

MacTeX Design Philosophy vs TeXShop Design Philosophy

Richard Koch

July 29, 2014

A Lesson from Apple

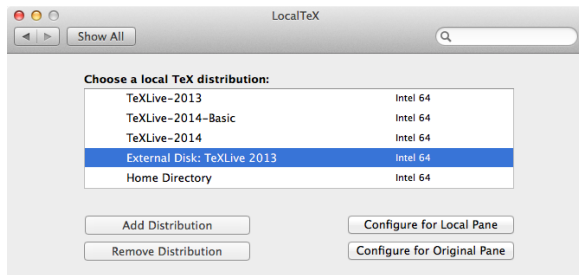
WWDC 2000

Steve Jobs Keynote:

- ▶ OS X Release Version renamed OS X Public Beta
- ▶ \$130 price will instead be \$15 handling fee
- ▶ But, today you can draw for a free piece of shlocky software

Schlock

- ▶ Free thumb drive
- ▶ LocalTeX Preference Pane



- ▶ TeX Live (simple scheme)
- ▶ TeXShop 3.38 + changes document

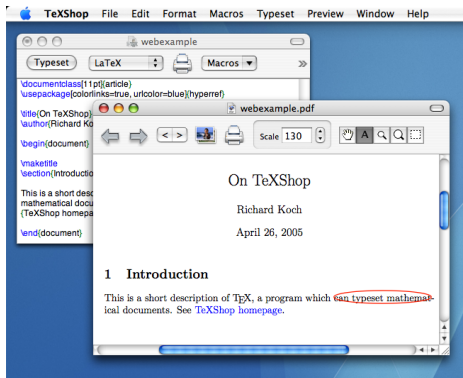
MacTeX Design Philosophy

- ▶ MacTeX installs TeX Live, Ghostscript, and various GUI apps without asking questions
- ▶ Everything is configured and ready to use
- ▶ TeX Live is a *completely unmodified* full version
- ▶ We would *never* reach into TeX Live and change something for the Mac
- ▶ *Cross platform rules; open source forever!*



TeXShop Design Philosophy

- ▶ TeXShop is a Macintosh front end for TeX, written in Cocoa
- ▶ *TeXShop is NOT cross platform.*
- ▶ Your front end mitigates between you (with the conventions of your operating system), and the TeX world (with its conventions). Front ends should not be cross platform.



Example 1

TeX on OS X Mailing List Messages:

From: Warren Nagourney <wnagourney@comcast.net>

I am using TeXshop 2.47 on a retina MBP and have noticed a slight tendency for the letters in the preview window to be slightly slanted from time to time.

From: Giovanni Dore <giovanni.dore@unibo.it>

I think that this is not a problem of TeXShop. I use Skim and sometimes I have the same problem.

From: Victor Ivrii <vivrii@gmail.com>

Try to check if the same distortion appears in TeXWorks and Adobe Reader: TS and Skim are PDFKit based, while TW is poppler based and AR has an Adobe engine.

Following Ivrii's Advice

- ▶ TeXShop and Skim exhibit the bug
- ▶ TeXWorks and Adobe Reader don't
- ▶ The bug is caused by a bug in Apple's PDFKit display code, later fixed
- ▶ It looks like this is an argument for TeXWorks and Adobe Reader, and against TeXShop and Skim
- ▶ But actually ...

Retina Display

The bug is a problem in Apple's Retina Display support code, later fixed by Apple. TeXShop and Skim support the Retina display. TeXWorks and Adobe Reader don't.

Where: /Applications/TeX
Created: Friday, June 6, 2014 at 2:31 PM
Modified: Friday, June 6, 2014 at 2:32 PM
Version: 3.36.2
Copyright: Copyright 2001–2014, Richard Koch
<http://pages.uoregon.edu/koch/texshop/texshop.html>
 Open in Low Resolution
 Locked

Kind: Application
Size: 298,219,817 bytes (301.2 MB on disk)
Where: /Applications
Created: January 3, 2012 at 4:15 AM
Modified: February 15, 2012 at 12:01 PM
Version: 10.1.10
Copyright: Copyright 1984–2012 Adobe Systems Incorporated. All rights reserved.
 Open in 32-bit mode
 Open in Low Resolution
 Prevent App Nap
 Locked

A Boast to my Student

- ▶ Koch: TeXShop supported the Retina display from the beginning.
- ▶ Splonskowski: Yeah? And how many lines of code did that take?

TeXShop, Skim, TeX Live Utility, BibDesk, and LaTeXiT are written with Cocoa. So are several commercial front ends in the Apple store. Cocoa programs automatically support the Retina display. Programs written with Carbon, or Java, or open source libraries, don't.

Cocoa

- ▶ Inherited from NeXT
- ▶ The modern way to program the Mac, and the *only* way to program the iPhone and iPad
- ▶ Led a precarious existence at Apple for many years; shunned by Microsoft, Adobe, and a host of others
- ▶ After ten years, achieved a total victory over competing technologies, all now deprecated (Carbon)
- ▶ Cocoa provides “base classes”; the programmer gives the code substance by “subclassing the base classes”; Apple can modify the base classes because they solved the “fragile base class problem” in objective C during the transition to 64 bits.
- ▶ Apple added ARC, automatic reference counting, to modernize memory management.
- ▶ Then Apple made obsolete all Macs which cannot use these modern features

Example 2: Remember Window Position and Size When Closing; Open Windows in the Same Positions When Opening

Example 3: Automatic Saving in TeXShop

The image shows a screenshot of the TeXShop application interface. Two LaTeX documents are open side-by-side. The left document, titled "LocalTeX-PrefPane.tex", contains a paragraph of text discussing Apple's transition to 64-bit libraries in the Tiger-Leopard timeframe. The right document, titled "D8D01560-BB7C-4FF2-A3C8-9910EBE500DE.tex", contains LaTeX source code for a document, including package declarations and sectioning commands. The interface includes a menu bar with options like "Typeset", "LaTeX", "Macros", "Tags", and "Templates". A toolbar at the bottom contains buttons for "Current Document", "Done", "Restore", and a clock showing "Today at 2:35 PM".

```
LocalTeX-PrefPane.tex -- Edited
Typeset LaTeX MacOS Tags Templates

318 wasn't quite as bad in objective C as elsewhere, because it had been designed to extra methods in base
319 classes are legal. But still: no extra instance variables.
320
321 When Apple added 64 bit libraries in the Tiger-Leopard timeframe, they realized that they had a once in a
322 lifetime opportunity to fix this problem [before that, there
323 were no 64 bit applications, so every 64 app would have to be compiled from scratch. So they made several
324 changes to objective C at that time, including completely solving the fragile base class problem. if your
325 Macintosh application runs in 64 bits, then the dream of improving everything by revising the base classes
326 can be realized.
327
328 Incidentally, at first the iPhone was programmed in 32 bits. But since no iPhone apps preexisted release, the
329 fragile base class problem was solved on phones even in 32 bits.
330
331 After this change, Apple rapidly made many older machines obsolete. Snow Leopard required intel
332 processors, and Lion required 64 bit processors, and
333 Mountain Lion required machines running the kernel in 64 bits. Notice that since then, the policy changed:
334 Mavericks and Yosemite run on all machines that
335 can run Mountain Lion, and are free to boot. I cannot confirm this, but I strongly believe that the reason for
336 these policies is not that 64 bit programs run
337 faster, but instead that Apple can now use all the extra added properties of objective C, including adding
338 instance variables and methods to base classes.
339
340
341
342
343
344 \section{Lion}
345
346 Lion is the first Apple system to make real this great dream of objective oriented programming. Programs
347 written in 64 bits with Cocoa got crucial
348 added functionality for free, essentially without even a recompile.
349
350 One of the standard requests for TeXShop was that it remember window sizes and positions when gull, and
351 restore these windows automatically when next restarted. To shame me into working on this, users told me of
352 other GUI's for TeX which already had the ability.
353
354 Imagine my surprise, then, when I discovered that TeXShop on Lion got the requested ability automatically for
355 free.
356
357 An advantage of letting Apple do this is that Apple had second thoughts and slightly modified the behavior in
```

```
D8D01560-BB7C-4FF2-A3C8-9910EBE500DE.tex
Typeset LaTeX MacOS Tags Templates

1 \documentclass[11pt,oneside]{article} % use "amsart" instead of "article" for AMSLaTeX format
2 \usepackage{geometry} % See geometry.pdf to learn the layout options. There are lots.
3 \geometry{letterpaper} % ... or a4paper or a5paper or ...
4 \usepackage{landscape} % Activate for for rotated page geometry
5 \usepackage{partfil} \partskip % Activate to begin paragraphs with an empty line rather than an indent
6 \usepackage{graphicx} % Use pdf, png, jpg, or eps with pdfLaTeX; use eps in DVI mode
7 % TeX will automatically convert eps -> pdf in pdfLaTeX
8
9 \usepackage{amssymb}
10 \usepackage{url}
11
12 \title{MacTeX Design Philosophy vs TeXShop Design Philosophy}
13 \author{Richard Koch}
14 %\date{} % Activate to display a given date or no date
15
16 \begin{document}
17 \mainmatter
18 \section{}
19 \subsection{}
20
21 \section{A Sop for the Audience}
22
23 I went to the Apple Developer Conference in May, 2000. Developers at this conference were supposed to
24 receive the release version of OS X.
25 In the keynote address, Steve Jobs announced
26 that the new release would be renamed OS X Public Beta with a price reduced from $130 to a handling fee
27 of $15. After the keynote,
28 a knowledgeable friend translated: "OS X has been delayed by a year."
29
30 As a sop to the audience, Apple held a software raffle during this conference, the only time I've heard of them
31 doing so. Every developer got something, but it soon transpired that there were
32 only a few copies of Adobe Illustrator and Photoshop, and everybody else got a slocky piece of software on
33 a CD, shrink wrapped against a
34 piece of cardboard. The documentation was on a stamp-sized folded paper between the CD and the
35 cardboard which had to be unfolded
36 like the documentation for a Timex watch.
37
38 So ... I was looking at this talk I promised to give, and there isn't much of interest here. Then I thought of
39 Apple's playbook and decided to
40 give each of you a free piece of software. Here it is.
41
42
```

Current Document Done Restore Today at 2:35 PM

Example 3: Big Changes Made by Automatic Saving

File	Edit	Source	Macros	T
New			⌘N	
New from Stationery...				
Open...			⌘O	
Open Recent				▶
Open for Preview...				
Show Console				
Show Log File				
Close			⌘W	
Save			⌘S	
Save As...			⇧⌘S	
Save To...				
Revert To Saved			⌘U	
Page Setup...			⇧⌘P	
Print...			⌘P	
Print Source...				

File	Edit	Source	Macros
New			⌘N
New from Stationery...			
Open...			⌘O
Open Recent			▶
Open for Preview...			
Show Console			
Show Log File			
Close			⌘W
Save			⌘S
Duplicate			⇧⌘S
Rename...			
Move To...			
Export...			
Revert To			▶
Page Setup...			⇧⌘P
Print...			⌘P
Print Source...			

Example 3: Implementing Automatic Saving

```
BOOL doAutoSave = [SUD boolForKey:AutoSaveEnabledKey];
```

```
+ (BOOL)autosavesInPlace  
{  
    return doAutoSave;  
}
```

Example 3: Uniform User Praise for Automatic Saving in TeXShop



+50



Amberg reviewed on 18 Dec 2012

This is a nice TeX editing environment, but these days it is marred by Apple's hideous Auto-Save feature. It would be nice to have a version of this program that used the normal load/save/revert file management system. Another great loss in recent history is the find and replace panel. It now just uses Apple's bog standard panel which is okay for non-coders but fairly worthless for anyone that needs to work around code spans.

So, overall, in its eagerness to adopt some of Apple's less enticing newer OS features, it's kind of dropped behind the pack for me.

[Version 3.11]

Example 4: Automatic Reference Counting (ARC) in 3.34 through 3.38

- ▶ In a Cocoa program, objects are created dynamically. When the program is done with an object, it must reclaim its memory. If this is done too conservatively, computer memory becomes clogged and the program becomes sluggish. If it is done too aggressively and objects still being used are thrown away, the program will crash.
- ▶ Garbage collection is an automatic way to handle memory management, but it proved inappropriate for the iPhone. So Apple invented “automatic reference counting,” a method in which the compiler automatically handles memory management without intervention of the developer.
- ▶ The most important (but invisible) change in TeXShop in the last six months has been the adoption of ARC. This was begun in version 3.34, and is finished in version 3.38 which was released to you today.