

# Enterprise-Friendly Software Development

The image shows the Roy E. Disney Animation Building at Walt Disney Animation Studios. The building is a large, multi-story structure with a prominent blue conical tower in the center featuring a white crescent moon and stars. To the right, there is a curved section with red and white vertical stripes. The main entrance is flanked by red brick pillars. The foreground is a paved plaza with a checkered pattern. A large green star sculpture is visible on the right side of the building.

ROY E. DISNEY ANIMATION BUILDING

**Greg Neagle**  
Sr. Systems Engineer  
Walt Disney Animation Studios

**Why?**

**Profit.**

# Enterprise-friendly:

- Sell more copies/seats to “enterprise”
- Lower your support costs

**= Profit.**

**What is “enterprise”?**

- Banks
- Insurance companies
- Schools
- Universities
- Internet search giants
- Animation studios

any environment with a large number of <sup>managed</sup> Macs



**What is “enterprise-friendly”?**

# What is “enterprise-friendly”?

- Can be easily purchased for large numbers of users
- Can be installed with automated tools
- When installed, works as expected for all users of a machine, even those without admin rights.

# Enterprise purchasing

- Rule #1: Make sure your customers can buy your product

# Enterprise purchasing

- The Mac App Store is not an enterprise-friendly store.
- Make sure your software is available outside the Mac App Store.
- You get to keep 30% more!



# Enterprise purchasing

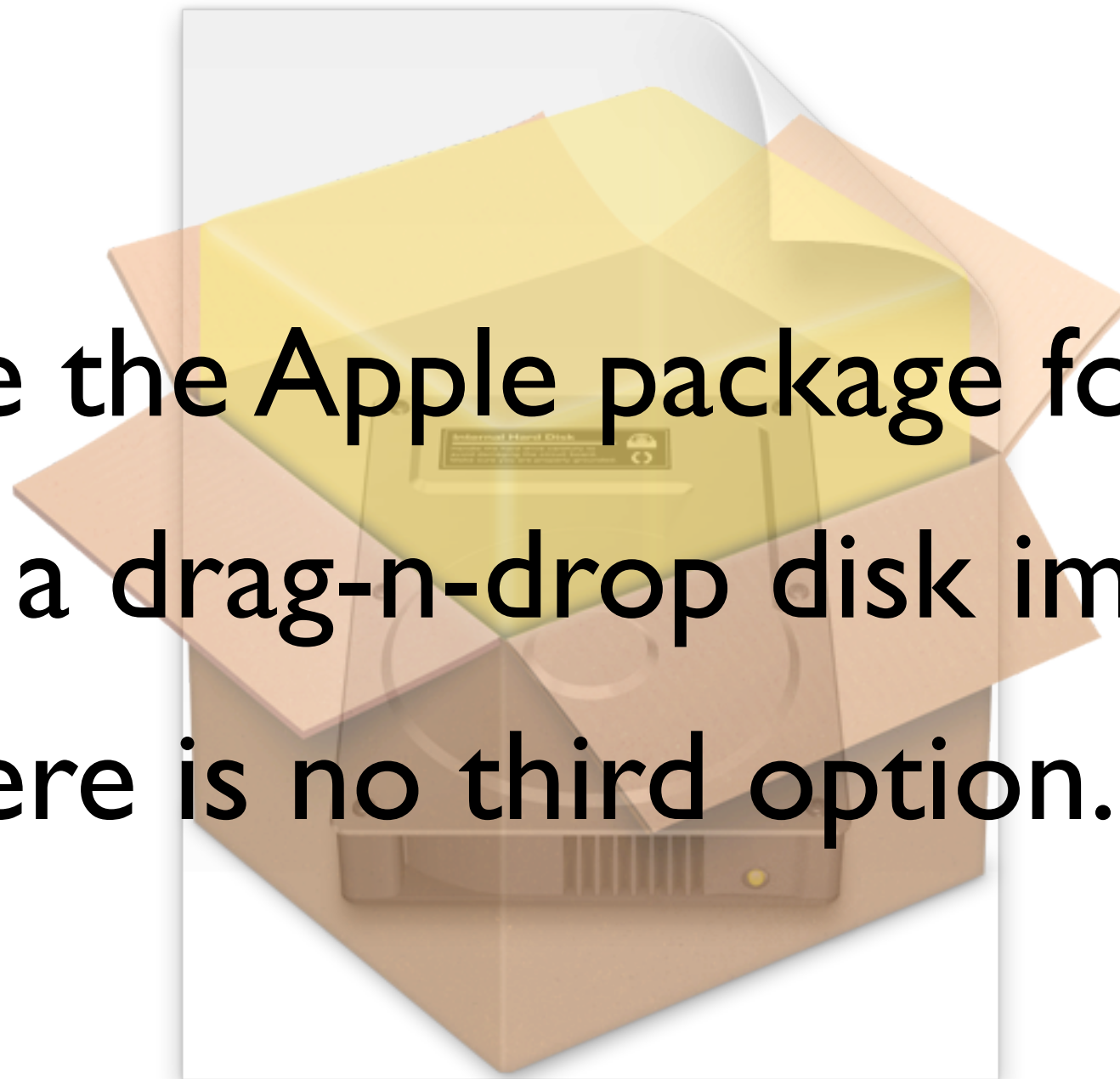
- Extra credit: support “site licenses”

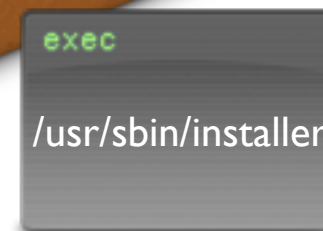
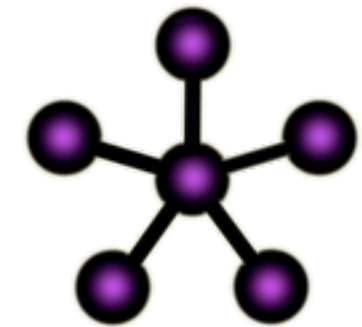
# What is “enterprise-friendly”?

- Can be easily purchased for large numbers of users
- Can be installed with automated tools
- When installed, works as expected for all users of a machine, even those without admin rights.

# Enterprise deployment

- Use the Apple package format
- Or a drag-n-drop disk image
- There is no third option.







**Repackaging considered harmful**

# Repackaging considered harmful

- Extra work for system administrators
- They will do it wrong
- You get to support their mistakes

# Packaging for enterprise

- Simpler is better
- Install all your parts
- Avoid package scripts

# Package script pitfalls

- Assuming there is a logged-in user
- Assuming the user installing the software will be the same user **using** the software
- Assuming you own the Dock, Desktop, etc
- Assuming the software will be installed to the startup disk

# if you must script...

- Use the `$COMMAND_LINE_INSTALL` environment variable
- Don't use `$USER`
- Use the `$3` argument
- Test your script when no-one is logged in

# if you must script...

```
# demonstrate $COMMAND_LINE_INSTALL
if [ "$COMMAND_LINE_INSTALL" = "" ]; then
    open "http://www.example.com/welcome"
fi
```

# please don't...

- Assign ownership to current user
- Launch an application after install
- Install everything as world-writable!

# What is “enterprise-friendly”?

- Can be easily purchased for large numbers of users
- Can be installed with automated tools
- When installed, works as expected for all users of a machine, even those without admin rights.



# What's different?

- Multiple users on one machine
- Directory Services/network users
- Most users may not have admin rights
- Network home directories

# Home folders

- May not be in /Users
- May be encrypted
- May be on network volumes
- May not exist yet!

# Home folders

- use `NSHomeDirectory`
- or `NSHomeDirectoryForUser`
- or even `$HOME`
- don't assume `/Users!`

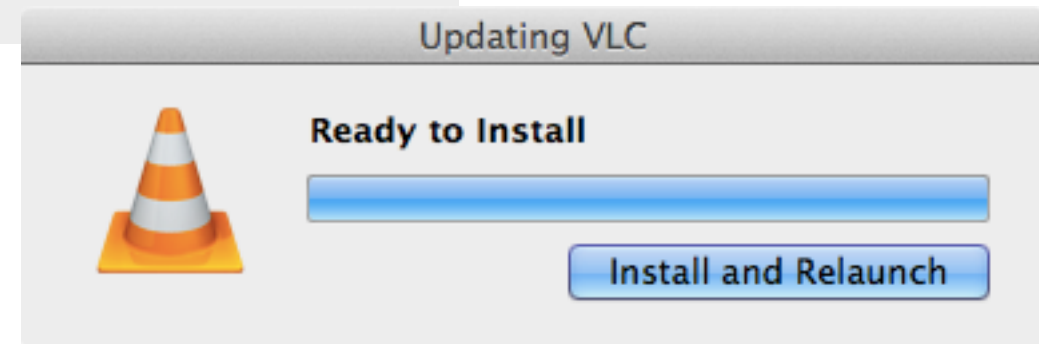
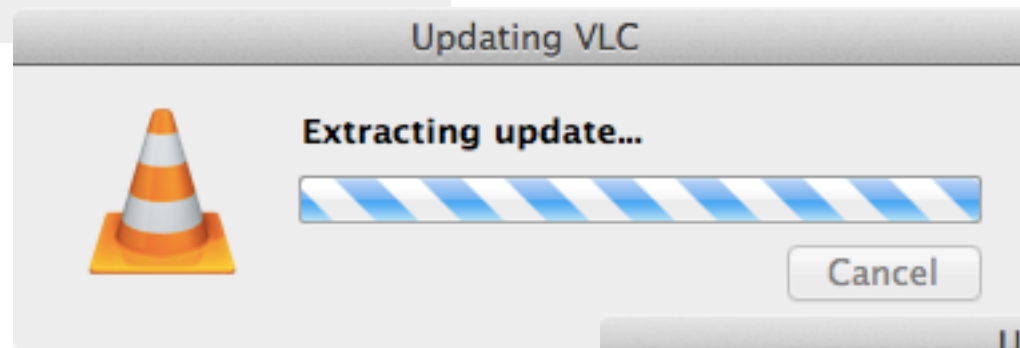
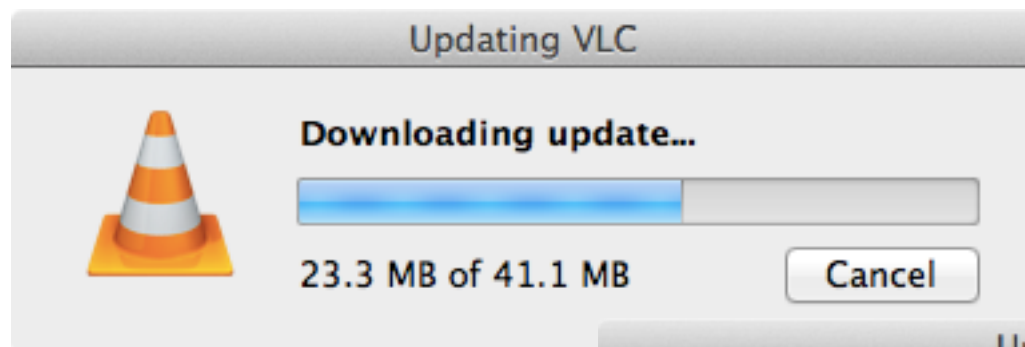
# Licensing/Registration

- Storing this info in user home folder:
  - Not ideal in the enterprise
- Support a global location:
  - /Library/Preferences/
  - /Library/Application Support/

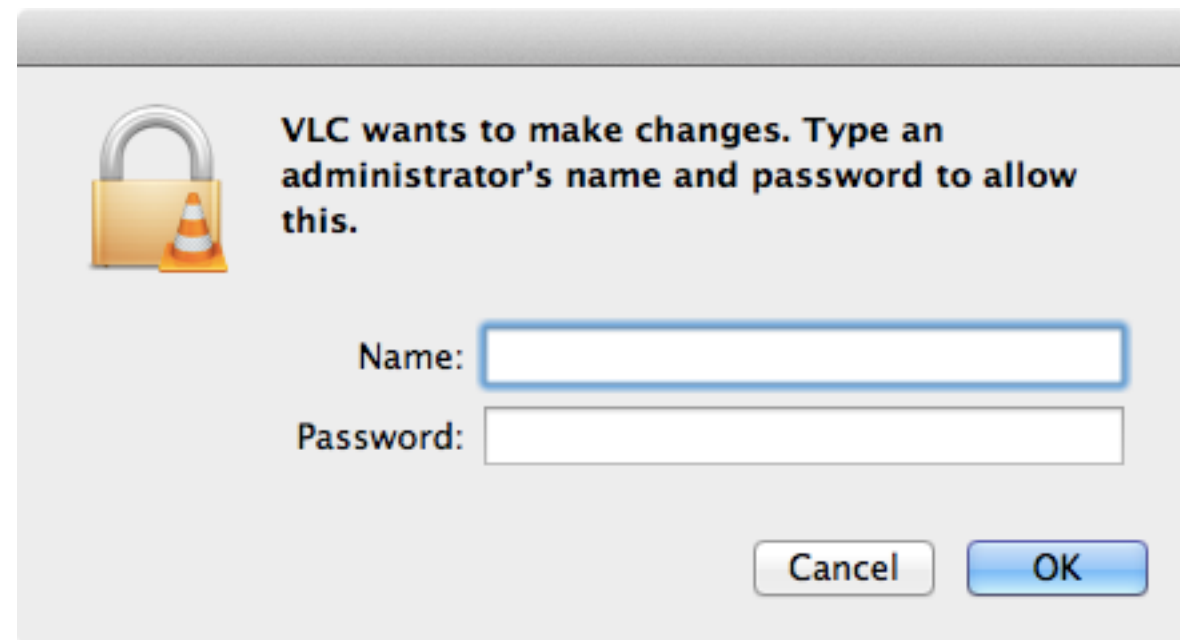
# Auto Updates



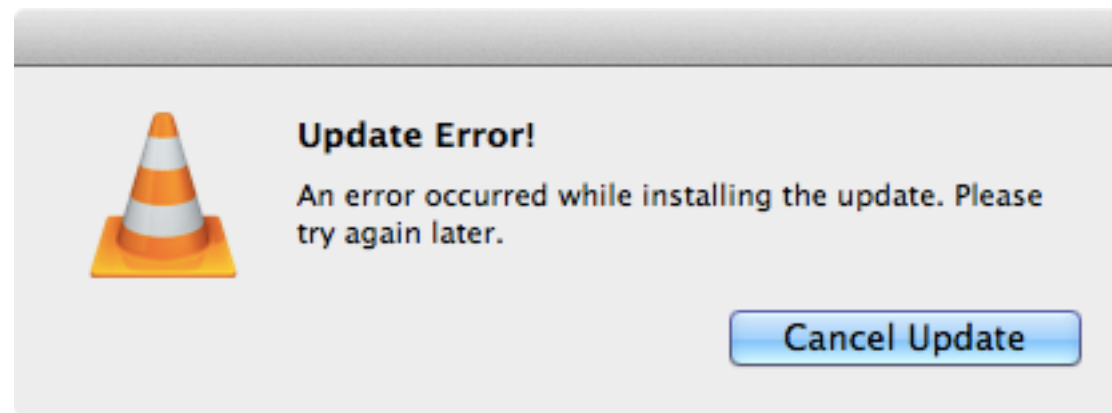
# Auto Updates



# Auto Updates



# Auto Updates





# Auto Updates

- User might not be an administrator!
- Organization may want to control timing
- Let us turn off auto-updates
- Make updates available as packages

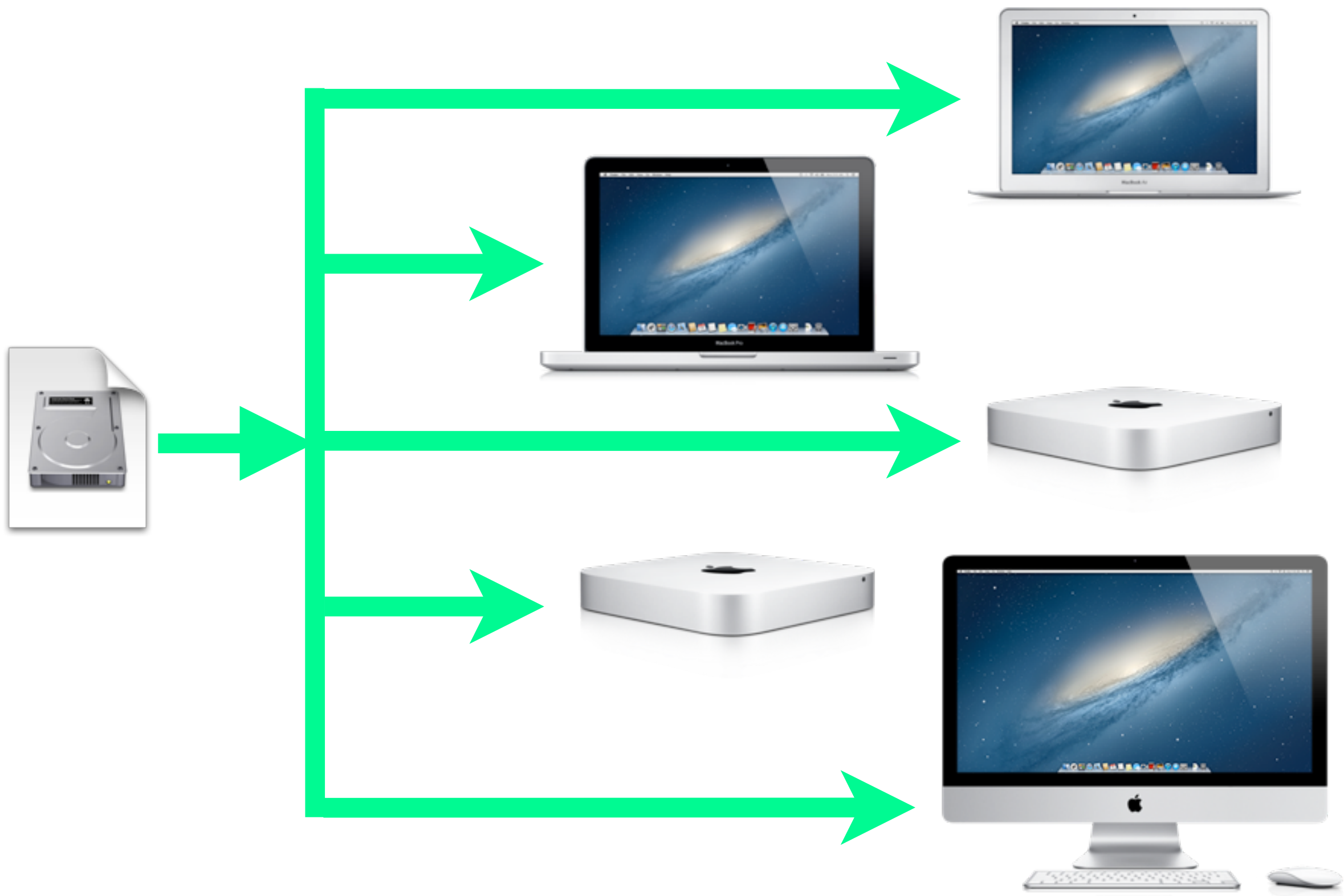
# Extra Credit

- Imaging
- MCX/Profiles

# Modular imaging



# Modular imaging



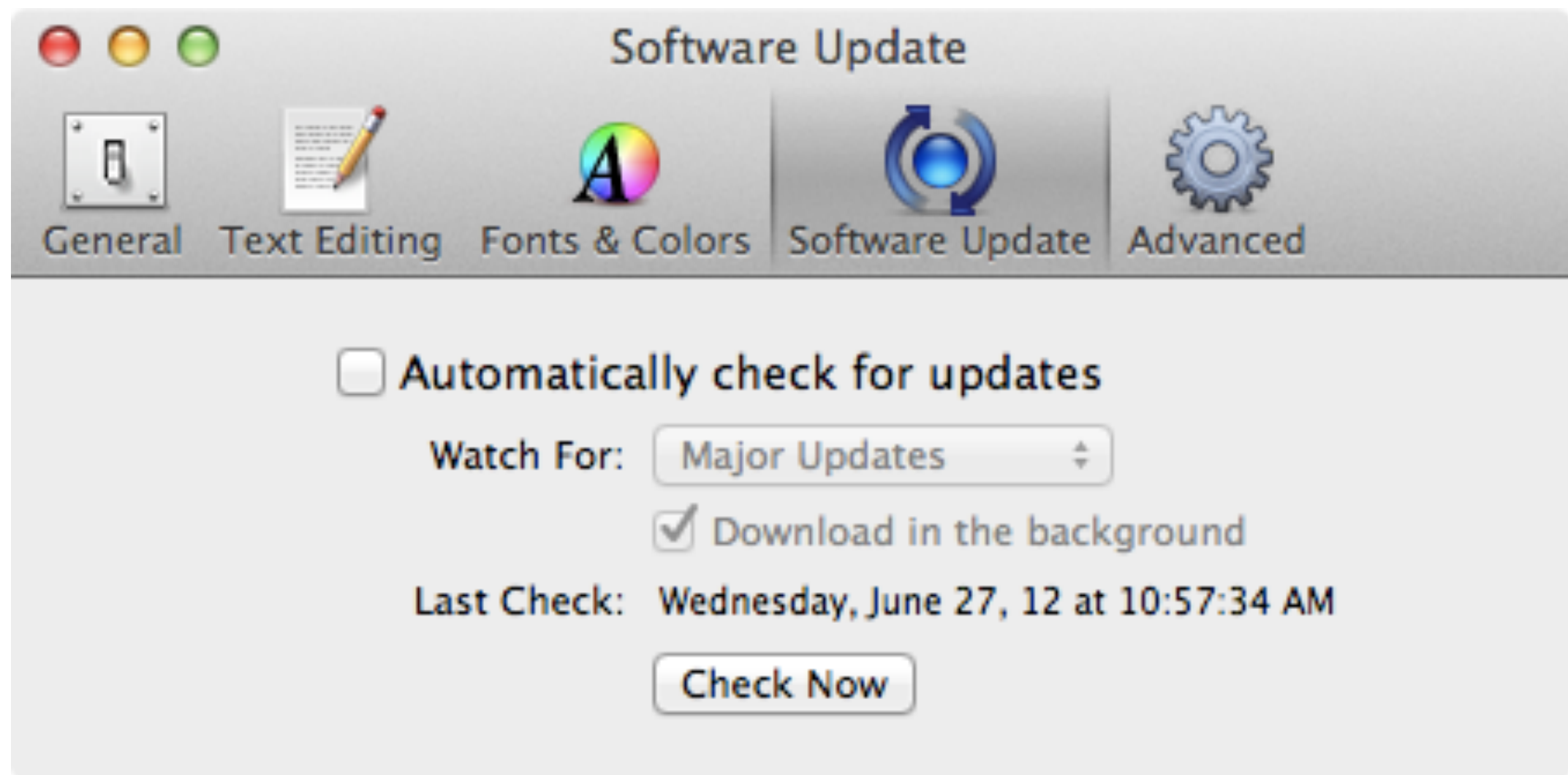
# Modular imaging

- Ensure your software can:
  - Install to non-boot disk
  - Install with no GUI user

# MCX/Profiles

- Allow admins to manage preferences/settings
- Examples:
  - Disable auto-update
  - Turn off registration prompts

# MCX/Profiles

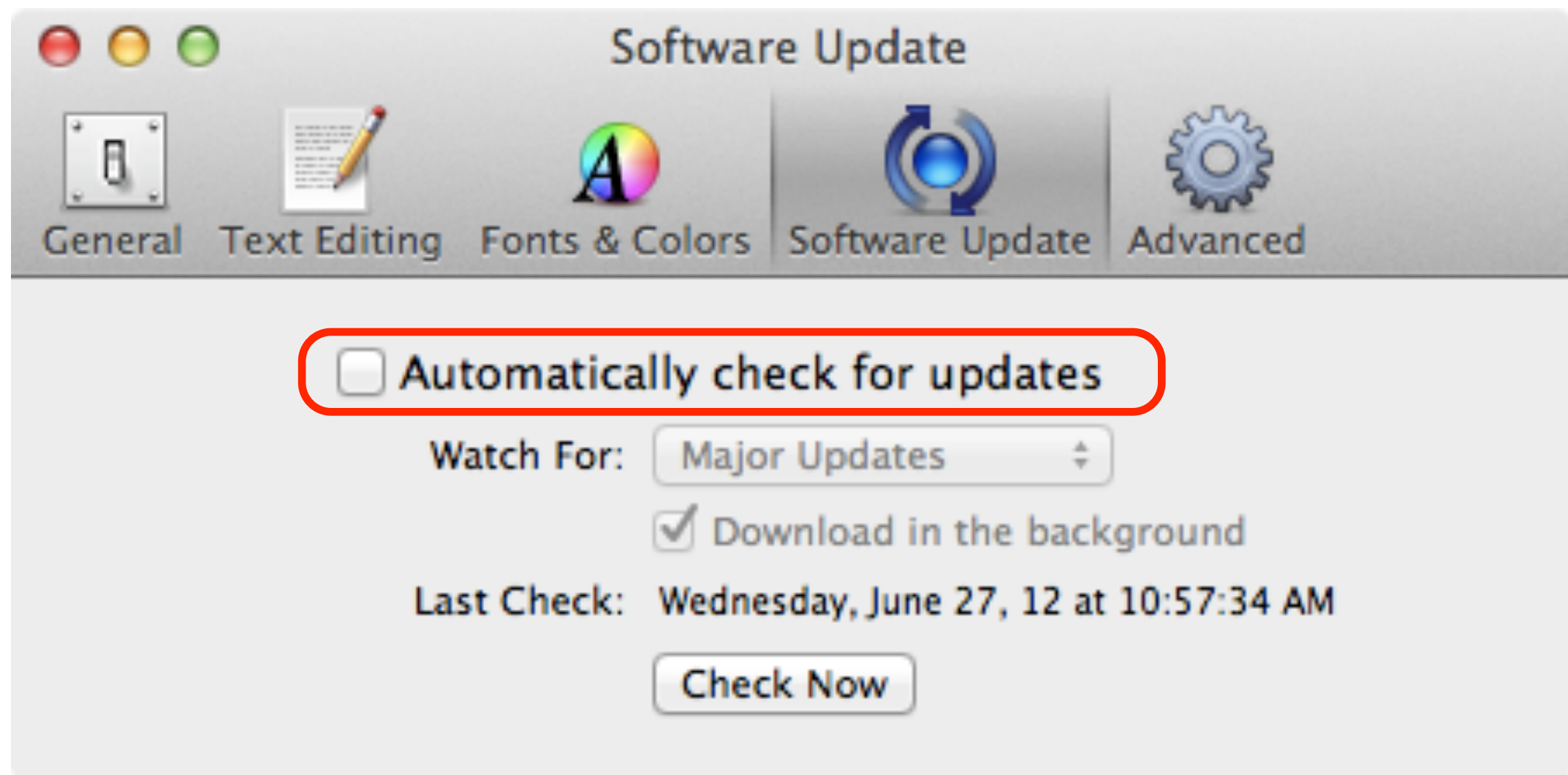


# MCX/Profiles

- Don't roll your own saving/restoring of preferences
- Use CoreFoundation CFPreferences methods



# MCX/Profiles



# MCX/Profiles

```
CFStringRef autoUpdateKey = CFSTR("SoftwareUpdateAutomaticCheckEnabled");
```

```
// Set up the preference.
```

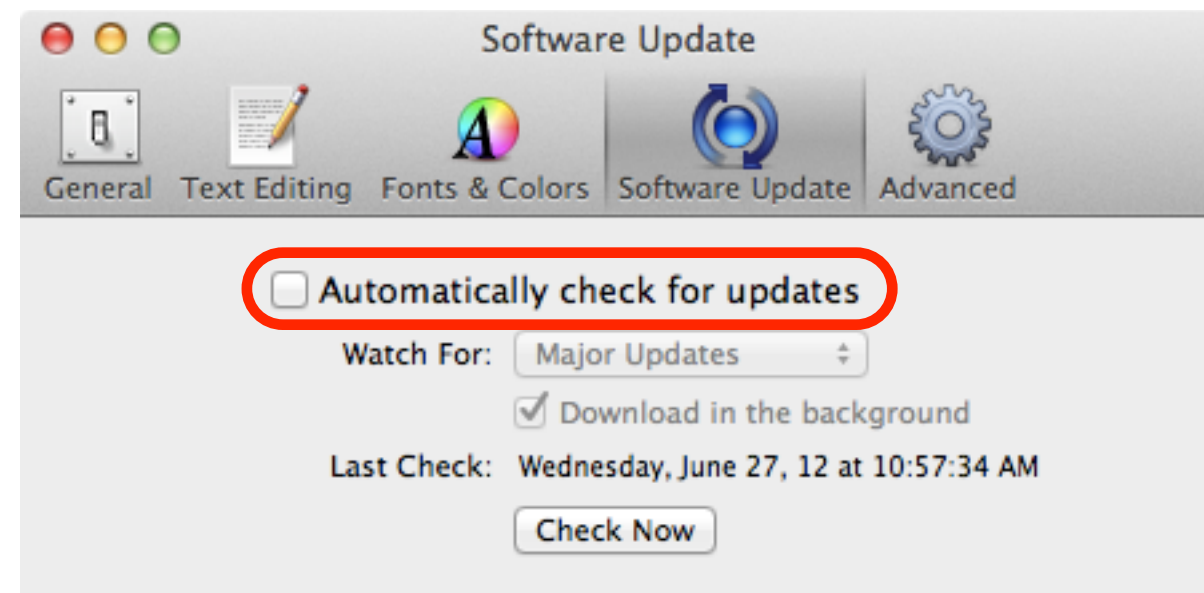
```
CFPreferencesSetAppValue(autoUpdateKey, @NO,  
    kCFPreferencesCurrentApplication);
```

```
// Write out the preference data.
```

```
CFPreferencesAppSynchronize(kCFPreferencesCurrentApplication);
```

```
// Read the preference.
```

```
autoUpdatesEnabled = (CFStringRef)CFPreferencesCopyAppValue(autoUpdateKey,  
    kCFPreferencesCurrentApplication);
```



# What if I don't?

- Systems administrators may recommend competitor's enterprise-friendly product
- Systems administrators may resort to repackaging or hacks to make your software work in their environment

# What if I don't?

- Fewer sales
- Higher support costs

# Review

# What is “enterprise-friendly”?

- Make it easy to purchase for large numbers of users/machines
- Make it easy to install with automated tools
- Make sure it works for all users of a machine, even those without admin rights

**...that's enterprise-friendly!**