

Documentation - Why All The Cool Kids Are Doing It

Rich Trouton

Howard Hughes Medical Institute,
Janelia Research Campus



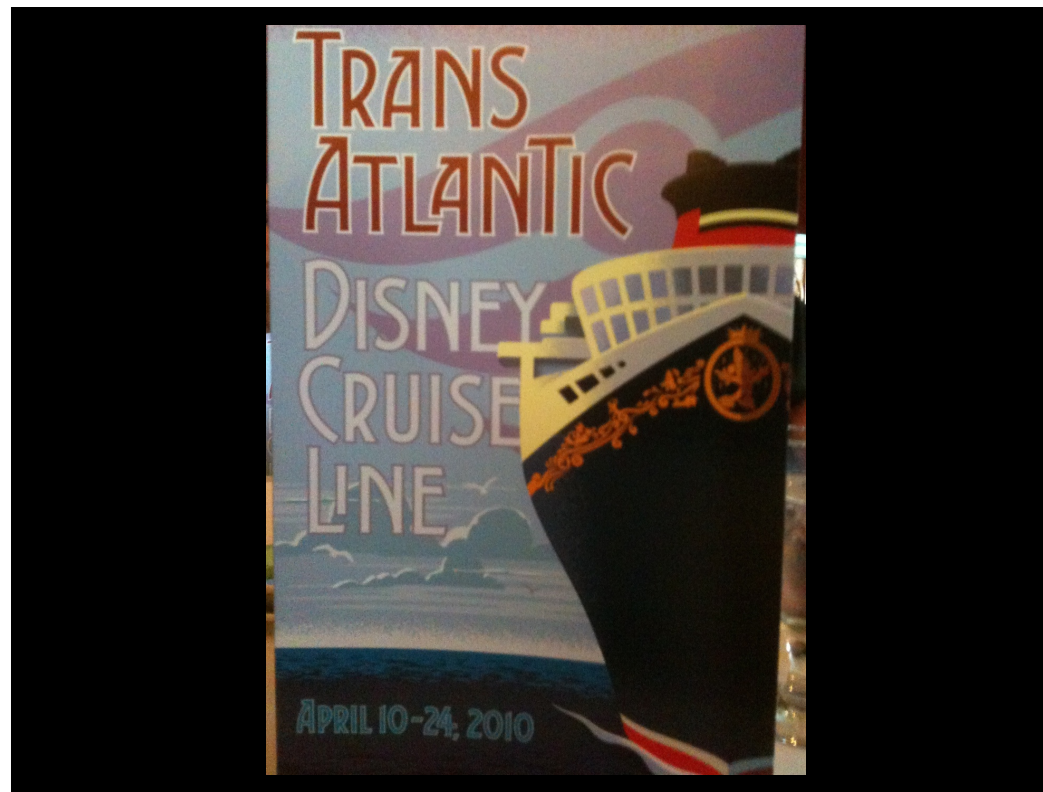
Howard Hughes
Medical Institute

Before we get started, there's two things I'd like to mention. The first is that, all of the slides and speakers' notes are available for download and I'll be providing a link at the end of the talk. I tend to be one of those folks who can't keep up with the speaker and take notes at the same time, so for those folks in the same situation, no need to take notes. Everything I'm covering is going to be available for download.

The second is to please hold all questions until the end. If you've got questions, make a note of them and talk with me afterwards. With luck, I'll be able to answer most of your questions during the talk itself.

Why Document?

So, why document? It's a lot of work, especially if you're someone who normally doesn't write a lot.



One reason for me is that I like to take vacations. Here's a picture from one I took a few years ago.



90 percent of the time, this was my view. My glorious, beautiful, thousands of miles from anywhere view.



Sample Rates for Connect@Sea		
Pay-As-You-Go	Get started with this package if you're not sure how much you'll be online. You can upgrade to a different package at any time.	\$0.25 per MB
Small	This package is a good choice for occasional use. Even though you're on vacation, you still want to keep tabs on things back home once in a while. Check your email—and then go back to enjoying your cruise.	100 MB \$19.00 (\$0.19 per MB)
Medium	A package for moderate use. You want to share your cruise experience with friends and family back home.	300 MB \$39.00 (\$0.13 per MB)
Large	All these packages can be shared, but this one makes it easy. Stream movies or your favorite music. This package covers all your Internet needs.	1,000 MB \$89.00 (\$0.09 per MB)

My internet connection was both slow and expensive, which focused my priorities towards using my expensive internet time to post photos to Facebook and away from checking my work email.



Onboard Extras

 [Print This Page](#)

Q: Can I contact people back home from the ship?

A: Your friends and family can contact the ship by calling (888) 724-7447. Or from outside the U.S. they can call (321) 953-9003. The cost is \$7.95 USD a minute and can be charged to their MasterCard® or Visa®. From outside the U.S. additional long distance charges will also apply.

You can make ship-to-shore calls from your stateroom, 24 hours a day. The cost is \$7.95 USD per minute and will be automatically charged to your SeaPass account.

You can also use our onboard Internet Cafés, stateroom Internet connection, or Hot Spots to send an e-mail to your friends or family for a minimal fee.

\$7.95 a minute?!?

It was possible for work to have called me on the boat, but as you can see, the cost would have likely pushed whatever problem work was having from “Call him right now” to “This can wait for him to get back”.



One of the reason this worked was that I left behind documentation covering what to do in case of emergencies as well as day-to-day tasks. In fact, as I was riding the airport shuttle on my way to a similar trip, I got a call from my workplace where the person calling said that they had a power outage at our main datacenter. I was gone, my alternate was stuck in traffic, what should they do? I said “Can you find the Mac server rack?” Yes, they found it. “Do you see the packet marked Emergency Server Startup and Shutdown Procedures?” Yes, they did. “OK, open that and start reading. It’ll walk you through the process.” I talked with them for a few more minutes to make sure that they were OK, then I said goodbye and ended the call.

Without that packet attached to the front of the server rack, which I had made sure was current the day before, I might have been trying to walk someone through the shutdown procedure for about fifteen servers and twelve RAID arrays over the phone up until the moment that the flight attendant yanked my phone out of my hand because the plane needed to take off. That is why I document.

How Does Documentation Help You?

Memory Aid

- Use documentation as your external memory.
- Documentation can be used to track the smallest details.

Teaching Tool

- Documentation can be given to others to help train them.
- Documentation can answer others' questions in place of them needing to ask you.

Everyone's memory is fallible. Moreover, Murphy's Law practically guarantees that you're more likely to forget things when it's most important that you remember them.

It's also a lot easier to teach something to someone else when you have it well documented. Hopefully, that will then save on people asking you about the same thing repeatedly, which conserves your time for other things. Hopefully.

When to Document?

Required Documentation	Common Tasks
Business Continuity	Unusual Situations

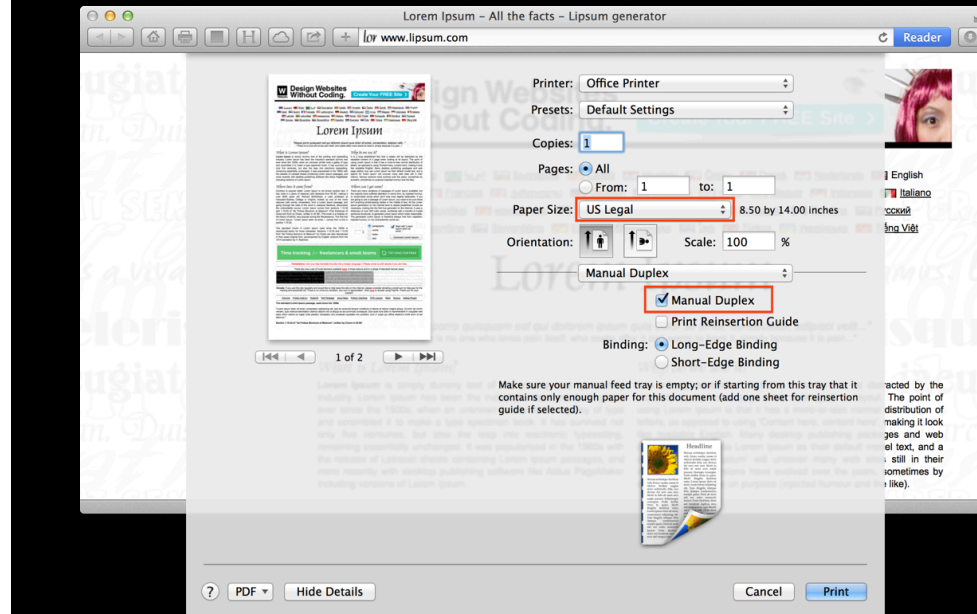
Now that we've discussed why you should be documenting, let's discuss when you should be writing documentation.

Required Documentation



Documentation of certain processes and procedures may be required by law or a policy at your workplace. In these cases, you must document to make sure you're in compliance.

Common Tasks



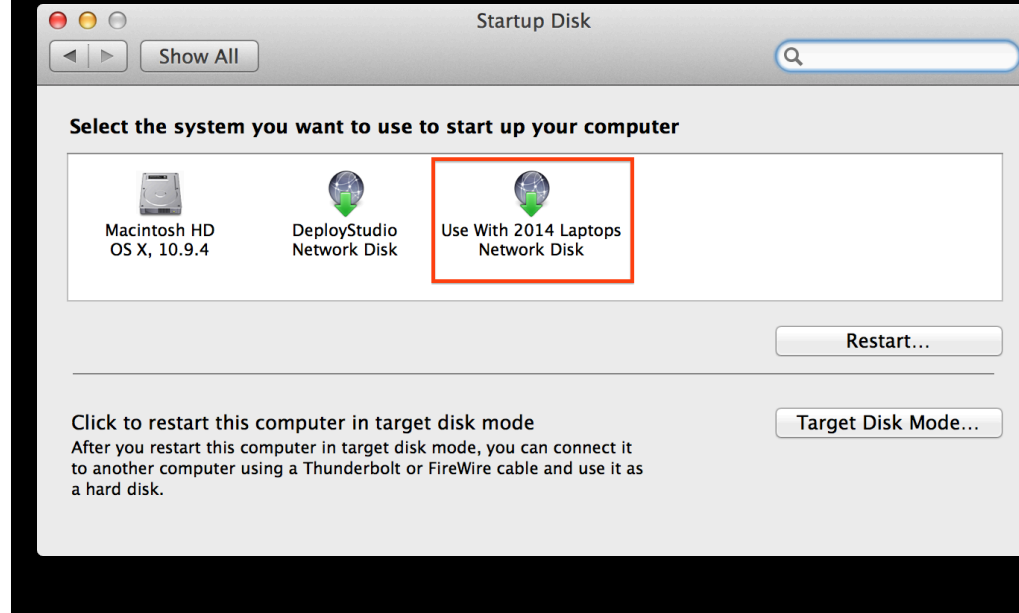
Check your common day-to-day processes to see if the available documentation covers common situations. For example, you generally don't need to write documentation to show someone how to print to the office printer. What may not be documented is how to set up the printer to print double-sided from the legal-sized tray.

Business Continuity



If you're not available, can others get the job done? Check your regular processes and procedures to see if the available documentation is sufficient to help others get them done in the event that you're not there. After all, even if you get hit by the lottery bus, the job still needs to get done.

Unusual Situations



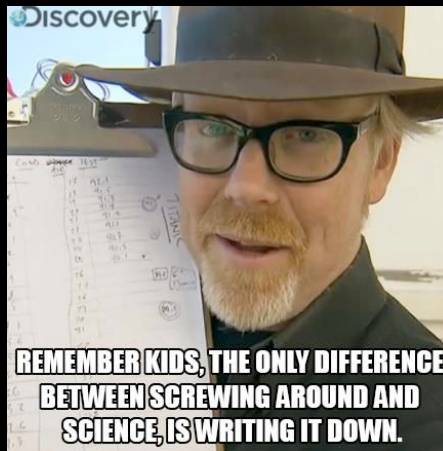
Documentation also allows you a way to communicate how to properly handle unusual situations. You may be able to assume that your coworkers know to hold down the N key on new machines' keyboards to NetBoot a machine from your work's default NetBoot set. However, do they know how to handle that new MacBook Pro that isn't booting up? Probably not unless you've documented that they should be using the non-default NetBoot set which you built to support new laptops.

What About Disaster Recovery?



Disaster recovery documentation may fit under documentation that you're required to do. If it isn't, you should certainly be doing it as business continuity documentation. Disasters happen regardless if you're in or out of the office.

Your disaster recovery plan should also not consist of this. If there's ever a time to get all the details included in documentation, disaster recovery is it. Remember, you may be out of the office and it may fall to someone completely untrained to safely bring your systems down and then back up later. It happened to me, it can happen to you as well.



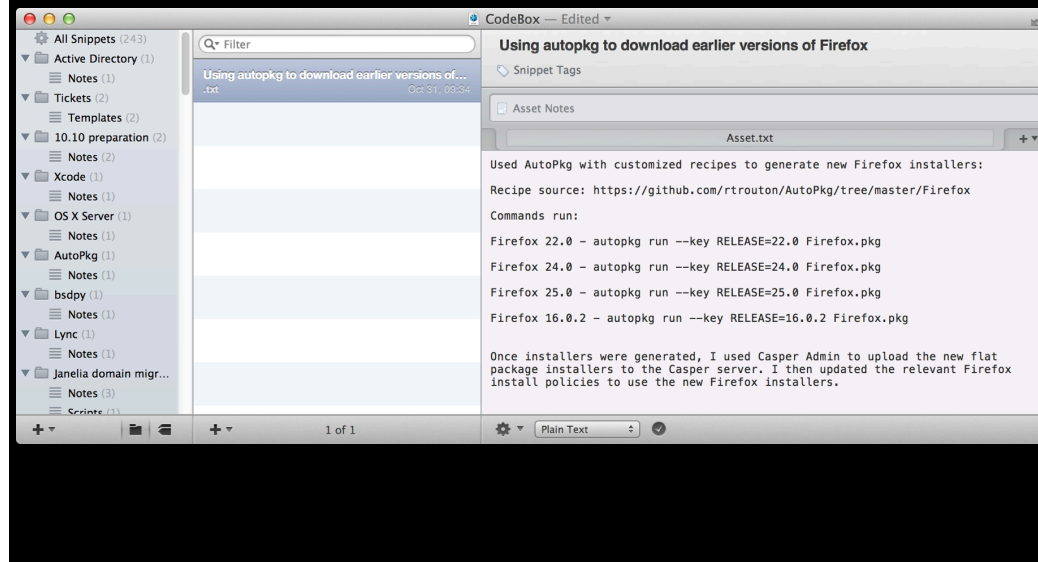
Once an assessment has been done, it should be straightforward to find where your documentation has gaps. When you find those gaps in your documentation, fill them.

Who's The Audience?

You
Colleagues
Anyone

Who is the audience for your documentation? Usually, this can be broken down into three main groups: You, your colleagues and everyone else.

You Are The Audience



The documentation that you write for your own use is first and foremost a memory aid. This kind of documentation can take all kinds of forms, from notes typed into a notebook app, emails sent to yourself, or information posted to a wiki. In my opinion, this is the only time where you can make assumptions about the knowledge level of the audience. Even there, we all have off days.

The other thing to keep in mind is that documentation that you write for yourself should have sufficient detail that you can turn it into documentation for other folks fairly easily.

Colleagues Are The Audience

Tell the story of the problem

- What It Is
- Implications
- Process Followed To Solve Problem
- Solution To Problem

When writing documentation for your colleagues, you want to tell them the story of the problem:

- How the problem was discovered
- The implications of the problem
- The process you followed when solving the problem. If appropriate, include where you initially took an incorrect approach and how you recognized and fixed that
- How you solved the problem

Add all of the details you can into the documentation, including the commented code if a script is involved.

Latest version of Skype now only available for Macs running 10.9.x or higher

May 22, 2014

trouton

[Go to comments](#)

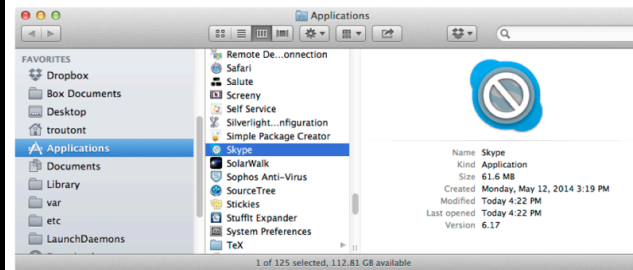
[Leave a comment](#)

[Edit](#)

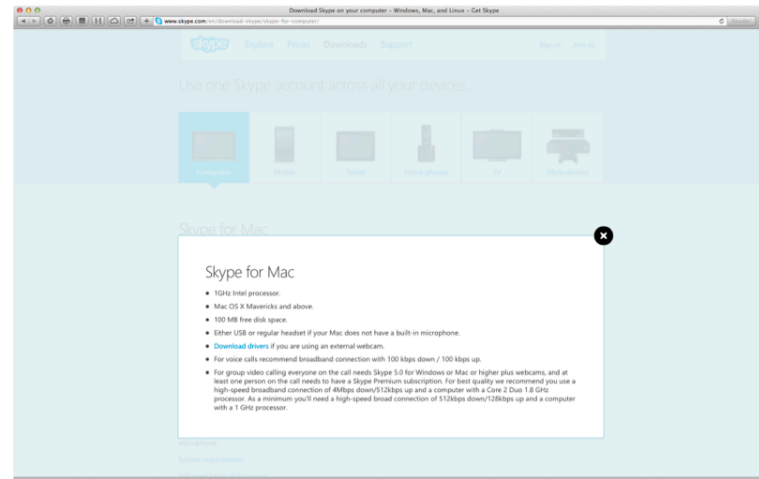
While assisting a user today whose workstation is running 10.8.5, I downloaded the latest version of Skype and tried to run it. To my surprise, I received the following message.



It looks like Skype 6.1.7 only supports 10.9 and higher, which is not something mentioned on the [Skype support site's system requirements page](#).



The new system requirements are available on the [Skype download page](#) by clicking the **System Requirements** link found there.



For those that need a version of Skype that works on 10.6.x – 10.8.x, it looks like Skype 6.1.5 (334) is the latest version available. It's available from the following link:

<http://www.skype.com/go/gets skype-macosx-106-107-108>

Hat tip to [@tvsutton](#) for finding the download link for the latest version of Skype that works on 10.6.x – 10.8.x.

Using /etc/auto_home on Mavericks to mount shares under /home

April 6, 2014  rtrouton

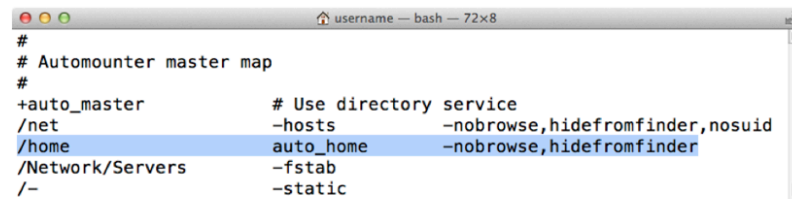
[Go to comments](#) [Leave a comment](#) [Edit](#)

One of my users at work asked me recently about symlinking his network home folder to **/home** on his Mac running 10.9.2 and wanted to check to see if it was safe to do so.

In this case, the person in question works on both [Fedora Linux](#), where his network home directory was mounted as **/home/username**, and on OS X. His network home directory was available via SMB on his Mac as **smb://servername/home\$/username**. He wanted to be able to mount **smb://servername/home\$/username** to **/home/username** on his Mac, so that it matched the mountpoint of his network home on his Fedora box.

At the time, here's what I knew about **/home**:

1. Nothing appears to be stored in it by default
2. It's listed in **/etc/auto_master** as a mountpoint



```
username — bash — 72x8
#
# Automounter master map
#
+auto_master      # Use directory service
/net              -hosts          -nobrowse,hidefromfinder,nosuid
/home             auto_home    -nobrowse,hidefromfinder
/Network/Servers  -fstab
/-                -static
```

3. [Time Machine](#) does not back it up

After talking with Apple's enterprise support folks and doing some additional research, the file that controls what's mounted in **/home** is the **/etc/auto_home** auto mount file.

```
username — bash — 82x8
#
# Automounter map for /home
#
+auto_home      # Use directory service
#
# Get /home records synthesized from user records
#
+/usr/libexec/od_user_homes
```

To do what my user wanted to do, the following entry could be added to **/etc/auto_home** to have the designated share mount as **/home/username**:

```
1 | username    -fstype=smbfs    :// 'DOMAIN;username':password@servername/home$/username
```

```
username — bash — 82x10
#
# Automounter map for /home
#
+auto_home      # Use directory service
#
# Get /home records synthesized from user records
#
+/usr/libexec/od_user_homes

username -fstype=smbfs :// 'DOMAIN;username':password@servername/home$/username
```

Note: If you have a password longer than 8 characters, or if the password has special characters in it (like `! # $ % & ' () * + , - . / : ; < = > ? @ [\] ^ _ { | } ~`), you may receive a **"No locks available"** error message and the share will not mount under **/home**. You will also receive a **"No locks available"** or similar **"Host is down"** error if the password is wrong or missing.

That said, having an SMB mount entry in **/etc/auto_home** may not a good idea for the following reasons:

1. The username and password will need to be stored in **/etc/auto_home** as part of the mount entry. [autofs](#) on OS X doesn't support using Kerberos authentication, so there's no good way to secure the connection.
2. The first person to access **/home/username** will have full access to the share as that user on the server. If this is a multi-user system, subsequent users will have no access to **/home/username** on that Mac.

If NFS is an available option, this may be better because no usernames or passwords will need to be stored in **/etc/auto_home** in order to mount an NFS share. If the network home is mountable from the server from a **home** NFS share, the following entry could be added to **/etc/auto_home** to have the designated share mount as **/home/username**:

```
1 | username    servername:/home/username
```

If you want to have an NFS mount show up under **/home** for each user that logs in, you can also use wildcards (*****) and ampersands (**&**) in **/etc/auto_home**. For example, the following entry could be added to **/etc/auto_home** to have appropriate shares mount as **/home/username_goes_here**:

```
1 | *    servername:/home/&
```

For more information on properly formatting entries for **/etc/auto_home**, including information on using wildcards (*****) and ampersands (**&**), I recommend reading the [man page for /etc/automaster](#). The wildcard information is in the **Wildcards** section, while information about using ampersands is available in the **Substituting the map key entry** section.

Colleagues Are The Audience

Why Go To All That Effort?

- You can use documentation as a way to teach your teammates how to solve issues.
- Hopefully, the example will inspire them to become better problem solvers themselves.
- Just giving the solution may mean you're the only problem solver.

The idea is not just to tell your colleagues how to solve the problem, but also show them how it got solved. After all, if you can teach them how to fish, maybe they can start catching fish on their own. Even better, maybe they'll be able to start arriving at better solutions.



Also, one thing I hear again and again is “Why should I document this? Nobody will read this except me.” When I hear that, I know that at least one person will be reading the documentation: you.

Treat your future self right and make sure to document the process you were following because walking through the process may help you get back into the mindset you were in when you wrote the documentation.

Colleagues Are The Audience

Does It Work?

Der Flounder

Seldom updated, occasionally insightful.

- I've been using this approach for years on Der Flounder
- Is it working?

For those who wonder if it works, I've been using this approach wherever possible for the past few years on my blog, Der Flounder. For my readers in the audience, can I get a show of hands if you think it's working?

Anyone Is The Audience

Fix the problem

- Anticipate the user's likely question to narrow down scope of problem.
- Provide step-by-step solution to problem
- If appropriate, information about the problem can be included after the step-by-step solution.

When writing documentation for outside your circle of colleagues, any assumptions you may have been making about knowledge level should be immediately discarded. Instead you should focus on helping the reader figure out if the problem you're covering in the documentation applies to them. If it does, provide a step-by-step solution to fix the problem they're having.

If appropriate, include the story of the problem in the documentation. However, that information should go in its own section after the solution. That way, the user doesn't need to read through it if they don't want to. In a case like this, it'd be nice if the user received education along the way, but the ultimate goal is to get the user back up and working as quickly as possible.

Knowledge Base Article Detail

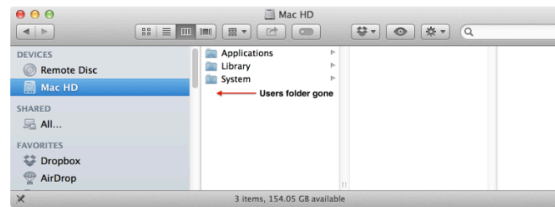
Below are the details of the Knowledge Base article you selected. Use the back button on your browser or navigation links at the top of the page to start another search.



Title: Unable to find Users folder on hard drive
Category: [Operating Systems Support](#)
Issue Type: [Macintosh](#)
Wiki Content:

Problem:

I'm unable to find my Users folder on my hard drive. How can I fix this?



Solution:

Please install iTunes 11.2.1 or later to fix this issue. iTunes 11.2.1 and later is available via [Apple's Software Update](#) or from Apple's [iTunes download site](#).

If you cannot update to iTunes 11.2.1 or later, please follow the steps below to fix this issue.

1. Open System Preferences
2. Open the **iCloud** preference pane



3. Check if **Find My Mac** is enabled



4. If **Find My Mac** is enabled, uncheck it to disable it



5. Verify that **Find My Mac** is disabled

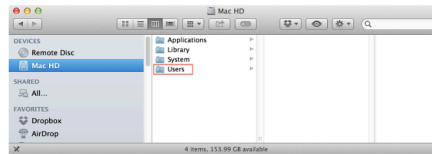
6. Download and run the installer package linked below.

 [unhide_users_and_fix_permissions.zip](#)

Note: Running the installer package may take ten minutes or so.

7. Reboot

8. After the reboot, verify that you can see the **Users** folder.



Background information:

One side effect of the **iTunes 11.2** update released on Thursday, May 15th 2014 has been that some but not all Macs were seeing the **/Users** and **/Users/Shared** folders disappear. The permissions on the **/Users** folder were also changed to be world-writable, so that anyone could read and write to the **/Users** folder.

The issue has been tied to two causes:

1. **iTunes 11.2** being installed
2. iCloud's **Find My Mac** being enabled.

The installer package linked above runs a script that looks for the **/Users** and **/Users/Shared** directory. If found, the **/Users** and **/Users/Shared** directories are unhidden. A permissions repair is then run using the **diskutil** command to fix the world-writable permission issue for the **/Users** directory.

Note: *The permissions repair may take up to 10 minutes to run.*

It is important that **Find My Mac** be disabled before the permissions fix and also that **Find My Mac** remain disabled. If **Find My Mac** is re-enabled, the **/Users** and **/Users/Shared** folder will disappear again and **/Users** will revert to being world-writable.

Knowledge Base Article Detail



Below are the details of the Knowledge Base article you selected. Use the back button on your browser or navigation links at the top of the page to start another search.

Title: Fixing repeated prompts to unlock the "Local Items" keychain

Category: [Password Reset & Unlock](#)

Issue Type:

Wiki Content:

Problem:

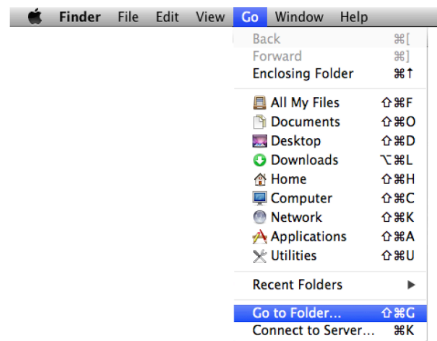
After logging in, I'm getting repeatedly prompted to unlock the **Local Items** keychain in multiple applications. How can I resolve this?



Note: This article is about the **Local Items** keychain, not the **Login** keychain. Please [see this article](#) for issues with the "Login" keychain password.

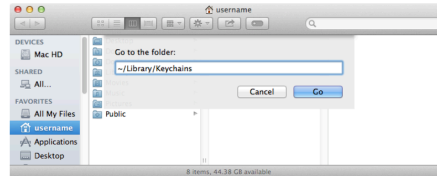
Solution:

1. In the Finder, select the **Go** menu.
2. Under the **Go** menu, select **Go to Folder...**



3. In the window that appears, type the following:

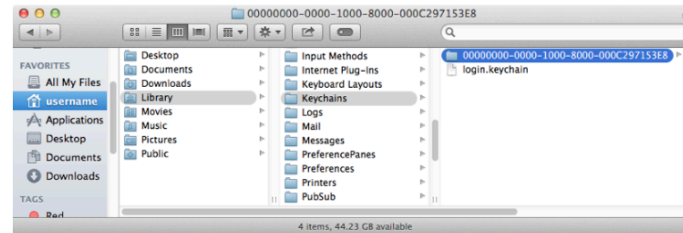
~/Library/Keychains/



4. Click **OK**.

5. In the **Keychains** folder, look for a folder with a name similar to this: **A8F5E7B8-CEC1-4479-A7DF-F23CB076C8B8**

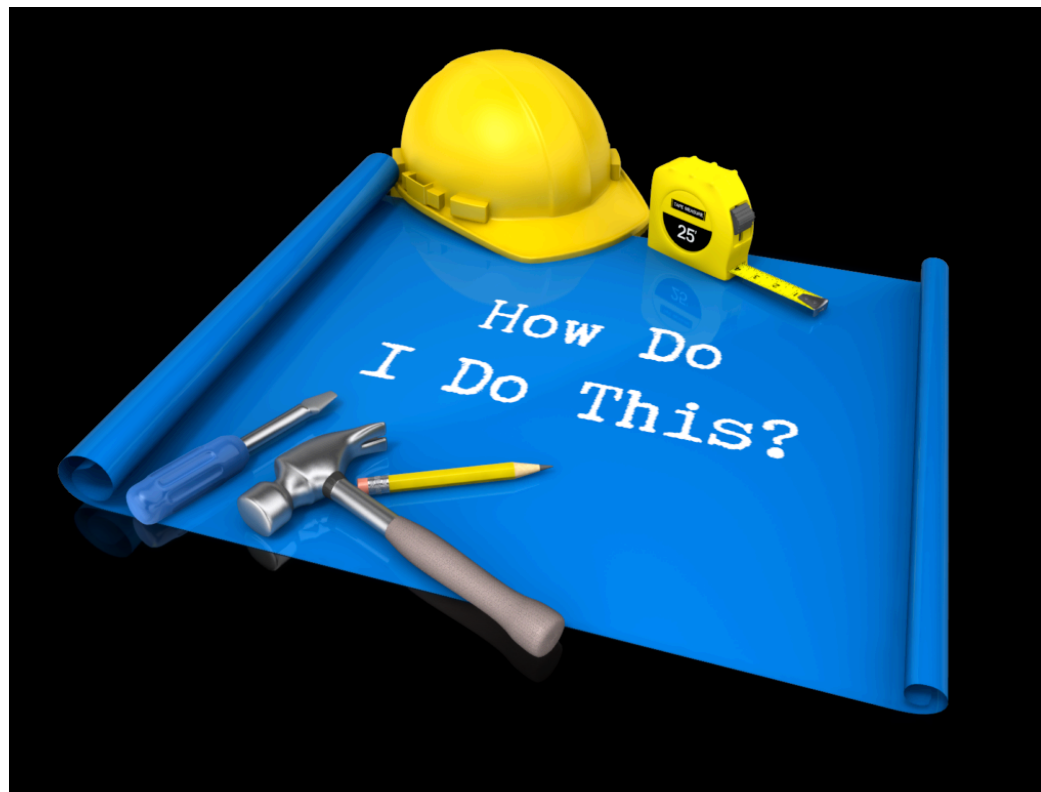
Note: Each folder has a unique number and will not appear exactly the same.



6. Move this folder to the Trash.

7. Restart your Mac.

After restarting the computer, a new folder will be created in the **Keychains** folder with a name similar to "**4B29A0BB-599D-47FC-A2D1-42B5592B130B**". The new folder is expected and will fix the prompting issue.



At this point, we've discussed why you should be documenting and the styles you may want to use when addressing different audiences. Now let's turn to how you can do this for your own shop.

Process	Tools
Generizicing	Media

How do to it breaks down for me in four major areas. There's the process you use to write documentation, the tools you use, genericizing, and the media you're using.

Process

- Everyone will have their own process
- My own process
 - Gathering raw data is top priority
 - Capturing as many details as I can into a plaintext document
 - Screenshotting everything I can
 - Once all available details have been gathered
 - Write up the details in preferred format.
 - Reshoot screenshots as needed.

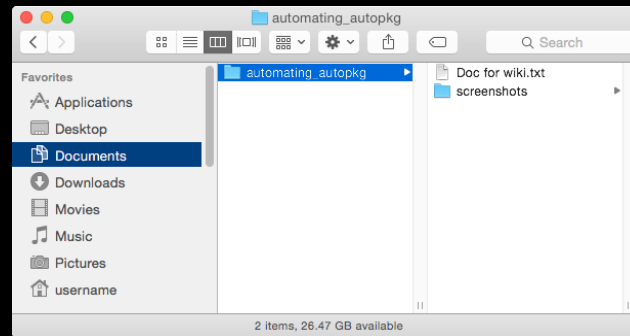
Process is probably where everybody's going to do things a little differently. For myself, my process of writing documentation involves getting the details down first and screenshotting every step I can. The screenshots I'm getting sometimes aren't what goes into the final documentation, but they're invaluable to me if I need to recreate something later.

In my process, the overriding initial priority is getting the raw data. It may turn out that I don't need something after all, but I'd rather have it and not need it than the reverse.

Without the details, all the polishing and wordsmithing in the world will not make your documentation better.

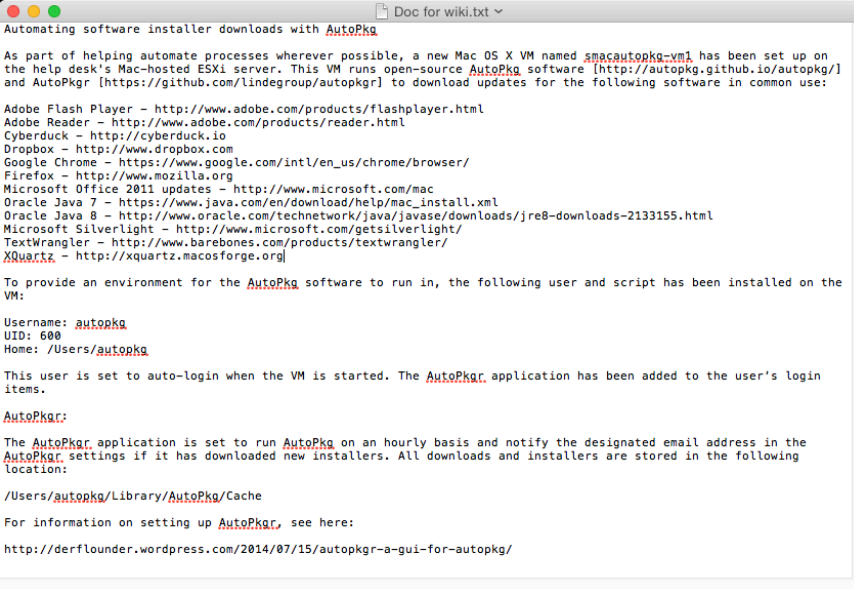
Process

Getting it all in one place



My usual process begins like this, by creating a folder where I'm going to store everything related to a particular piece of documentation.

Process



```
Automating software installer downloads with AutoPkg

As part of helping automate processes wherever possible, a new Mac OS X VM named amacautopkg-vm1 has been set up on the help desk's Mac-hosted ESXi server. This VM runs open-source AutoPkg software (http://autopkg.github.io/autopkg/) and AutoPkgR (https://github.com/lindegroup/autopkgR) to download updates for the following software in common use:

Adobe Flash Player - http://www.adobe.com/products/flashplayer.html
Adobe Reader - http://www.adobe.com/products/reader.html
Cyberduck - http://cyberduck.io
Dropbox - http://www.dropbox.com
Google Chrome - https://www.google.com/intl/en\_us/chrome/browser/
Firefox - http://www.mozilla.org
Microsoft Office 2011 updates - http://www.microsoft.com/mac
Oracle Java 7 - https://www.java.com/en/download/help/mac\_install.xml
Oracle Java 8 - https://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html
Microsoft Silverlight - https://www.microsoft.com/getsilverlight/
TextWrangler - http://www.barebones.com/products/textwrangler/
XQuartz - http://xquartz.macosforge.org/

To provide an environment for the AutoPkg software to run in, the following user and script has been installed on the VM:

Username: autopkg
UID: 600
Home: /Users/autopkg

This user is set to auto-login when the VM is started. The AutoPkgR application has been added to the user's login items.

AutoPkgR:

The AutoPkgR application is set to run AutoPkg on an hourly basis and notify the designated email address in the AutoPkgR settings if it has downloaded new installers. All downloads and installers are stored in the following location:

/Users/autopkg/Library/AutoPkg/Cache

For information on setting up AutoPkgR, see here:

http://derflounder.wordpress.com/2014/07/15/autopkgR-a-gui-for-autopkg/
```

Next, I do a dump of all of the relevant information into a document. What the documentation will likely be titled, the various components, links to those components, any and all important details.

Process

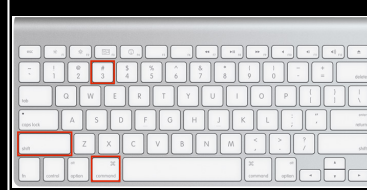
Saving screenshots to a file

Command (⌘)-Shift-3	Captures a screenshot of your entire screen, then saves it as a file.
Command (⌘)-Shift-4	Turns cursor into a crosshair, which you can drag to select an area of the screen to capture. Once selected, area will be saved as a file.
Command (⌘)-Shift-4, then selecting space bar, then clicking on a open window	Hitting the space bar turns the crosshair into a little camera icon, which you can move over any open window. Click on the window to take a screenshot of a window and save it as a file.

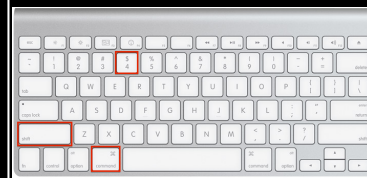
As mentioned, part of my process is screenshotting as much as I can. For this, I rely on OS X's built-in capability to take screenshots via keyboard shortcuts.

Screenshots

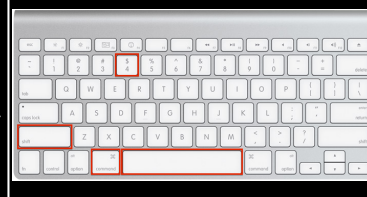
Command (⌘)-Shift-3



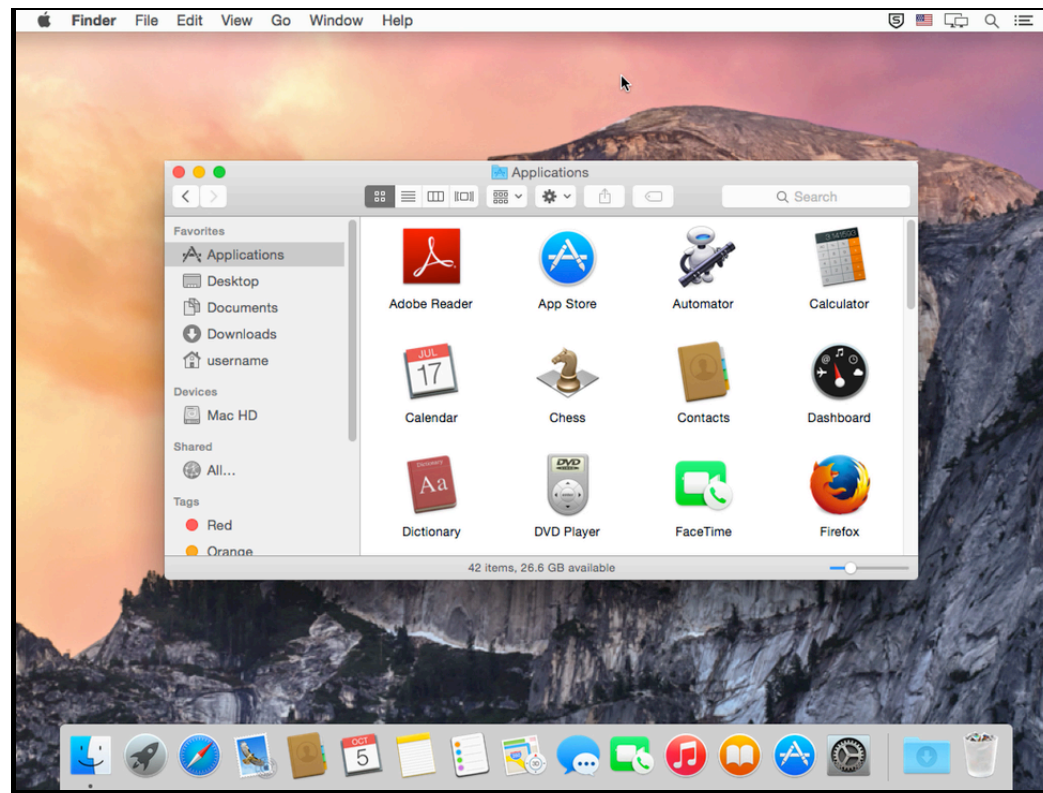
Command (⌘)-Shift-4



Command (⌘)-Shift-4,
then selecting space bar



More often than not, I'm using Command-Shift-Four, then clicking on the spacebar to get a camera icon.



Once I have the camera icon, I can move it onto a window that I want to get a screenshot of and click the window to make the screenshot.




Screenshots

Saving screenshots to the clipboard

Command (⌘)-Control-Shift-3	Captures a screenshot of your entire screen, then saves it to the Mac's clipboard.
Command (⌘)-Control-Shift-4	Turns cursor into a crosshair, which you can drag to select an area of the screen to capture. Once selected, area will be saved to the Mac's clipboard.
Command (⌘)-Control-Shift-4, then selecting space bar, then clicking on a open window	Hitting the space bar turns the crosshair into a little camera icon, which you can move over any open window. Click on the window to take a screenshot of a window and save it to the Mac's clipboard.

If you prefer to save directly to the clipboard, add the Control key to the keyboard shortcut sequence.

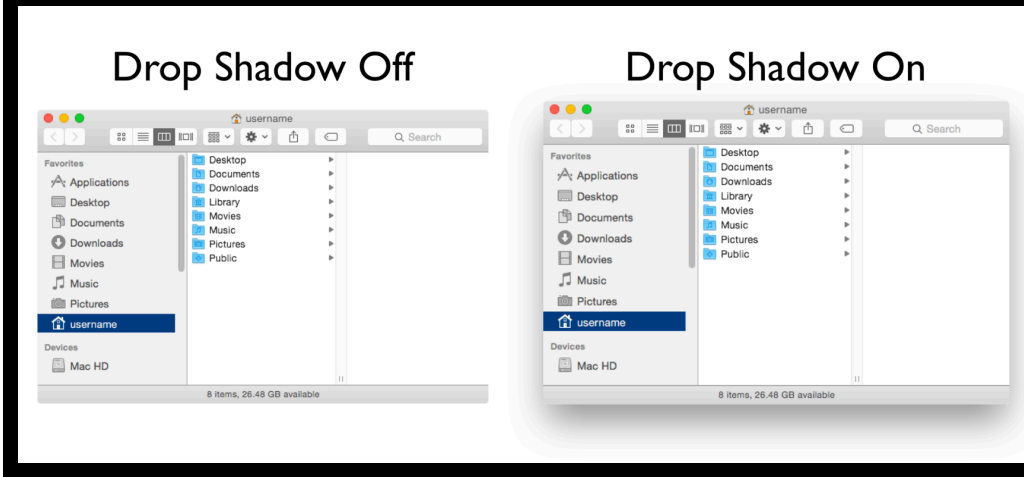
Screenshots

Command (⌘)-Control-Shift-3	
Command (⌘)-Control-Shift-4	
Command (⌘)-Control-Shift-4, then selecting space bar	

One advantage to this method is that saving screenshots into the clipboard will allow you to paste directly into a new document.

Screenshots

Drop Shadow, or No Drop Shadow?



By default, when you take a screenshot of a window with Command–Shift–Four–Spacebar, a drop shadow is added to the image. However, you can turn this drop shadow on or off as you need to. My own preference is to have the drop shadow off.

Screenshots

Changing Screenshot Settings

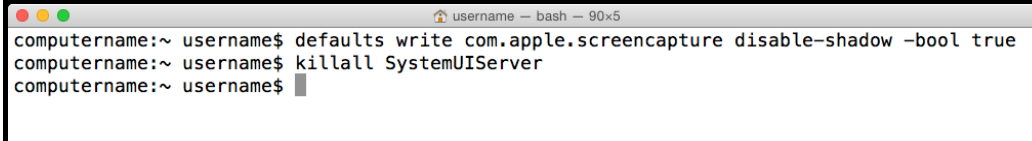
Setting	Key	Parameters	Example
Screenshot format	type	image format: Options: <u>jpg</u> , <u>tiff</u> , <u>pdf</u> , <u>png</u> , <u>bmp</u> or <u>pict</u>	defaults write com.apple.screencapture type png Results: Screenshots saved as PNG image format files
Location to save screenshots	location	directory: <u>/Users/username/Desktop</u>	defaults write com.apple.screencapture location ~/Pictures Results: Screenshot files saved to the Pictures folder in the user's home folder.
Drop Shadow	disable-shadow	-bool true: disable <u>-bool false</u> : enable	defaults write com.apple.screencapture disable-shadow -bool true Results: Drop shadow not being applied to screenshots
Filename	name	Filename: <u>"Screen Shot"</u>	defaults write com.apple.screencapture name "Screen Capture" Results: Screenshot filenames similar to "Screen Capture 2014-11-02 at 2.24PM"

Default formats and settings are underlined.

To turn on and off the drop shadow, you can use the defaults command in the Terminal to edit the screenshot settings for the drop shadow. In fact, you can use OS X's default commands to make a number of changes to how screenshot files are saved.

Screenshots

Applying Screenshot Setting Changes

A screenshot of a macOS terminal window. The window has a title bar with three colored buttons (red, yellow, green) on the left and a title "username - bash - 90x5" on the right. The terminal text shows a user at a computer prompt running two commands: "defaults write com.apple.screencapture disable-shadow -bool true" and "killall SystemUIServer".

```
computername:~ username$ defaults write com.apple.screencapture disable-shadow -bool true
computername:~ username$ killall SystemUIServer
computername:~ username$
```

killall SystemUIServer

Once you've made changes to the screen capture settings using the defaults command, you'll need to restart the SytemUIServer process, which is the part of the OS responsible for doing things like taking screenshots and drawing drop shadows. To do this, run the killall SytemUIServer command shown on the screen. That will cause the SytemUIServer process to restart and pick up the new screenshot settings.

SCREENCAPTURE(1)	BSD General Commands Manual	SCREENCAPTURE(1)
NAME		
screencapture — capture images from the screen and save them to a file or the clipboard		
SYNOPSIS		
screencapture [-SWCTMPcimsxto] file		
DESCRIPTION		
The screencapture utility is not very well documented to date. A list of options follows.		
-c Force screen capture to go to the clipboard.		
-C Capture the cursor as well as the screen. Only allowed in non-interactive modes.		
-i Capture screen interactively, by selection or window. The control key will cause the screen shot to go to the clipboard. The space key will toggle between mouse selection and window selection modes. The escape key will cancel the interactive screen shot.		
-m Only capture the main monitor, undefined if -i is set.		
-M Open the taken picture in a new Mail message.		
-o In window capture mode, do not capture the shadow of the window.		
-P Open the taken picture in a Preview window.		
-s Only allow mouse selection mode.		
-S In window capture mode, capture the screen instead of the window.		
-t <format> Image format to create, default is png (other options include pdf, jpg, tiff and other formats).		
-T <seconds> Take the picture after a delay of <seconds>, default is 5.		
-w Only allow window selection mode.		
-W Start interaction in window selection mode.		
-x Do not play sounds.		
-a Do not capture attached windows.		
-r Do not add screen dpi meta data to captured file.		
files where to save the screen capture, 1 file per screen		
BUGS		
Better documentation is needed for this utility.		
SECURITY CONSIDERATIONS		
To capture screen content while logged in via ssh, you must launch screencapture in the same mach bootstrap hierarchy as loginwindow:		
<pre>PID=\$(pid of loginwindow) sudo launchctl bsexec \$PID screencapture [options]</pre>		
HISTORY		
A screencapture utility first appeared in Mac OS X v10.2.		
Mac OS	June 16, 2004	Mac OS

For those who want to capture screenshots via the command line, OS X also includes a screencapture command line tool. The options for that are shown on the screen. For those in the audience who can't read this, this may be a good time to remind everyone that this will be available for download afterwards.


```
username — less — 71x16
BUGS
  Better documentation is needed for this utility.

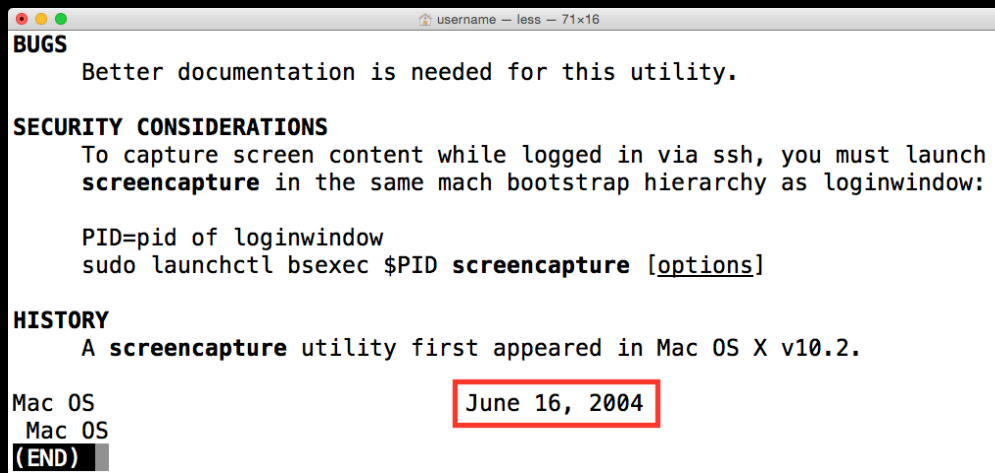
SECURITY CONSIDERATIONS
  To capture screen content while logged in via ssh, you must launch
  screencapture in the same mach bootstrap hierarchy as loginwindow:

  PID=pid of loginwindow
  sudo launchctl bsexec $PID screencapture [options]

HISTORY
  A screencapture utility first appeared in Mac OS X v10.2.

Mac OS                                     June 16, 2004
Mac OS
(END)
```

One particular thing I always find funny is that this documentation tool has a bug report right in the documentation that better documentation is needed.



A screenshot of a macOS terminal window with a title bar that reads "username — less — 71x16". The terminal displays a bug report for the "screencapture" utility. The report is structured with sections: "BUGS", "SECURITY CONSIDERATIONS", and "HISTORY". The "BUGS" section states that better documentation is needed. The "SECURITY CONSIDERATIONS" section explains that to capture screen content via SSH, the utility must be launched in the same Mach bootstrap hierarchy as the login window, and provides a command: "sudo launchctl bsexec \$PID screencapture [options]". The "HISTORY" section notes that the utility first appeared in Mac OS X v10.2. At the bottom, the text "Mac OS" is repeated twice, and "(END)" is shown in a box. A red rectangular box highlights the date "June 16, 2004".

```
BUGS
  Better documentation is needed for this utility.

SECURITY CONSIDERATIONS
  To capture screen content while logged in via ssh, you must launch
  screencapture in the same mach bootstrap hierarchy as loginwindow:

  PID=pid of loginwindow
  sudo launchctl bsexec $PID screencapture [options]

HISTORY
  A screencapture utility first appeared in Mac OS X v10.2.

Mac OS
Mac OS
(END)
```

Meanwhile, as proof that merely wishing for better documentation doesn't make it so, this bug report is apparently still open over ten years later.

Tools






Text Editor

Graphics Editor

Video Editor

When it comes to writing documentation, you'll need at the very minimum a text editor. You may also want to use a graphics editor and a video editor. All of these needs have a variety of tools available, and I'm not going to tell you which ones to use because everybody's going to have different ones that work well for them. It's also been my experience that people feel strongly about the right tool for them, which is another reason why I'm not going to tell anyone which tools to use.

Tools

Text Editor	 
Graphics Editor	
Video Editor	 

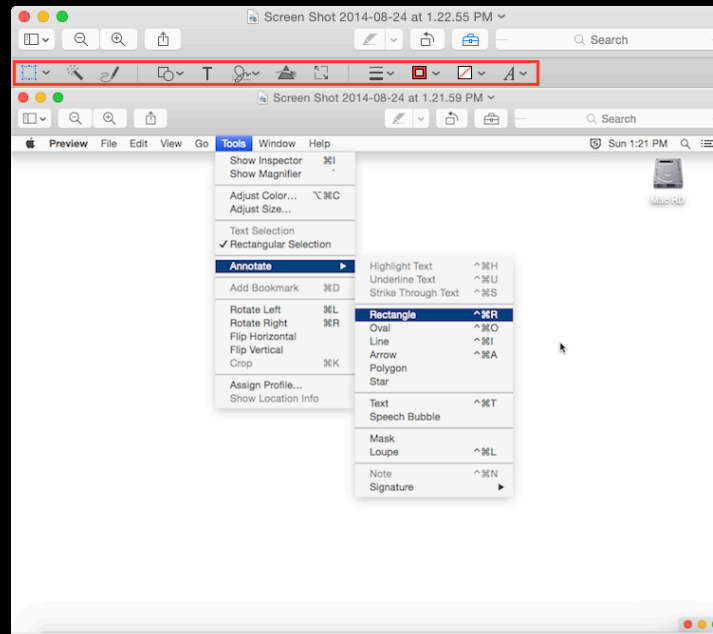
Here are the tools that I use. For text editing, I use Apple's TextEdit or Bare Bones Software's TextWrangler. For my graphics editing, I normally just use Apple's Preview as it has what may be a surprising number of options when it comes to editing screenshots. When I'm shooting video, I use an application called Screeny that I purchased from the App Store to take screencapture videos and then use Apple's Quicktime Pro to edit them.

Preview: Screenshot Editor Extraordinaire



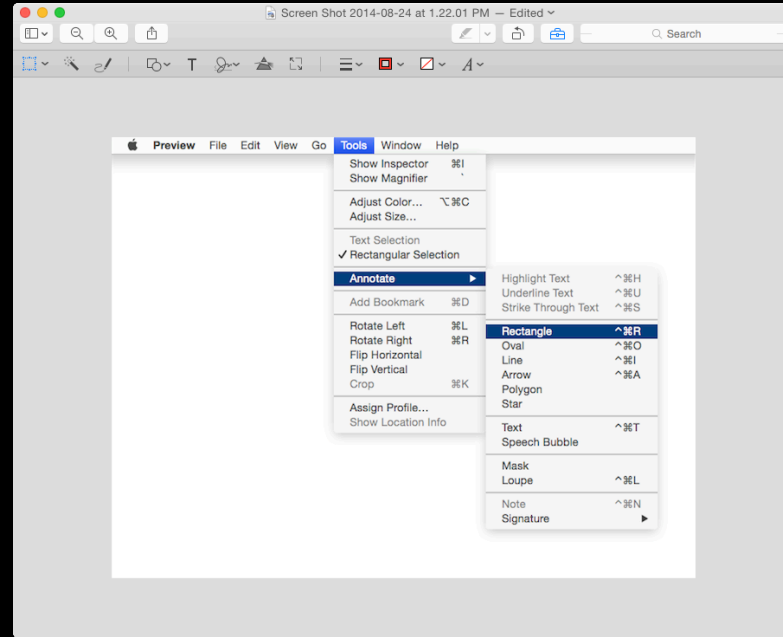
Preview in particular is an indispensable tool for me. While most folks don't use it for more than viewing PDFs and images, it's also got a lot of options for editing and refining images as well.

Preview: Screenshot Editor Extraordinaire



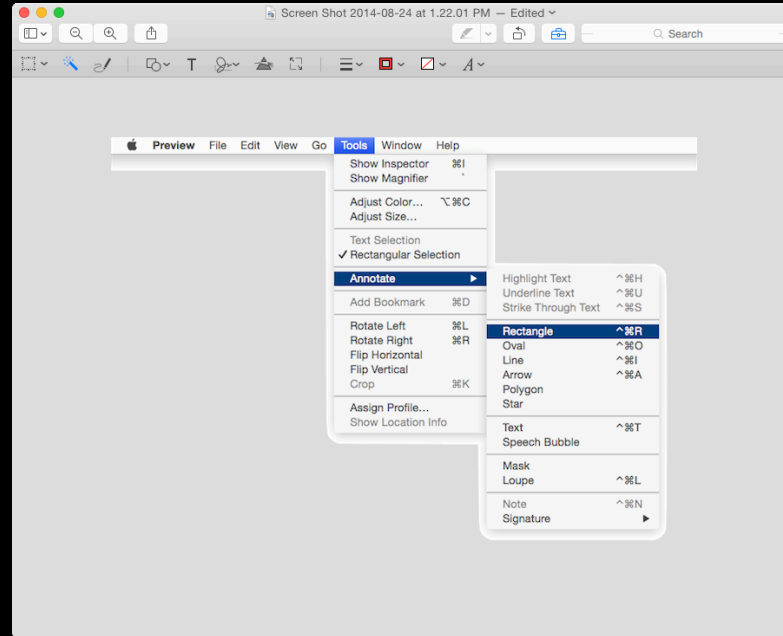
To access its versatile array of tools, open a screenshot file and click on the Edit Toolbar button, which has a toolbox icon in Yosemite.

Preview: Screenshot Editor Extraordinaire



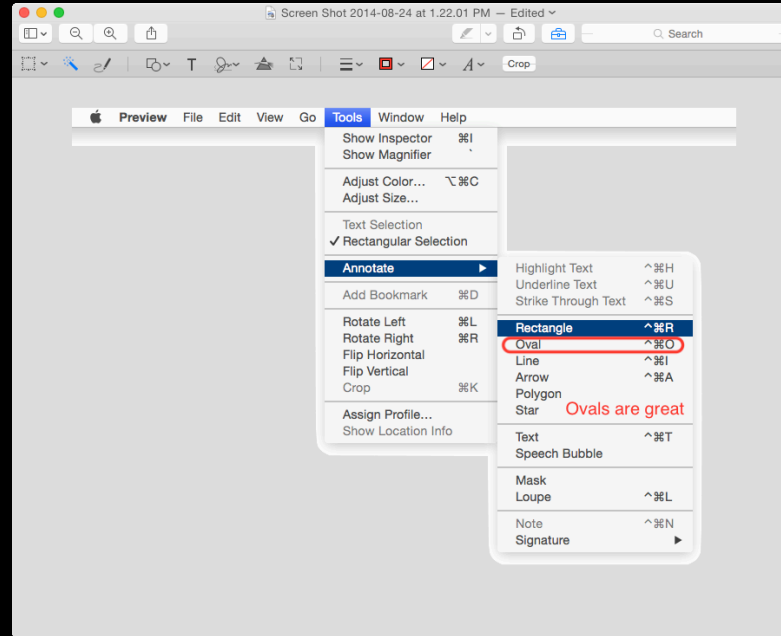
Among its various tools is the ability to crop images, to remove unwanted details from a screenshot.

Preview: Screenshot Editor Extraordinaire



The instant alpha tool in Preview can be used to remove unwanted backgrounds.

Preview: Screenshot Editor Extraordinaire



Preview can also edit images to add things like shapes and text.

Preview: Screenshot Editor Extraordinaire



<http://tinyurl.com/osx-preview-tutorial>

This only scratches the surface of how useful Preview has been to me when it comes to editing screenshots. For a more in-depth tutorial, I encourage you to check out the link on the screen.

Screeny - Simple Screencapture Videos



<http://screenyapp.com>

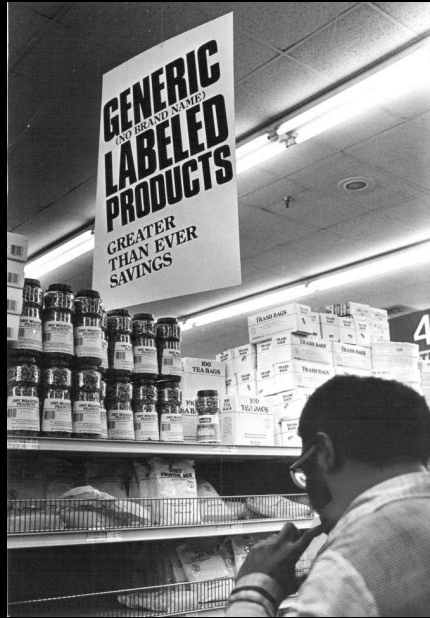
Another tool that I've come to rely on is Screeny, a simple-to-use application for creating screencapture videos.

Screeny - Simple Screencapture Videos



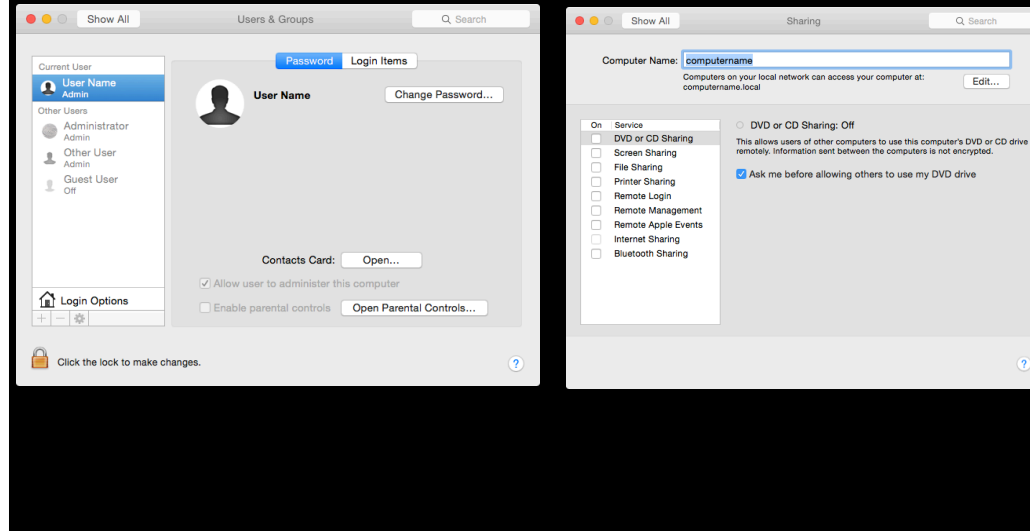
Among the reasons I like it is that it records to Quicktime movie files, it has simple controls and allows me to select how much of the screen I want to record. This has been especially useful to me because it allows me to record what's happening in a virtual machine and keep a tight focus on just the parts that I want to include in the video. As an added bonus, since I bought it via Apple's App Store, it's also easy for me to install it wherever I need it.

Genericizing



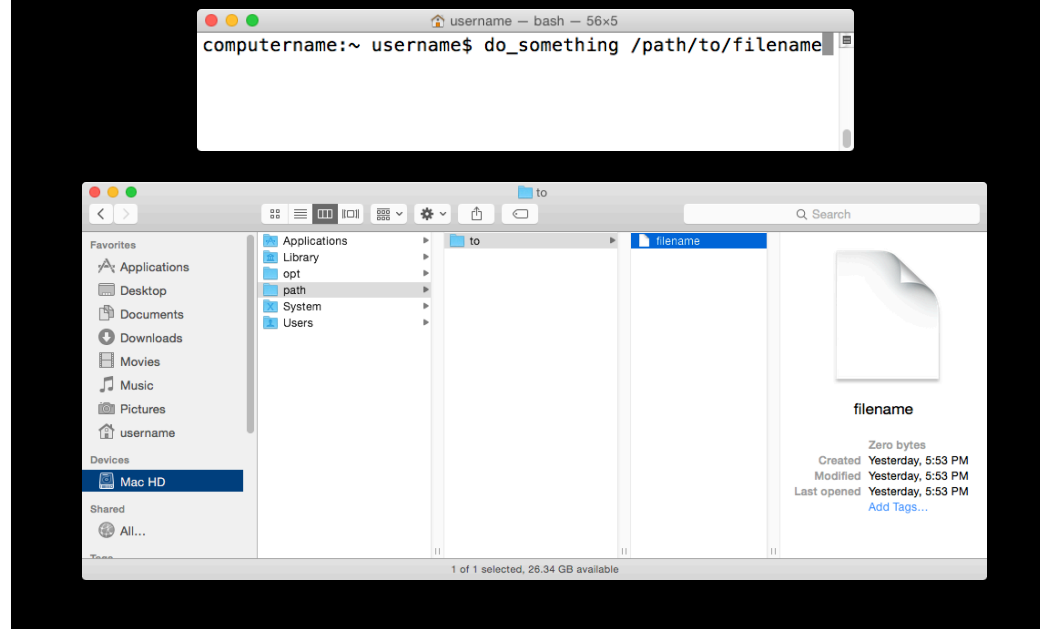
Another important part of my documentation method is what I call genericizing. This is the fine art of removing as much specific identifying information from your documentation as possible, while hopefully still making it clear what you mean.

Genericizing



A good example of this is creating a user account which is named “username” and having the computer be named “computername”.

Genericizing



One other thing I like to do to genericize is to remove path names in the event that the directory path I'm referring to in the documentation will be different for everyone. A good example of that is when referring to a directory path that will likely include the person's username. Instead, I replace the actual path with `"/path/to"`. When writing the documentation, it's often useful for an actual directory structure correspond to that, so I will create a folder called "path" and a folder inside named "to".

The reason I do this is two-fold and is to accomodate two different groups of people. Folks who are experienced with computers should be able to look at that and automatically substitute in the values that they need to put in. Folks who are not experienced with computers but blindly follow directions they found on the internet will get "file not found" errors instead of potentially breaking their machines.

Genericizing

```
/var/log/firstbootpackageinstall.log

2014-08-23 08:11:43 Checking for active network connection.
2014-08-23 08:11:43 =====
2014-08-23 08:11:43 This check will automatically run every five
2014-08-23 08:11:43 seconds until an active network connection is
2014-08-23 08:11:43 detected or until sixty minutes have passed.
2014-08-23 08:11:43 =====
2014-08-23 08:12:18 Active network connection detected. Proceeding.
2014-08-23 08:12:18 /Users/Shared/fb_installers present on Mac
2014-08-23 08:12:18 Found 2 packages to install.
2014-08-23 08:12:18 Please be patient. This process may take a while to complete.
2014-08-23 08:12:18 Once all installations are finished, this machine will automatically reboot.
2014-08-23 08:12:18 Installing /Users/Shared/fb_installers/00/Failed Install.pkg on this Mac.
2014-08-23 08:12:51 FAILURE: /Users/Shared/fb_installers/00/Failed Install.pkg did not install correctly.
2014-08-23 08:12:51 Installing /Users/Shared/fb_installers/01/Successful Install.pkg on this Mac.
2014-08-23 08:13:23 SUCCESS: /Users/Shared/fb_installers/01/Successful Install.pkg has been successfully installed.
2014-08-23 08:13:23 Finished with all installations.
2014-08-23 08:13:23 If any installations were reported as not installing correctly,
2014-08-23 08:13:23 please check /var/log/install.log on this Mac for details.
2014-08-23 08:13:23 Removing /Users/Shared/fb_installers from this Mac.
2014-08-23 08:13:23 Restarting Mac.
```

Another method I've developed is to name things for their expected result. A good example of that is this screenshot I took for a tool I've developed called First Boot Package Install. In this case, I was documenting that I had added error logging to display information on installations that had failed. To help demonstrate the new logging, I built two installers. The first was named "Failed Install" and had a deliberate problem with it to make sure the installer tool would report failure. The other was named "Successful Install" and designed to work fine. When I went to take the screenshot, "Failed Install" showed an installation failure and "Successful Install" showed as having succeeded.

Genericizing



All of the genericizing is generally a lot of work to do in advance, so I've been leveraging automation whenever possible. I'm now using automated deployment tools like DeployStudio with virtual machines to quickly spin up a ready-to-go machine with the usernames, computer names and other settings that I need to make the screenshots and screencapture videos I want when I need them.

Media



Last but not least, documentation has to be posted somewhere.
Hopefully, not in this fashion.

Media

Wikis	 
Blogs	
Other	 

As with creating documentation, there are a number of tools used to post electronic documentation. For my own shop, I'm using a Confluence wiki and a Sharepoint site to host internal documentation. My blog is hosted up on Wordpress.com, so I'm using Wordpress there.

In those instances where I'm not posting to either a wiki or a blog, my usual tool is to use Microsoft Word to write and format it and then use OS X's built-in PDF creation tool to create a PDF of it. That way, I have the documentation that I wrote in Word available in both Word and PDF format.

Media

Why Make a PDF?

Universal Readability	PDF viewing applications are available for all OSs produced since the late 1990s.
Font and Image Encapsulation	PDF files include all the document information, including the text, fonts, graphics, and other information needed to display it.
Open Specification	The PDF specification is an open standard, published as ISO 32000-1:2008

In the last case, where I'm writing it up in Word, why am I also making a PDF? The reason is twofold. First, I know all platforms will be able to read the PDF, as PDF viewers are available for every modern OS. Second, because PDFs encapsulate not just text but also images, fonts and formatting, I know that the person receiving the PDF will be able to read it exactly the way that I wrote it. I can't make that assumption about someone who receives a Microsoft Word document from me.



*"I wrote them down in my diary so that I wouldn't have to remember."
Professor Henry Jones, Indiana Jones and the Last Crusade.*

Finally, I'd like to close with a tribute to the fictional character who first got me interested in documentation, Dr. Henry Jones, Senior. Dr. Jones Senior was exhaustively thorough when it came to documentation and it was because he wanted to free up his brain for the Grail Quest and not have to spare mental cycles to remember that Saint Whatisname liked oatmeal instead of mutton for breakfast. Meanwhile, when Indy needed it most, Dad had made sure that all the needed details were in the Grail diary. This allowed Indy to save the day by reading the documentation as well as drawing on his own knowledge of their shared field.

**DOCUMENT ALL
THE THINGS!!!!**



Getting into the habit of documentation has helped me immensely in many ways and being able to help others through what I've documented has been rewarding, both professionally and personally.

If you've been inspired to start documenting all the things, hopefully I've now equipped you with some ideas and tools that will help you to do so.

Downloads

PDF available from the following link:

<http://tinyurl.com/MT2014DocPDF>

Keynote slides available from the
following link:

<http://tinyurl.com/MT2014DocKeynote>