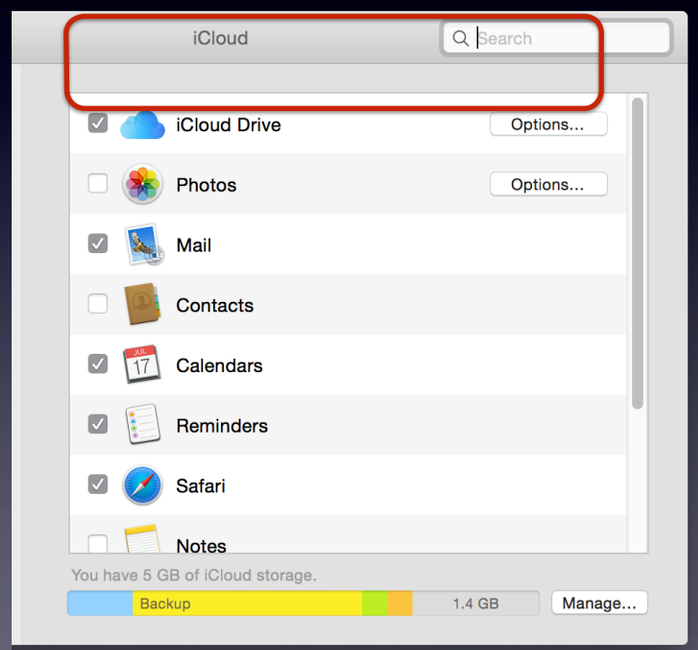
Deconstructing iCloud Drive

MacTech Pro, Atlanta GA
May 6, 2015

iCloud Drive does a great job of keeping your stuff on all of your devices. But how? The problem with “magic” solutions is, when they break you don’t know how to fix them. In this session, we fix that by completely pulling apart iCloud Drive and really understanding where files are located, how backups capture the data and what can go wrong and how to fix it if they do. Be one of the only consultants in your market who really understands iCloud Drive.

iCloud Drive, Specifically



Not talking iCloud calendars, mail, contacts, Find My Mac, etc. Just iCloud Drive.

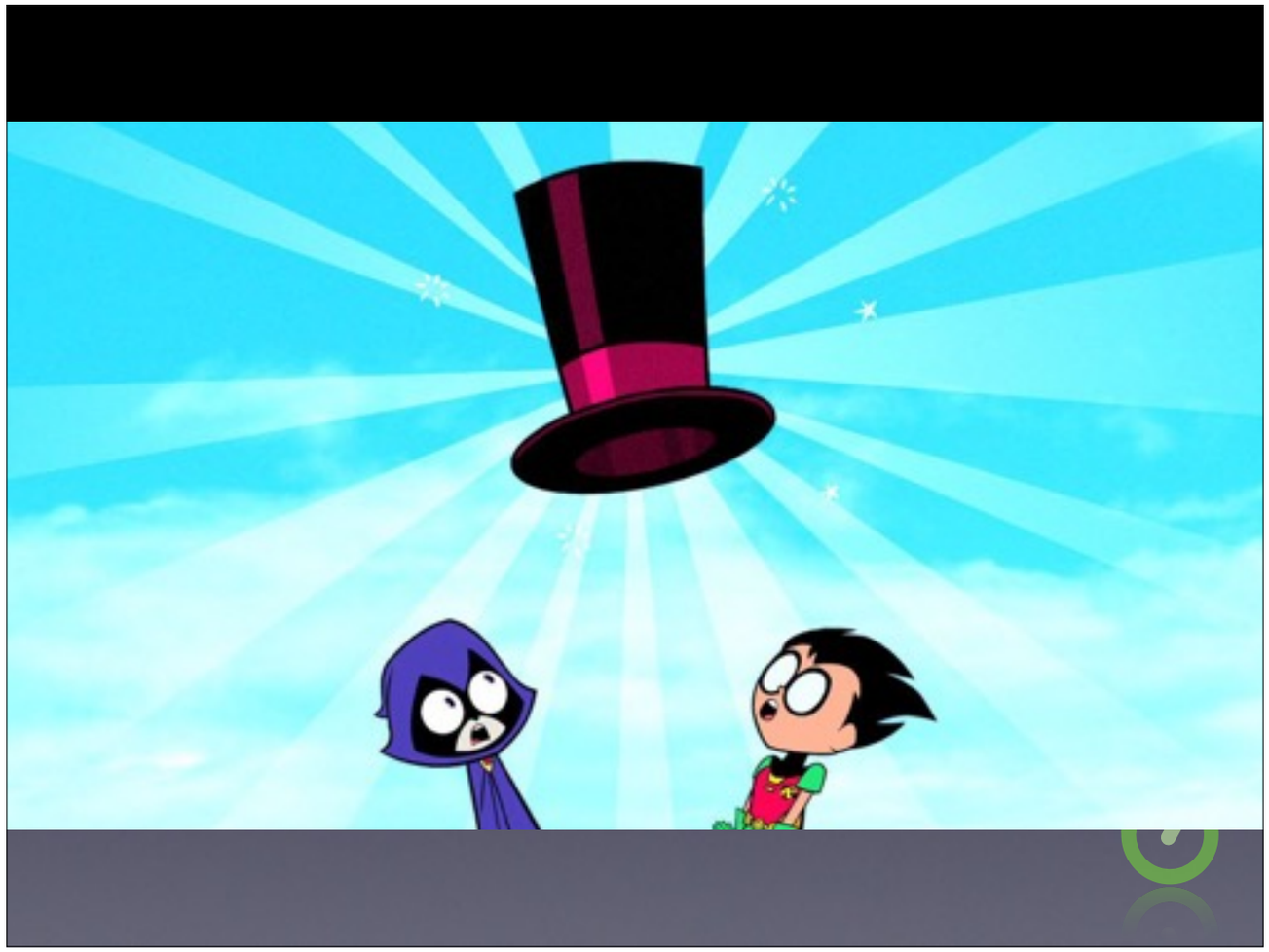
Quick show of hands: who uses it?

Why?

https://developer.apple.com/library/ios/documentation/General/Conceptual/iCloudDesignGuide/Chapters/iCloudFundametals.html

From the perspective of users, iCloud is a simple feature that automatically makes their personal content available on all their devices. To allow your app participate in this "magic," you design and implement your app somewhat differently

Why should we learn this?
MAGIC!!

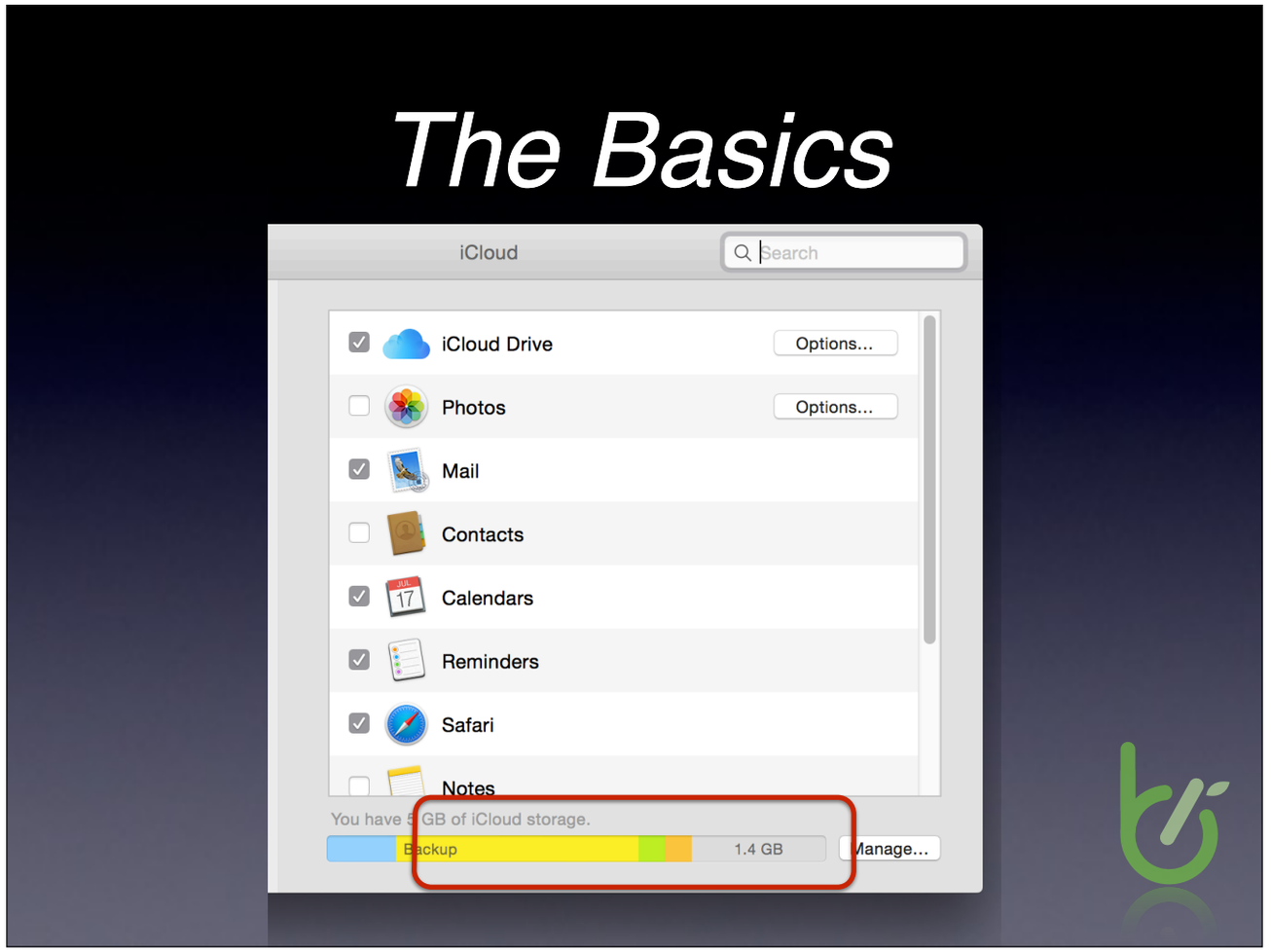


The Basics

File < 15GB

Each file must be under 15GB in size.

The Basics



Files must fit in your iCloud storage pool, so 15GB not possible for me.

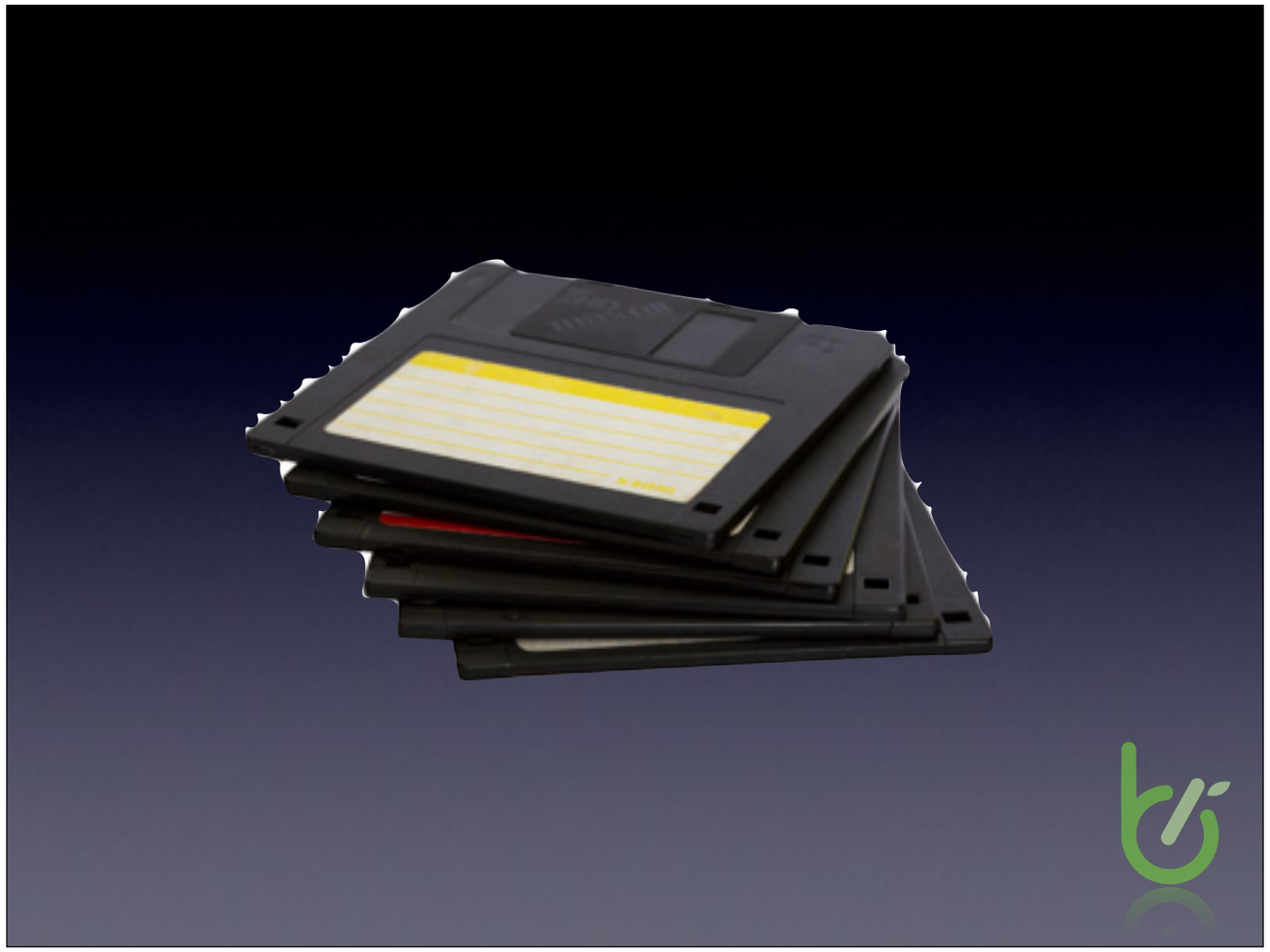
iOS 8

Not that Dos Equis.

The Basics

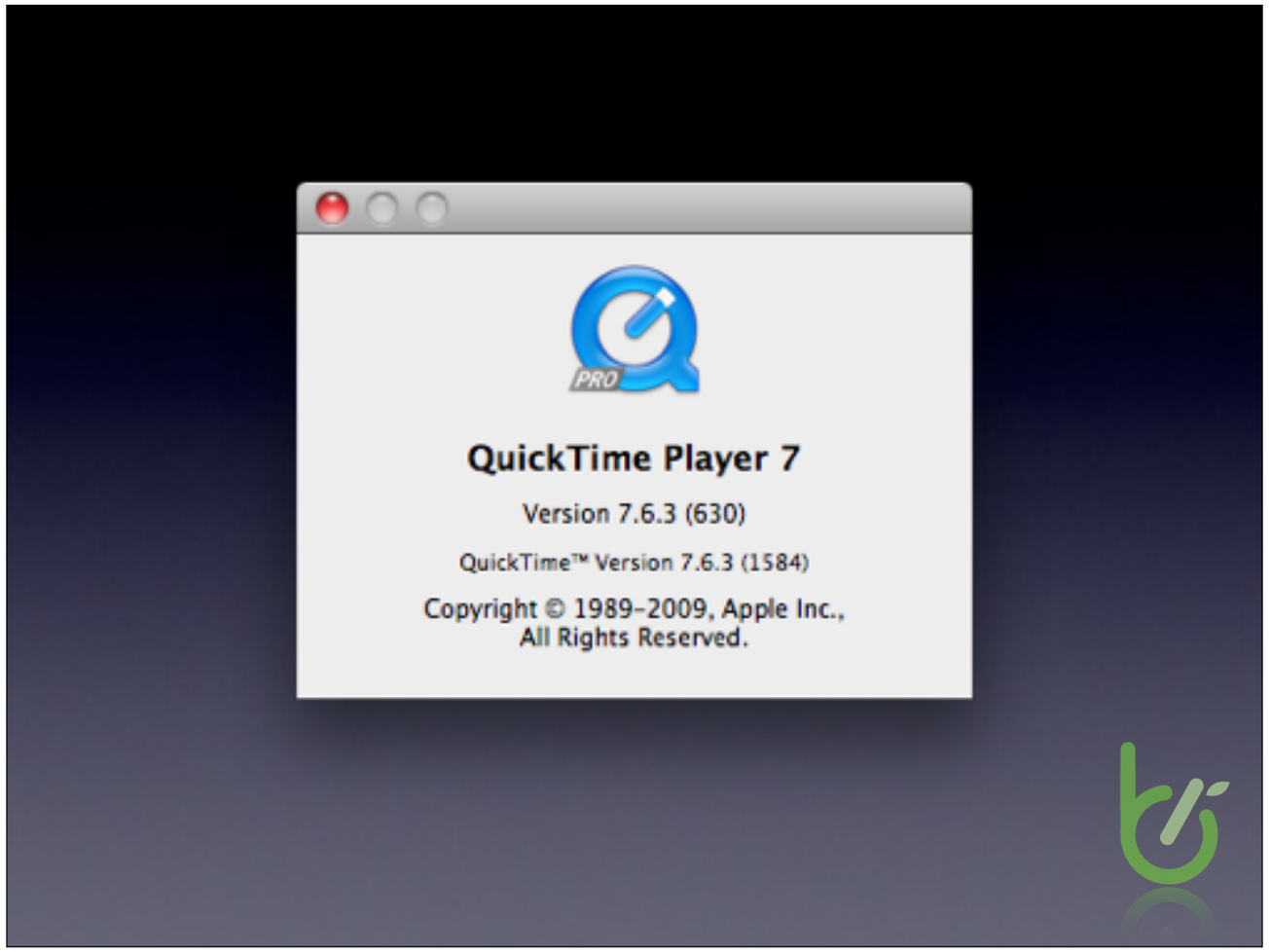
This Dos Equis. X.X 10.10 (Ya get it? Ya like it?!)

No kitties. (And why would you run Mavericks?!)



If you’re running anything but 10.10.3, you might as well be using this.

Or this.


QuickTime Player 7
Version 7.6.3 (630)
QuickTime™ Version 7.6.3 (1584)
Copyright © 1989–2009, Apple Inc.,
All Rights Reserved.

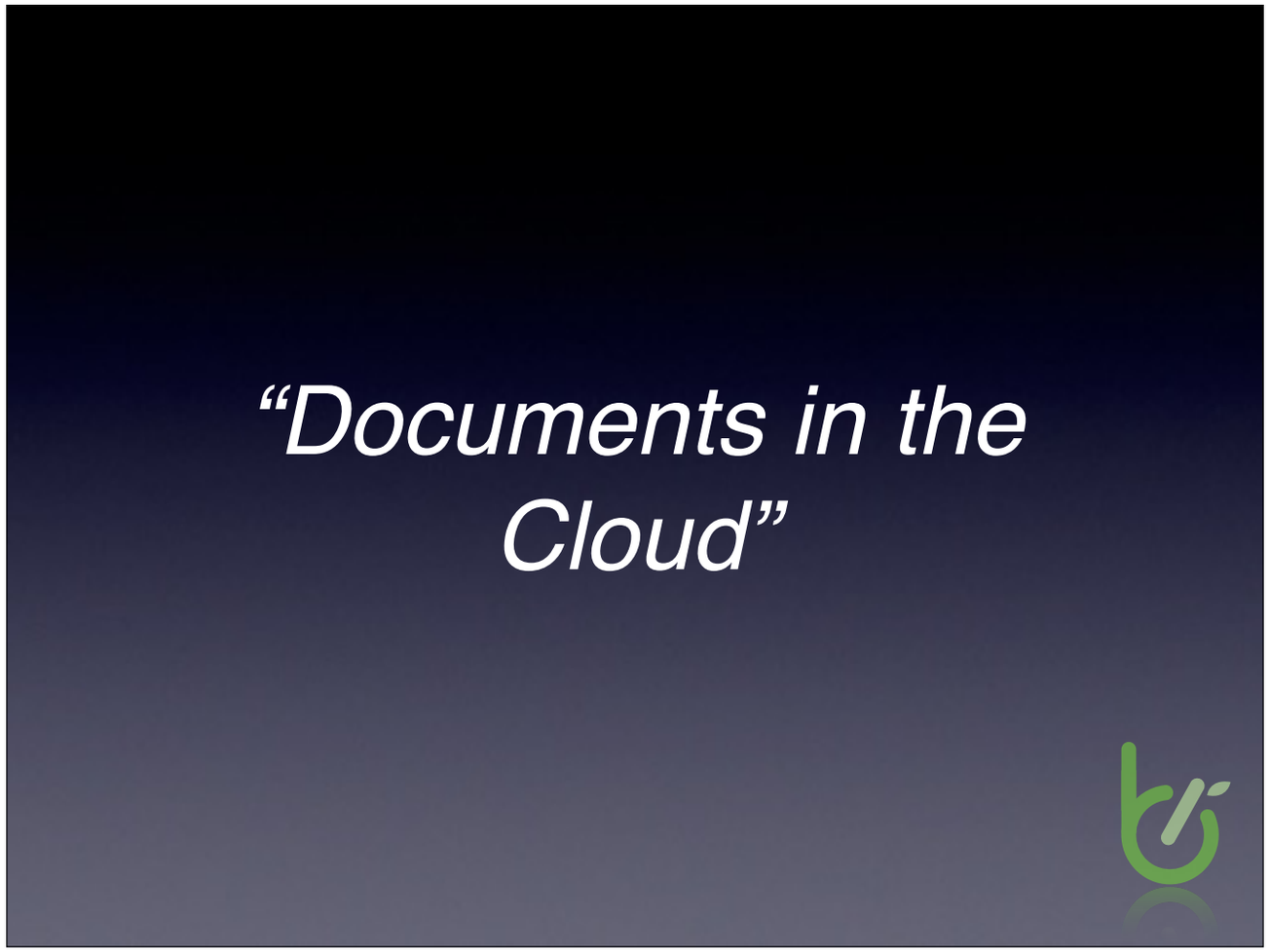
Or this.

Or this.

10.10.2 and Earlier

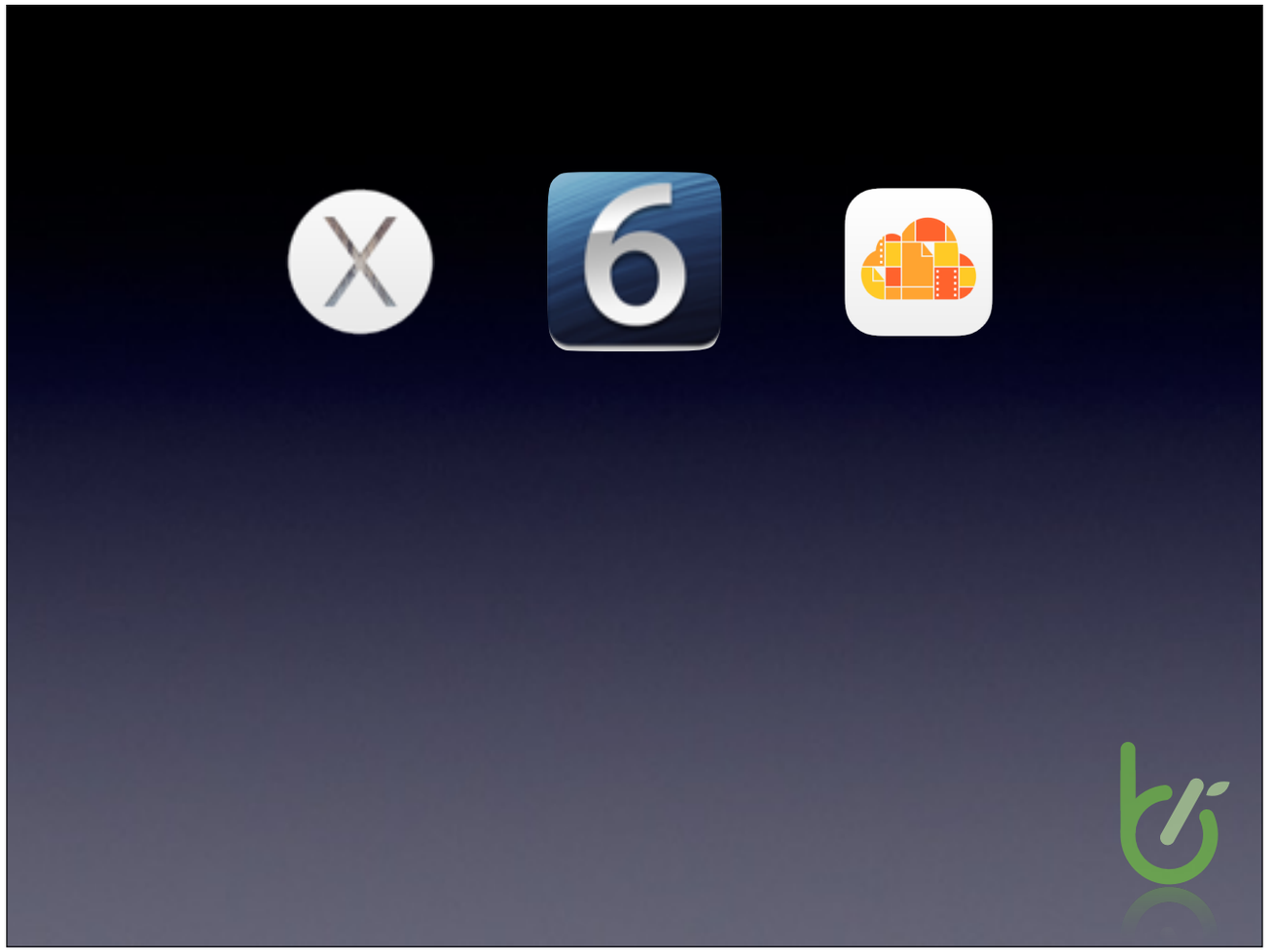
```
##
# This module requires Metasploit: http://metasploit.com/download
# Current source: https://github.com/rapid7/metasploit-framework
##

require 'msf/core'

class Metasploit4 < Msf::Exploit::Local

  Rank = GreatRanking

  include Msf::Post::OSX::System
  include Msf::Exploit::EXE
  include Msf::Exploit::FileDropper

  def initialize(info = {})
    super(update_info(info,
      'Name'           => 'Mac OS X "Rootpipe" Privilege Escalation',
      'Description'    => %q{
        This module exploits a hidden backdoor API in Apple's Admin framework on
        Mac OS X to escalate privileges to root. Dubbed "Rootpipe."

        Tested on Yosemite 10.10.2 and should work on previous versions.

        The patch for this issue was not backported to older releases.

        Note: you must run this exploit as an admin user to escalate to root.
      },
      'Author'         => [
        'Emil Kvarnhammar', # Vulnerability discovery and PoC
        'joev',             # Copy/paste monkey
        'wvu'               # Meta copy/paste monkey
      ],
      'References'     => [
        ['CVE',   '2015-1130'],
        ['OSVDB', '114114'],
        ['EDB',   '36692'],
        ['URL',   'https://truesecdev.wordpress.com/2015/04/09/hidden-backdoor-api-to-root-privileges-in-apple-
```

Or this.
Root Pipe vulnerability isn't getting fixed for old OS'es. There's yer sign.



The old name for what iCloud Drive does.

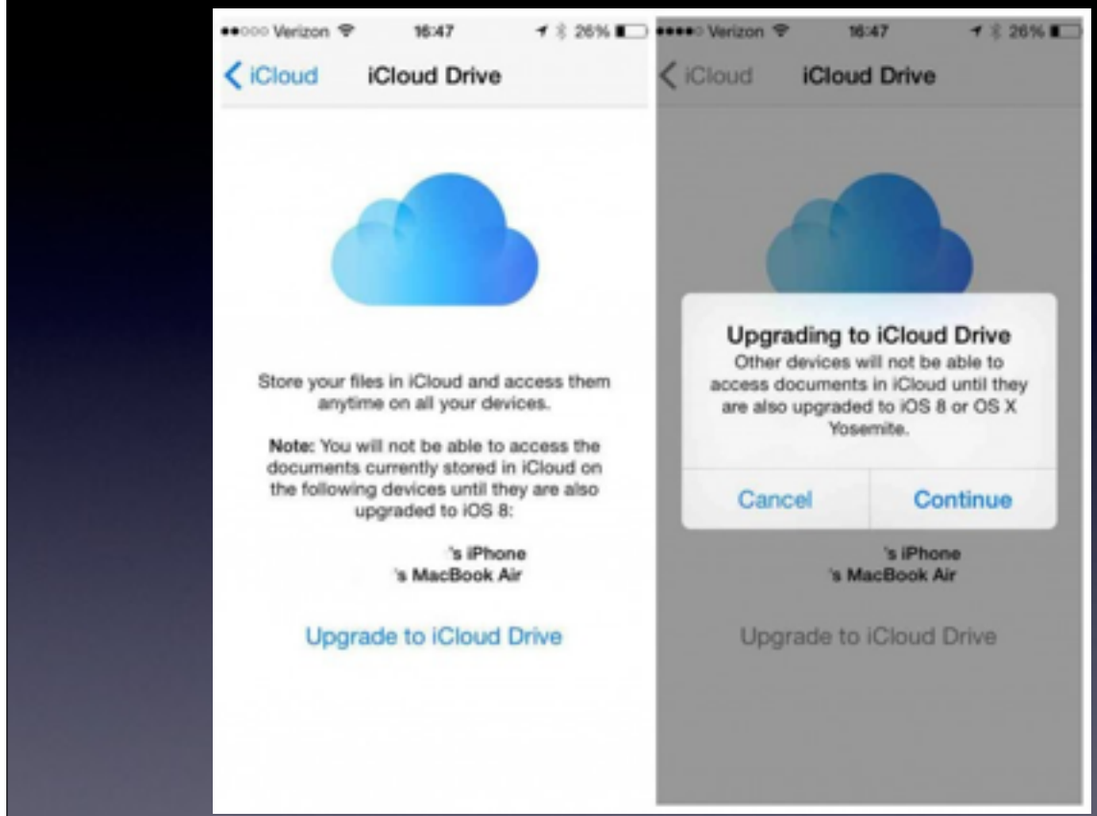
“Same age” concept



Doing this? Survey SAYS….

buzz

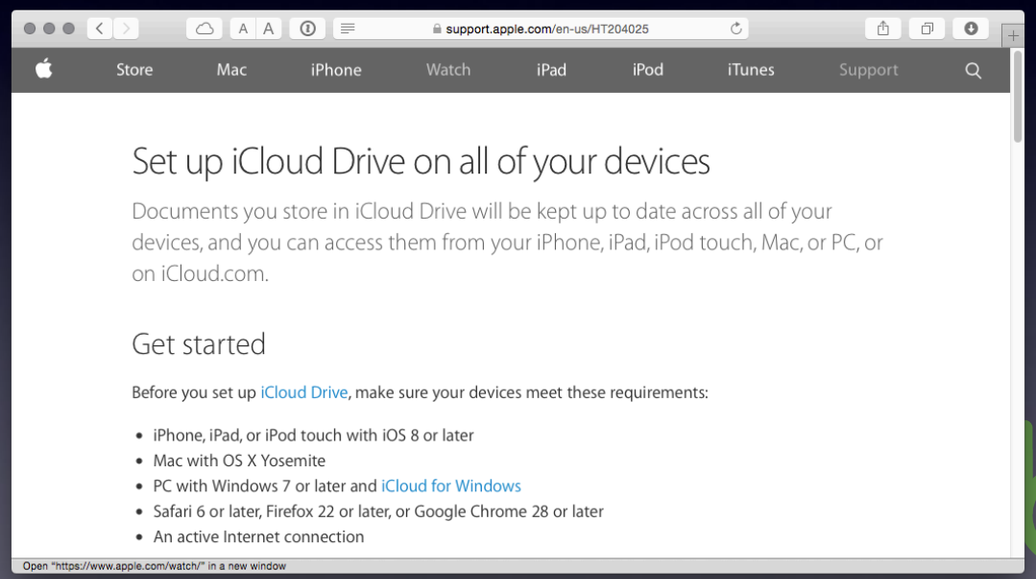
And Steve Harvey looks at you like this.
In reality there won’t be data loss, but your stuff won’t sync. Bummer.



No going back.

Setup Steps

https://support.apple.com/en-us/HT204025

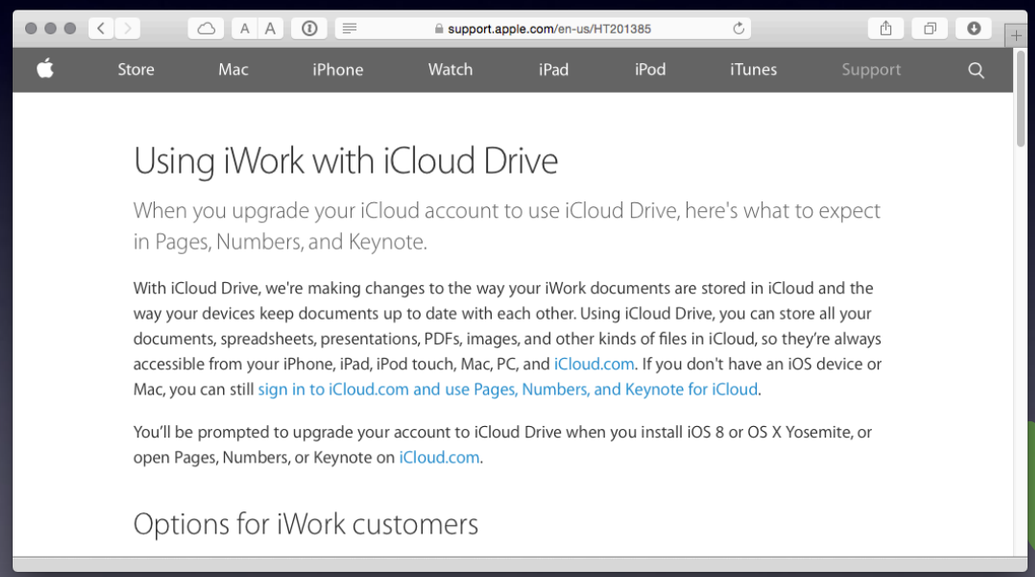


https://support.apple.com/en-us/HT204025

Neil, we are getting the slides right?

iWork and iCloud Drive

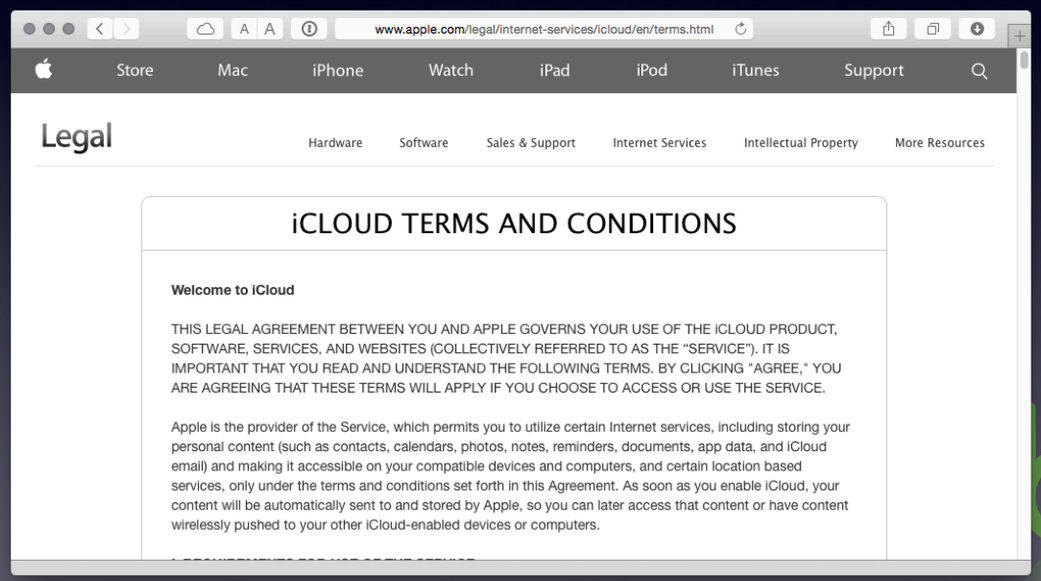
https://support.apple.com/en-us/HT201385



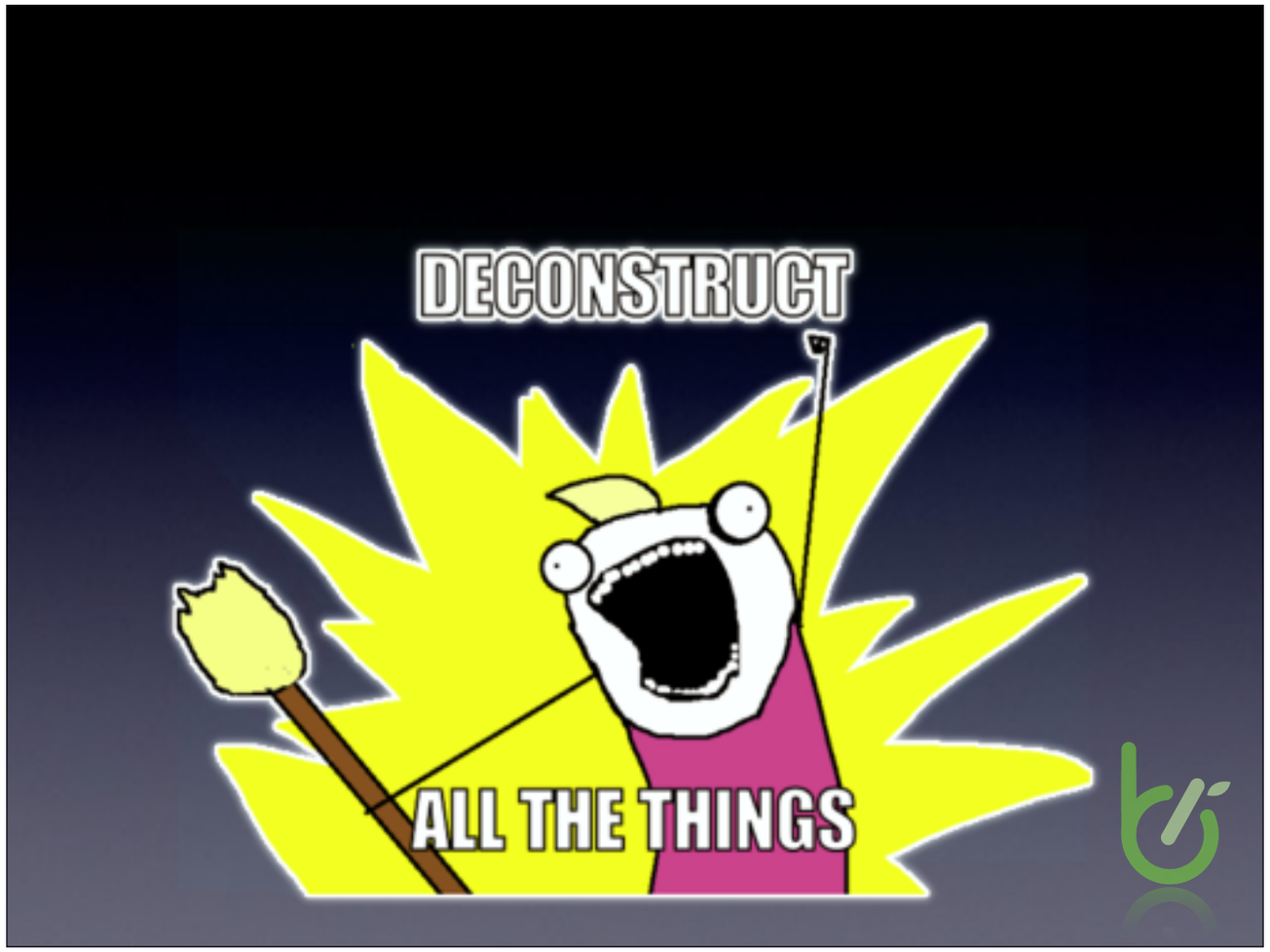
https://support.apple.com/en-us/HT201385

Terms & Conditions

http://www.apple.com/legal/internet-services/icloud/en/terms.html



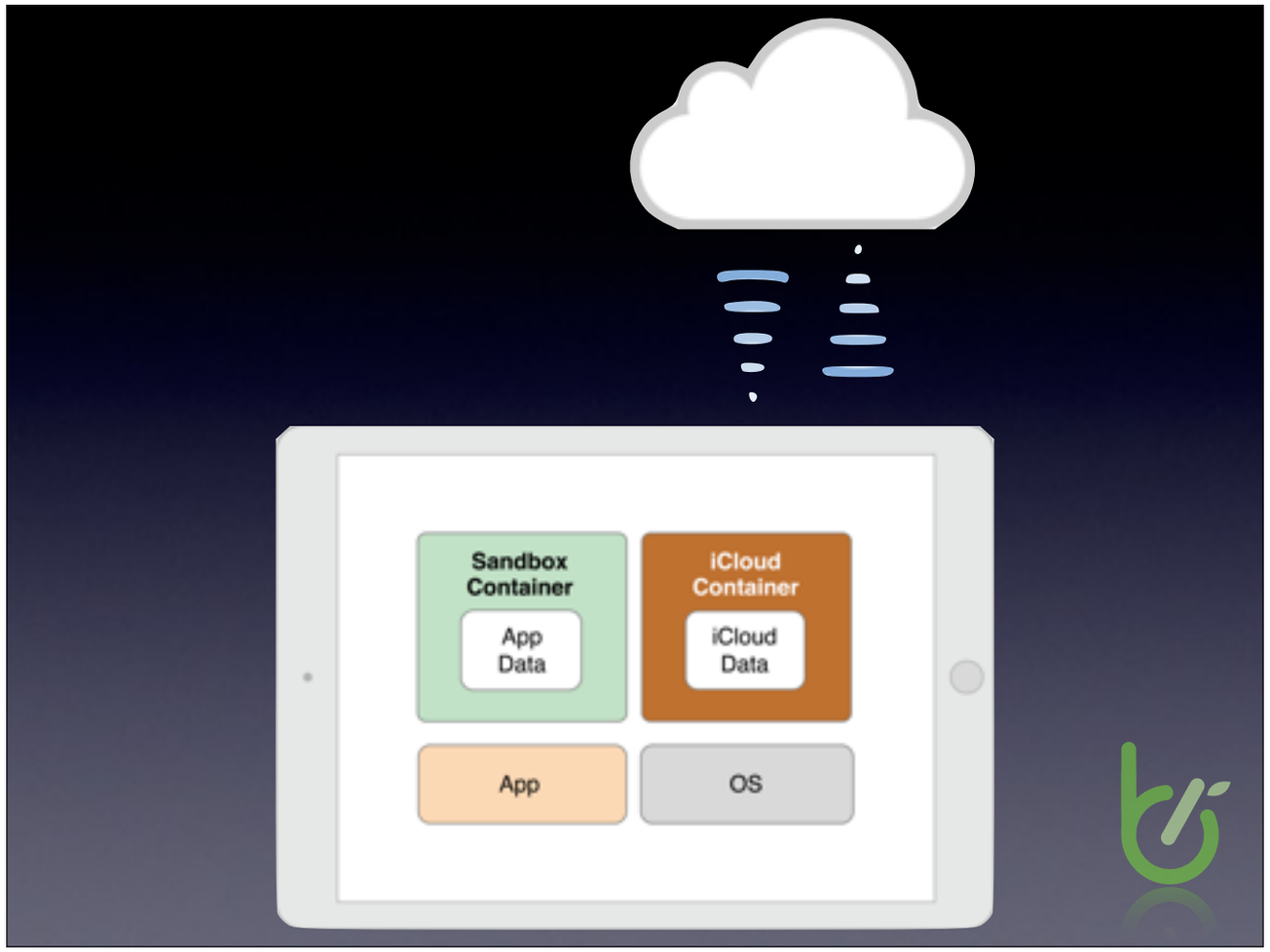
http://www.apple.com/legal/internet-services/icloud/en/terms.html



Let's have some fun.

Monitor iCloud connections with Little Snitch or nettop
brctl log -w
Change this slide and watch it change on iOS
And back again watching the network monitors

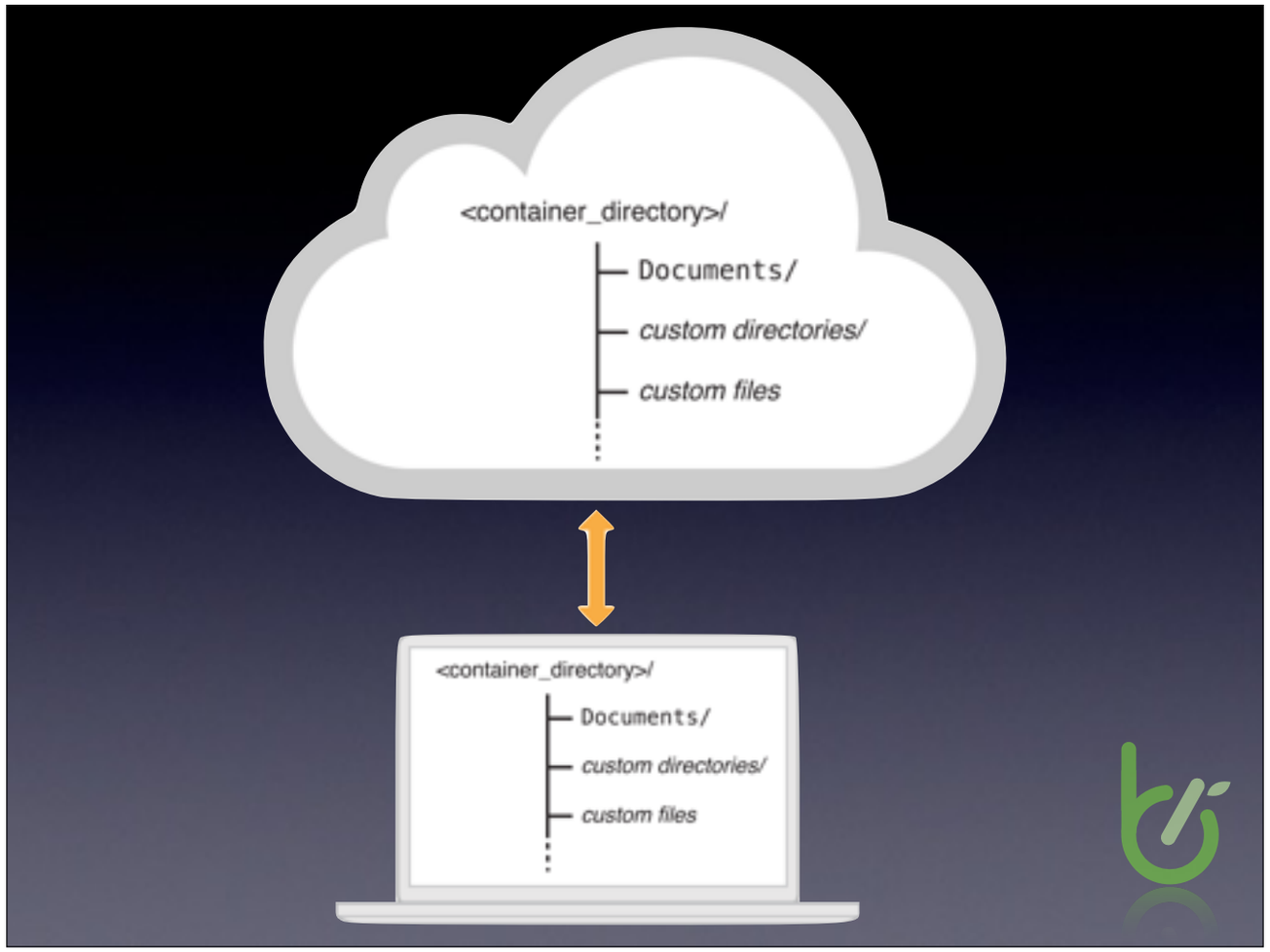
Open Textastic on iOS and iCloud Drive in Finder, same process with changes and new files



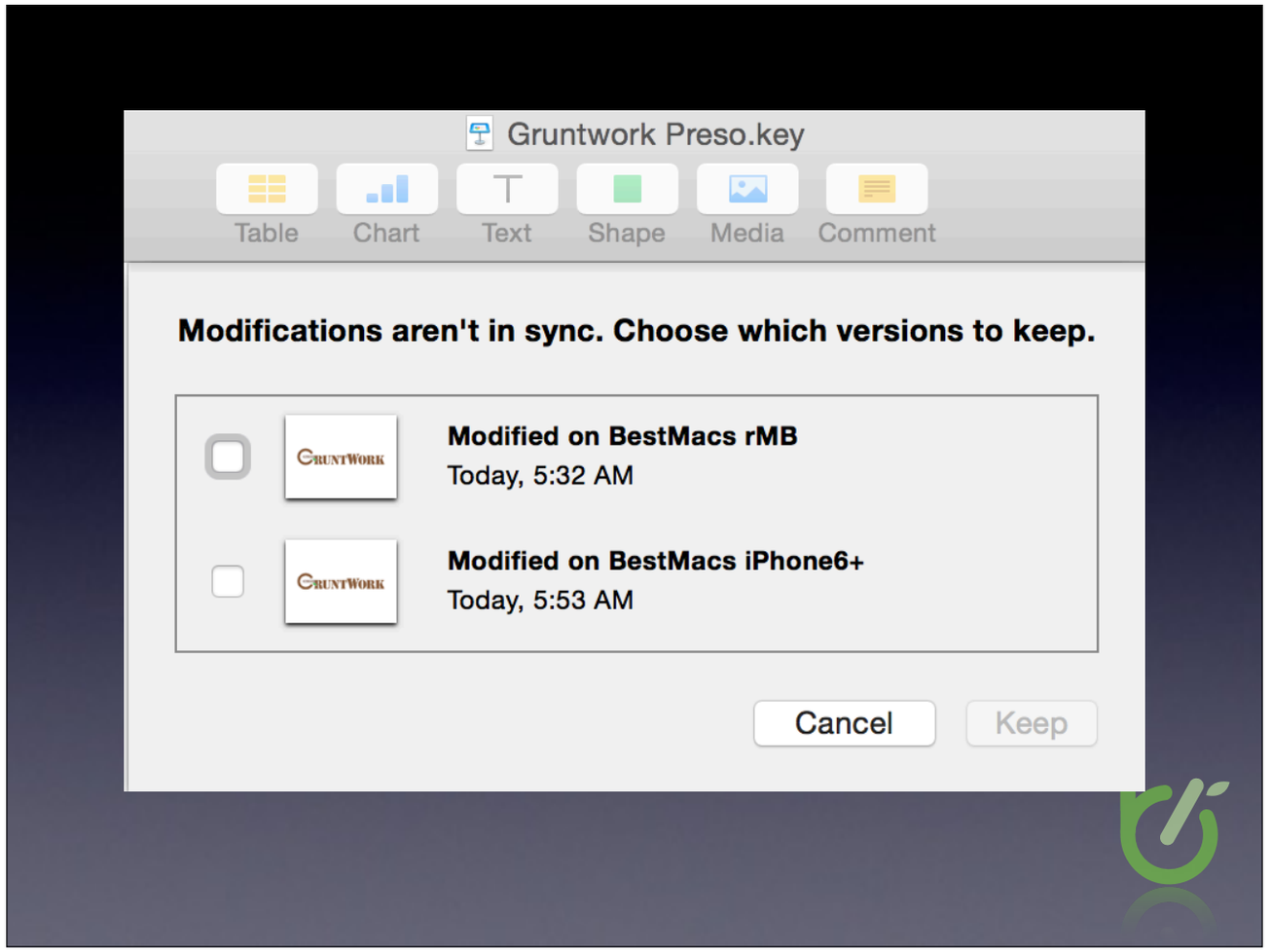
Textastic makes it a little more clear what's going on with iCloud containers.
Note diff between "local files" (App Data) and "iCloud" (iCould Data)
Other apps let you look at it more closely, some don't…

Plant a test folder/file in iCloud Drive

Look at iFiles and Cloud Opener.
Look at how it's displayed in Terminal



container/Documents is presented to you as “public face” of folder
everything outside of Documents is considered private app data
Note ~/Mobile Documents/com~apple~CloudData



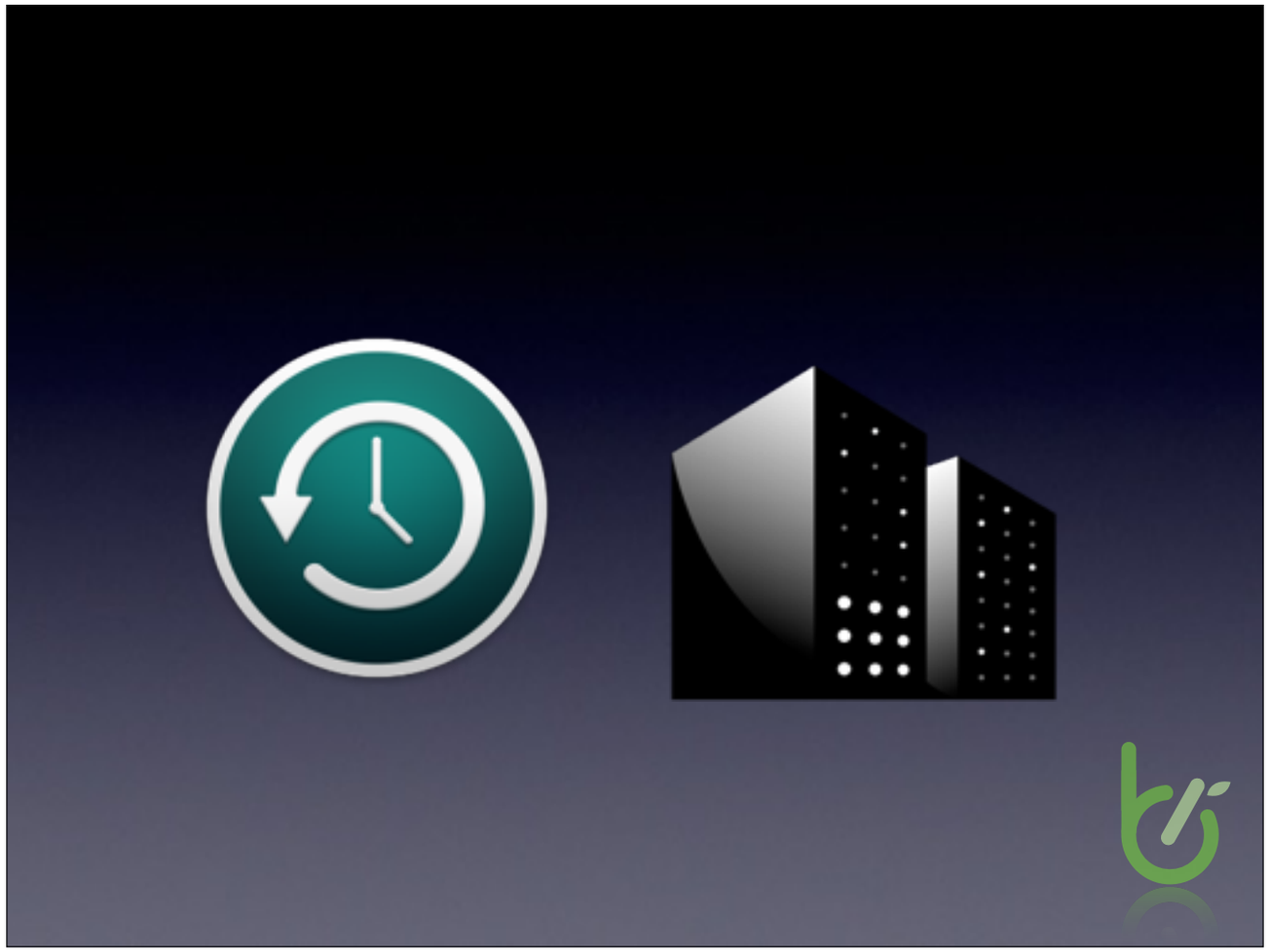
Conflicts can and do happen. Pick one and go with it.

SYNC
IS
NOT
BACKUP

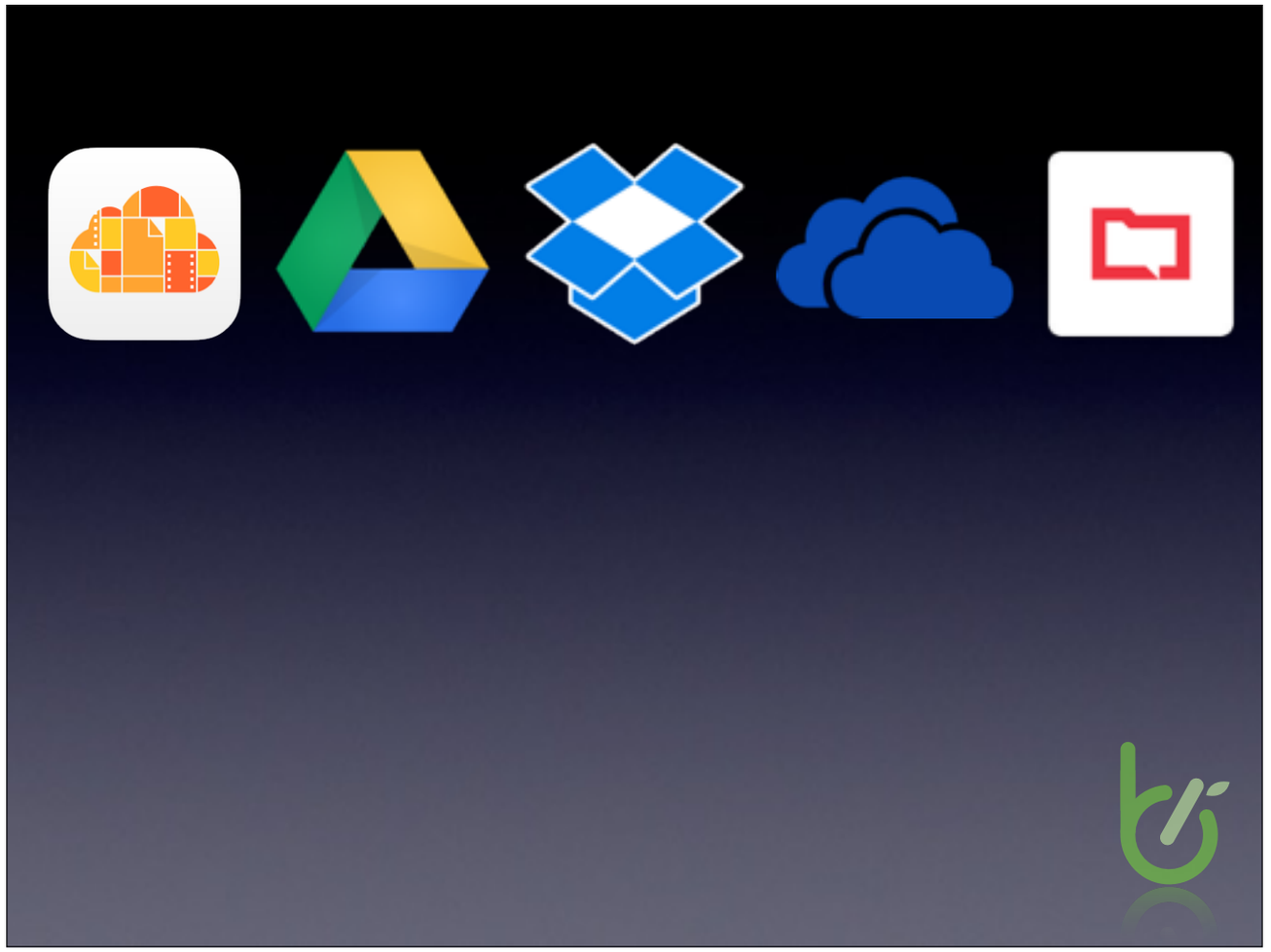
SYNC
IS
NOT
BACKUP

SYNC
IS
NOT
BACKUP

Catastrophic loss or fail, sure you've got copies. More common: file deletion, unwanted changes, corruption.



Let’s explore Time Machine and CrashPlan



iCloud Drive vs. Google Drive vs. DropBox (et al) vs. Microsoft OneDrive vs. Code42 SharePlan

Too Many Clouds

http://www.macworld.com/article/2844929/how-to-simplify-overlapping-cloud-storage-services.html

Home / Cloud & Services

How to simplify overlapping cloud storage services

Joe Kissell | *@joekissell*
Senior Contributor, Macworld

Nov 10, 2014 3:47 AM

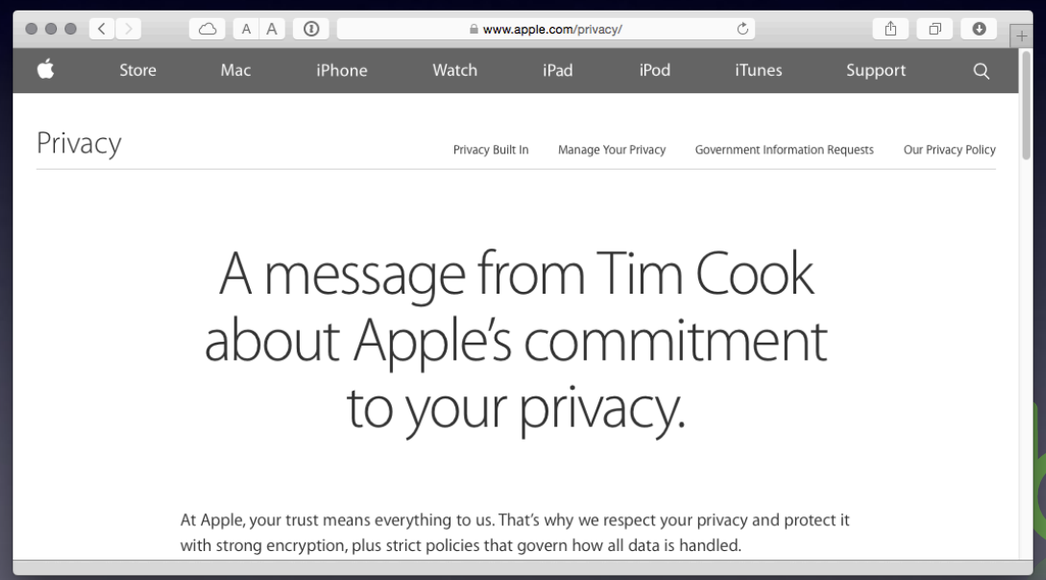
There's no shortage of choices for cloud storage, but that leads to another problem: how do you decide which services you truly need, and which files to

http://www.macworld.com/article/2844929/how-to-simplify-overlapping-cloud-storage-services.html


TIN FOIL
FREE YOUR MIND

Privacy Policy

https://www.apple.com/privacy/



https://www.apple.com/privacy/

My good friend Tim…