

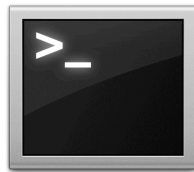
InstaDMG 1.6rc1



Mountain Lion Quickstart:
*(See Page 5 for **Snow Leopard** Version)*

step 1

Open Terminal,



and paste the following*:

svn checkout <http://instadmng.googlecode.com/svn/trunk> instadmng

step 2

open /Applications/Install\ OS\ X\ Mountain\ Lion.app/Contents/SharedSupport

Drag InstallESD.dmg to  /instadmng/InstallerFiles/BaseOS

step 3

sudo instadmng/AddOns/InstaUp2Date/instaUp2Date.py 10.8_vanilla --process

When it says 'Done', you'll see the fully patched image in

 /instadmng/OutputFiles

*for 10.8 you'll need to install svn, e.g. from [here](http://afp548.com/forums/forum/software/instadmng/)

Overview

Quickstart Explained

Steps 1 & 2, SVN through finished preparation	3
(Step 2 Details)	4
Step 3 - one last command before results	5

For 10.6 and earlier, rewind to step 2

Lightweight Architectural Discussion

InstaUp2Date	5
Recovery Partition	6

AddOns

CreateUserPkg.app	6
Wait, how do I incorporate it in the creation of an image?	6
DisableSetupAssistant.pkg	7

Appendix

Community/Discussion Resources	7
Other Documentation Sources	7
Complimentary Tool Resources	7
Mac Management Presentations	7

Overview

When setting up many new computers or refreshing existing workstations, it is important to have a 'known good state' you can rely on. Imaging commonly refers to the process of overwriting the entire contents of a disk, which improves consistency and can reduce the amount of steps we would otherwise have to perform manually.

Instead of the 'Golden Master' method of manually configuring everything on a specific piece of hardware (with the inherent possibly of introducing human error,) and then worrying about cleaning that image before distributing it, InstaDMG uses a programmatic method that generates 'hardware-agnostic' disk images (e.g. you can use the same image for many different Mac models.) Updates, software, and/or other customizations and enhancements can be [slipstreamed](#) (loosely speaking) into a baseline image through the addition of installer packages, which can be added to images in a 'Modular' manner.

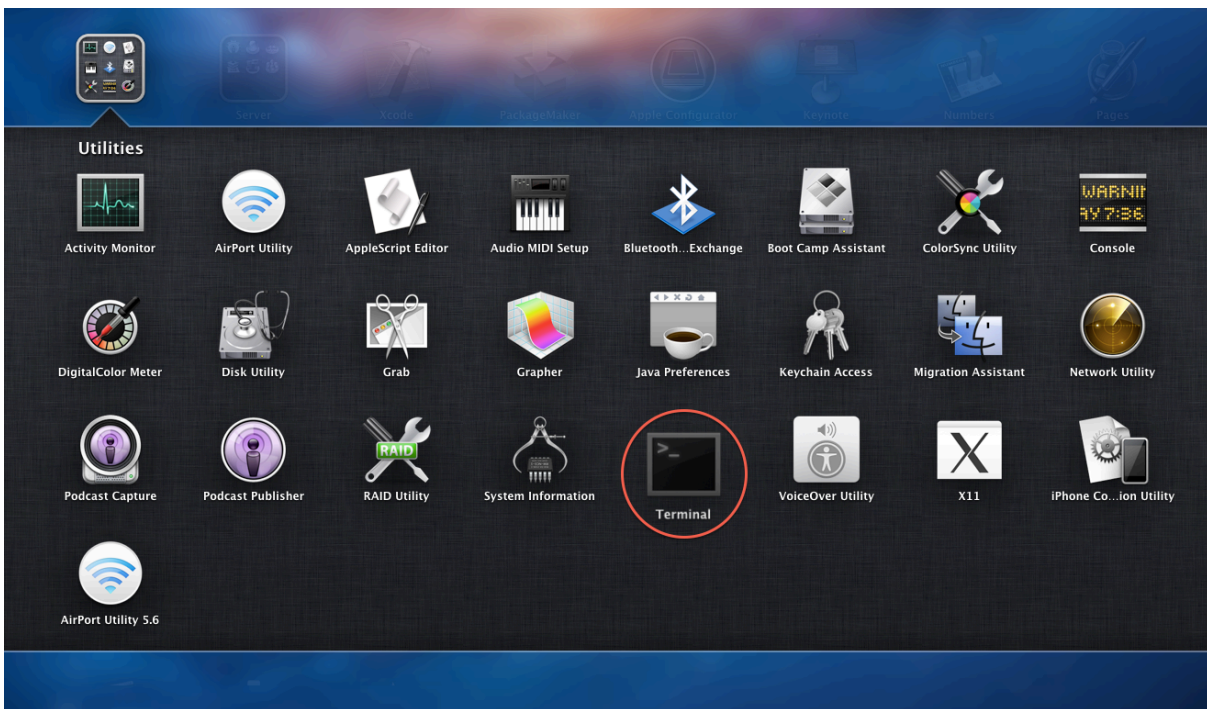
Many of the common questions of how to effectively apply those customizations, that could also be performed via patch management or other maintenance procedures, has been attacked and in many ways solved by the community on the [afp548.com forums](http://afp548.com/forums/). We won't be covering all of the common challenges and techniques we would encounter, instead focusing on an overview of all of the parts of InstaDMG that can help sysadmins get the baseline they need.

Before we move forward, though, let's walk through the quickstart InstaDMG usage scenario on the first page.

Quickstart Explained

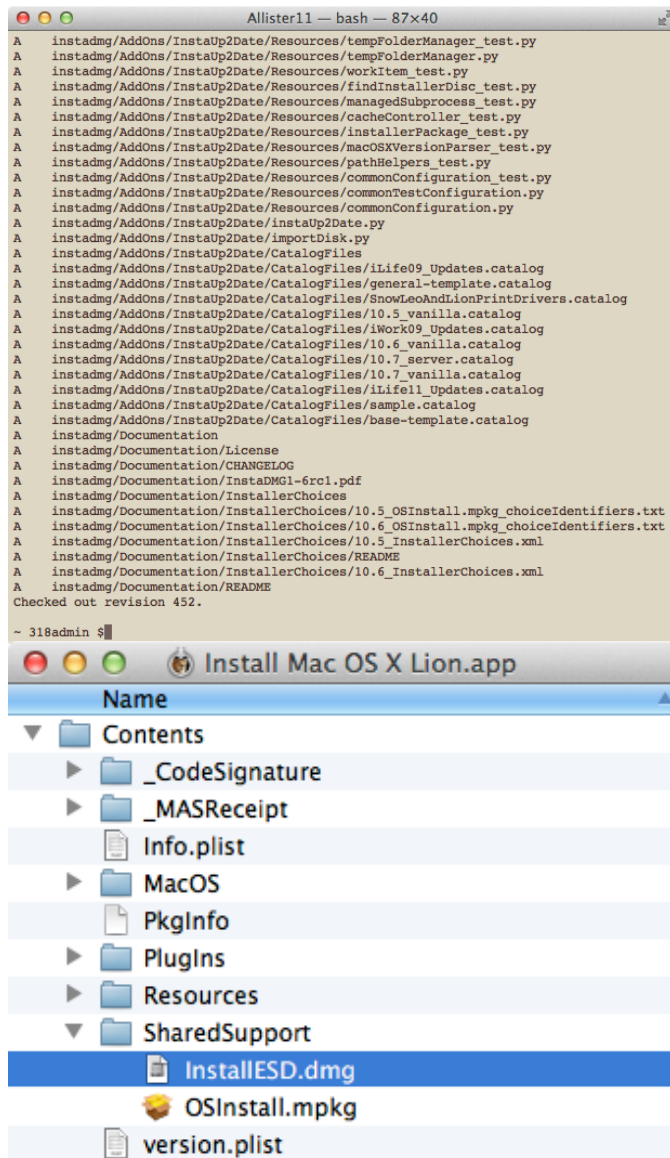
Steps 1 & 2, SVN through finished preparation

To get started, we want the newest possible version of the InstaDMG software, along with its often-maintained helper files. Like most software, keeping track of versions is entrusted to an application like git or svn. These are commonly interacted with using a command line interface, so we open the Terminal application from the Utilities folder, its location in Lion is pictured below:



```
svn checkout http://instadmg.googlecode.com/svn/trunk instadmg
```

You may need to install svn first, which you can find [here](#). When we copy the above line into Terminal and hit return, lines of text start filling the screen, eventually looking something like the picture on the following page on the left. What is actually happening is a new folder called “instadmg” is created, with the files and folders you see on the following page on the right.

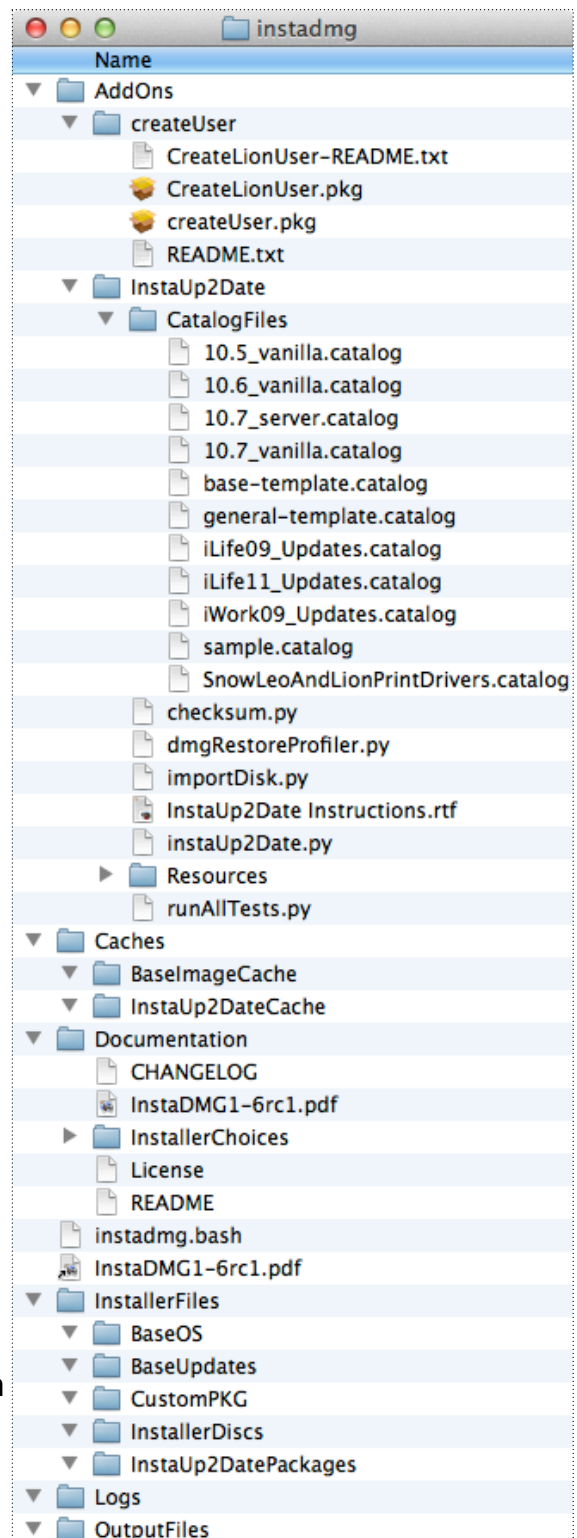


(Step 2 Details)

We can't actually do much without the install 'media' for Lion, which you would have downloaded from the [Mac App Store](#). (Hardware-specific builds can be obtained [like this](#), but let's move on for now.)

Inside of the application 'bundle' (meaning folder structure) for "Install Mac OS X Mountain Lion.app", we dig out the part that's useful for our purposes, `InstallESD.dmg`, as pictured above.

Once `InstallESD.dmg` is moved into the `BaseOS` folder, we're ready to move to the next and final step.



Step 3 - one last command before results

```
sudo instadmg/AddOns/InstaUp2Date/instaUp2Date.py 10.8_vanilla --process
```

Along with everything else pulled down by our original “svn checkout” command, you may have noticed the files that end in ‘.catalog’ on the previous page. Those are ‘catalogs’ of software patches that can be queued up for InstaDMG to use when building the image.

That queueing up is accomplished by this second command we’ve run, which uses the helper application `instaUp2Date.py` with the `-p` option to go through with the *process* of generating a fully updated image (otherwise it would just download and cache the updates listed for later use.) `InstaUp2Date` is always fed a catalog file to work with, and then triggers the transition from the `instaUp2Date` functionality (which pulls down the updates specified in the `10.8_vanilla.catalog` file,) to the running of our key player, the `instadmg` bash script. The end result will be a ready-to-restore dmg in the `./instadmg/OutputFiles` folder, which depending on how recently the `10.8_vanilla` catalog was updated will be fully patched and compatible with many if not all of the hardware models that OS supports.

For 10.6 and earlier, rewind to step 2

Remember optical media? Well we want to access the installer a little faster than ~30MB/s, since the process of image creation is rather I/O bound. For 10.6 and earlier OS’s, a tool called `importdisk.py` was created to simplify this process. This changes step 1.5 from our quickstart to the following:

```
sudo ./instadmg/AddOns/InstaUp2Date/importDisk.py --automatic
```

After prompting you for your (currently-logged-in admin) password, this prepares the inserted installer disk (DVD) by creating a dmg of it and placing that in the `./instadmg/InstallerFiles/InstallerDiscs` folder. The “automatic” option makes it skip prompting you for which disk you’d like to import, since we’re assuming there is only one installer disk mounted. Expect this step to take around 45 minutes or so. You’d then run the applicable catalog file like so:

```
sudo ./instadmg/AddOns/InstaUp2Date/instaUp2Date.py 10.6_vanilla --process
```

Lightweight Architectural Discussion

InstaUp2Date

As so many community members now collaborate on catalog files and other methods of taking care of software and settings, `InstaUp2Date` is clearly the future. When it comes to the stability of the software, it should be considered that fixes aren’t usually prioritized for things

outside of the quickstart workflow presented. Please pardon and be patient with the devs, as they are in a transition period moving the project to pure python.

Recovery Partition

InstaDMG was faced with the option of not supporting the recovery partition generated by the install process on 10.7 and greater, but the difficulties involved generating updated versions of that recovery image made that option worth going through the trouble of shoehorning into the pre-existing process. This does not require a special process to restore, as deployment tools that are powered by the asr command line binary just put the recovery partition in place after the main image. This could be undesirable in certain instances, for example if dual booting Windows, so that can currently be disabled manually in the instadmg.bash script by modifying line 48 (as of this writing) to read DMG_FS=HFS+ instead of DMG_FS=HFS+J

AddOns

CreateUserPkg.app

One of the most common and obvious things to customize about an image built with InstaDMG is to have an admin user account pre-existing, although the contents of the actual home folder are unnecessary. CreateUserPkg can be found on the [MacAppStore](#), but is also bundled directly with the InstaDMG project due to the grace of its author, Per Oloffson. It is remarkably straightforward to use, but for more information please consult the [Github project page](#).

Wait, how do I incorporate it in the creation of an image?

The following workflow would allow you to incorporate any desired package into an InstaUp2Date run. First, you need to 'fingerprint' the pkg you'd like to incorporate with the checksum.py tool. For example, if you save your CreateUserPkg-generated pkg to the root of your home folder, the command would look something like the following:

```
./instadmg/AddOns/InstaUp2Date/checksum.py create_ladmin-1.0.pkg
```

Terminal will then spit back something like this:

```
create_ladmin-1.0      create_ladmin-1.0.pkg sha1:ad378ac3d18e64.....
```

You would then copy-paste that line (making sure TRUE tabs exist at the beginning of the line and between each section - python is picky like that) into your vanilla catalog directly, or use a 'nesting' method described in the base-template catalog file included in the instadmg checkout, an example can be found [here](#).

DisableSetupAssistant.pkg

Once you start customizing a user account, you probably don't want a new admin created when the setup assistant runs. While not currently included with the InstaDMG project, we leave it as an exercise to the reader to find more info on [how to accomplish that](#).

Appendix

Community/Discussion Resources

The AFP548 forum has traditionally been the place to ask questions or collaborate on techniques, although it's been very quiet in recent times. It is still around for those that prefer that format: <http://afp548.com/forums/forum/software/instadmg/>

For general discussion and immediate feedback, an IRC chat room exists for those that are savvy, the channel name is **##osx-server** on Freenode

The Mac Enterprise mailing list has traditionally been a great place to discuss things as well: <http://lists.psu.edu/archives/macenterprise.html>

Other Documentation Sources

Another take on explaining InstaUp2Date and the common workflow steps surrounding InstaDMG can be found here: http://afp548.com/mediawiki/index.php/Main_Page

Here's a recent three-part discussion of InstaDMG that some may find helpful: <http://www.coolestguyplanetech.com/create-clean-image-os-x-mountain-lion-10-8-instadmg/>

Complimentary Tool Resources

The preferred method to install trickier software or patch an out-of date image is Munki, also known as Managed Software Update. More details about bootstrapping an image with it can be found here: <https://code.google.com/p/munki/wiki/BootstrappingWithMunki>

To deploy your image, a tool like DeployStudio is recommended. Quick start and architecture guides for it can be found here: <http://www.deploystudio.com/Doc/Doc.html>

Mac Management Presentations

Penn State University has hosted a conference for various IT professionals covering many recent practices in Mac management. <http://youtube.com/user/psumacconf/>
Previously on iTunes: <http://itunes.apple.com/itunes-u/penn-state-mac-admins-podcasts/id429872255>