

Deployment

MacTech BootCamp II

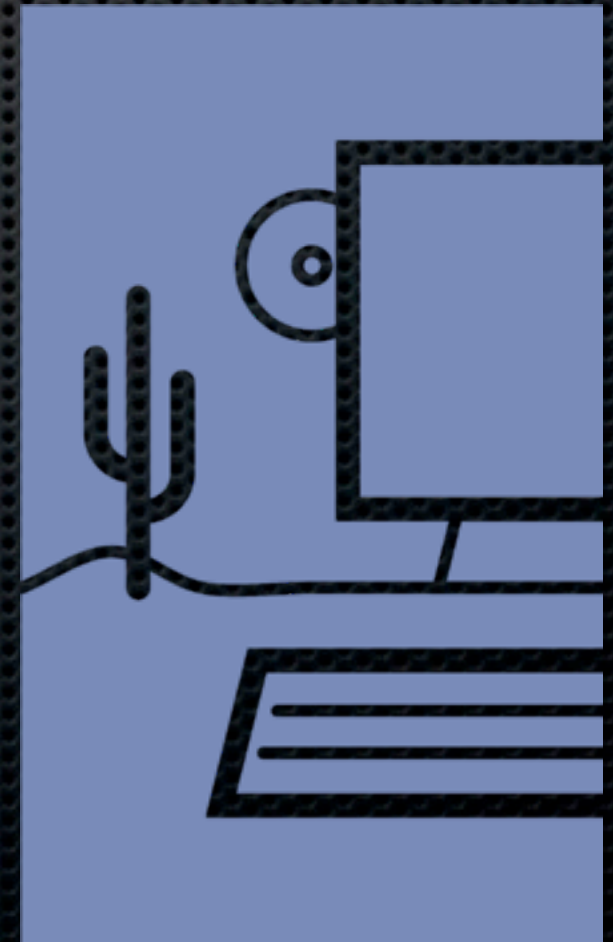
Dallas, 2013-04-17

Zack Williams

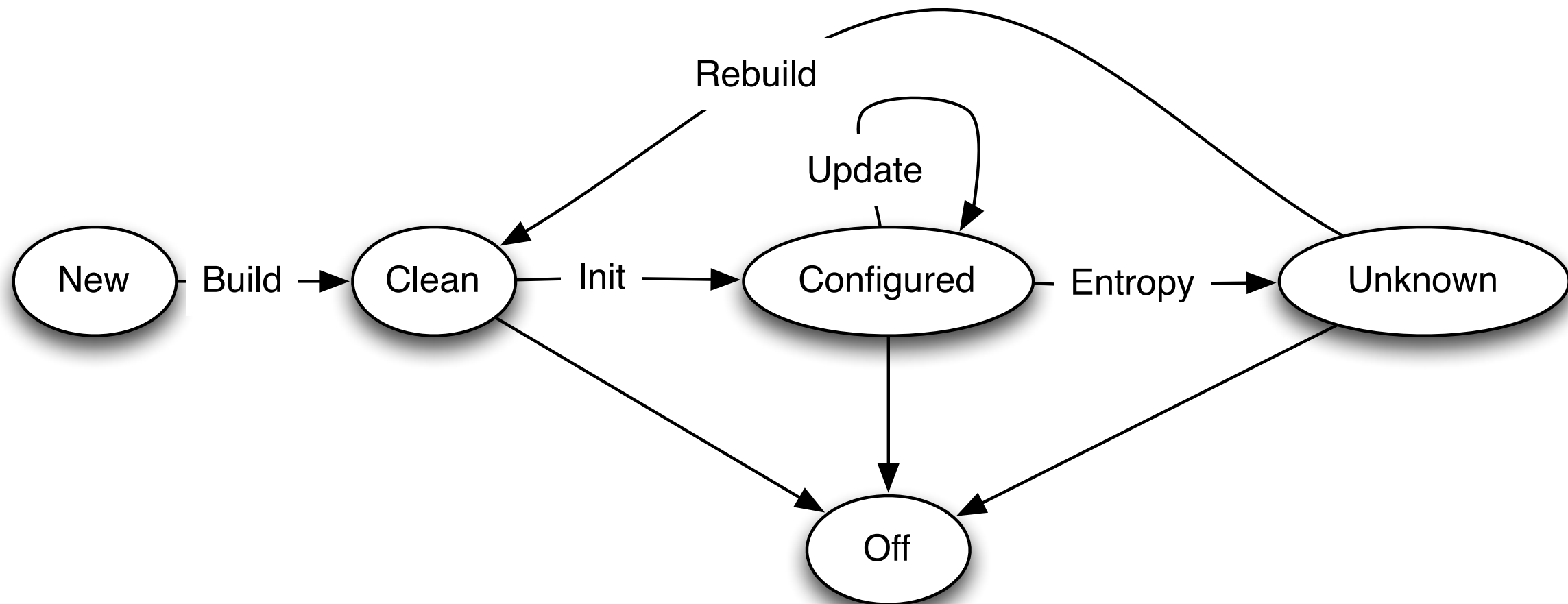
Artisan Computer

zdw@artisancomputer.com

@zdw



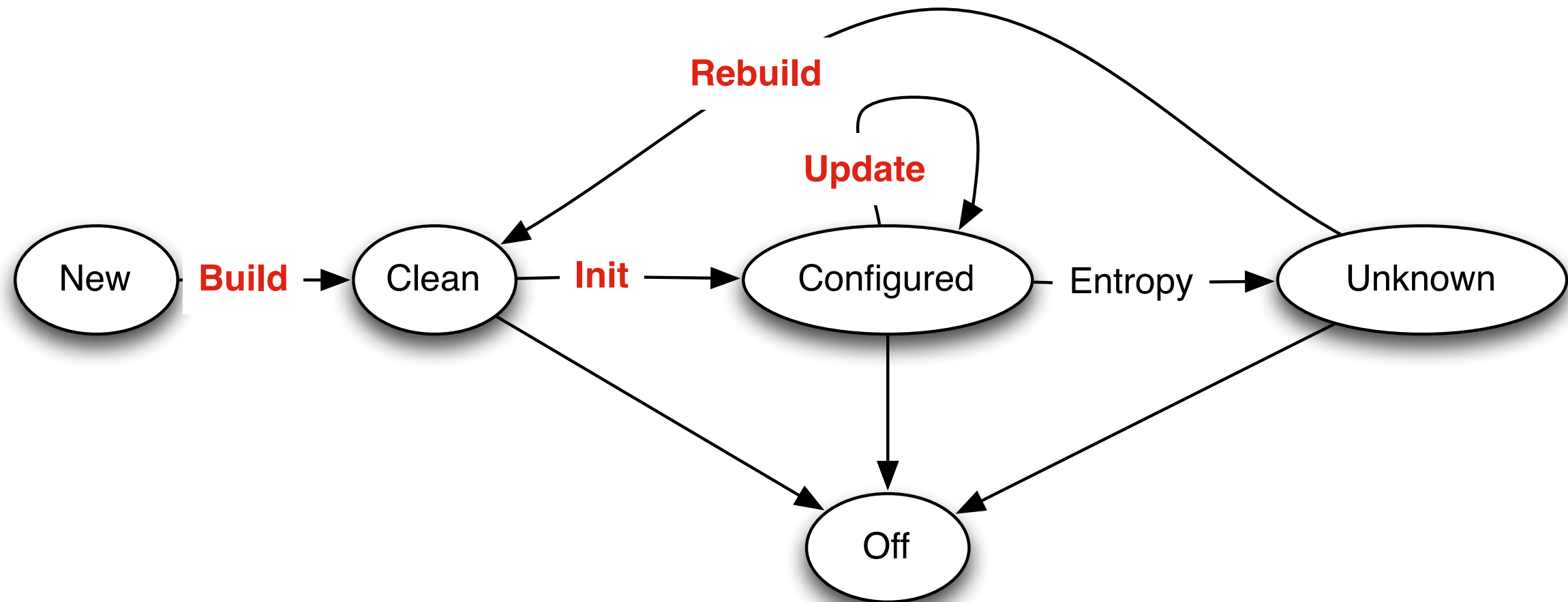
Device Lifecycle



Adapted from: "An analysis of Unix System Configuration", Rémy Evard, 1997

Referenced in "The Practice of System and Network Administration, 2nd ed.", Limoncelli, Hogan, Chalup 2007

Deployment Is:



Adapted from: "An analysis of Unix System Configuration", Rémy Evard, 1997

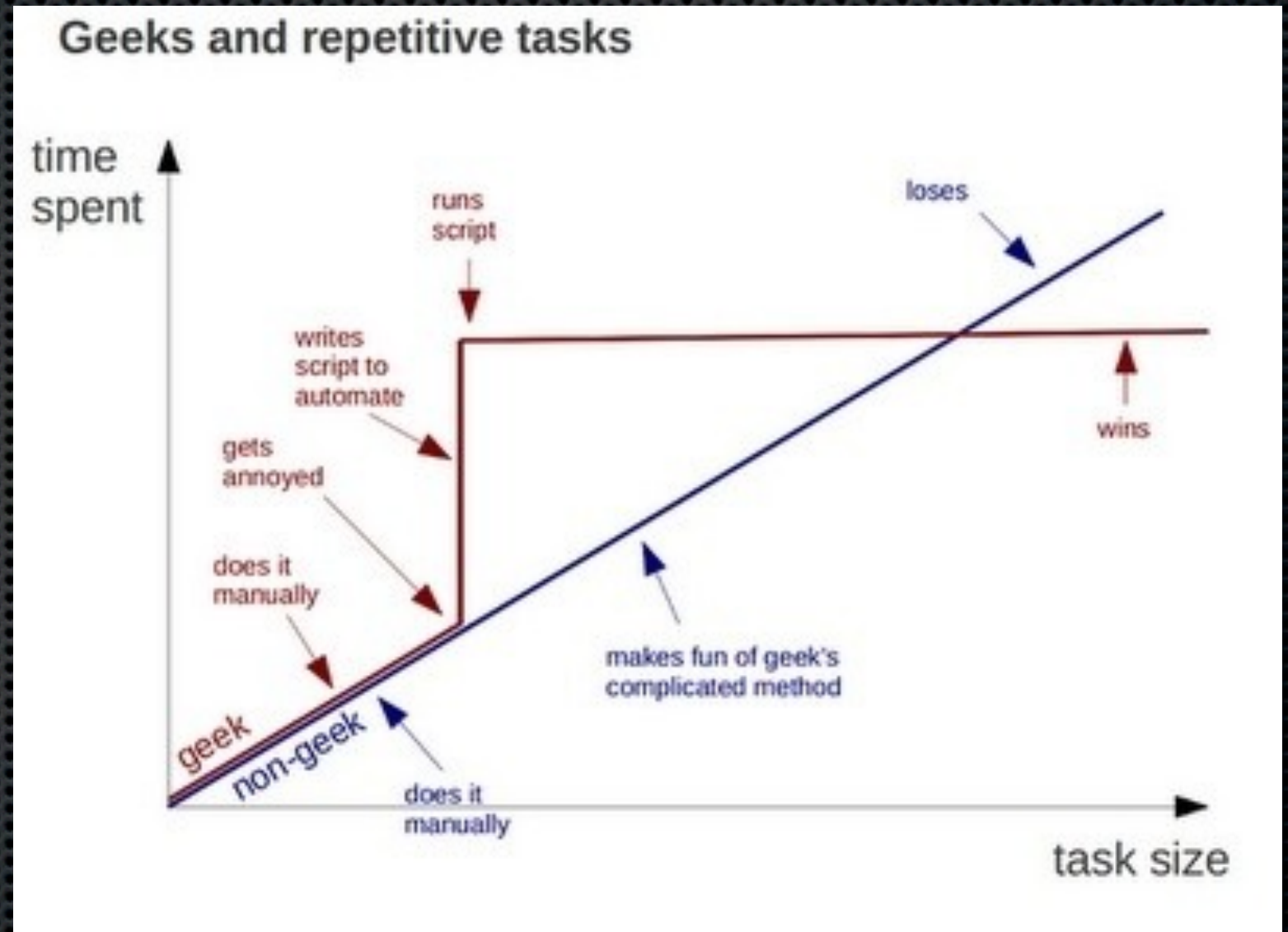
Referenced in "The Practice of System and Network Administration, 2nd ed.", Limoncelli, Hogan, Chalup 2007

Deployment Is:

- ✦ Build/Rebuild
 - ✦ Physical Setup, Install OS
- ✦ Init
 - ✦ Apply configuration
 - ✦ Install 3rd party software
- ✦ Update
 - ✦ Apply updates to software

Why Deploy?

- ✦ Repeatability
- ✦ Fewer Errors
- ✦ Eventual, massive time savings



“Golden Master” Deployment

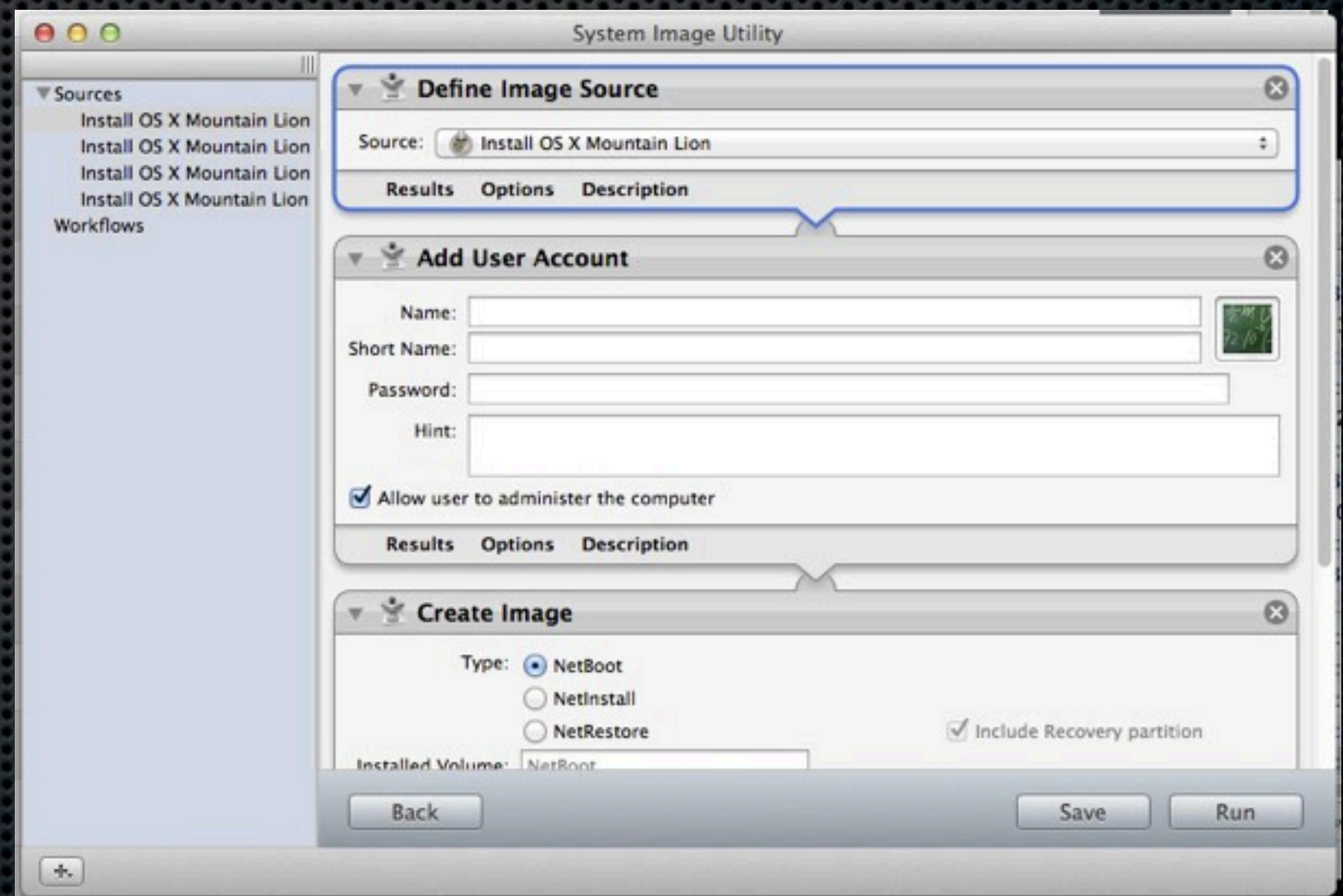
- ✦ “Perfect” computer -> image -> deployed
- ✦ Common in Windows and Virtualization
- ✦ Problems:
 - ✦ Any updates = new master
 - ✦ Need a master per role
 - ✦ Compatibility issues with new HW
 - ✦ Big disk images, long restores

Modular Imaging

- ✦ Lay down base OS
- ✦ Install automation tools, which:
 - ✦ Apply configuration
 - ✦ Install additional software
- ✦ New Software/OS?
Replace just that component!

Building OS Images

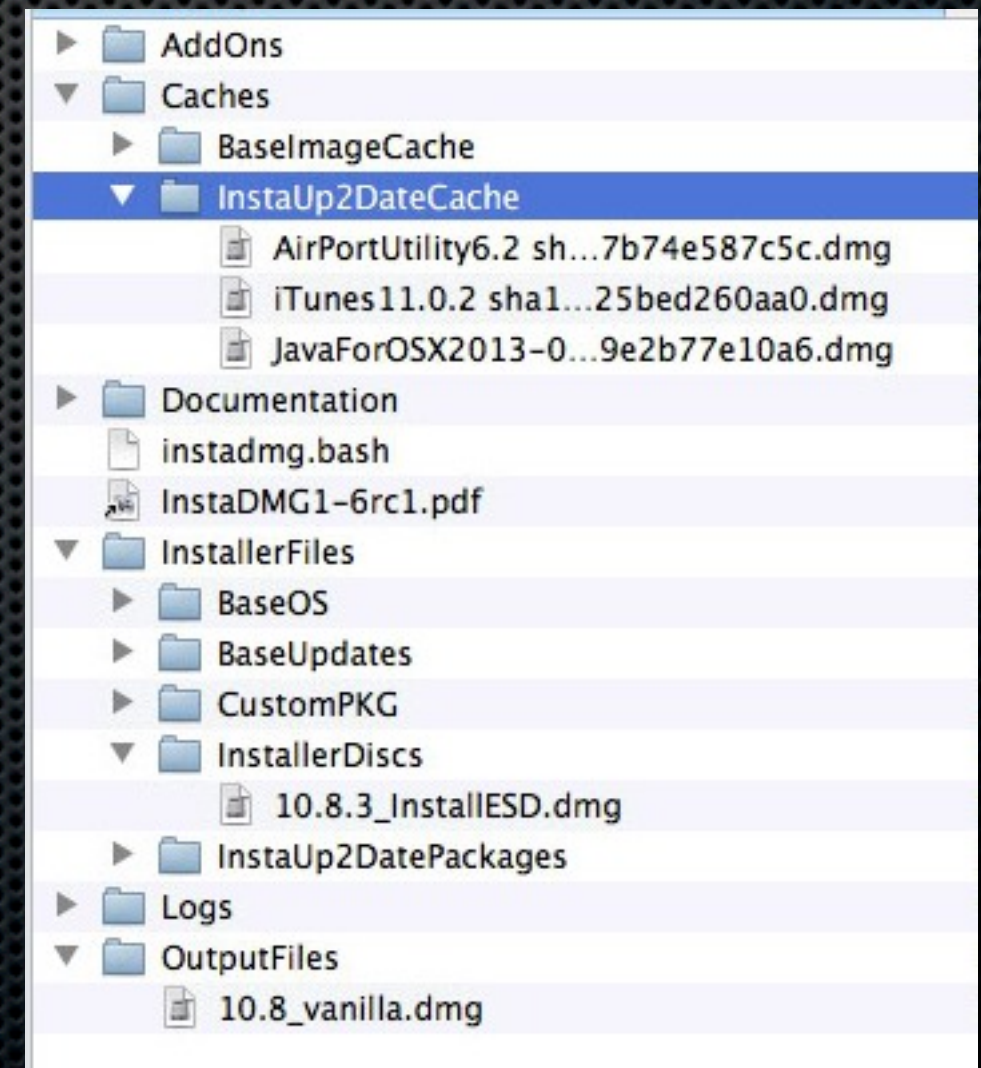
- ✦ System Image Utility
- ✦ InstaDMG
- ✦ Cloned drives



InstaDMG

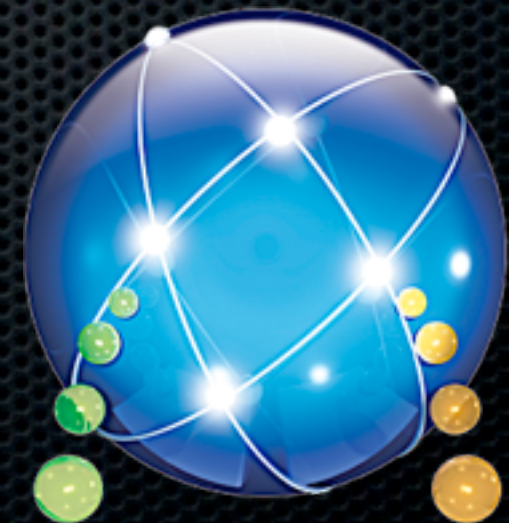


- ✦ CLI tool
- ✦ Builds updated, unbooted OS disk images
- ✦ Preps for deployment



Lay the Foundation

- ✦ NetInstall/System Image Utility
- ✦ DeployStudio



DeployStudio

- ✦ Requires OS X Server (Open Directory)
- ✦ Workflows to:
 - ✦ Capture/Restore disk images
 - ✦ Partition drives
 - ✦ Apply settings
 - ✦ Run scripts

 Active Directory binding Binds a computer to an Active Directory server.	 Alert Pauses the workflow by displaying an alert message.
 Automatic enrollment Enrolls a computer to a MDM server at first boot.	 Configure Applies computer's settings defined in the DeployStudio database.
 Disk image Creates a disk image of a volume or partition.	 File copy Copies files and folders to a volume.
 Firmware Security Enables Firmware Security at boot.	 Generic Runs a script or any executable file.
 Hostname form Updates computer's name in the DeployStudio database.	 Licenses form Updates computer's Server & Xsan licenses in the DeployStudio database.
 Localization Configures the default system language & keyboard layout.	 Network location form Updates computer's network locations in the DeployStudio database.
 New user form Updates computer's local users list in the DeployStudio database.	 Open Directory binding Binds a computer to an Open Directory server.
 Package install Installs a package to a volume.	 Partition Creates multiple partitions on a specific disk.
 Periodic NetBoot Configures the computer to NetBoot periodically.	 Restore Restores a disk image to a specific partition or disk.
 Shut down Shuts down the computer at the end of the workflow (NetBoot only).	 Software update Customizes the Software update client.
 Time Configures the default network time server & time zone.	 Workflow Inserts all the tasks of another workflow at runtime.

Packaging

- ✦ The software vendor!
- ✦ PackageMaker
- ✦ **pkgbuild/pkgutil**
- ✦ The Luggage
- ✦ Iceberg, JAMF Composer, etc.
- ✦ Can run scripts (type depending on OS version)



The Luggage Example

```
$ cat Makefile
# copy "Final Cut Pro X Unleashed.pdf" to the /Users/Shared directory

include /usr/local/share/luggage/luggage.make

TITLE=install_fcpx_unleashed_pdf
REVERSE_DOMAIN=com.artcs
PAYLOAD=fcpx_unleashed_pdf

prep_Users_Shared:
    @sudo mkdir -p ${WORK_D}/Users/Shared

fcpx_unleashed_pdf: prep_Users_Shared Final_Cut_Pro_X_Unleashed.pdf
    @sudo ${CP} ./Final_Cut_Pro_X_Unleashed.pdf ${WORK_D}/Users/Shared
```


Install 3rd Party Software

- ✦ Push packages with ARD
- ✦ Wrap into “Golden Master”
- ✦ Munki
- ✦ 3rd party solutions (Casper, Absolute Manage, etc.)

Munki



- ✦ Automates 3rd party package installation
- ✦ No Admin rights required
- ✦ Can run **softwareupdate**
- ✦ Self Service app installation
- ✦ Server doesn't need to be OS X
- ✦ <http://code.google.com/p/munki/wiki/HowMunkiDecidesWhatNeedsToBeInstalled>

Munki Example

```
$ munkiimport Office\ Installer.pkg
  Item name [Office Installer]:
  Display name [Microsoft Office for Mac 2011]:
  Description []: Office 2011, Volume Licensing
  Version [14.3.0]:
  Catalogs [testing]:
```

```
  Item name: Office Installer
  Display name: Microsoft Office for Mac 2011
  Description: Office 2011, Volume Licensing
  Version: 14.3.0
  Catalogs: testing
```

```
Import this item? [y/n]
```


Munki Ecosystem

- ✦ automunki
- ✦ macautopkg
- ✦ Simian, and other web interfaces
- ✦ Read MacTech!

Apply Configuration

- ✦ Run commands via ARD
- ✦ Apple Profile Manager, Managed Client (MCX)
- ✦ Configuration Management Systems
- ✦ Payload-free packages
 - ✦ Run scripts, write config files, etc.

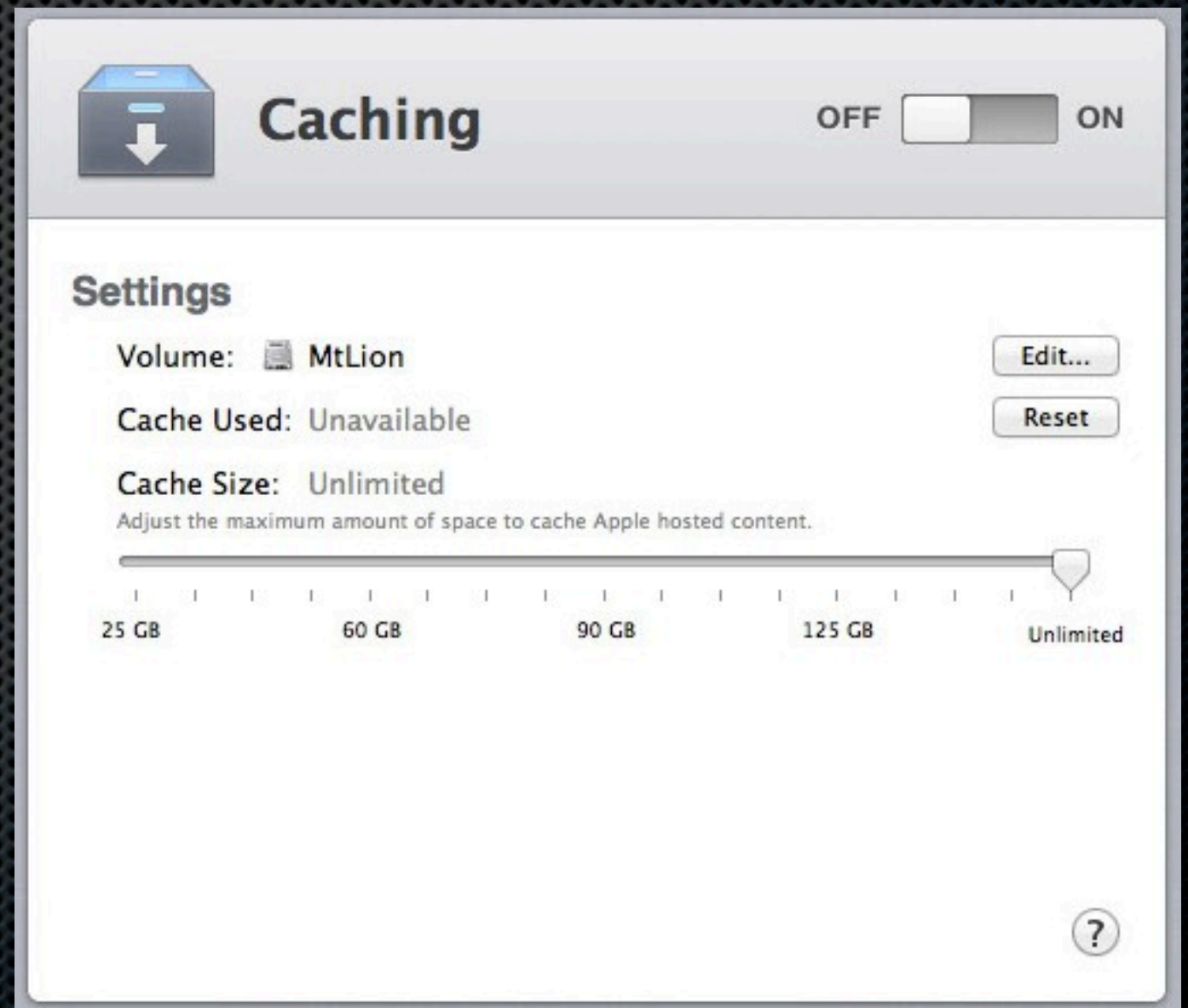
Updating the OS

- ✦ Push out update packages
- ✦ Apple Software Update Server
 - ✦ Requires same or newer OS X Server
- ✦ Reposado
 - ✦ Python script, runs on nearly anything
- ✦ Large space requirement!



Mac App Store?

- ✦ Deployment challenges
- ✦ OS X Server's Caching server



What I use

- ✦ InstaDMG to create OS disk image
- ✦ DeployStudio to lay down image, configure Munki
- ✦ Munki to customize installation, apply updates
- ✦ Reposado to provide OS updates

Best Practices

- ✦ Get Volume Licensing
- ✦ Test your deployments
 - ✦ Virtual Machines are great... but still test on real HW
- ✦ Have a rollback plan