Jun 2010
Vol. XXI, No 6

KEYSTONE MacCentral

printout

Keystone MacCentral Macintosh Users Group ❖ http://www.keystonemac.com

Be back in September

Meet us at

Giant Food

Corner of Trindle Road & 32nd St (Route 15)
3301 East Trindle Road, Camp Hill, PA 17011

Tuesday, September 21, 2010, 6:30 p.m.

Attendance is free and open to all interested persons.

Contents

Keystone MacCentral is a not-for-profit group of Macintosh enthusiasts who generally meet the third Tuesday of every month to exchange information, participate in question-and-answer sessions, view product demonstrations, and obtain resource materials that will help them get the most out of their computer systems. Meetings are free and open to the public. The *Keystone MacCentral Printout* is the official newsletter of Keystone MacCentral and an independent publication not affiliated or otherwise associated with or sponsored or sanctioned by any for-profit organization, including Apple Computer, Inc. Copyright © 2010, Keystone MacCentral, 305 Somerset Drive, Shiresmanstown, PA 17011.

Nonprofit user groups may reproduce articles form the Printout only if the copyright notice is included, the articles have not been edited, are clearly attributed to the original author and to the Keystone MacCentral Printout, and a copy of the publication is mailed to the editor of this newsletter.

The opinions, statements, positions, and views stated herein are those of the author(s) or publisher and are not intended to be the opinions, statements, positions, or views of Apple Computer, Inc.

Throughout this publication, trademarked names are used. Rather than include a trademark symbol in every occurrence of a trademarked name, we are using the trademarked names only for editorial purposes and to the benefit of the trademark owner with no intent of trademark infringement.

Board of Directors

President
Linda J Cober

Vice President
Tom Owad

Recorder
Gary Brandt

Treasurer
Ron Rankin

Program Director
Gary Brandt

Membership Chair
Eric Adams

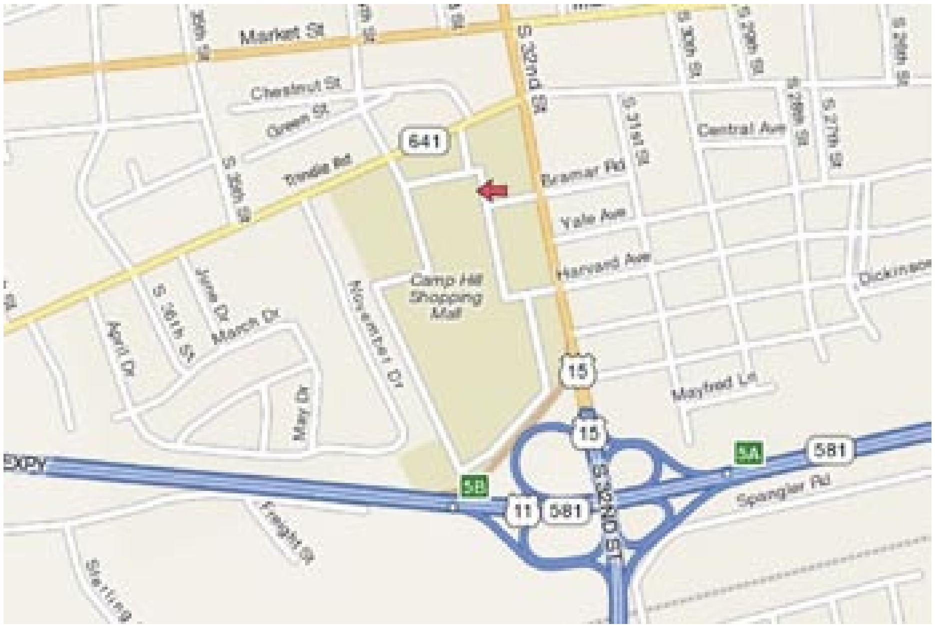
Correspondence Secretary
Sandra Cober

Newsletter Editor
Tim Sullivan

Industry Liaison
Wendy Adams

Web Master
Tom Bank II

Librarian
Tim Sullivan



Keystone MacCentral Essentials

Meeting Place
Giant Food (upstairs)
Corner of Trindle Road & 32nd St
(Route 15)
Camp Hill

Web Site
http://www.keystonemac.com

Mailing Address
305 Somerset Drive
Shiresmanstown, PA 17011

by Gary Brandt, Recorder

Keystone MacCentral Minutes

May 18, 2010

Business Meeting

President Linda Cober began the May meeting by asking board members for their reports. Eric Adams reported that most issues relating to software downloads from the auction had been resolved

Q&A & Comments

The first question was about creating stationery with Adobe Photoshop, using logos from the internet. They could be converted into a usable format with Apple's Preview application but they would probably be too low resolution for attractive printing. It would be best to use the original logo file. Someone mentioned an application to convert photos to look like drawings. In fact, we auctioned off a copy of Akvis Sketch during the auction in April. You can check it out with the link on our vendors page. A member reported problems opening Word files received as email attachments. They do not always open and sometimes do not look right. File size limits could be one problem. Word files might also lose their links to embedded photos when sent by email. In response to a question, we searched the internet for a free site to convert PDF files on a PC. Primopdf. com is one such site. Someone needed help bypassing the Empty Trash warning that prevented a file from being emptied from the Trash. A suggestion to unlock the item from the Get Info window was made. You can also Option-Empty Trash to bypass the warning window. Linda Cober mentioned that we had received a copy of the "Teach Yourself iPhone Application Development In 24 Hours" book. She asked if anyone would like to review the book. Eric Adams mentioned one of his favorite features of Snow Leopard. That is Quick Look, which also works in Leopard. Selecting a file and pressing the Spacebar opens a preview of many different file types without actually opening the associated application. This can come in handy when trying to remember what a photo or a text file is and it also works with music and movie files. Linda Cober asked for recommendations for a laptop. A matte screen would be one consideration. Jim Carey suggested getting the most powerful processor and at least 4 GB of memory. A laptop hard drive could be either 5400 or 7200 rpms. Jim buys refurbished Macs from the online Apple Store. Tom Owad checks Apple Insider for deals. You can sometimes also find discounted deals on AppleCare coverage. Those of you interested can find show notes of the program done by Bob Levitus at Hershey Apple Core on HAC's web site.

Election

Linda Cober asked if there were any nominations from the floor for the election of board officers. A slate of candidates had been published in the May newsletter. No additional nominations were made. Bill Galbraith made a motion to approve the published slate of candidates to serve a two year term. Jim Carey seconded and the motion to elect board officers was passed. The following were elected.

President: Linda Cober
Vice President: Tom Owad
Recorder: Gary Brandt
Treasurer: Ron Rankin
Program Director: Gary Brandt
Membership Chair: Eric Adams
Correspondence Secretary: Sandra Cober
Newsletter Editor: Tim Sullivan
Industry Liaison: Wendy Adams
Web Master: Tom Bank II
Librarian: Tim Sullivan

Program Notes

We had talked briefly about Pandora during Q & A so Eric Adams navigated to their web site to give us a look at their service. Eric and Jim Carey were Pandora users prior to the meeting so they were already familiar with the site. Registered users can create their own radio stations based on an artist or a song they like. We logged into Jim's account to view his existing stations and demonstrate how to add new ones. Pandora's algorithms create a play queue on the fly based on your initial selection. This is enhanced when you rate the songs being played. You can choose to add variety to a station. The Pandora service is available on the web and also for smart phones. We next looked at the Apple Store and its selections for music, movies, TV shows, podcasts, audiobooks, and the App Store section. Eric recommended the Photoshop User TV podcast. Jim noted some

Continued on page 4

Continued from page 3

Keystone MacCentral Minutes

of his favorites, including podcasts done by Terry White, an Adobe employee who leads a Detroit users group that tapes its meetings and then uploads the videos to iTunes.

Jim mentioned some of his favorite iPhone applications. The TomTom GPS app costs about $40 and Jim said it works better than a Tom Tom unit for him. Dragon Dictation is a free app to convert speech to text. Red Laser is an application to scan bar codes and check for best prices. Yelp is useful for restaurant recommendations. Pinball Wizard is a fun diversion. Eric described how he and his family use Skype to make free calls overseas. Calls are free between two registered Skype users. You can also use Skype to call a phone number for reasonable rates.

by Tim Sullivan

Rumors and Reality

Predicting disk failure: On Google there is a rather long and formal discussion of disk failure. There are few things more disheartening that having 500Gb of good stuff (family photos, recipes, financial stuff) go Poof and disappear. Even if most of it was backed up elsewhere, restoring is a really, really big pain.

This dissertation was the result of a study of a large number of hard disks. The findings are worth taking into consideration.

First, the study found high temperature did not contribute to failure. This is true even of older drives.

On the other hand, there are some drive models, manufacturers and vintages that you should avooid. Use good stuff.

Second, Self-Monitoring, Analysis, and Reporting Technology, or S.M.A.R.T. (sometimes written as SMART), is a monitoring system for computer hard disks to detect and report on various indicators of reliability, in the hope of anticipating failures. You can check the status of your computer's hard disk by opening "Disk Utility," which can be found by going to Application > Utilities. When you click on the disk in the left panel, all will be revealed at the bottom: "S.M.A.R.T. Status: Verified." Or "Not Supported" in the case of external hard drives. Wikipedia has page showing the information being monitored. I sometimes wonder how anything gets done with all the background processes going on. In any event, the software keeps track of a bunch of things, which get condensed into "Verified" or whatever comes up when your disk is about to crash and burn. SMARTReporter is also available. It places a small colored icon on the menu bar. Green is good; red is bad.

The study found that, in general, SMART reports were not of great value in predicting impeding failure except when scan errors are detected. Normally, the software abandons the sector that got the error and uses a spare sector. One would think that would be the end of the problem. But once scan errors happen, the chance the drive will fail goes up dramatically. Unfortunately, neither Disk Utility nor SMARTReporter specify the problem. For the average home user there does not seem be anything that specifies which parameters have been exceeded.

Curiouser and curiouser: Apparently some iPhones are transferring large amounts (20Mb) of date to somewhere on their own on a regular basis. Not much has been discovered about what is being transferred, nor where it's going. It happens late a night -- around 2 a.m. ATT&T considers the transfer billable.

There is speculation that Apple might be responsible. One user reported that he had opened a case with AT&T and talked to a manager at tech support there. He said that when you register your phone there is something that you check where you allow Apple access to it, and no way later to turn this off. Apparently these data transfers could be going to Apple!

by Adam C. Engst

Beware Tabnabbing, a New Type of Phishing Attack

I can never decide whether I'm happy when a good guy discovers and publicizes a new way of potentially exploiting Internet users. After all, it's better that we learn about the problem before it appears in the wild, but there's always a worry that the bad guys wouldn't have figured it out on their own without the hint. The latest trick, dubbed "tabnabbing," comes from Aza Raskin, Creative Lead for Firefox (and son of Jef Raskin).

Here's how it works, and you can watch it happen yourself by loading the proof-of-concept (which is also the page where Raskin explains the exploit). Although Aza Raskin tested primarily with Firefox, I was able to verify that the exploit also works in the Mac versions of Safari, Camino, Opera, and OmniWeb, though not quite in the same way in each. The current version of Google Chrome (5.0.375.55) appears to be immune to the problem, though it's possible that Google fixed it quickly, since others have previously reported Chrome as vulnerable.

Imagine you're browsing the Web and you end up at a particular page, call it SneakyPage. It doesn't look evil, and it may in fact be a totally legitimate site that has been compromised by a bad guy. But it contains a tiny bit of malicious JavaScript that loads with the page, and that JavaScript does nothing unless you switch to another tab, leaving the tab holding SneakyPage open.

At that point, the malicious JavaScript springs into action, replacing the SneakyPage tab's favicon, title, and page content. Remember, you're off in another tab, or even in another program, so you're not paying attention at this point.

SneakyPage could pretend to be Gmail or Hotmail or Citibank or any other commonly used site. The specifics don't matter; all it has to do is make you believe that the tab contains a legitimate login form for a service you use.

At some point later, you come back to the tab, see the login form, and decide that yes, you do want to log back in to check your e-mail or your account balance. Once you do so, SneakyPage's JavaScript snags your login credentials for future nefarious purposes and redirects you to the actual site, so you're none the wiser that you've just fallen victim to a phishing attack.

"But," I can hear you saying, "how would the malicious script be able to guess that I use Gmail or Citibank or whatever?" The problem is that it's possible to figure out if a user has visited specific sites, thanks to the way most sites identify visited links by changing their colors via CSS. So the malicious JavaScript we're postulating could determine if you use any of a set of particular Web sites, and then fake an appropriate one. LWN.net has an article describing this browser history leak in more detail, and if you don't believe it, visit StartPanic.com for a personalized demonstration.

The elephant gun solution is to turn off JavaScript entirely, or, for Firefox users, run the NoScript extension, which enables you to block JavaScript on all sites but those you allow (Google Chrome has this capability too). Unfortunately, turning off JavaScript entirely renders the modern Web nearly unusable. And NoScript is an option only for Firefox users, and even then, many people find it — or Google Chrome's similar feature — too intrusive for everyday use.

Worse, security researcher Aviv Raff has figured out a way to simulate the exploit without using JavaScript. Brian Krebs links to Raff's proof-of-concept from his Krebs on Security blog post; it's best to start there since the proof-of-concept morphs a mockup of Krebs's post into a Gmail login screen. The NoScript extension may protect against Raff's approach as well, but regardless, the type of users who would be fooled by tabnabbing aren't as likely to be the sort of people who would be running NoScript.

So how much of a worry is tabnabbing, and what can you do? My gut feeling is that if you stick to mainstream legitimate Web sites, you have little to worry about. However, that doesn't mean that avoiding sleazy destinations like file download sites is a guarantee of safety. In September 2009, the New York Times Web site served a rogue advertisement that purported to scan for viruses. If a criminal organization was somehow able to sneak a tabnabbing JavaScript into an ad and place it on legitimate sites via an ad network, it could wreak havoc.

If there's no guarantee of safety — at least until browser makers figure out a solution — how can you protect yourself? I see a few realistic options that don't require extra effort and could even make your life easier:

- If you ever switch to a tab and it's displaying a login screen, be very wary. No, scratch that. Just close the tab — it's not worth thinking about whether it might be an attack.

- Rely on auto-fill, either via the browser's own auto-fill feature or a program like 1Password, to enter login

Continued on page 6

Continued from page 5

Beware Tabnabbing, a New Type of Phishing Attack

credentials, and if the auto-fill doesn't work (as it wouldn't in the case of a faked login page because the domain wouldn't match), close the tab, access the site again from a bookmark or manually typed URL, and try again.

• Create bookmarks for sites that require logins, and always use your bookmark to visit those sites. Even if you see a login form just waiting for you in a tab, load your bookmark instead.

• Better yet, make site-specific browsers for sites that require logins to protect sensitive data, and use those sites only via their site-specific browsers. A site-specific browser enables you to turn any Web app into a standalone Mac application with its own windows and menus and Dock icon. For instance, I have a site-specific browser for Google Docs, and another for the Manymoon project collaboration site. The main site-specific browsers I know of are Fluid, which relies on Apple's WebKit and thus works like Safari, and Mozilla's Prism, which works like Firefox; both are free. As an added bonus, using site-specific browsers reduces the confusion that can occur when you have too many tabs open; it also lets you think of and interact with a Web app like any other desktop application.

• Use a dedicated client for login-based sites where possible. This is merely an extension of the site-specific browser suggestion, but there are dedicated applications for certain Web sites, like Mailplane for Gmail and Waveboard for Google Wave. If you like the idea of breaking Web apps out into Mac applications, why not get extra features from a dedicated client?

Meanwhile, back at the conundrum I posed at the beginning of this article, what is a good guy who discovers such a trick to do? This isn't the same as finding a browser bug that enables a security exploit, since in that case it makes sense to report the bug privately so the browser maker can fix the bug before the bad guys exploit it. Browser makers don't always do this quickly enough, but that's the theory.

In this situation, though, the browsers are acting largely as they're supposed to, which is why tabnabbing works across multiple browsers. Similarly, the CSS browser history leak isn't new, and it too works across multiple browsers. So I suppose that full public disclosure, as a way of encouraging multiple browser makers to agree on ways of blocking these vulnerabilities, does make the most sense, especially in situations like this, where user education is the best defense. Consider yourself educated, and do what you can to encourage Apple and Mozilla and the others to prevent tabnabbing.

Still, it does make one long for the early days of the Internet when it wasn't necessary to worry about such things.

by Glenn Fleishman

Jobs Explains Apple's Position on Adobe Flash

The notoriously private Steve Jobs, who keeps Apple within the same public comment straitjacket, has found his voice lately. In a break from answering e-mail messages sent to sjobs@apple.com with one-word responses (generally, "No"), the Apple CEO posted a candid assessment of why Adobe's Flash technology has no place in the iPhone OS.

Jobs's "Thoughts on Flash" reiterates points made by many of us who have a long experience with Adobe, Apple, and the capabilities of mobile devices. Notably, John Gruber has written extensively about this issue, including an essay on a change in the iPhone Developer Program agreement that Jobs told one e-mail correspondent was "very insightful and not negative."

There are two areas that Jobs splits a bit across his six points that I want to discuss: open standards and Flash's suitability for the mobile world.

Open Standards — Jobs writes that Adobe is pushing a proprietary technology, Flash, while Apple is backing the open standards of HTML5, CSS, and JavaScript. Apple also developed and put considerable effort into the open-source WebKit browser engine, which Google, Nokia, and Palm use as the basis of browsers on their smartphones. Research In Motion will be moving to WebKit for the BlackBerry, too.

This bleeds a bit into Jobs's second point, in which he rebuts Adobe's claim that by not having Flash support, the iPhone OS lacks "the full Web."

In an ideal world, content created targeting the three standards that Jobs cites would display similarly on any device regardless of browser. HTML5 includes descriptions — but not prescriptions — for graphical animation, video playback, audio playback, and quite a bit else, all of which encroach on some of the main uses of Flash.

Adobe's CEO, Shantanu Narayen, told the Wall Street Journal a few hours after Jobs's essay hit that "Flash is an open specification," chuckling while talking to the reporter. It is true that Adobe has published the details of Flash — as the SWF File Format Specification — but that's more than a little disingenuous.

Unlike any version of HTML, CSS, or JavaScript, SWF was a closed spec until 1 May 2008. Also unlike those other standards, there was not and remains no explosion of Flash-oriented applications across many platforms.

True, Adobe's Flash player is available for many systems, and some non-Adobe software can create SWF files. Adobe has released some SWF code under an open-source license. And there's even a mature effort, Gnash, to bring Flash playback to the GNU/Linux and BSD world.

There are issues that remain about how a third-party Flash player could process DRM that's embedded in content — like the BBC Player. But my reading of the open-source Flash code documentation is that the 10.1 release of Flash will allow third parties to work with protected content.

What the Wall Street Journal reporter should have asked Narayen was, "Tell me about all the applications and players not from Adobe that handle Flash?" or "How are other individuals and companies actively involved in Flash format development?"

That's because none of the movement in the Flash world holds a candle to the variety of Web browsers available. Nor is Adobe's control of the Flash spec anything but a sharp contrast with the ability to influence browser capabilities that the W3C (World Wide Web Consortium) or other standards body has.

Flash may be an "open specification," but Adobe remains in complete control of development, saying only that "Adobe seriously considers all feedback to the SWF file format specification." Notably, Adobe has no requirement to incorporate the ideas of others or to reach consensus on how the Flash community would like to see the technology evolve.

In comparison, HTML, CSS, and JavaScript are the result of well over a decade (more than 15 years in HTML's case) of experimentation, pushback, browser differentiation, compromise, debate, and not a little hostility among competing interests.

The reporter should also have asked about Adobe's alleged role in stalling HTML5 development through procedural moves related to the scope of HTML5's work. HTML5 incorporates a video tag that will allow direct embedding of video in a standard way. (The underlying video format, whether H.264, Ogg Theora, or Google's VP8, will likely not be specified in the HTML5 spec, but H.264 will likely be the most widely supported, given Microsoft's commitment to H.264 in Internet Explorer 9.) HTML5's video tag threatens Flash, even though Adobe's tools can output H.264 and other video formats. Adobe denied holding up HTML5 draft progress, but the Web's father, Tim Berners-Lee, stepped in to move things along by declaring that HTML5's scope was just fine, and work should proceed.

Of course, Jobs is also being disingenuous about Apple's commitment to openness when he focuses on Web-based standards. Outside the browser, the iPhone OS is a completely closed platform, and the changes that John Gruber discussed in the iPhone Developer Program License Agreement require that apps be written only using tools supplied by and controlled by Apple. (According to a New York Post article, that move may prompt an antitrust inquiry.)

The license agreement change prevents Adobe from letting developers compile Flash programs into iPhone OS apps using a tool in Adobe Creative Suite 5 that has already been used for 100 programs in the App Store.

It may also prevent the use of some other development tools for creating iPhone apps; it's not yet known whether Apple will enforce the rule change uniformly. With 200,000 apps in the App Store, the company can afford to lose a few, despite the impact on those developers.

Apple and Adobe are intentionally arguing at cross purposes. Jobs supports open formats for Web-based resources, but a closed development and distribution system for everything outside the browser.

Adobe is arguing for a quasi-proprietary embedded use of Flash on Web pages, which requires that everyone have Adobe's Flash Player installed instead of relying on browser makers to shape a consistent experience independent of any one company.

But in the realm of apps, Adobe is arguing that Flash developers shouldn't be forced to use a separate development process for the iPhone OS, which Apple has now required via the iPhone Developer Program License Agreement.

In terms of tools, Adobe makes the only rich Flash design and development software, while Apple makes the only viable iPhone OS development environment. Developers would likely prefer Adobe's hegemony, despite the limits it imposes, if only because Flash's reach is much broader than that of the iPhone OS.

(Whether Adobe's packaging tool can turn Flash content into an app that provides as good a user experience as a native app is a separate question, and one that Jobs strongly disagrees with in a different section of the essay.)

What the two firms are really arguing about is control. Apple wants near-total control over the iPhone OS platform, but is willing to accept open standards in the realm of the browser, because even the best Web apps don't challenge iPhone OS apps. The best thing a browser can do is let Web

Continued on page 8

Continued from page 7

Jobs Explains Apple's Position on Adobe Flash

pages appear and work the same as every other browser. It took Microsoft a decade to accept that.

On the other side of the coin, Adobe wants control over embedded Web content, seeing itself as a conduit for developers to create a single product that can span dozens of disparate environments.

Each company clearly sees itself as the developers' friend: Apple in encouraging developers to rely on open standards that will render well everywhere, and Adobe in providing developers with tools to create rich content that will render well everywhere that Flash is supported.

Mobile Performance and Suitability — Jobs devotes his other two points to whether Flash even makes sense on a phone as it now stands.

Despite the focus on the iPhone, iPod touch, and iPad, Flash isn't widely available on any other mobile platform. Nokia includes Flash 9.4 on its N900, but no Android, Windows Mobile, BlackBerry, webOS, or Symbian phone handles Flash content. (There is a Flash Light player that works on many basic phones, but which requires Flash content that has been designed and optimized for Flash Light. The Flash Light site appears to be have been last updated in 2008.)

While Adobe has complained about the lack of Flash support from the start on the iPhone, it has only recently demonstrated reasonable mobile performance with a beta of Flash 10.1 on Android.

As Jobs writes, "We have routinely asked Adobe to show us Flash performing well on a mobile device, any mobile device, for a few years now. We have never seen it."

That will soon change. Adobe has commitments for Flash Player 10.1 to work in Android (after the 2.2 release later this year), Palm (the Flash beta works in WebOS 1.4), and Research in Motion for some of its BlackBerry phones. Flash 10.1 for Symbian devices is likely, but I have heard nothing about preinstalled support.

Microsoft at one point said that Flash would be included with its Windows Phone 7 system due out in time for the holidays in 2010, but the company has since backed away from that timeline. A future release will support Flash.

Adobe's complaints seem to be that Apple won't include support for a technology that nobody else supports and that isn't ready for prime-time mobile use until mid-2010 at the earliest, after an 18-month delay. Jobs writes, "We think it will eventually ship, but we're glad we didn't hold our breath. Who knows how it will perform?"

Jobs also makes public several claims that have been reported as being aired privately within Apple. First, that Flash "is the number one reason Macs crash." Adobe's CEO's response to the Journal: that it has something "to do with the Apple operating system."

We can't evaluate either claim. The only browser plug-in that most users rely on regularly is Flash, and thus browser crashes not caused by browser bugs would be due to Flash. Apple has said this without referring to Adobe before, and it gathers such information directly from automatically generated crash reports.

Second, Jobs claims that Flash is a battery hog due to a lack of the necessary low-level hardware support on mobile devices. Jobs says plain H.264 video burns through power at half the rate that Flash video does. Adobe's CEO says that's "patently false."

It's also hard to evaluate this claim. On a Mac, Flash often chews up processor power, but the Flash 10.1 release looks to be substantially more efficient. Someone could benchmark H.264 playback against Flash video with the 10.1 beta on an Android phone to test the current state of those claims.

The Future — Jobs's essay and Adobe's Wall Street Journal interview make it clear that neither embedded Flash Web content nor Flash-as-apps are in the iPhone OS future. Adobe is pushing hard on other platforms, but to the extent it has made this a battle with Apple, Adobe simply can't win.

Adobe's CEO says the company is working on "dozens" of tablet projects with other companies. But those projects aren't in the market — a market where Apple has already sold 85 million iPhones and iPod touches and, as of 3 May 2010, 1 million iPads.

And just this last week, Microsoft canned the Courier tablet concept project, a well-received set of new ideas for a two-screen slate-sized computer, and TechCrunch reported that HP, in the wake of its Palm acquisition, has opted not to release a Windows 7-based tablet demonstrated by Microsoft's CEO at CES in January 2010.

Adobe has made a strategic error. Its customers aren't Web users; its customers are developers and designers. Adobe should have been spending its time and effort to make a superb creative and production environment for creating content that would appear on every smartphone, tablet, laptop, and desktop. Its customers care vastly more about being able to create something that works in Mobile Safari and other mobile browsers than the reasons why Adobe hasn't yet been able to get Flash working on a single smartphone platform.

Adobe deserves enormous praise for developing Flash for all these years and managing to get it embedded in most browsers. Without Flash, the Web would have been much less interactive, much less fun, and much less useful over the last few years. Video on the Web would have been a non-starter. YouTube, Vimeo, Hulu, and all the rest would probably not have come into existence at the time they did.

Let's give Adobe all that. But it's also clear that the necessity for Flash as the sole enabling technology to bring rich media and rich interaction to the Web is over. HTML5's support for animation, video, and audio, coupled with improvements in JavaScript rendering engines, means that most of what Flash was absolutely needed for will be generally available without buying into a particular company's technology or tools. Adobe's tools will have to evolve to match this new reality.

Beyond this sea change in Web development, going head-to-head with Steve Jobs never works. The only way Adobe could have gotten Flash on the iPhone was to convince Jobs that Apple's customers — we end users — desperately wanted content that was available only from Flash developers. That didn't happen.

Apple is happy to force technological change on the industry, and if HTML5 will meet the industry's needs, few companies will resist adding support for it if they want to serve their own iPhone OS-using customers.

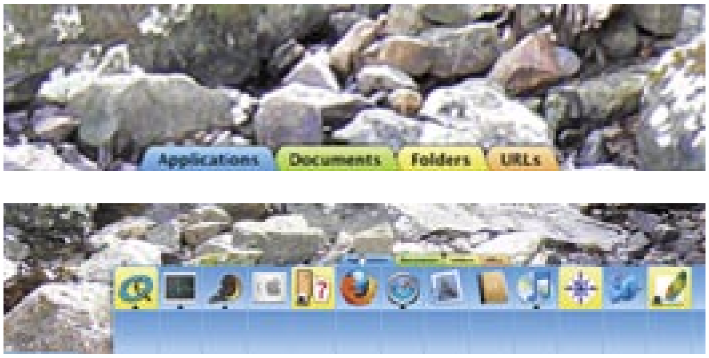
by Tim Sullivan

June
Software Review

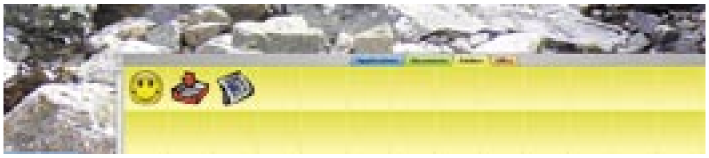
DragThing 5.9.6
http://dragthing.com/

Requires: different version depending upon your version of OS X. Universal binary. $29

This is a replacement for your dock. The view can be customized. I have mine set to dock at the bottom of the screen and using large icons.



DragThing configured as a drawer. The normal, collpsed view is shown at the top. Clicking on the blue (Applications) tab causes the drawer to open. Double click on an icon to launch the program.



DragThing configured as a window. The Folders tab has been selected.

It puts all your documents, folders, and applications just a single click away. Highly flexible, it allows multiple docks, each customized to suit your exact needs.

It stores frequently-used clippings such as text and pictures, and lets you easily paste them into other applications with just a click.

DragThing also allows you to define a wide variety of hot keys. These hot keys can open items in docks, control various other aspects of DragThing and even control the visibility of and switching between other applications running on your Mac.

DragThing can also provide you with a Trash on your desktop, play sounds in response to actions like ejecting a CD, and a lot more besides.

Note: Installing DragThing does not make your dock go away. There are several things that you could do:
1) command-option-D toggles the dock to hide or reappear.
2) Go to System Preferences and move the dock to the side of your screen. 3) Do both.