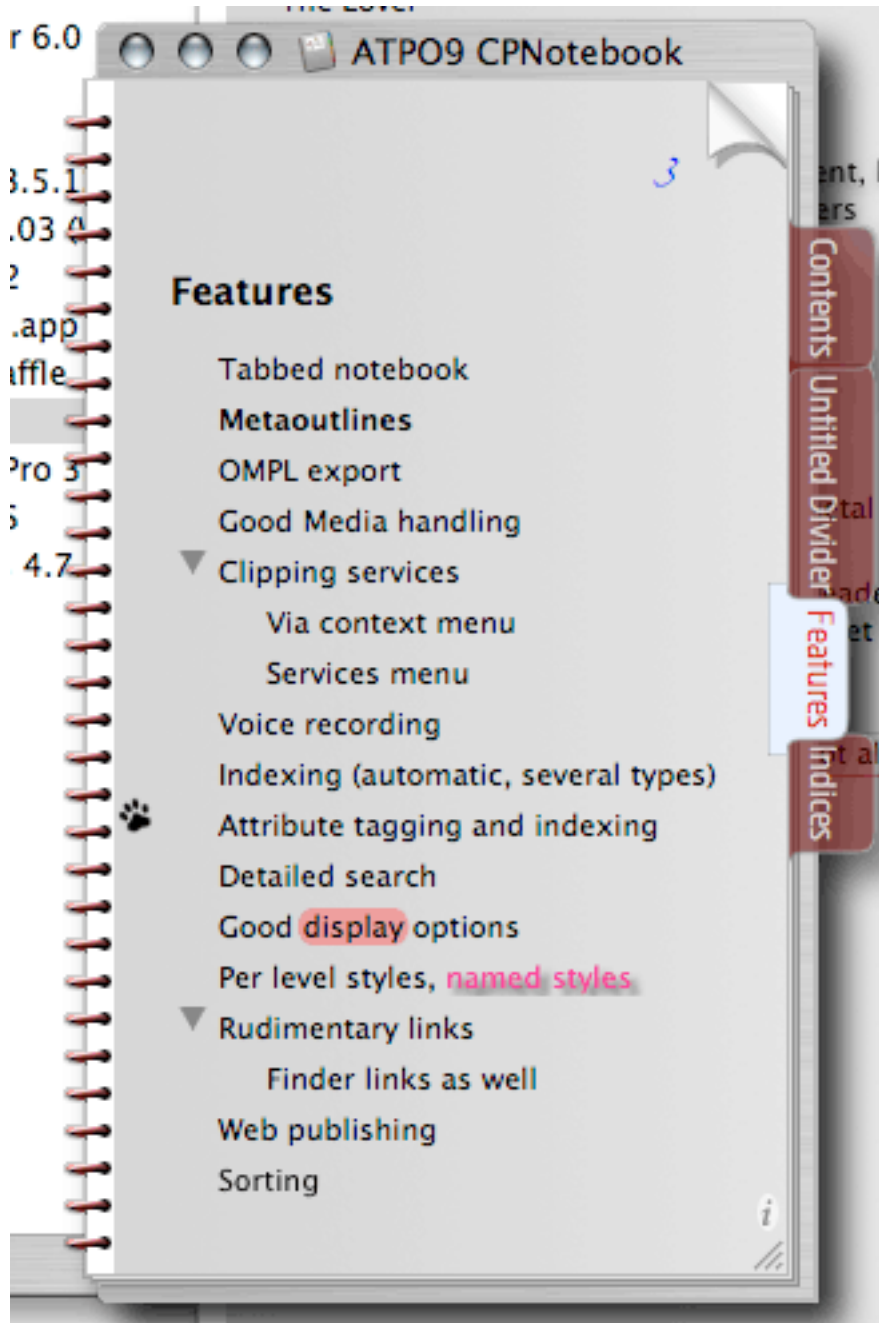
Current version: 2.1



NoteBook (Circus Ponies)

Environment: Cocoa

Price: \$50

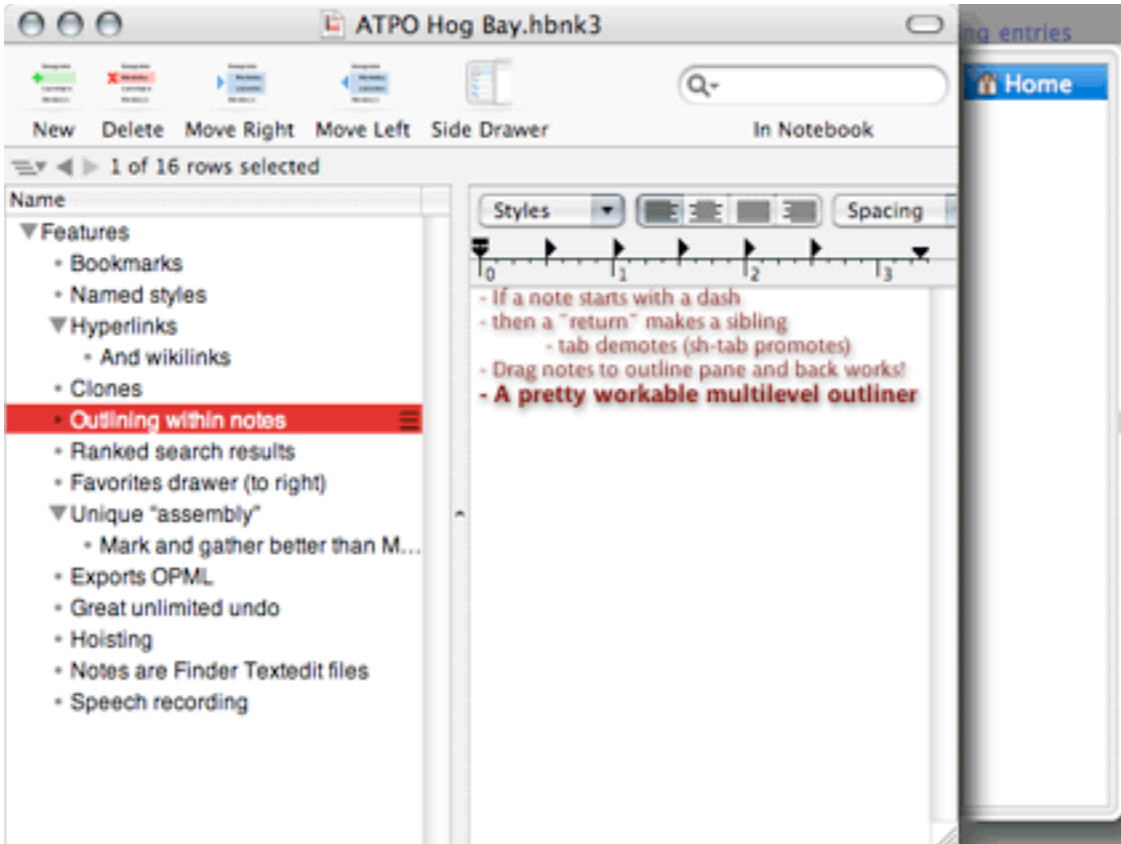


Notebook (Hog Bay)

Environment: Cocoa

Price: \$20

Current version: 3.1.1 (v112)

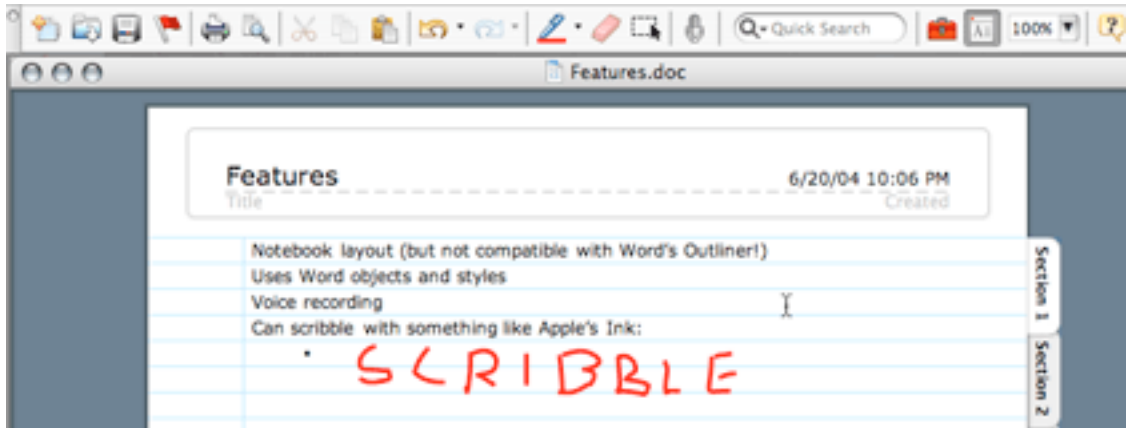


Notebook (Microsoft)

Environment: Carbon

Price: \$239 (as part of Word); \$399 (as part of Office)

Current version: 2004

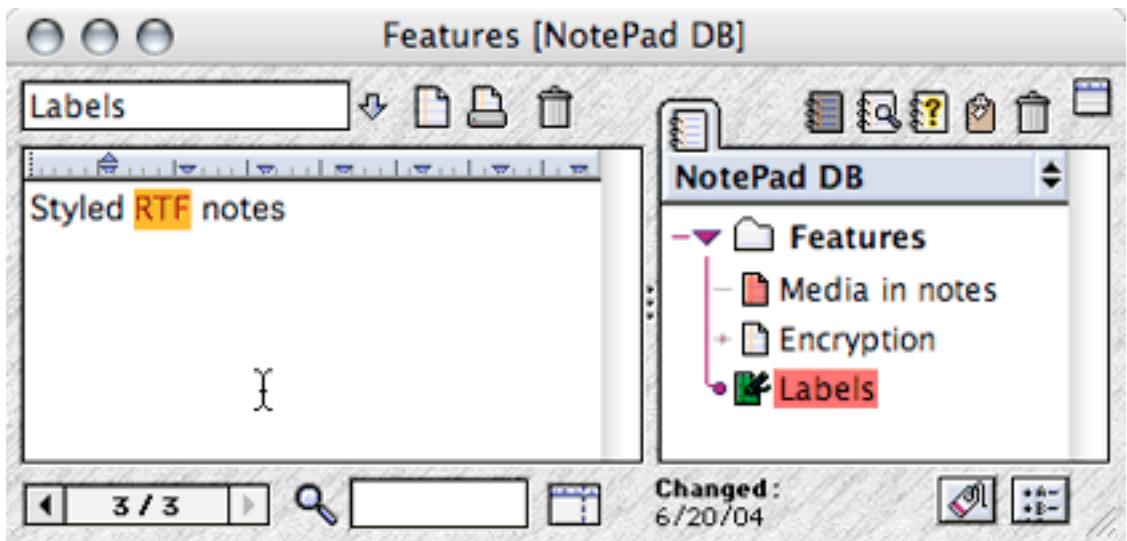


NotePad Deluxe (Ibrium)

Environment: Carbon

Price: \$25

Current version: 2.3.4

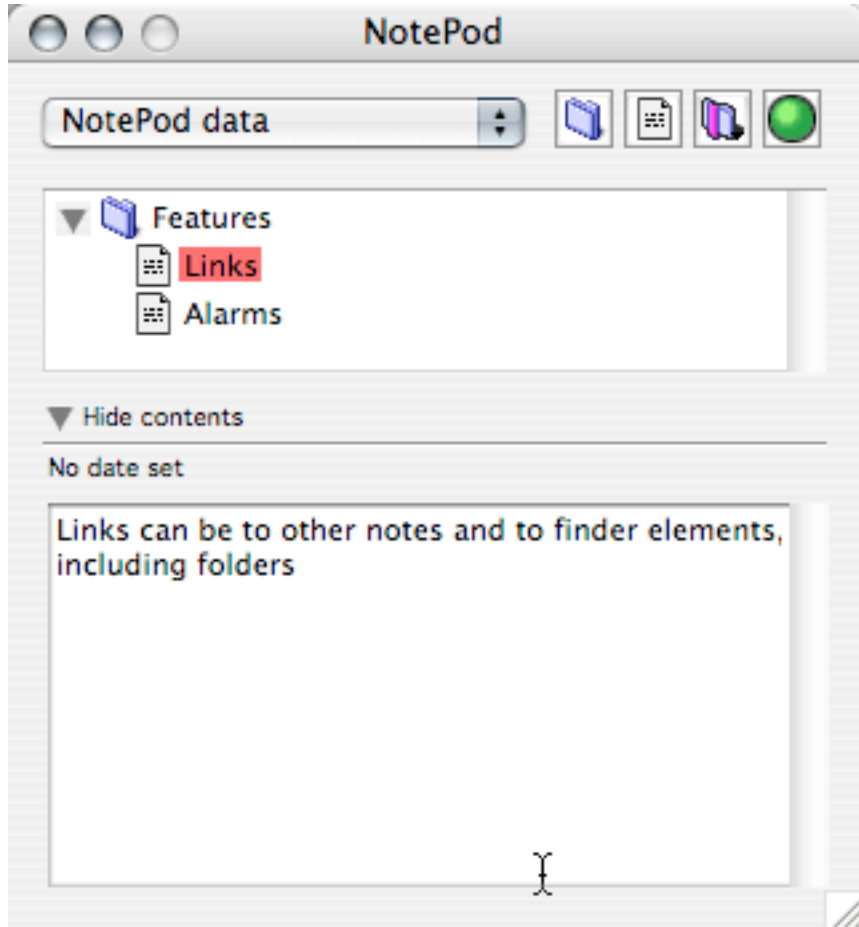


NotePod (Graffix)

Environment: Carbon

Price: \$15

Current version: 1.1

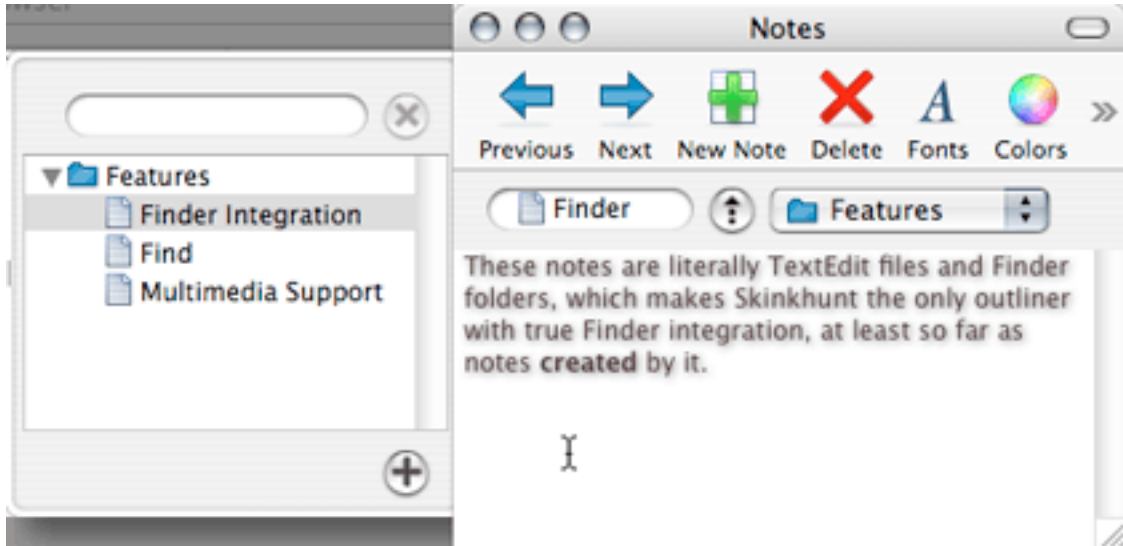


Notes (SkinkHunt Software)

Environment: Cocoa

Price: \$20 (site currently down)

Current version: 3.0.4

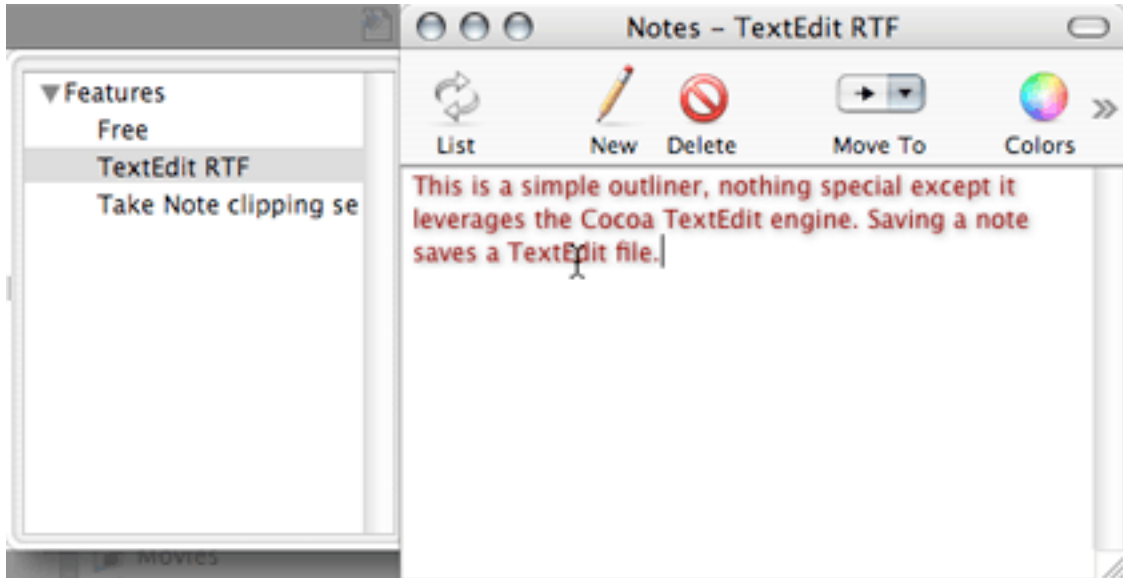


Notes (Richard Schreyer)

Environment: Cocoa

Price: free

Current version: 1.3.4

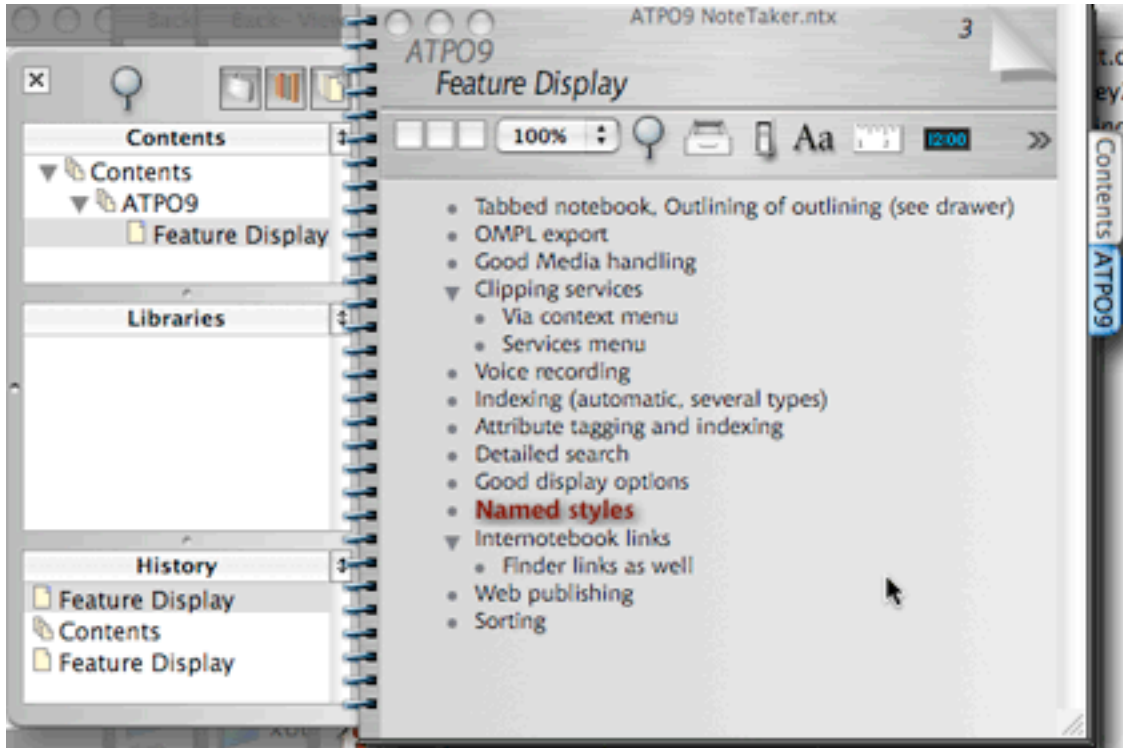


NoteTaker (AquaMinds)

Environment: Cocoa

Price: \$70

Current version: 1.8.9

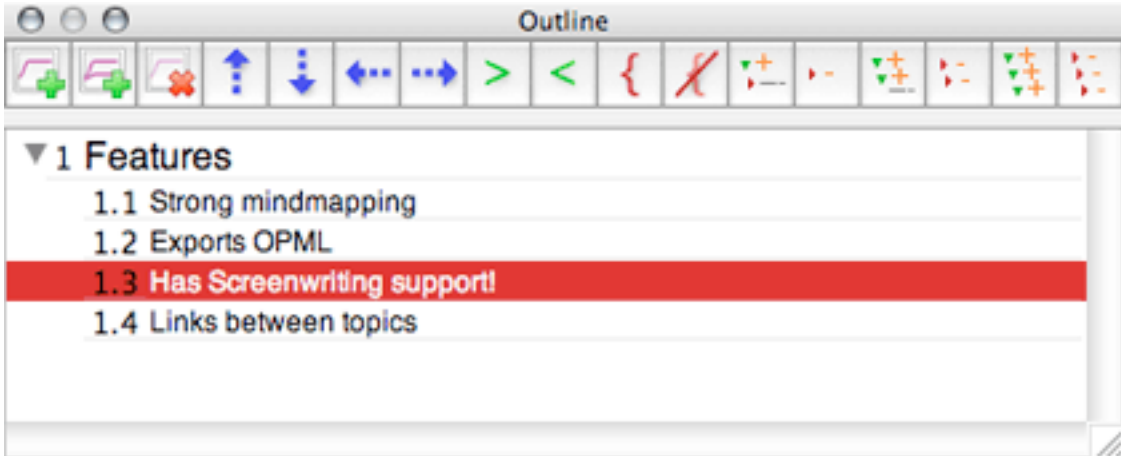


NovaMind (NovaMind Software)

Environment: Cocoa

Price: \$60

Current version: 2.1

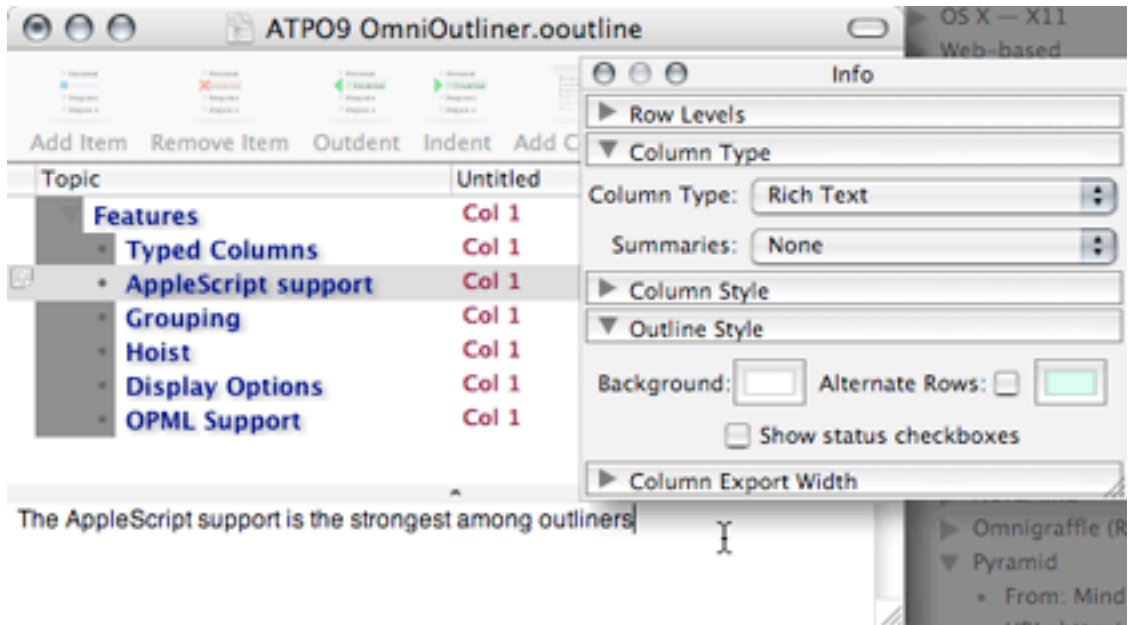


OmniOutliner (OmniGroup)

Environment: Cocoa

Price: \$30

Current version: 2.2.6 (69.6); version 3 announced

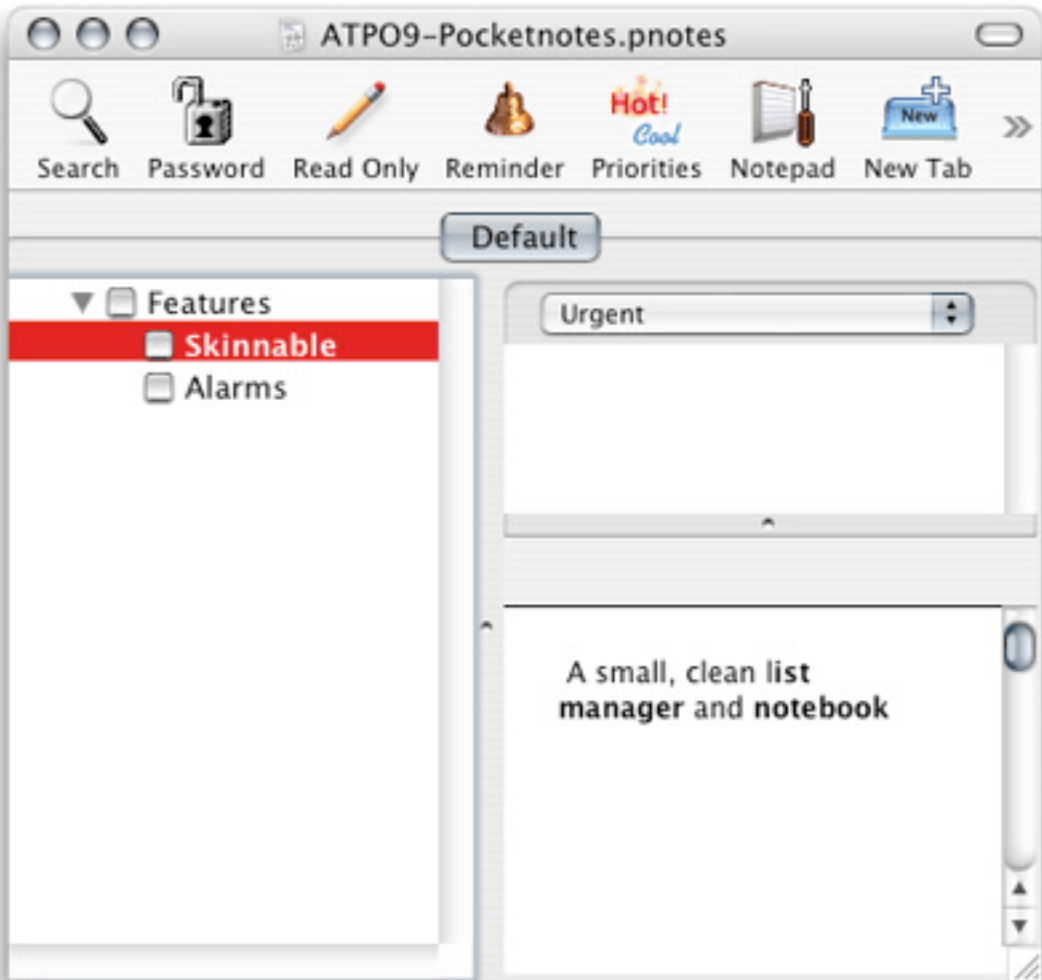


[PocketNotes \(Pocket Software\)](#)

Environment: Cocoa

Price: \$20

Current version: 2.1

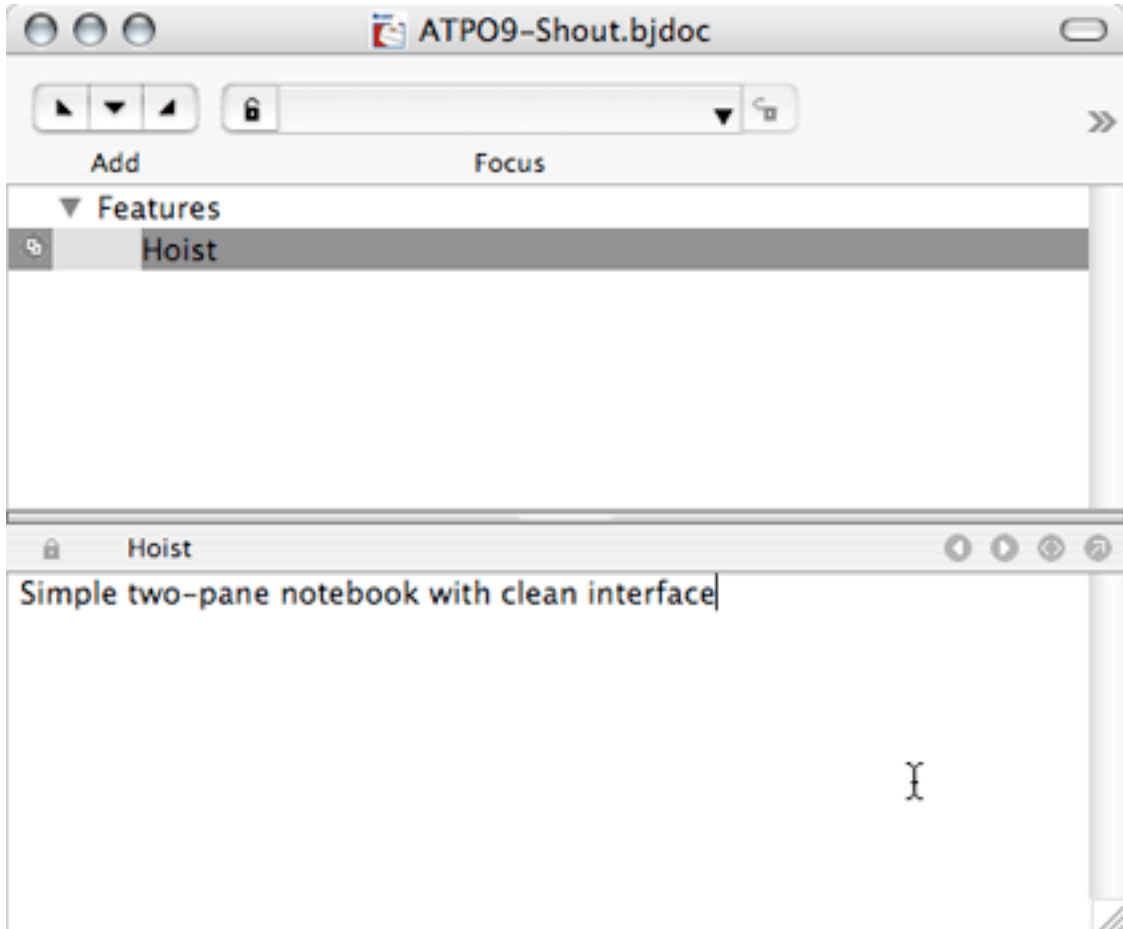


Sh-Out ! My Brain (Strawberry Heads)

Environment: Cocoa

Price: \$20

Current version: 2.1.3



Shadow (Code Jedi)

Environment: Carbon

Price: free public beta (Window/Palm version is \$23)

Current version: 2.0.6

- ▼ ATPO9
 - ▼ Features
 - Good To Do list management
 - Windows, Palm versions
 - Native XML

Title Text:

Native XML

Memo:

A popular task manager on the Windows/Palm side
Native XML means it could collaborate with other
tools, but the Mac OSX version is clunky and
unfinished

Attributes:

Priority: Checked

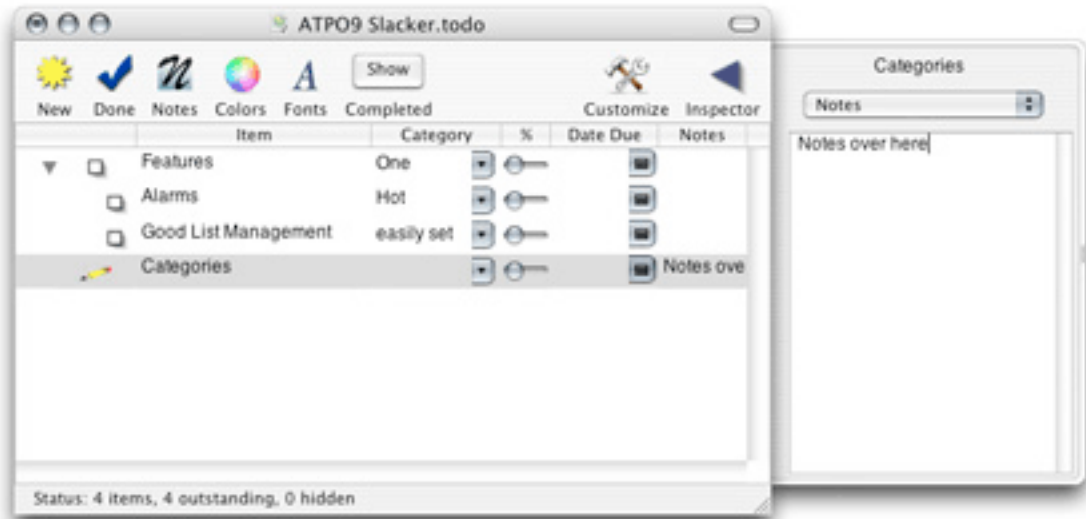
Progress: Bold

Slacker ToDo Lists (Mindlube Software)

Environment: Cocoa

Price: \$25

Current version: 2.1.6

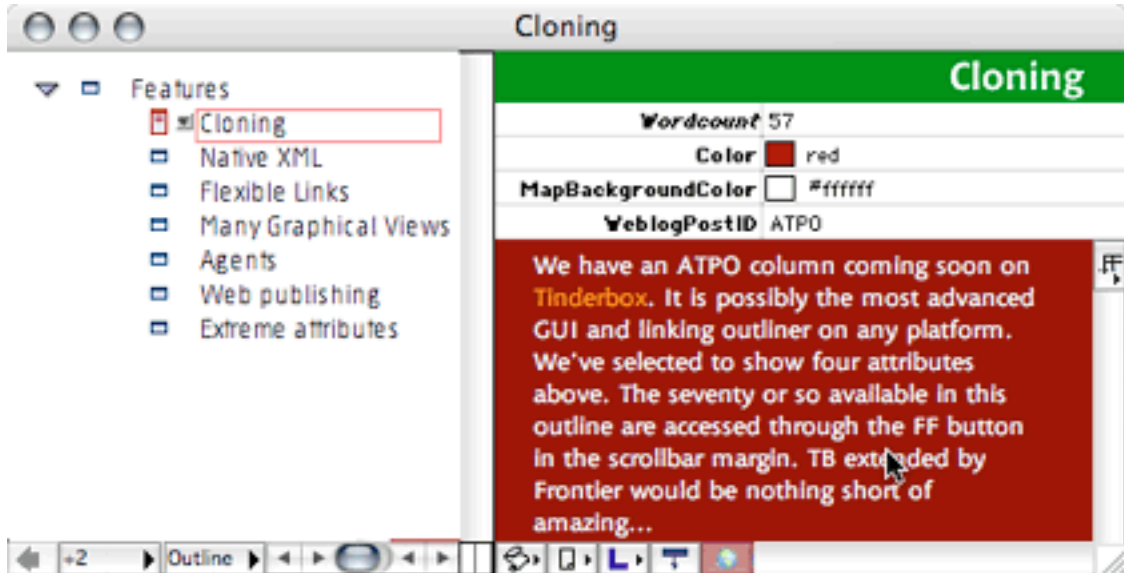


Tinderbox (Eastgate Systems)

Environment: Carbon

Price: \$145

Current version: 2.2

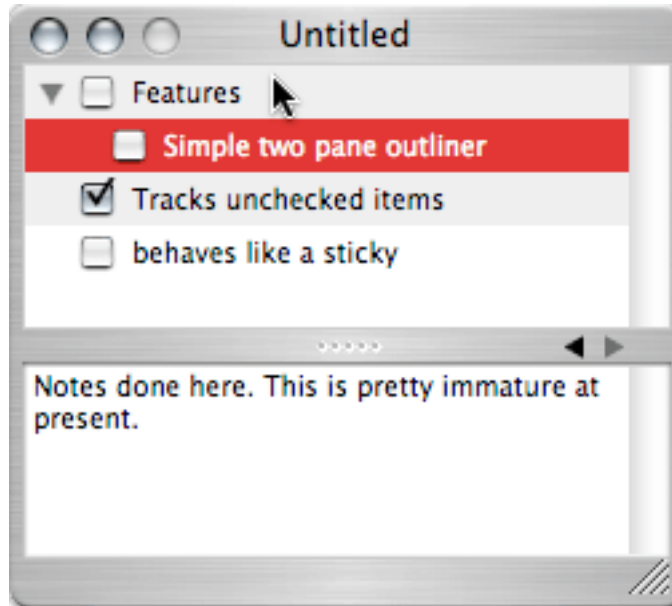


To-Do Stickies (Joseph Sharp)

Environment: Carbon

Price: \$15

Current version: 1.0.1



If you just are tracking a few on your own, look at these for promising new ideas:

- Curio for blackboard outlines
- Tinderbox for metadata richness
- Hog Bay for outline outlining
- FO as a possible writer's outliner
- [Jedit](#), which is being used as a basis for a next-generation open source Arrange (described in our *ATPO* column on [legacy outliners](#))
- [Alpaca](#) with [Clotho](#) as the possible framework for the open source Mac OS X molecule-level tinkerable outliner
- [PathFinder](#) as a potential Finder-integrated outliner

Next month: Mindmapping outliners. Send me your views and recommendations.

Copyright © 2004 Ted Goranson, tgoranson@atpm.com. Ted Goranson has been thirty years in the visualization and model abstraction field. He is slowly beginning a [new user interface project](#).



Using WEP Security on an AirPort Network

Introduction

You and your wireless network get along very well. It moves files where you need them, checks Internet weather when asked, and says nice things about your friends and relatives. It's your metaphorical family dog that runs on electricity. This dog is friendly, though. *Very* friendly. Did you know that anyone near your house, even across the street, can call and ask your dog to fetch stuff for them? If not, it's time for network obedience school.

Wired Equivalent Privacy

Wired Equivalent Privacy (WEP) is a way to protect network data moving from the AirPort Base Station (or ad hoc connection) to your computers. Using a specially generated software "key," it scrambles the data. You need the key to unlock the data, so even if someone is listening in, it'll look like garbage. WEP doesn't keep them from receiving the wireless signal (nothing can, outside of shielding your walls and windows), but the data is useless without the key.

Someone Wrote It; It Must Be True

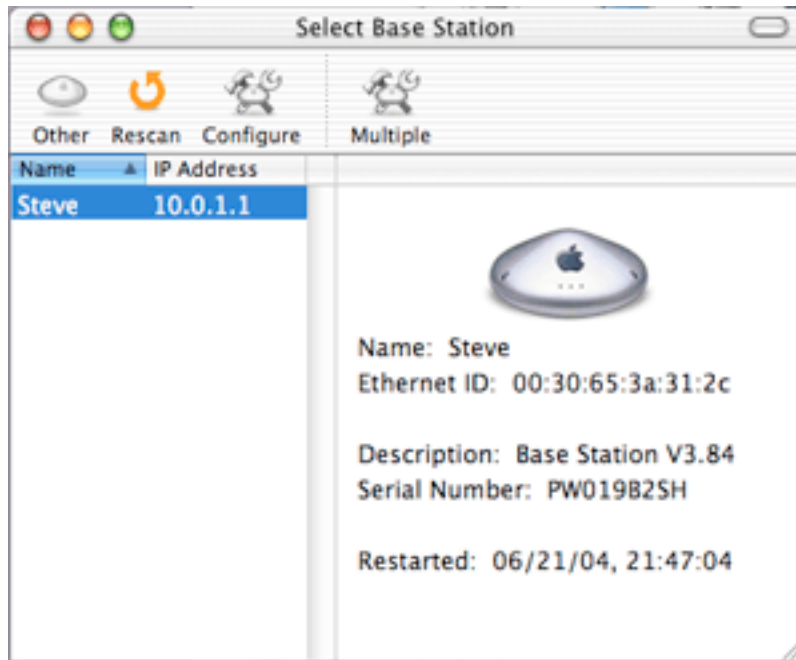
Many know WEP only by rumors of its design flaws. No lie—it's not the world's best security mechanism. However, it's still practical for the standard home or small office user. Consider that the people you're protecting against will steer clear of networks with any kind of security. Why break in through an upstairs window when the next house's door is open? Just *enabling* WEP is most of the battle. Change the network key every week for even more security.

WEP and AirPort

Wireless products must support 64-bit WEP to be certified Wi-Fi compliant. There is also 128-bit WEP, but this article sticks to 64-bit, which all AirPort models support. Wi-Fi compliance shows as a logo on the box or Internet product description.



Assuming your network is already running, open the AirPort Admin Utility from the Utilities folder of your Applications directory. You should see your Base Station. If you don't, verify that the Base Station's power is on, then push Rescan.



Double-click the AirPort name (Steve in my example) or select Configure to open the preferences window. This shows the summary page, which includes WEP details when they're enabled. For now, it should say "not configured."

To enable WEP, select the Name and Password tab, then check Enable Encryption. Select Change Password... to open a window where you enter the new password twice. Then select Update to save the changes. Once the Base Station restarts, select it from the AirPort menu. If it asks for the password, enter it and save it to the keychain. Now the summary screen shows your Wireless Hex Equivalent Password.



Apple AirPort Base Station

Base Station Name: Steve

Wireless Network Name: Steve

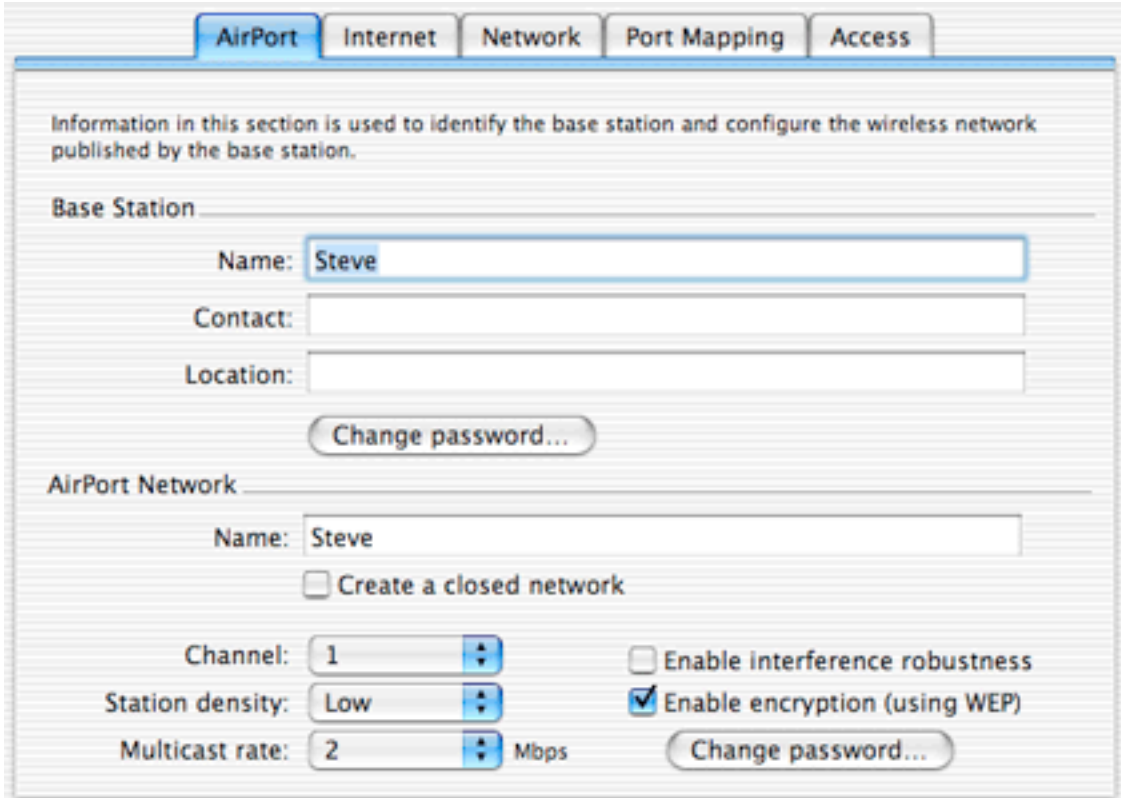
Firmware Version: V3.84

Wireless Channel: 1

Wireless ASCII Equivalent Password: (not printable)

Wireless Hex Equivalent Password: CFC2E9E26D

Other Macs should do likewise—select the AirPort from the menu and enter the password to join. The password should be saved in the keychain, so you only need enter it once. Change this password any time you like using the same steps, or from the Show All Settings page.



Connecting a Windows 2000 system to my AirPort network required entering the Wireless Hex Equivalent Password, so you may need to write this down and carry it to the other computers. Macs should be able to use the password instead. (Your AirPort uses the password to generate the hex sequence, so the same password always generates the same hex values.)

Change your password weekly on home networks and more often on ones that see heavy use. This prevents snoopers from sitting and listening to traffic long enough to break the encryption scheme (very few will attempt this). Each new password renders any previous snooping useless.

Summary

WEP is a snap to enable, and its mere presence defends you from the majority of drive-by wireless thieves. Don't let this security tool just sit in the box—protect your data before someone makes you wish you had.

Copyright © 2004 Matthew Glidden, mglidden@atpm.com. Matthew Glidden is the webmaster of Threemacs.com, a guide to constructing and maintaining home and small-office Macintosh networks. He can also tango and juggle, not necessarily at the same time.



How To

by Charles Wu, ccwu@mac.com

Summer Photo Fun

Summer means vacations, and vacations mean taking lots and lots of pictures with your digital camera. No matter how fun it is to take them, it's even more fun to share them. I'll cover a few fun ways to share your summer memories. The Mac is great since it comes bundled with iPhoto, which lets you organize and touch up your photos with amazing ease. However, once you have a digital camera you may find yourself buried in photos, and instead of your shoebox being overstuffed with photos, your iPhoto library is bursting at the seams.

It's too bad iPhoto lets you have only one library, but Brian Webster's [iPhoto Library Manager](#) solves this problem by letting you create more than one photo library with an interface that allows you to "seamlessly" switch between them. So you might create a library for family photos, one for vacations, or whatever you choose. Now that you've organized the tons of photos, what are you going to do with them? Obviously you'll choose a few to share with friends and family, but which ones? How about all of them all at once?

You may remember a few years ago seeing pictures made up of thousands of little pictures were all the rage, often sold under the trade name Photomosaic®. These clever montages were developed by a student at MIT's Media Lab. Wouldn't it be great if you could create your own photo montage of your favorite vacation picture, made up of your other vacation pictures? Well, now you can by using Frank M. Midgeley's [MacOSaiX](#).

With MacOSaix, you choose the picture you want for your montage and then select image sources to provide the individual tiles that make up your picture. You can choose folders from your own computer, or downloaded from Google Images based on keywords, or choose glyphs (which are different characters and fonts chosen at random from your system). Select how detailed you want your picture to be by choosing the number of tiles you want your montage to have. While it is true that the more tiles the better looking the montage, the downside is that it takes longer. Once these options have been chosen, let

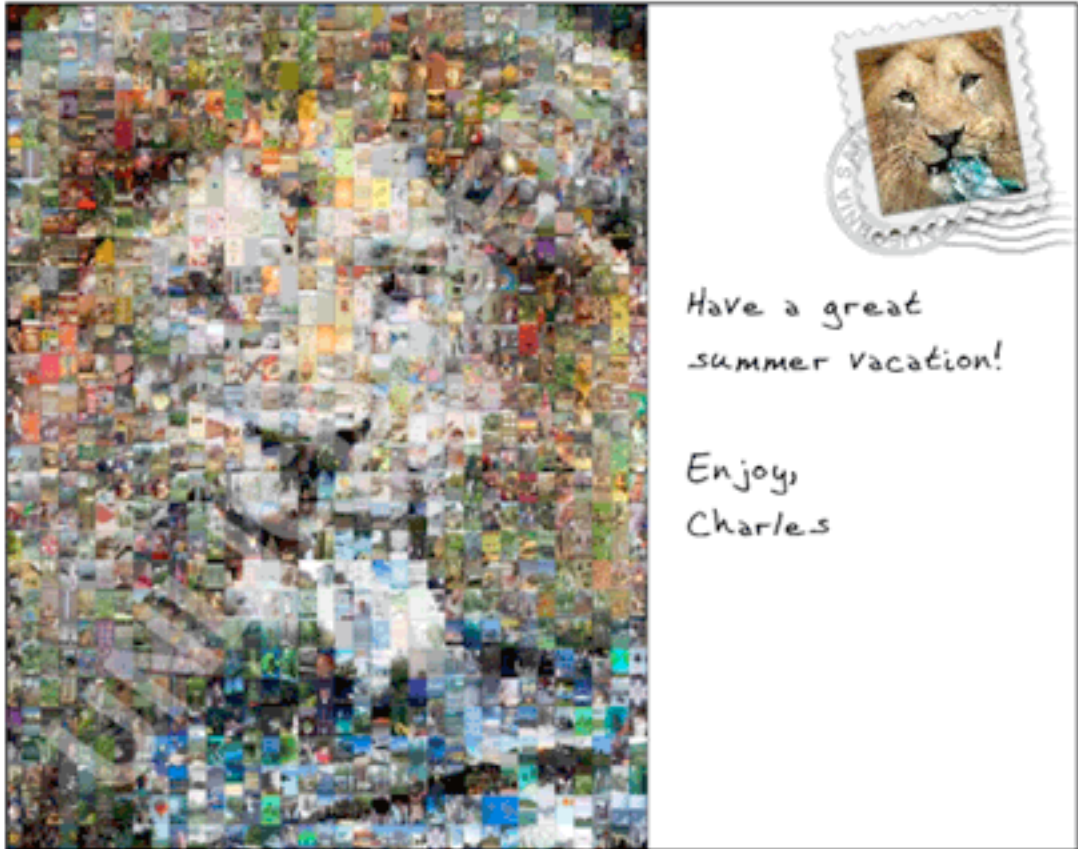
MacOSaix go at it. It's a fun process to watch as your image slowly gets made as MacOSaix chooses the best tiles. I found that a combination of local pictures and Google pictures works best.

Now that your vacation is captured all in one picture, what's next? You can e-mail it to your friends, create an iCard with it if you have .Mac, or post it on a Web site. If you don't have .Mac, you can use Steve DeKorte's [Postcard](#) to make graphical postcards of your own, even letting you customize the stamp and postmark and type a short message. It's a simple application, but lots of fun if you want to create personalized postcards for holidays or special occasions.

The great thing about cameras is that they are the most personal piece of technology around. They don't offer anything that you're not personally involved in, and that is a really neat thing when you think about all the images and media you can buy today. The pictures you take can truly be priceless—and truly your own.

So it doesn't seem right that you should choose to use a regular font that everyone else can get when you create your postcard. [Fontifier](#) is a service that creates a font of your own handwriting. The service is easy to use. Visit the Web site to download a template

and print it out. Using a felt-tipped pen, write down your version of each letter and rescan the template into your Mac. Upload the scan to the Fontifier Web site. You can then preview your new font to decide if you want to purchase it or not.



Take a look at the sample image and you can see everything put together, a MacOSaix photo montage of a lion photo from the AP, in a Postcard. The stamp has the original image so you can compare, and the text of the postcard is in my own personal font.

Have fun with your pictures and have a great summer.

Copyright © 2004 Charles Wu, ccwu@mac.com. Charles is a member of the [North Coast Mac Users Group](#).



What to Do When You Need a Little More Room

Last month, as the rest of the editorial staff was putting the final touches on another stellar issue, I was struggling with some work-related deadlines. In the few moments I took off for fun, my iBook decided to assert itself and prove which of us was boss. Usually a rock-solid workhorse, it suddenly began balking at the more intensive tasks. Near-crashes and a kernel panic or two were involved. It was close for a while there, but I finally regained the upper hand.

In reality, it did not take too long to sort out the problem and the techniques that I used were simple. Many of you will consider them too simple for a column such as this; however, since I had forgotten these pearls of wisdom, chances are others have as well.

The Problem

Even during its worst temper tantrum, my iBook performed well. Its DVD drive no longer works, but that is not much of an issue since I have an external CD-RW drive that I can boot from when necessary. It continued to crank out word processing files, read e-mail, and view smaller graphics files without missing a beat. Nevertheless, all was not well in my little corner of the world.

The first problem that I noticed was Photoshop complaining about scratch disks being full. That was not much of a problem: I had forgotten to connect the external drive that I use for Photoshop's scratch disk. The more troubling problem was that sometimes, in the middle of some intensive task, the Finder would complain about low memory and warn me of dire consequences if I didn't act immediately. The system didn't crash during those times, so a quick restart left me happily computing again, only to have the problem rear its ugly head again several days later. Lack of memory did not seem to be the problem; after all, I had used this same system to perform the same tasks several times. About the third time it happened, I took the time to troubleshoot and solve this problem.

The Cause

After a bit of detective work I was able to resolve what had appeared to be a random problem. The first clue was that these problems always occurred either in Photoshop, or in iTunes while converting music files. This was unusual since neither program had a history of such behavior. Upon closer inspection, I discovered that the problem always occurred while working with large files or large numbers of files. Apply a filter to an eight-by-ten photograph or convert several albums purchased from the [iTunes Music Store](#), and boom. No crash, just dire warnings of things to come. From these clues, some of you already have a culprit in mind. Everyone else can keep reading.

On a good day, my iBook's boot partition has about two GB of available space. That sounds like a lot, until you realize that OS X makes extensive use of swap files. If you are not familiar with swap files, think of them in much the same terms that you would OS 9's virtual memory files. In both cases, by default these files can be found on the boot drive or drive partition.

For our purposes, there is an important difference between these two types of files. While OS 9 allowed you to control the size of virtual memory files manually via the Memory control panel, OS X controls the sizes of these files dynamically. If you do not have enough free drive space, the Finder will complain.

In addition to the Finder complaining, most applications that make extensive use of temporary files will also begin to complain. The only reason I had not noticed until now was that I had not spent enough time with these applications recently; this would explain why neither Toast nor iTunes wanted to convert the music files in my library. I could convert a few at a time, but the minute I ran low on drive space for temporary files there was a general complaint heard throughout the land.

A quick glance at my hard drive revealed that at the time the problems started, I had about half a GB of free space on the drive. For some reason, a song about needing elbow room kept running through my head (I think that's from Scholastic Rock). That settled it: something had to go. A larger drive just was not in the budget. I had already moved Photoshop to an external drive, and at first glance I could not see anything else that would free up enough space to make a difference.

The Solution

The first step is to get rid of large files that you no longer need. You could spend your time searching for these files manually but that would seriously interfere with time better spent enjoying your Mac. Why not let a utility do the job for you? [OmniDiskSweeper](#) searches a hard drive and shows you a list of all files and folders ranked by size. You may then delete files you no longer need. With the free version, you must delete the files manually. Paying the \$15 purchase price enables the delete button. This option has the added advantage that OmniDiskSweeper will not let you delete system files.

For many users who still need additional space the answer is simple. Move some data files to another location. Under OS X, many applications encourage you to store data inside subfolders in your home folder. Word processors, for example, may default to the Documents folder inside your home folder. By navigating through the Save As dialog box you can choose to save files in another location if you wish. I have never been in the habit of storing data in my home files so that was not an answer for me. It may also not be the best option in a multiple user setting.

One day, during my third attempt to finish converting the files in my iTunes Music folder I realized that these were the only type of data files stored inside my home directory. Performing a quick Get Info on my Music folder revealed that several albums located therein were consuming well over one GB of precious hard drive real estate. Those files had to go.

I was able to burn these albums to a CD and remove them from my hard drive. This of course meant that playing the files back on the computer was only possible when I had access to the CD. This was at best a temporary solution, and sure enough after a recent downloading spree at the iTunes Music Store I was once again faced with the need to move music files off my primary drive.

This time I found a different solution. I used a small external hard drive, but this should also work if you move the folder to another internal drive or partition. The basic steps to accomplish this are as follows:

- Use a disk utility to check the health of the drive or partition that will house your new music folder. This is mainly a precaution, but there is no point putting the folder on a bad drive.

- Go to your home folder and locate your iTunes Music folder. Copy the iTunes Music folder to its new location. In order to avoid any problems I made sure to use a utility that is aware of OS X file permissions: there are several such utilities out there, but I used [Carbon Copy Cloner](#).
- Once you have successfully copied your iTunes Music folder to its new location, make an alias of that folder. Do this by selecting the folder (in its new location), and either select “Make Alias” from the File menu or press Command-L. Delete the iTunes Music folder within your Home folder, and replace it with the alias that you just created. Be sure to remove the word “alias” (including the space character in front) from the alias file name.
- The last step is to launch iTunes and tell it where its new Music folder is located. In the current version of iTunes, you can do this by launching the program and pressing Command-Comma to open the Preferences dialog window. Once there, click on Advanced. You will see the pathname to the old music folder location. Click the “Change” button, and you will have an opportunity to select your iTunes Music folder in its new location.

I am unsure whether the step of creating the alias is necessary; I did it mainly so that I would not look there one day and wonder where the music files were.

Conclusion

If a lack of drive space is causing you some problems and a larger hard drive is not practical then these techniques may help. I have listed them in the order that seems to be most beneficial.

Without running any applications that use temporary files or scratch disks, see how much free drive space is available on your boot drive. If there is not very much space then it might be time to move some data files to a new location. When you do this, use a utility that preserves file permissions.

If you seem to have enough space on the boot drive and the problem only affects certain programs, tackle the problem that way. Launch each program and see if there is a preference that controls how large the temporary files are or where they are located.

Theoretically, the techniques applied to the iTunes Music folder should work with other types of data as well. Which folders you need to move will depend upon the type and size of the documents that you create. Keep track of the original locations for any folders that you relocate, as it may prove necessary to relocate them back.

Be aware that if you move any files to an external drive, then that drive must remain switched on, attached to your computer, and made available to the Finder for its files to be available. Depending upon which files you have moved, some programs may not function correctly if they are unable to access necessary files.

Although I have seen procedures to move OS X's swap files to new locations, there appear to be numerous problems or side effects that can occur as a result. The same can be said for moving the entire Users folder to a new location. If the steps I have outlined have failed to free up enough space to resolve the problem and you have enough RAM installed for OS X to function acceptably, then either something else is going wrong or it's time to buy a larger hard drive.

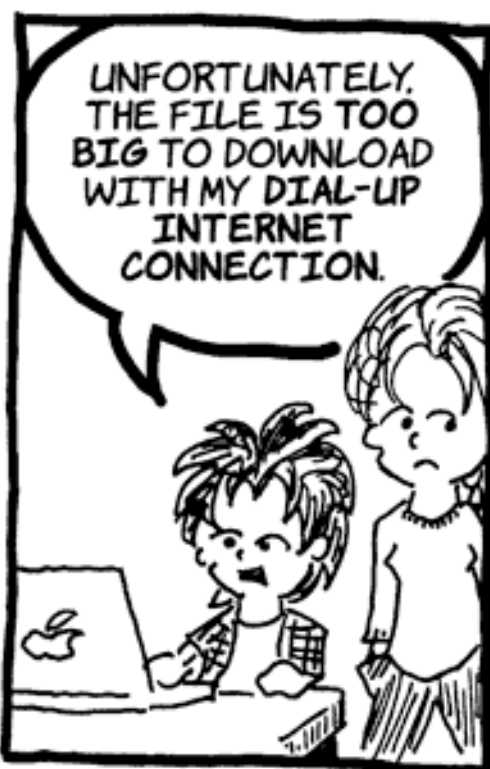
Copyright © 2004 Sylvester Roque, sroque@atpm.com.



Cartoon: Cortland

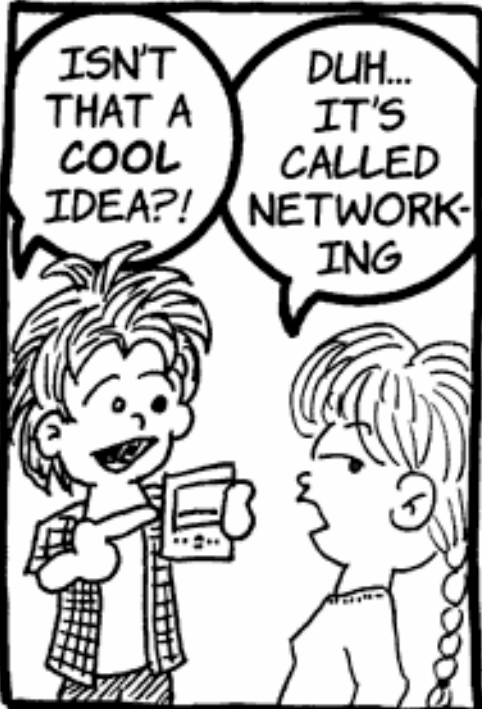
by Matt Johnson, mjohnson@atpm.com

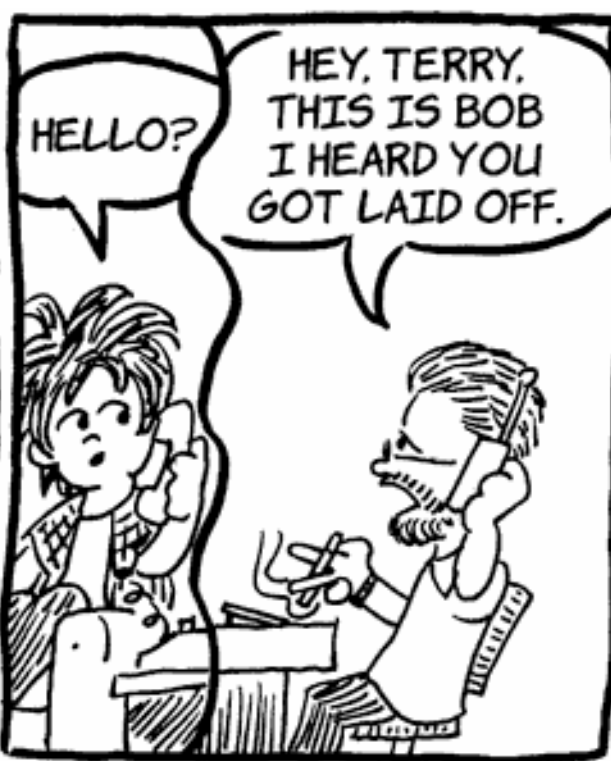
















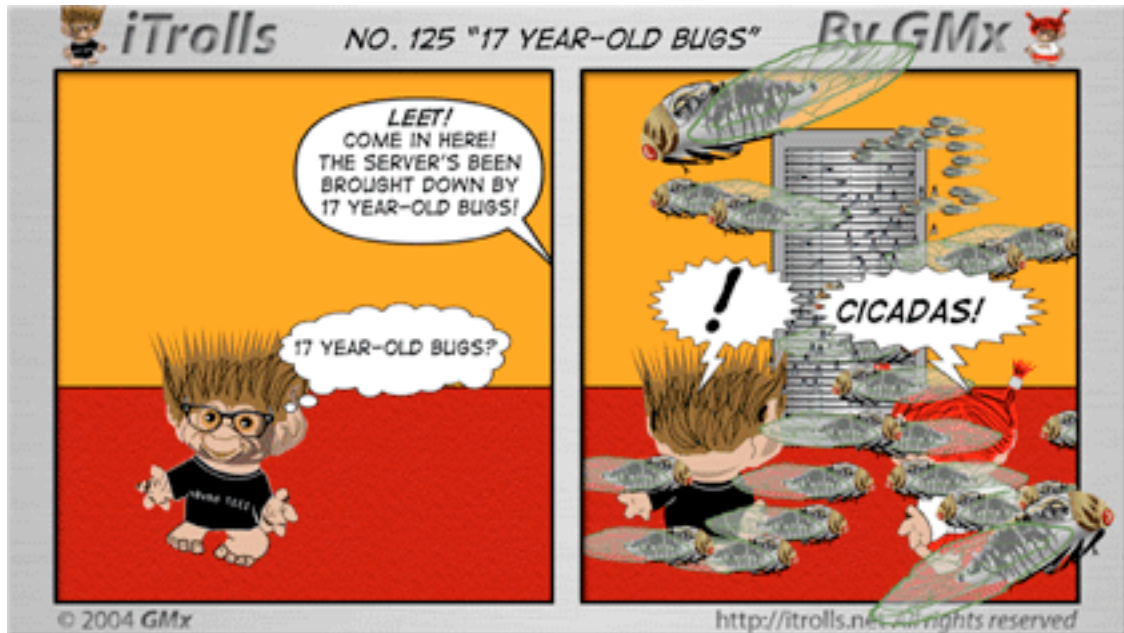


Copyright © 2004 Matt Johnson, mjohnson@atpm.com.



Cartoon: iTrolls

by GMx, <http://homepage.mac.com/gregmx>





iTrolls

NO. 126 "WORK NECESSITY"

By GMx



© 2004 GMx



<http://itrolls.net> All rights reserved



iTrolls

No. 127 "Forty Days..."

By GMx



© 2004 GMx



<http://itrolls.net> All rights reserved



Copyright © 2004 GMx, <http://homepage.mac.com/gregmx>.



Volcano National Park

Volcano National Park

This month features the second part of Christopher Turner's 2001 visit to the Big Island of Hawai'i and Kilauea Volcano National Park. The theme for July is lava, lava, lava!

There are several different sizes of lava tubes on the island. Lava tubes are created during volcanic eruptions when the lava flows, not to the surface, but rather through less-voluminous rock, in its race to the sea. One tube is so large that there is a trail through it, the opening of which is featured in the first photo. Other pictures of a lava tube are also present.

The lava desert of the eastern side of the island is showcased, as well as a distant show of live lava flowing in to the Pacific, throwing up clouds of steam. Finally, a massive sea arch is depicted, as well as the small black-sand beach near its base.

Previous Months' Desktop Pictures

Pictures from previous months are listed in the [desktop pictures archives](#).

Downloading all the Pictures at Once

iCab and Interarchy (formerly Anarchie) can download an entire set of desktop pictures at once. In iCab, use the Download command to download "Get all files in same path." In Interarchy, use HTTP Mirror feature.

Contributing Your Own Desktop Pictures

If you have a picture, whether a small series or just one fabulous or funny shot, feel free to send it to editor@atpm.com and we'll consider publishing it in next month's issue. Have a regular print but no scanner? Don't worry. E-mail us, and we tell you where to send it so we can scan it for you. Note that we cannot return the original print, so send us a copy.

Placing Desktop Pictures

Mac OS X 10.3.x

Choose “System Preferences...” from the Apple menu, click the Desktop & Screen Saver button, then choose the Desktop tab. In the left-side menu, select the desktop pictures folder you want to use.

You can also use the pictures with Mac OS X’s built-in screen saver. Select the Screen Saver tab which is also in the Desktop & Screen Saver System Preferences pane. If you put the ATPM pictures in your Pictures folder, click on the Pictures Folder in the list of screen savers. Otherwise, click Choose Folder to tell the screen saver which pictures to use.

Mac OS X 10.1.x and 10.2.x

Choose “System Preferences...” from the Apple menu and click the Desktop button. With the popup menu, select the desktop pictures folder you want to use.

You can also use the pictures with Mac OS X’s built-in screen saver. Choose “System Preferences...” from the Apple menu. Click the Screen Saver (10.1.x) or Screen Effects (10.2.x) button. Then click on Custom Slide Show in the list of screen savers. If you put the ATPM pictures in your Pictures folder, you’re all set. Otherwise, click Configure to tell the screen saver which pictures to use.

Mac OS X 10.0.x

Switch to the Finder. Choose “Preferences...” from the “Finder” menu. Click on the “Select Picture...” button on the right. In the Open Panel, select the desktop picture you want to use. The panel defaults to your “~/Library/Desktop Pictures” folder. Close the “Finder Preferences” window when you are done.

Mac OS 8.5–9.x

Go to the Appearance control panel. Click on the “Desktop” tab at the top of the window. Press the “Place Picture...” button in the bottom right corner, then select the desired image. By default, it will show you the images in the “Desktop Pictures” subfolder of your “Appearance” folder in the System Folder, however you can select images from anywhere on your hard disk.

After you select the desired image file and press “Choose,” a preview will appear in the Appearance window. The “Position Automatically” selection is usually fine. You can play with the settings to see if you like the others better. You will see the result in the little preview screen.

Once you are satisfied with the selection, click on “Set Desktop” in the lower right corner of the window. That’s it! Should you ever want to get rid of it, just go to the desktop settings again and press “Remove Picture.”

Mac OS 8.0 and 8.1

Go to the “Desktop Patterns” control panel. Click on “Desktop Pictures” in the list on the left of the window, and follow steps similar to the ones above.

Random Desktop Pictures

If you drag a folder of pictures onto the miniature desktop in the Appearance or Desktop Pictures control panel, your Mac will choose one from the folder at random when it starts up.

DeskPicture

An alternative to Mac OS’s Appearance control panel is Pierce Software’s DeskPicture, [reviewed](#) in issue 5.10 and available for [download](#).



Review: BOOQ BP3 System

by Christopher Turner, cturner@atpm.com

Developer: [BOOQ](#)

Price: \$160

Requirements: a 15-inch laptop for the PowerSleeve 15



BOOQ (pronounced like “Luke”) has quietly emerged from the recesses of California to become a leading manufacturer of quality computer bags. Its goal is “the fusion of toughness and comfort in design,” and I would have to say it has accomplished that in its flagship product, the [BP3 System](#).

Building off the [BP3 backpack](#), the System adds the PowerSleeve of your choice, up to 15 inches in size, the PDAPaq PDA case, and the FonePaq mobile phone case. Individually, these items combined would cost between \$197 and \$212, so the System is a good value at \$160.

The BP3

The backpack itself can be used without the PowerSleeve, and I have done so many times, when I either did not require or did not have the PowerBook. While it won’t compare to the [Brain Bag](#) in its overall capacity, the BP3 is loaded with pockets and storage space. I

was able to cram everything I “must” have with me while “Book-totin” and still had room left over to wedge my camera bag in on top. Again, it doesn’t match up to the Brain Bag in capacity, but I favor the BP3 because of its pocket and zipper layouts.



One bit of customization the BP3 offers over other bags is the [Shell](#), a single pocket padded piece that snaps onto the backpack. BP3 Shells are offered in a variety of colors, with even a camouflage pattern for the militarily deployed PowerBook driver. My Shell is black, on a gray BP3.



The BP3 with its Shell detached.

The BP3 also features a multimedia pocket, which has a pass-through for your iPod's headphones or earbuds. The pocket is covered up when the Shell is in place, but you can still run the line for your cans without issue. Personally, I never used this feature, as my iPod tends to be located on my hip in its [Contour Showcase](#).

A PowerSleeve for a PowerBook

My BP3 System arrived with a [PowerSleeve 15](#), for my Titanium PowerBook G4/500. The PowerSleeve is solidly built from ripstop ballistic nylon and can be used as a carrying case in its own right, if all you want to carry is your PowerBook, its AC adapter, and a magazine or two.



PowerSleeve

The PowerSleeve comes with its own strap for over-the-shoulder carry and has a comfortable handle at the top for quick transport to that meeting in the executive boardroom. There is a small zippered pocket on the front flap, which the yo-yo AC adapter for my PowerBook easily slid into. The large zippered pocket on the back holds the adapter-to-wall cable that goes with the adapter. It can also accommodate a couple of magazines or file folders.

I normally do not care for vertical carrying sleeves, but the PowerSleeve is among the best-looking of the lot and feels hefty enough to me to be used as a stand-alone carrying case.

I no longer have the PowerBook G4/500, as it belonged to my former employer. After being laid off (along with 100 other fellow IT workers), I did acquire a 12-inch PowerBook G4/1 GHz. The PowerSleeve 15 holds the smaller PowerBook sibling ably, though I slide it in horizontally, as opposed to the vertical alignment of the 15-inch PowerBook. Still, I would recommend the [PowerSleeve 12](#) for those with smaller Mac portables.

Pack Paqs

Two other pieces of the BP3 System are the [PDApaq](#) and the [FonePaq](#), small pocketed cases for, you guessed it, PDAs and mobile phones. The FonePaq can also be used as an iPod sleeve. Both are designed to be used either on your belt or on the BP3's shoulder straps.

The versions of these two pieces are different from the original ones I received, pictured below. I have not had a chance to use the current versions, but it appears they fix my only druthers with the BP3 System. The original PDApaq and FonePaq did not have Velcroed flaps, but rather the flaps were zippered on top, and I found it difficult to use them with my Palm Tungsten T2 and Motorola flip-phone. The new versions make access to these mobile devices much easier and are welcome replacements.



Booq'em, Mac-o

The BP3 System has become my new favorite backpack system for PowerBook portability. It accomplishes the dual mission of form and function set forth by the company, offering individual customization through different bag and shell colors, as well as having a layout that makes it easy to get to anything quickly. My BP3 has been on numerous cross-country trips and has held up very well in getting tossed about in planes, trains, and automobiles.

BOOQ offers a 100% satisfaction guarantee (within 30 days of purchase), a 100% price guarantee (within 90 days of purchase), and a 100% lifetime warranty on its products. There is a 0% chance of you going wrong with this pack system.

Copyright © 2004 Christopher Turner, cturner@atpm.com. Managing Editor Christopher Turner is enjoying the ultra-portability and power of the 12-inch PowerBook G4 and looks forward to the first opportunity to burn a DVD while iChatting at 35,000 feet. Reviewing in ATPM is open to anyone. If you're interested, write to us at reviews@atpm.com.



Review: iTalk

by Lee Bennett, lbennett@atpm.com

Developer: [Griffin Technology](#)

Price: \$40

Requirements: 3rd-generation iPod with dock connector. (iPod Minis currently have no recording capabilities.)

Trial: None



After [dissing Griffin Technology's SightLight](#) last month, a few colleagues of mine got the mistaken impression that I don't like any of the company's products. Nothing could be further from the truth. I own a PowerMate and love it. I've also tried out the iFire and iMic, and would certainly purchase them if I really were in the market for them.

The iTalk is yet another addition to the list of awesome Griffin products.



The Ideal iTalk Customer

As a worker in the field of communication, on occasion I need to take information from someone to use in a story. I'm not exactly a journalist, but I pretend to be one once in a while. My ability to scribble notes quickly is sorely lacking. Sometimes, I remember to carry my department's micro-cassette recorder with me, but usually not. Instead, I now keep an iTalk in the center console of my car—and, of course, my iPod is rarely very far away!

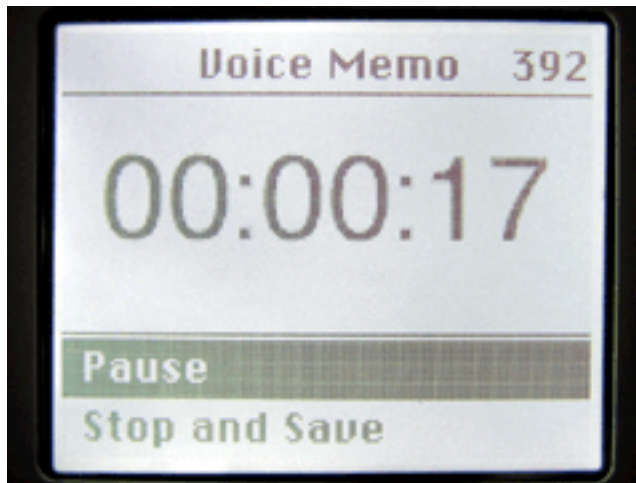
To be clear, the iTalk is not something people should run out and purchase an iPod for. Rather, the iTalk is an extremely useful device for specific categories of people who already own a third-generation iPod. Those people mainly include journalists who are frequently taking interviews, students who like to record lectures, bosses whose secretaries are tech-savvy enough to work with the WAV files for dictation, and any geek who likes to record voice reminders. (For what it's worth, dedicated digital voice recorders that hold more than just a couple minutes of audio don't cost much, if any, less than the iTalk.)

Using the iTalk

The iTalk couldn't be simpler to use. Just attach it to the top of your iPod and you are immediately presented with the screen to begin a recording.



Access the recording screen either by attaching the iTalk or via the Extras>Voice Memos>Record Now menu.



A large-print timer is displayed during recording, along with menu options to pause or to stop and save your memo.



Each recording appears chronologically in the Extras>Voice Memos menu. Selecting a recording offers the choice to play it or delete it.



Memo playback is functionally identical to song playback.

Wish List

Admittedly, I wavered several times between rating the iTalk as Excellent or Very Nice. If ATPM rating scale were numerical, 5 being Excellent, I'd have given the iTalk a 4.8. Thus, Excellent is the closest rating. There are three ever-so-minor points for improvement.

First, while Griffin conveniently added a headphone pass-through jack on the top of the iTalk, which doubles as an auto-switching mono microphone input, the iPod remote jack is not duplicated. If your conditions permit you to just leave the iTalk attached at all times, you won't be able to use Apple's iPod remote control. (The iTalk also covers the hold switch with no way to access it without removing the iTalk.)

The second gripe may just be inherent with how the iPod operates. When you click the option to begin recording, about two seconds elapse before recording actually begins. This is not readily apparent to the user at the time recording is started. Apparently, this is the time needed for the hard drive to spin up. Since there is really no confirmation between when you initiate recording and when you actually are recording, my advice is to always wait until you see the counter display one second elapsed before speaking.

Third, Griffin's product page states that the "built-in speaker is amazingly loud for its size." This is true—I once let 3–4 friends listen to a new song using the speaker and everyone had no problem hearing once I turned the volume up to somewhere around

90–95 percent. However, quality and volume are two entirely different things—a differentiation Griffin’s iTalk product page doesn’t seem to make. Sure, the speaker is great for listening to a reminder you previously recorded or for other brief playback needs. Its quality is exactly what I’d expect for something so small. Perhaps even a bit better. But I really wouldn’t enjoy listening to an entire lecture or a playlist of songs from it.

Audio Quality

Conversely, the recording quality is quite stunning when heard through headphones or decent computer speakers. The files it produces are 8-kilohertz, 16-bit, mono WAV files. These specs are dictated by Apple, not Griffin—indeed, it’s actually the iPod that performs the recording, not the iTalk. It only manages the recordings. This level of sound quality creates audio files that are roughly a megabyte per minute, give or take. So even if you have less than a GB free on your iPod, you probably have room for several hours of recording. It’s a safe bet you’ll run out of battery before you run out of space.

To test the quality of the recording specs, I held the iPod/iTalk at a typical distance that I would hold it for navigating the menu—around 8 inches from my mouth—and recorded a few seconds of talking. When I played back the file through my computer’s attached speakers, even though there was a slight hiss that is to be expected from an 8-kilohertz recording, the clarity was perfect. Even at a point when I raised my voice a bit, there was no distortion. Thus, for interviews, the iTalk passes with flying colors.

[normal.wav](#): A typical iTalk WAV file, recorded from 6-8 inches.

[close.wav](#): Resist the temptation to use the iTalk an inch or two away from your mouth, like a phone. Note, be sure your computer speakers are not set too high when playing this file. The distortion is in the recording—not your system.

To simulate a classroom/lecture setting, I took my iPod and iTrip into my weekly office staff meeting, which is held in a room that seats around 100. There were probably about 35 people in the room at the time. I sat about 3/4 the way to the back—maybe 20–25 feet from the person talking up front—and recorded for a few minutes. During the recording, two different people spoke. One was rather quiet and the other had one of those voices that really carry. In spite of obvious ambient noise in the recording, the second person’s voice came through loud and clear. The first person was noticeably quieter in the

recording, but raising my laptop's volume setting all the way (my external Yamaha powered speakers generally always remain set at about half power) enabled me to hear every word.

[far.wav](#): An iTalk recording from about 15-20 feet.

So, if you're in a good-sized lecture hall and sitting more than 4–5 rows from the speaker, you may need to either move closer, or ask permission to set your iPod on the lecturer's table.

Conceivable Feature

If there was only one thing I really wished were a feature of the iTalk, it would be for it to accept a stereo line-level input and record 44-kilohertz stereo WAVs or AIFFs. As I previously mentioned, the 8-kilohertz spec is dictated by Apple, so Griffin has no way to incorporate a better quality setting, even at the expense of using more space on the iPod. I occasionally record seminars (with the presenter's permission) sponsored by my office and make audio tapes available for sale. Many have inquired if we're going to start doing CDs instead of tapes. I'm liable to purchase an iMic soon and digitally record the seminars directly to my computer. But, recording CD-quality through an iTalk to an iPod, connected to the room's sound system, would be far more convenient if this were possible. The recordings would sync into iTunes, from where I could easily start burning CDs.

Interaction With iTunes

Speaking of moving the recordings to and from your computer, there is a small caveat to keep in mind. Recordings initially are accessed in the Extras>Voice Memos menu. When you sync your iPod, they show up in iTunes as a playlist named Voice Memos. The "gotcha" is that, after synchronization, all prior recordings no longer show up under the Extras menu. They now appear in your iPod, as you have probably guessed, as a regular playlist named Voice Memos.

The Verdict

Don't expect the iTalk to make worthwhile recordings at concerts or large outdoor events. But, if close-proximity recordings could easily become a part of your normal routine, the iTrip is small enough, cheap enough, and convenient enough to possibly be the best digital voice recorder you can buy.

Copyright © 2004 Lee Bennett, lbennett@atpm.com. Reviewing in ATPM is open to anyone. If you're interested, write to us at reviews@atpm.com.



Review: Parliant PhoneValet

by Andrew Kator, akator@atpm.com

Developer: [Parliant](#)

Price: \$200

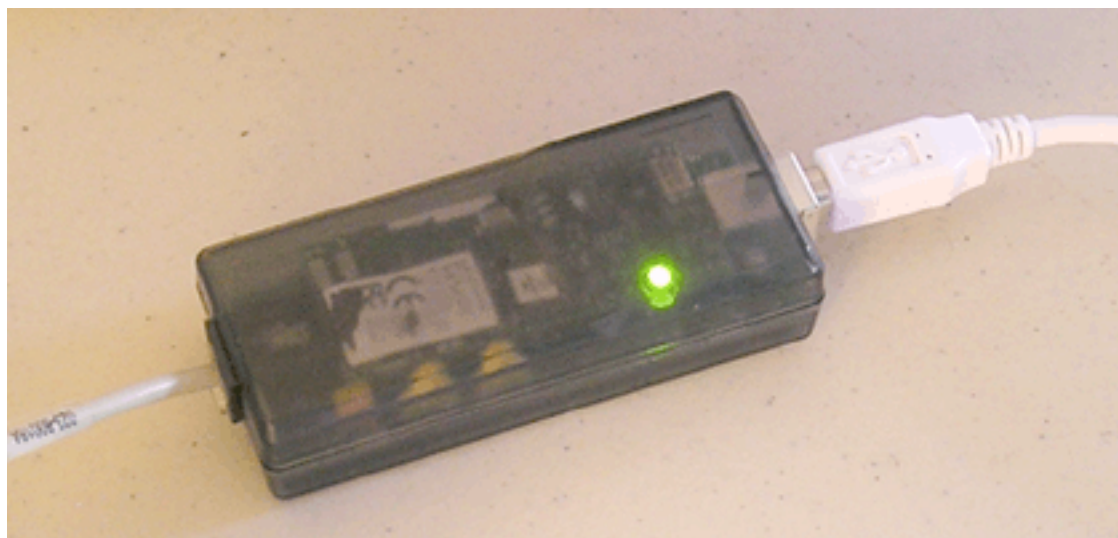
Requirements: Mac with USB port, Mac OS X 10.1.5, QuickTime 6.3.1, analog phone line.

Trial: None (30-day money-back guarantee).



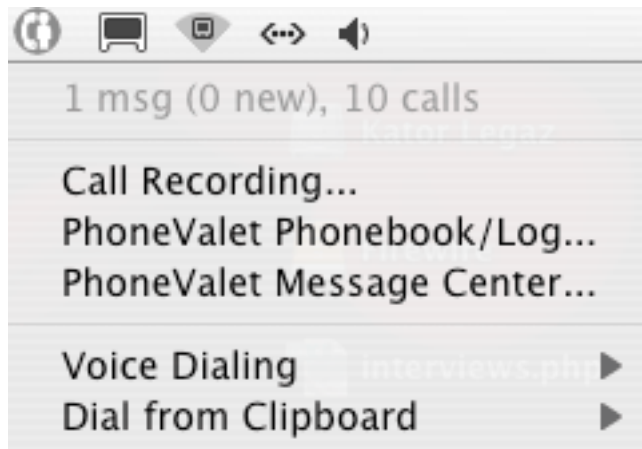
For many years, Mac users have been looking for a decent telephony solution to allow their computers to manage voice mail, caller ID, and fax data. With Mac OS X 10.3 Apple introduced fax capabilities, but OS X has to date been found to be lacking by way of voice features. How can a Mac truly be a digital hub without these features?

Parliant's PhoneValet resolves these missing capabilities. Using this hardware/software solution, small businesses and individuals can use their Mac as a telephone communications hub. The package includes a USB phone adapter that connects a Mac and a phone line (a standard household RJ-10 plug); a USB cable; and a software CD.



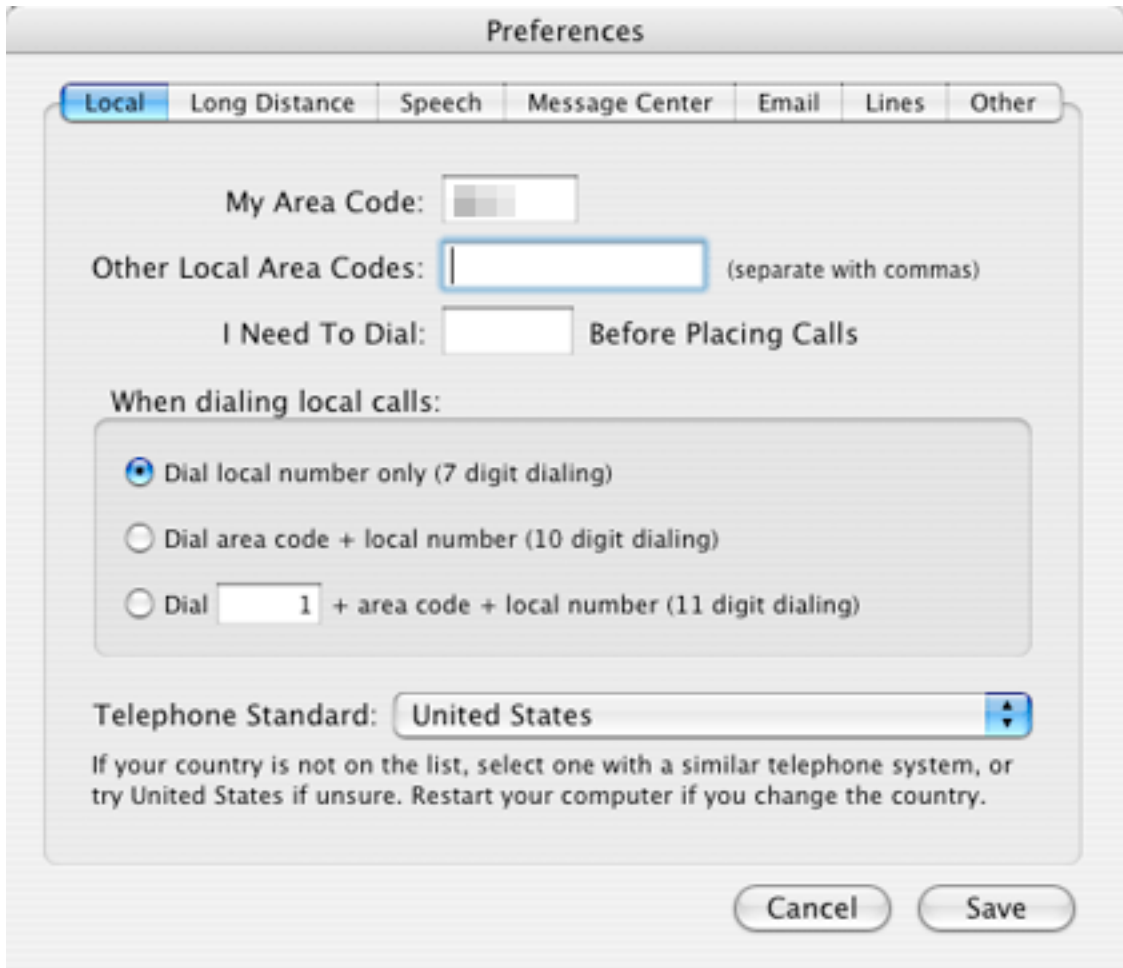
Installing PhoneValet is simple, and takes less than five minutes. The USB phone adapter should first be plugged into your telephone line, then into your Mac using the supplied USB cable. Install the software from the CD and you're ready to go.

PhoneValet is its own voice mail system. It lets you use your phone to remote control your Mac, record your phone calls, and use one telephone line to take both voice and fax calls. The software breaks these capabilities down into three basic functions, all accessible from a menu in the menu bar: Call Recording, Phonebook and Call Log, and Message Center.



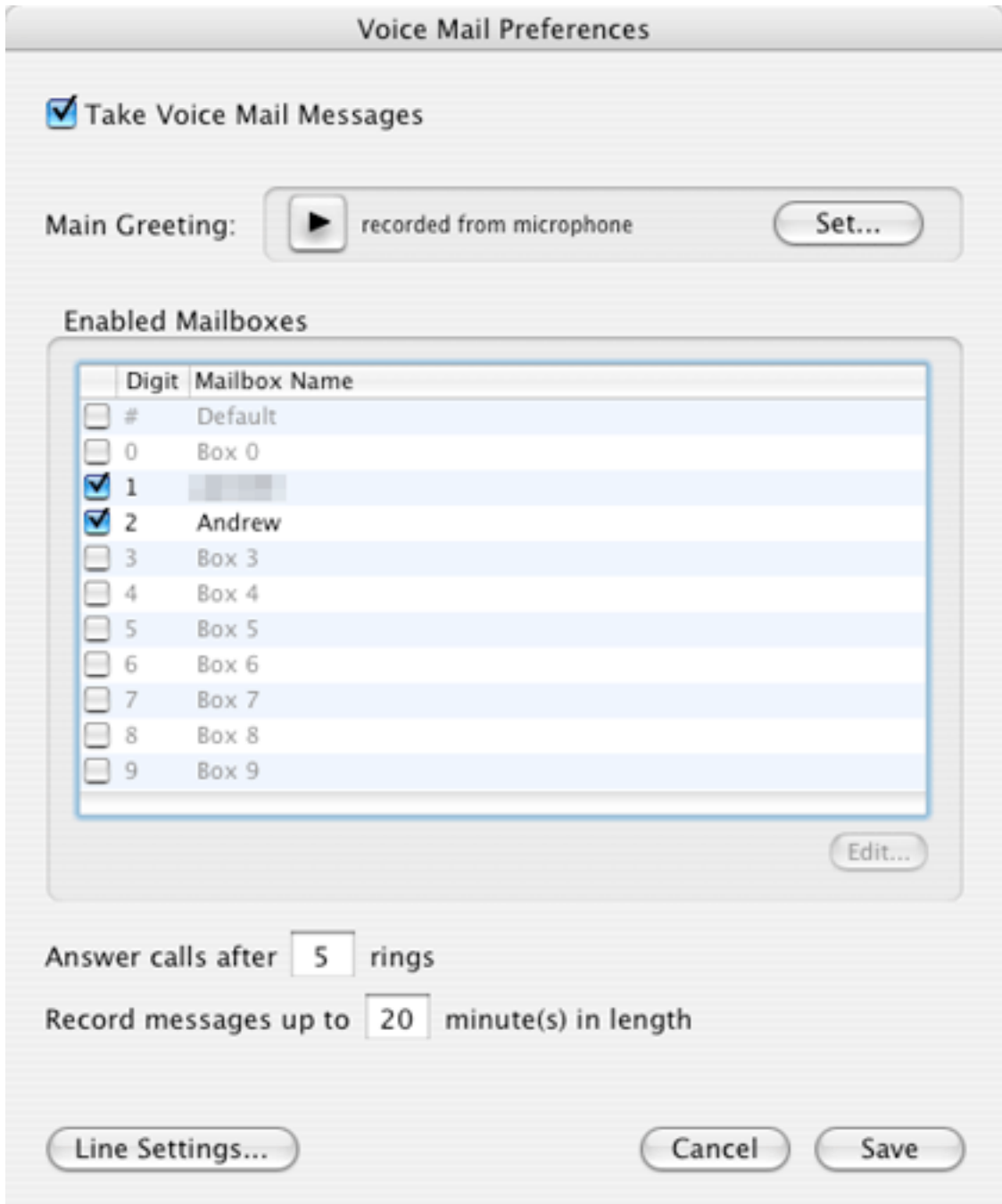
Software Setup

The first steps after installation involve using PhoneValet's Preferences, Voice Mail Preferences, and Remote Control Preferences to determine how the software should function. Setup is extremely easy, and while the printed manual offers instructions that are easy to follow, the software was self-explanatory and the manual was not required.



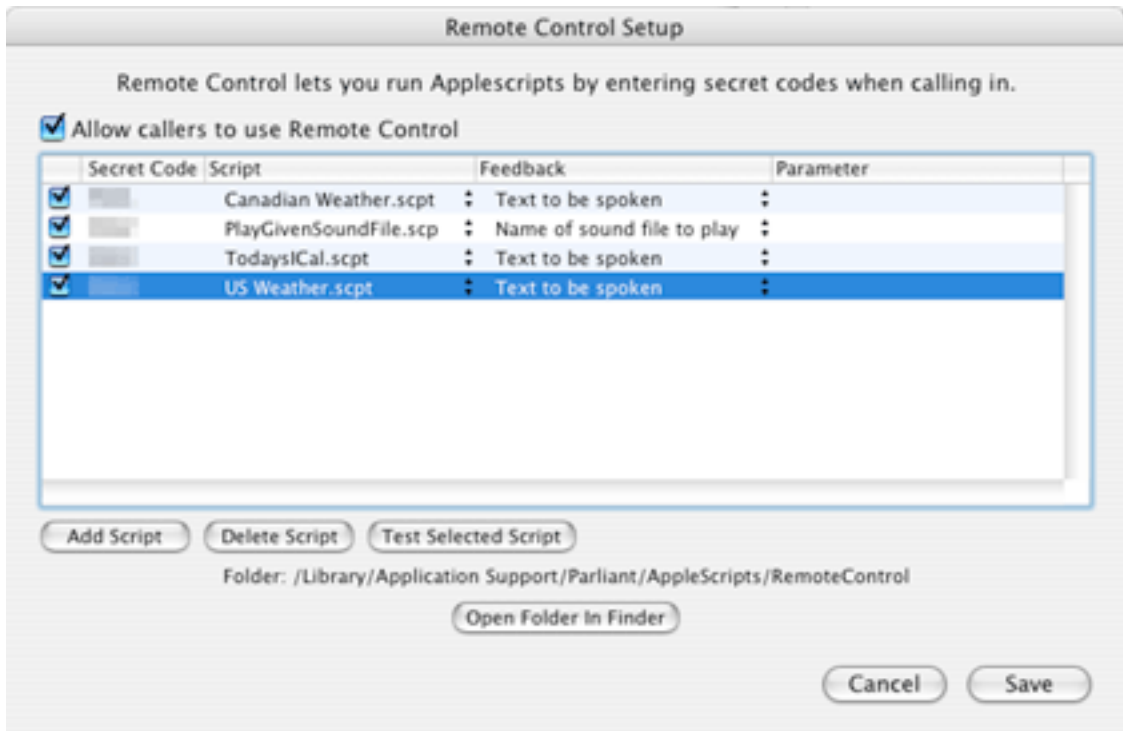
The basic PhoneValet preferences provide options for long distance, incoming call announcements with caller identity, message recording quality, e-mail options (including e-mail notifications and e-mailed phone messages), management of multiple phone adapters for multiple phone lines, and fax handling.

Fax setup offers a choice between using OS X's built-in fax abilities and using PageSender (separate shareware software).

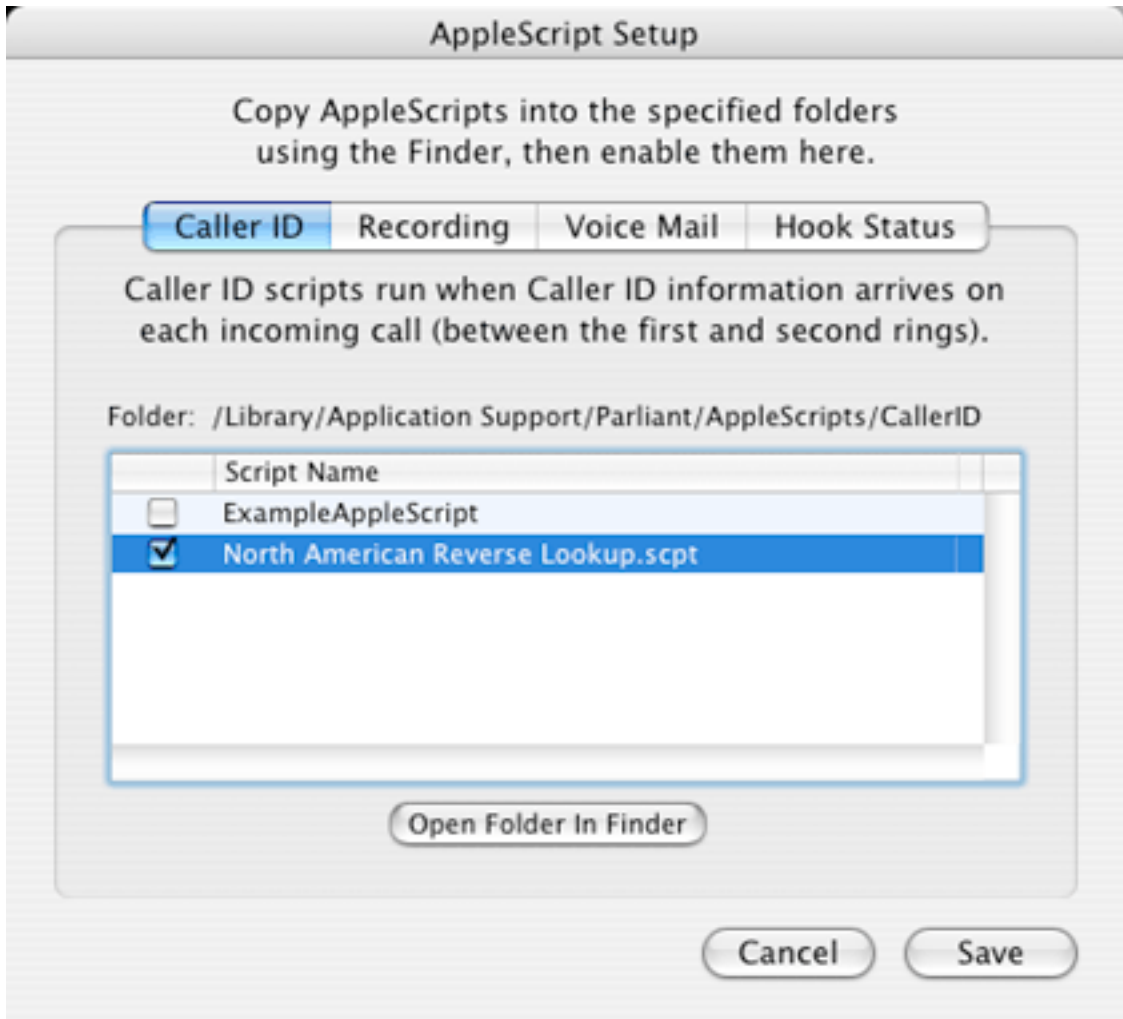


The Voice Mail Preferences dialog lets you change the number of times the phone rings before PhoneValet answers and lets you manage your voice mailboxes, greetings, and phone menu options. Phone menu clips and individual greetings can consist of voice recordings, or can be read out by OS X's Text-To-Speech (TTS) engine.

All of the voice mail options worked as expected, but the automated OS X Speech did not work that well. This was not due to problems with PhoneValet, however: the limited signal range offered by standard phone lines makes the computer-generated voice less coherent. Even so, the use of Apple's TTS achieved the positive effects of scaring off telemarketers and piquing the interest of legitimate callers.



One of PhoneValet's more interesting features is the ability to remotely administer a Mac using a telephone. The Remote Control preferences dialog allows you to trigger AppleScripts using special digit codes. Four example scripts are included which give an indication of this feature's potential: the most useful and impressive of these four is one that reads iCal's daily schedule back to you over the phone.



AppleScripts can also be triggered based on PhoneValet's status. For example, iTunes can be automatically muted during incoming phone calls, and a Caller ID reverse lookup can be accompanied by a Web page, loaded automatically to display additional caller information.

Additional Configuration

According to the documentation, the one computer can manage more than one phone line, simply by adding more hardware phone adapters. I was unable to test this feature because I only had the one phone line and phone adapter for testing purposes.

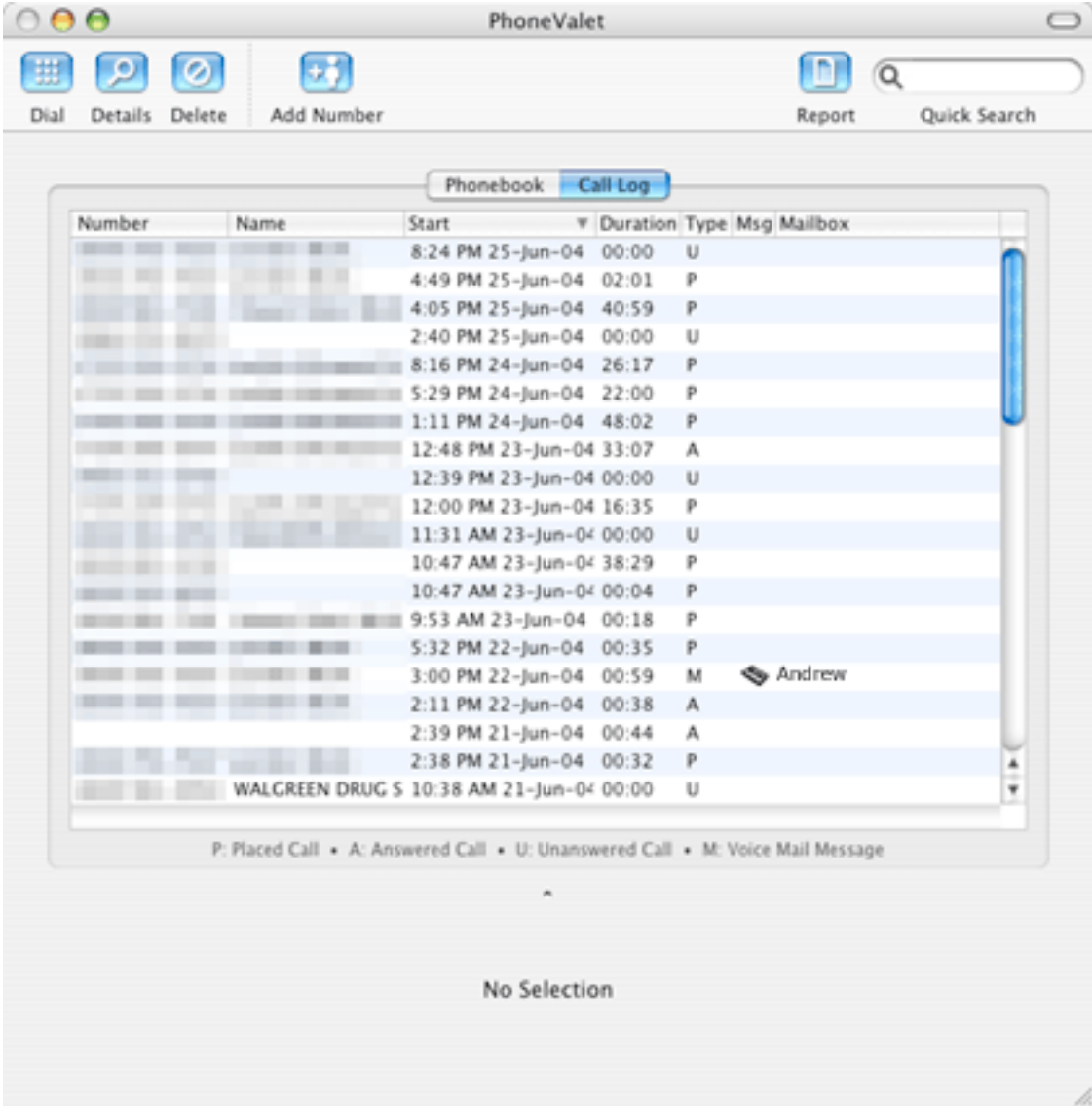
While the only step needed for basic functions is to plug the phone line into the phone adapter, a slightly more complex setup is required to benefit from the adapter's full capabilities. I used a three-way socket splitter (\$5 from most stores) to share the phone line between the phone adapter, a regular telephone, and the Mac's built-in modem (for fax purposes).

Phonebook and Call Log

The Phonebook/Call Log can import names and numbers from Apple's Address Book. You can either import its entire contents or restrict the import to a selection of data. As welcome as this feature is, it is disappointing that the PhoneValet Phonebook does not feature dynamic, two-way interaction with Address Book's database.

Once numbers are added and they appear in the Phonebook, double-clicking on a number places a call, which only requires that a phone on the same line be picked up before the number will be dialed.

The Call Log automatically records all information about the phone line usage. This includes outgoing calls (even ones placed without using the computer), answered calls, unanswered calls, and messages. Messages can be accessed from the Call Log without loading the separate Message Center application.



Even without a specific PhoneValet application running, a dialog box will automatically appear during incoming calls, showing Caller ID information. Unfortunately, this feature only attaches the identity of callers for numbers already listed in the PhoneValet Phonebook. If specified in the preferences, number and identity information will also be verbally announced using TTS.

The aforementioned reverse lookup AppleScript included with the software loads a Web page that provides additional Caller ID information; however, this process proved to be too slow to determine if calls from unknown numbers should be answered. It would be extremely helpful if the software could perform this task by itself, a feature common among other caller ID devices, without resorting to AppleScript.

The only bug experienced with the software occurred when the Phonebook/Call Log application was running, in that the computer would hang while attempting to restart. This did not occur with the other applications in the PhoneValet application suite, nor did it occur when the PhoneValet services menu was present in the menu bar.

Message Center

As with the other features, the Message Center is straightforward and easy to use. Calls are recorded in the standard format specified: 3GPP, AAC, or WAV. The Message Center allows for more detailed management of voice mail messages.

Call Recording

Nowadays almost everyone is recording our everyday telemarketing, sales, and customer service conversations. Unfortunately, the average person doesn't have a similar opportunity to record any misinformation that may be presented by the often

unscrupulous people or organizations at the other end of our phone line. While this may seem paranoid, many of us have experienced situations where we just wish we had a copy of a phone conversation to prove something.



PhoneValet solves this problem simply and easily. Any phone call can be recorded, whether dialed from the computer or manually. The recorded messages are legible and available in the same formats specified in Message Preferences, so conversations can be archived or even e-mailed if necessary.



PhoneValet is an extremely reliable and easy-to-use product that works exactly as advertised. The interface makes the manual almost unnecessary, something Mac users have come to expect from the better products available. Thankfully, the software imports contacts and info from Address Book, and for larger organizations the system is scalable for multiple phone lines with the purchase of additional phone adapters.

It would be highly useful if the PhoneValet Phonebook were dynamically integrated with Apple's Address Book, and a plus if there were direct display of Caller ID reverse lookup from within the PhoneValet software. These criticisms are however minor, and overall the PhoneValet system is a great, high-quality solution.

Copyright © 2004 Andrew Kator, akator@atpm.com. Reviewing in ATPM is open to anyone. If you're interested, write to us at reviews@atpm.com.



Review: SymmetryWorks and Tessella

by Ted Goranson, tgoranson@atpm.com

SymmetryWorks 3.1

Developer: [Artlandia, Inc.](#)

Price: \$215 (download); \$241 (3 CD-ROMs and extras); \$108 (student)

Requirements: Adobe Illustrator 8, Mac OS 8 through 10.3, excluding 10.0.

Trial: None (30 day money-back guarantee on purchase)



Tessella 1.21

Developer: [Stephen Vincent](#)

Price: \$40

Requirements: Adobe Illustrator 10 (older versions still available).

Trial: Fully-featured (30 days)



Some regular ATPM readers will know that I usually write the [outlining column](#) for ATPM. I've had many requests for a look at "MindMapping" outliners, those that switch back and forth between a conventional outline and a "whiteboard" diagram of the same set of concepts and relationships.

This mindmapping column will be no trivial matter, because it touches on the relationship between graphic layout and how concepts and concept groupings are perceived. My goal in the past has been to provide as much background as possible to help you make your own decisions about the tools you use. In this case, we move into complex territory.

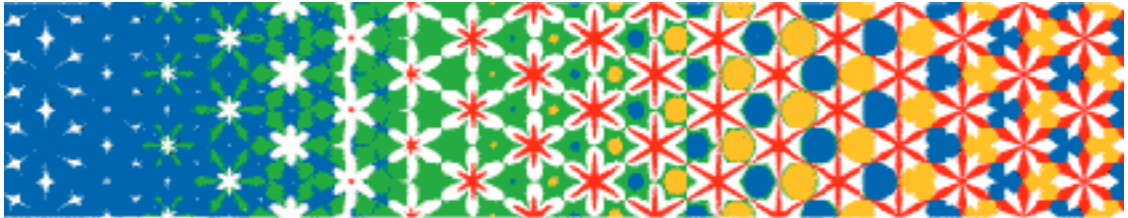
So this month, I take a little detour. That's because ordinary outliners are a relatively simple beast; they are basically linear text flows that have some graphic properties: indentations, collapsing, and possibly styles. The idea is that you can "see" the structure and deal with whatever you are creating on two levels—local and global.

When you move to the other side of the line, when the graphical layout dominates, the theories of perception get quite a bit more numerous and controversial. I have done lots of work in this area for the military and have some personal notions about graphic structure, which will color the mindmapping column to come.

So let's dive in. If you work with graphics, chances are you work with [Adobe Illustrator](#). If you are in the business of making charts (for instance with [OmniGraffle](#)) or diagrams for [Keynote](#) or even visual models for yourself using a mindmapper, you need to consider these two symmetry plug-ins.

Illustrator is one of a few "platform" applications that relies on third-party developers to supply extensions as plug-ins to address niche uses and vertical markets. (There are many sites that provide listings and reviews of Illustrator plug-ins. Search on "Illustrator Plug-ins.")

In this case, we will be looking at plug-ins designed to make patterns. You can use them for that of course, including making patterns for elements within drawings like textile patterns on curtains. We'll review them for their intended purpose, but we'll also note how useful they can be to generate grids and layouts for concept maps and compositions for art.



The Tessella Home Page Header

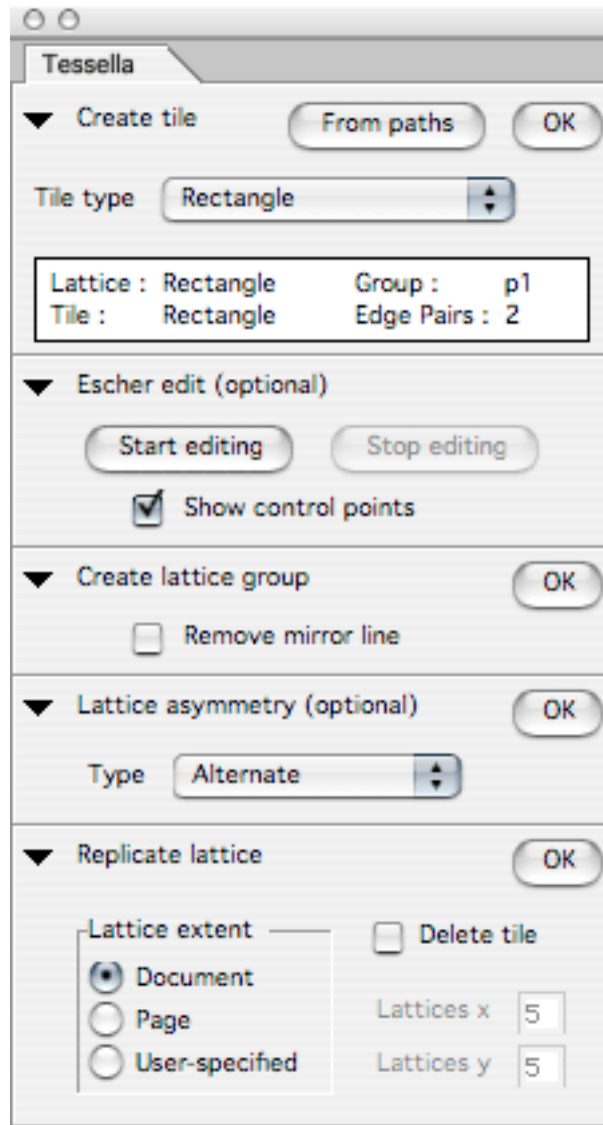
Tessella is the simpler and less expensive. Installation is trivial (as it is for nearly all such plug-ins); you simply drag a file to the "plug-in" folder. Tessella's approach to interface design and documentation is very task-oriented and avoids all the mathematics and notation that accompanies "tiling" software.

And that's basically what we are talking about: you draw a shape, and the plug-in makes copies that infinitely fill the canvas according to arrangement rules based on the mathematical laws of symmetry. With only a little practice, you can make "tiles" like you've no doubt seen by Escher. This shot is also from Tessella's site.



An Escher Example

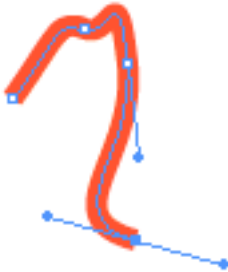
Tessella follows a simple five-step process where two of the steps are optional. Here is its palette, with the five sections expanded. (Plug-in palettes appear as ordinary Illustrator palettes, inheriting all the nice Adobe palette management tricks.)



The Tessella Palette

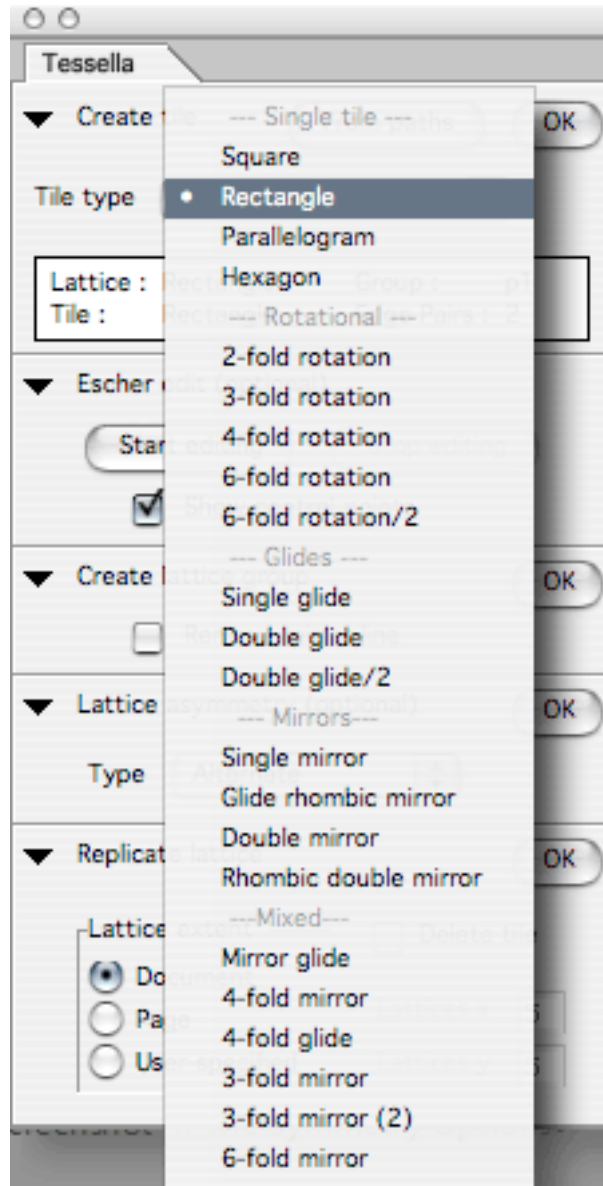
The controls are arranged so that you step through them from top to bottom. Tessella has an informative on-line [quickstart tutorial](#), which is almost sufficient for a review by itself. The plug-in comes with a 24-page PDF that explains all the options in detail.

You draw or select something to be the raw material for your "tile..."



Simple Illustrator Path

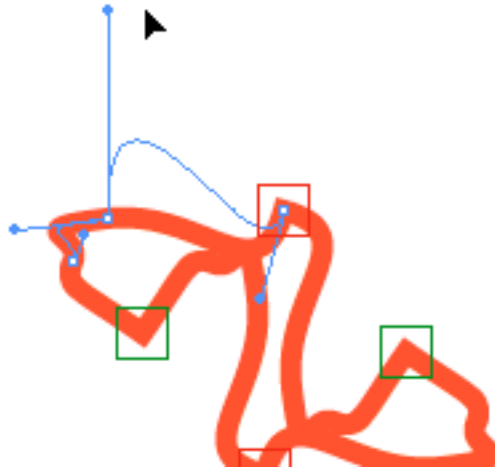
then take the first step by choosing the way it is replicated. In order to know which tilings have what effect, you have to play with it a while. This is the only hard part.



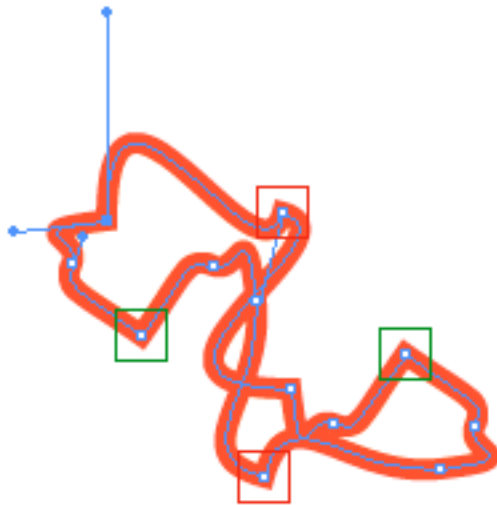
The Symmetry Options

Then you can enter Escher mode if you wish and edit the tile. Not all of Illustrator's editing commands are available, only those that work on the basic shape. If you change the shape to "stick out" in one side, it will mean a "stick in" on another so that all your tiles will fit together. This is really very cool and addictive.

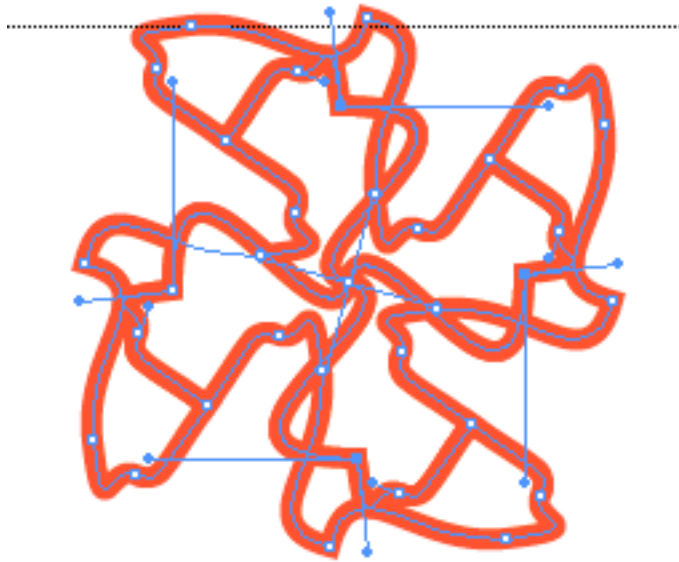
The next step allows you to change colors of alternating tiles if the tiling has them. And then you can make your tiling.



A Tile Element, Edited Using the Escher Edit



Now See How the Shape Has Changed Elsewhere



The Four-Tile Basic Unit



The Whole Tiling

It is easy to quickly become very clever with this. After you are finished with the tool, you can do anything in Illustrator or Photoshop that you might do with any image. The demo time period is more than enough to have fun. Since the demo is free, I highly recommend downloading it and noodling about a bit. You may want to look at [Tessella's gallery](#) to get an idea of how far you can go with this relatively limited approach.

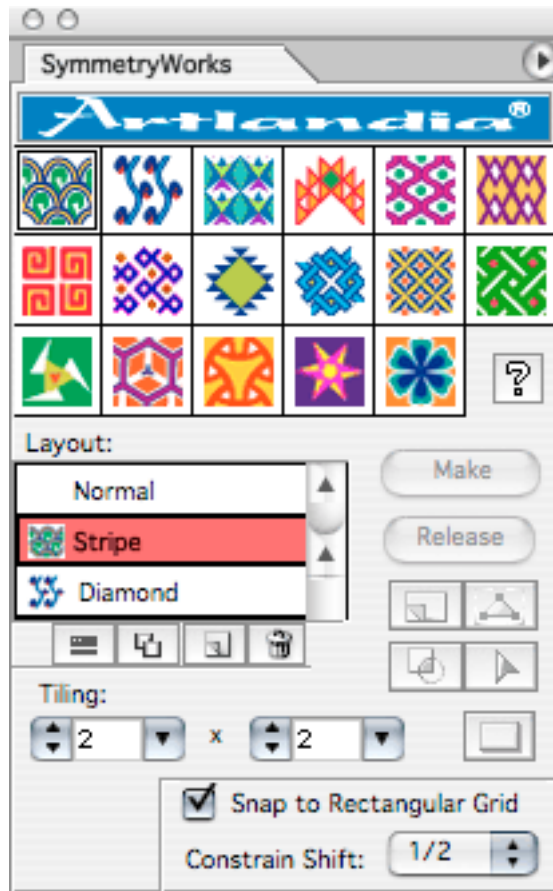
SymmetryWorks is a much more powerful (and expensive) tool.

It does not have a demo. Actually, it is set up so it can be used in demo mode with some restrictions, but there seems to be no way to download the package without paying. It does, however, have a great [online overview](#). With Tessella, you edit the tile and make the tiling. With SymmetryWorks, you can edit the tile after you have made the tiling, which is much handier.

Also, you have more of the Illustrator drawing tools at your disposal. In fact, SymmetryWorks can handle raster objects, type, 3D effects, gradients, transparency, filters, and masks. And you can “blend” tiles as well. This makes it both much easier for a beginner and more unlimited for an expert.

It comes with several demos, a quick reference PDF, and an 87-page PDF manual that is among the best (and most attractive) user manuals I have seen. They really should make that a free download as it convinces well.

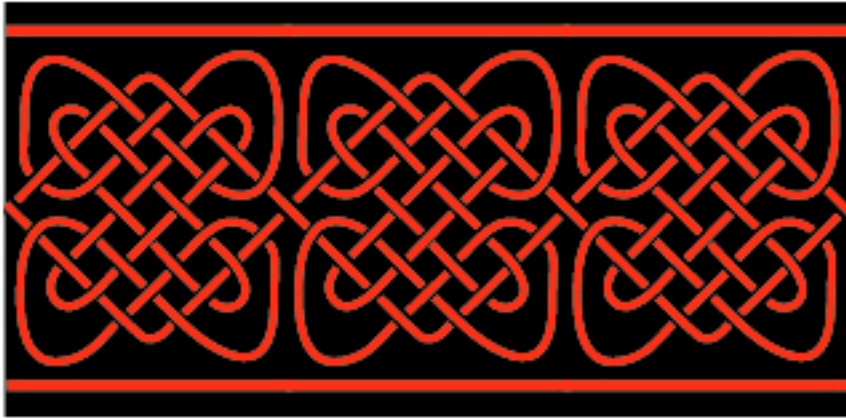
SymmetryWorks has its own imposing palette shown below.



The SymmetryWorks Palette

Those 17 colorful choices at the top are the same as those in Tessella's popup in its first step. Mathematically, there are only 17 distinct symmetry choices in two dimensional "paper." I won't go through all the controls on the palette. Many are for mundane tasks like make a bounding box—something you will want to do to change the SymmetryWorks pattern to an Illustrator pattern.

The design of this palette is in my opinion far less elegant than the design of the program overall, but no matter, because you actually use it very little. All the real work is done in Illustrator's tools excepting setting the symmetries—which you can toggle from one to another for a hair-raising adventure. You also have the ability to select a “ribbon” layout that allows you do patterns like the Celtic knot shown in the screenshot.



A Celtic Knot Example

SymmetryWorks is *the* tool to have if you take this stuff seriously. Artlandia the company also sells a version (confusingly but perhaps appropriately called “Artlandia”) that integrates with [Mathematica](#), a mathematically-based programming environment. In that case, you can program the effects instead of drawing freehand. This is *extreme* symmetry drawing power.

Now why would an outlining geek be interested in drawing symmetry patterns? Sure, these are nice if you want a decorative pattern, or if you are drawing something (like a tablecloth) that itself has a pattern.

But there is a deeper reason, and that brings us to the mindmapping column of next month.

The brain is hardwired to perceive things spatially. For many people, that can be tapped to comprehend and even reason spatially, according to visual relationships. This much is pretty universally agreed, and is why visual brainstorming is so common.

But experts disagree about just what spatial arrangements are most natural for certain purposes. I imagine we'll get into some of this next month when we review tools. A very strong community believes that symmetrical arrangements, even complex symmetrical arrangements, can be useful—sometimes extraordinarily useful—in this context.

In other words, it might be useful if you are an artist to consider certain symmetry-related principles in how the elements of your drawing are laid out and framed. Or more to the point, if you are charting concepts on a page, you may want to consider grids other than squares. If you are serious about communicating to yourself and others, you may even want to have a vocabulary of such grids.

I recommend both products as worth exploring. Tessella is Very Nice because it is an good bargain: all the basic symmetry editing tools for a low price and it has a generous demonstration. It just lacks access to all rich tools of Illustrator.

SymmetryWorks is just plain awesome in what it can do, and how it leverages Illustrator. I rate it Excellent. My only complaint is that the palette is inelegant.

Copyright © 2004 Ted Goranson, tgoranson@atpm.com. Ted writes a regular [column on outliners](#) for ATPM. He is the North American treasurer for the [International Society for the Interdisciplinary Study of Symmetry](#), which has an upcoming gathering in Budapest, October 22–29. These [Interdisciplinary Congresses](#), held every 2–3 years, are truly amazing, worthy of creative Mac users. [Reviewing in ATPM](#) is open to anyone. If you're interested, write to us at reviews@atpm.com.



Review: Wicked Cool Shell Scripts (book)

by Eric Blair, eblair@atpm.com

Author: [Dave Taylor](#)

Publisher: [No Starch Press](#)

Price: \$30

Trial: [Sample Shell Scripts \(1, 2, 3, 4\)](#)



It's been several years since the initial release of Mac OS X formally introduced Mac users to the world of Unix. Since then I've stumbled across a handful of Unix books and articles that claim to cover the Mac, and it often seems like the extent of Mac coverage can be summed up by the phrase, "By this way, this should work on OS X."

That said, the subtitle for *Wicked Cool Shell Scripts*—"101 Scripts for Linux, Mac OS X, and Unix Systems"—made me slightly wary. I know you're not supposed to judge a book by its cover (which, by the way, has a stylish retro thing going on), but a statement like that has a lot to live up to in my mind.

In spite of this, the book left me pleasantly surprised. With a few exceptions, the vast majority of the scripts in this book work under Mac OS X: those that don't are noted and, in many cases, alternatives are provided in a Mac-specific chapter. Chalk that up as one promise kept.

One of the strengths of this book is its fairly consistent layout. Most of the 101 shell scripts come with detailed descriptions of how the interesting code portions work, sample input and output to verify that the scripts work, and ideas for further modifying the scripts to increase their functionality. Scripts that lack one or more of these sections are usually the extremely simple ones, which don't require any explanation beyond the code.

Starting Slowly

When I started reading *Wicked Cool Shell Scripts*, I wasn't filled with such warmth. While the book does jump directly into writing scripts, as opposed to first spending time teaching the scripting language, the first two chapters are rather bland. Validating input types? Finding tools accessible through the `PATH` variable? Converting DOS commands to their Unix counterparts? Ugh, wake me up when it's over.

Truthfully, it took me longer to read the first two chapters than it took to read the remaining ten. That's not to say these first two chapters aren't worthwhile; if you haven't done any shell scripting before, you have a chance to ease into the process with some shorter scripts.

The scripts in these chapters also act as building blocks for the rest of the book. In some cases, ideas from the early chapters will show up in more advanced scripts. In other cases, the more advanced scripts will simply run one of the earlier scripts.

If you can avoid jumping ahead, you'll also be rewarded with a few diamonds in the rough from these introductory chapters. My personal favorites are the scripts that archive files when you remove them. Essentially, these scripts give you a command-line version of the Trash, as opposed to the `rm` command, which simply eradicates files from your hard drive.

Picking Up Steam

My impression of this book changed for the better once I got into chapter three and started working through the final ten chapters. The scripts in these chapters were often interesting or useful (or in some cases, both).

Chapters three and four both deal with creating and modifying Unix tools for various purposes, either making a tool more user-friendly or using a specialized tool to create a more generalized tool. A good example of this is the script that builds a spell checker, which uses both the `ispell` tool and your own customizable user dictionary.

The concept of using specialized tools inside of shell scripts is actually pretty important, and these chapters do a decent job of driving home this point.

System Administration

The next two chapters deal with various aspects of system administration. If you're the only person using your computer, a lot of these scripts won't apply to you. Also, the Mac's graphical user interface makes some of these redundant; for instance, who's going to set the system date from the command-line when Apple provides a preference pane that can automatically do this over the Internet?

There are a couple of scripts that anybody who dips into the command-line environment could find useful, though. There's one script for checking entire crontab files for errors, instead of waiting to find out about errors individually, and another script for managing log files before they get too unwieldy.

Of course, if you manage Macs in a multi-user environment, you could learn even more from these sections. I don't run any labs, so I'm not really in a position to judge this.

The system administration chapter is where most of the Mac-incompatible scripts can be found. The blatantly incompatible scripts are those that deal with adding, deleting, and modifying users, as OS X uses a completely different user management system from any other Unix operating system.

You should also be careful if you decide to use either of the backup scripts provided in these chapters. Both scripts use standard Unix file compression tools, which will not preserve any resource forks (or other Mac metadata) that might be present in your files.

Networking and the Internet

The scripts in chapters seven through ten all deal with various networking and Internet-related tasks, starting with accessing the Internet and working up to server administration.

The majority of the scripts for Internet users deal with gathering content from Web sites, for instance movie information from the [IMDb](#), stock information from [Lycos](#), or headlines from [BBC News](#). As these sites change their designs, the scripts will likely break, but the ideas behind the scripts are still valid.

Chapter eight covers scripts, many in the form of CGIs, for webmasters. This section includes examples of common Web elements like counters, photo albums, and guest books. Few of these scripts result in the most sophisticated Web pages, but they do provide a solid basis for future growth. For instance, Dave Taylor's own Web site incorporates a [fully-featured photo album](#) built with shell scripts.

Putting your Web site on the Internet is only the first step; after that, you need to keep it up and running. Fortunately, chapter nine features a number of scripts for maintaining your Web site, from those that check your spelling and links to those that synchronize your local and remote files by FTP or SFTP.

The final Internet-related chapter deals with actual server administration. Topics covered include keeping your server running smoothly and monitoring information about who is viewing your Web site.

The Good Stuff

Here's the part of the book that justifies including "Mac OS X" in the subtitle: a half dozen scripts that won't work on any Unix system other than OS X (although there is one that might work under Windows, with some coaxing). Linux users don't get Linux-only scripts, BSD users don't get BSD-only scripts. Just us OS X users.

The first three of these scripts deal with the NetInfo database that OS X uses to store all of its user data. There's a script for listing all of the user accounts present on the system (including the accounts that don't show up in the login panel), there's a script for adding new users, and there's a script for setting up e-mail aliases if you're using your Mac as a mail server.

Granted, NetInfo Manager provides a graphical interface for doing all of these things, but this might be one of those rare cases where using the command-line is actually more user-friendly than a graphical application. If you haven't had the opportunity to use NetInfo Manager, let's just say it's not the best example of user interface design.

The fourth script is a quickie that dynamically sets the title of your Terminal window. On the one hand, it's a bit of a gimmick. However, if you like having a short command prompt (like me), this lets you display the current working directory in the window's title bar.

The fifth script deals with iTunes. Don't get too excited about this one; it simply reads your iTunes Music Library.xml file and generates human-readable output. The output from the script is less dense, and more legible, than the output from iTunes' "Export Song List..." option.

The final OS X script is a customizable replacement for the `open` tool. This newer version lets you create shortcuts for applications so you can do things like launch QuickTime Player by typing `open2 qt` instead of `open -a QuickTime\ Player`.

There's also a bonus script that shows you how to take a screenshot every minute, using the `screencapture` tool. It doesn't count towards your quota of 101 shell scripts, but you get what you pay for, since the script implies that `screencapture` generates TIFF files instead of PDFs (Apple switched to using the PDF format as of Mac OS X 10.2).

Taking a Breather

If you make it to chapter 12, you're rewarded with games in the form of the final three shell scripts: a Jumble clone, a Hangman clone, and a state capital quiz. No Quake here, but a nice reminder that shell scripts can be used for a number of different things, not just system tools.

Some Thoughts

Wicked Cool Shell Scripts isn't perfect. As a selfish Mac user, I'd certainly like more Mac-specific content, perhaps a backup script that preserves resource forks by creating a compressed disk image instead of using `tar`. A few examples of shell scripts using the `osascript` command, which lets you run AppleScript from the command-line, would have been nice.

I noticed a few typos in some of the code included in the book. There wasn't anything major and some of the problems were fixed on the book's Web site, but it was a little annoying.

The other issue I had with the code was with the choice of font. The big problem was that the font used made it hard to distinguish a zero (0) and o (capital-oh). I'm a firm believer that any code examples should use a font that places a slash through the center of the 0, as it makes the code much more readable.

Minor quibbles aside, I thoroughly enjoyed reading this book. By the time I was finished, I had ideas for expanding some of the scripts in the book, ideas for adapting some of the scripts for other tasks, and ideas for writing completely new scripts. I'd say that any book capable of motivating people three-fold will prove quite successful.

Copyright © 2004 Eric Blair, eblair@atpm.com. Reviewing in ATPM is open to anyone. If you're interested, write to us at reviews@atpm.com.



FAQ: Frequently Asked Questions

What Is ATPM?

About This Particular Macintosh (ATPM) is, among other things, a monthly Internet magazine or “e-zine.” ATPM was created to celebrate the personal computing experience. For us this means the most personal of all personal computers—the Apple Macintosh. About This Particular Macintosh is intended to be about your Macintosh, our Macintoshes, and the creative, personal ideas and experiences of everyone who uses a Mac. We hope that we will continue to be faithful to our mission.

Are You Looking for New Staff Members?

We currently need several **Contributing Editors**. Please [contact us](#) if you’re interested.

How Can I Subscribe to ATPM?

Visit the [subscriptions page](#).

Which Format Is Best for Me?



The **Online Webzine** edition is for people who want to view ATPM in their Web browser, while connected to the Internet. It provides sharp text, lots of navigation options, and live links to ATPM back issues and other Web pages. You can use Sherlock to [search](#) all of the online issues at once.



The **Offline Webzine** is a HTML version of ATPM that is formatted for viewing offline and made available in a StuffIt archive to reduce file size. The graphics, content, and navigation elements are the same as with the Online Webzine, but you can view it without being connected to the Internet. It requires a Web browser.



The **Print PDF** edition is saved in Adobe Acrobat format. It has a two-column layout with smaller text and higher-resolution graphics that are optimized for printing. It may be viewed online in a browser, or downloaded and viewed in Adobe’s free Acrobat Reader on Macintosh or Windows. PDFs may be magnified to any size and searched with ease.



The **Screen PDF** edition is also saved in Adobe Acrobat format. It's a one-column layout with larger text that's optimized for reading on-screen. It may be viewed online in a browser, or downloaded and viewed in Adobe's free Acrobat Reader on Macintosh or Windows. PDFs may be magnified to any size and searched with ease.

What Are Some Tips for Viewing PDFs?

- You can [download](#) Adobe Acrobat Reader for free. If you have a Power Macintosh, Acrobat Reader 5 has better quality and performance. ATPM is also compatible with Acrobat Reader 3, for those with 680x0 Macs.
- You can zoom the PDF to full window width and scroll through articles simply by single-clicking anywhere in the article text (except underlined links).
- You can quickly navigate between articles using the bookmarks pane at the left of the main viewing window.
- For best results on small screens, be sure to hide the bookmarks pane; that way you'll be able to see the entire page width at 100%.
- Try turning Font Smoothing on and off in Acrobat Reader's preferences to see which setting you prefer.
- All blue-underlined links are clickable.
- You can hold down option while hovering over a link to see where it will lead.
- For best results, turn off Acrobat's "Fit to Page" option before printing.

Why Are Some Links Double-Underlined?

In the PDF editions of ATPM, links that are double-underlined lead to other pages in the same PDF. Links that are single-underlined will open in your Web browser.

What If I Get Errors Decoding ATPM?

ATPM and MacFixIt readers have reported problems decoding MacBinary files using early versions of StuffIt Expander 5.x. If you encounter problems decoding ATPM, we recommend upgrading to [StuffIt Expander 5.1.4 or later](#).

How Can I Submit Cover Art?

We enjoy the opportunity to display new, original cover art every month. We're also very proud of the people who have come forward to offer us cover art for each issue. If you're a Macintosh artist and interested in preparing a cover for ATPM, please e-mail us. The way

the process works is pretty simple. As soon as we have a topic or theme for the upcoming issue we let you know about it. Then, it's up to you. We do not pay for cover art but we are an international publication with a broad readership and we give appropriate credit alongside your work. There's space for an e-mail address and a Web page URL, too. Write to editor@atpm.com for more information.

How Can I Send a Letter to the Editor?

Got a comment about an article that you read in ATPM? Is there something you'd like us to write about in a future issue? We'd love to hear from you. Send your e-mail to editor@atpm.com. We often publish the e-mail that comes our way.

Do You Answer Technical Support Questions?

Of course. E-mail our Help Department at help@atpm.com.

How Can I Contribute to ATPM?

There are several sections of ATPM to which readers frequently contribute:

Segments: Slices from the Macintosh Life

This is one of our most successful spaces and one of our favorite places. We think of it as kind of the ATPM "guest room." This is where we will publish that sentimental Macintosh story that you promised yourself you would one day write. It's that special place in ATPM that's specifically designated for your stories. We'd really like to hear from you. Several Segments contributors have gone on to become ATPM columnists. Send your stuff to editor@atpm.com.

Hardware and Software Reviews

ATPM publishes hardware and software reviews. However, we do things in a rather unique way. Techno-jargon can be useful to engineers but is not always a help to most Mac users. We like reviews that inform our readers about how a particular piece of hardware or software will help their Macintosh lives. We want them to know what works, how it may help them in their work, and how enthusiastic they are about recommending it to others. If you have a new piece of hardware or software that you'd like to review, contact our reviews editor at reviews@atpm.com for more information.

Shareware Reviews

Most of us have been there; we find that special piece of shareware that significantly improves the quality our Macintosh life and we wonder why the entire world hasn't heard about it. Now here's the chance to tell them! Simply let us know by writing up a short review for our shareware section. Send your reviews to reviews@atpm.com.

Which Products Have You Reviewed?

Check our [reviews index](#) for the complete list.

What is Your Rating Scale?

ATPM uses the following ratings (in order from best to worst): Excellent, Very Nice, Good, Okay, Rotten.

Will You Review My Product?

If you or your company has a product that you'd like to see reviewed, send a copy our way. We're always looking for interesting pieces of software to try out. Contact reviews@atpm.com for shipping information. You can send press releases to news@atpm.com.

Can I Sponsor ATPM?

About This Particular Macintosh is free, and we intend to keep it this way. Our editors and staff are volunteers with "real" jobs who believe in the Macintosh way of computing. We don't make a profit, nor do we plan to. As such, we rely on advertisers to help us pay for our Web site and other expenses. Please consider supporting ATPM by advertising in our issues and on our web site. Contact advertise@atpm.com for more information.

Where Can I Find Back Issues of ATPM?

[Back issues](#) of ATPM, dating since April 1995, are available in DOCMaker stand-alone format. In addition, all issues since ATPM 2.05 (May 1996) are available in HTML format. You can [search](#) all of our back issues.

What If My Question Isn't Answered Above?

We hope by now that you've found what you're looking for (We can't imagine there's something else about ATPM that you'd like to know.). But just in case you've read this far (We appreciate your tenacity.) and still haven't found that little piece of information about ATPM that you came here to find, please feel free to e-mail us at (You guessed it.) editor@atpm.com.

