

iCloud Storage

No matter where you go, there you are

Session 501

Dallas De Atley

Core OS

These are confidential sessions—please refrain from streaming, blogging, or taking pictures

My Stuff

My Stuff



My Stuff



My Stuff



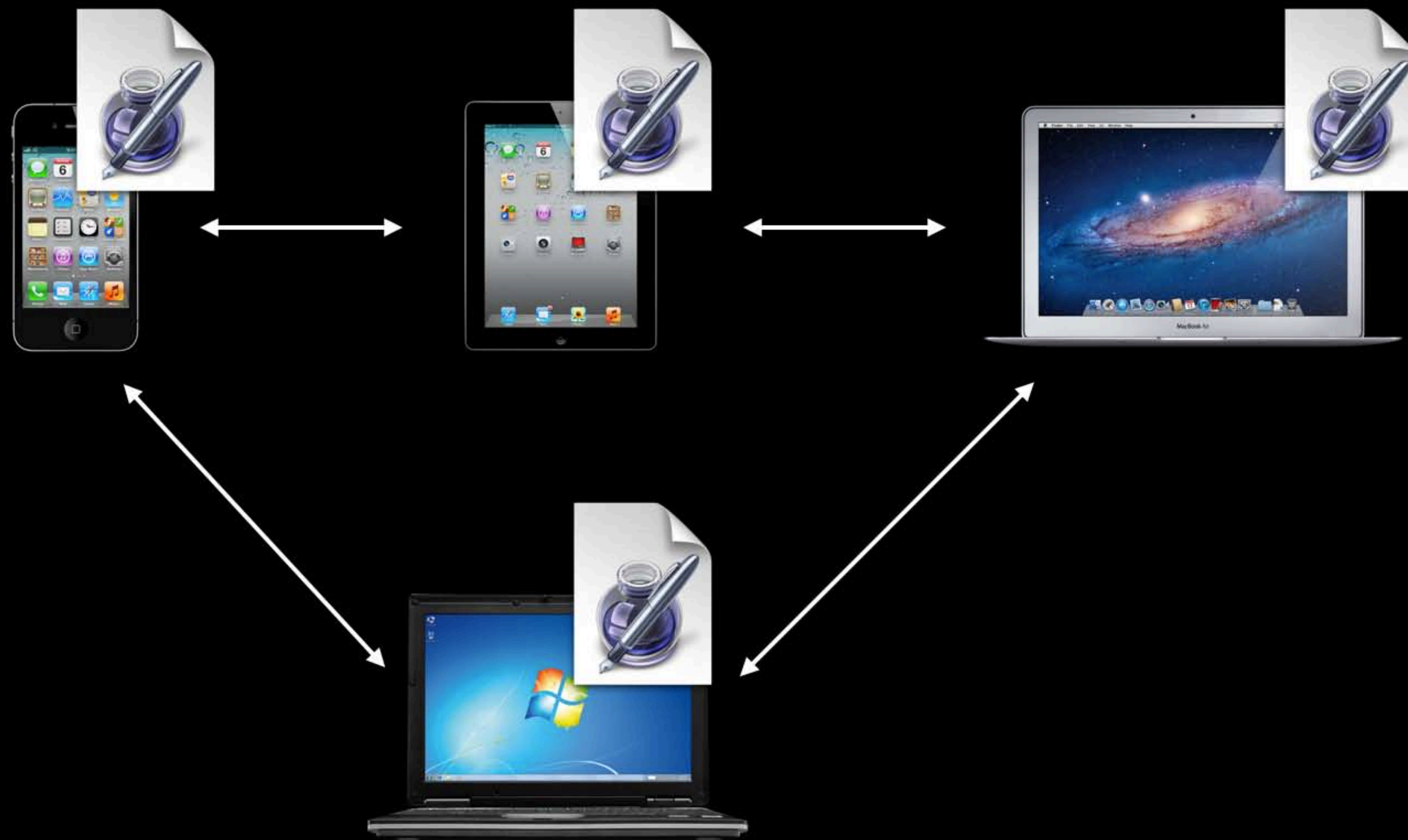
My Stuff



My Stuff



My Stuff



Existing Solutions



Email
via IMAP



Contacts
via CardDAV



Calendars
via CalDAV



iTunes
media and files
to devices



Custom Solution?

- Regular files
- Packages
- Shoebox apps
- Preferences

iCloud Storage



- New API available for App Store apps
- Provides an iCloud sandbox per app
 - Uploads files and data stored in the sandbox
- Uploads simple key value data
- Works continuously and in real time

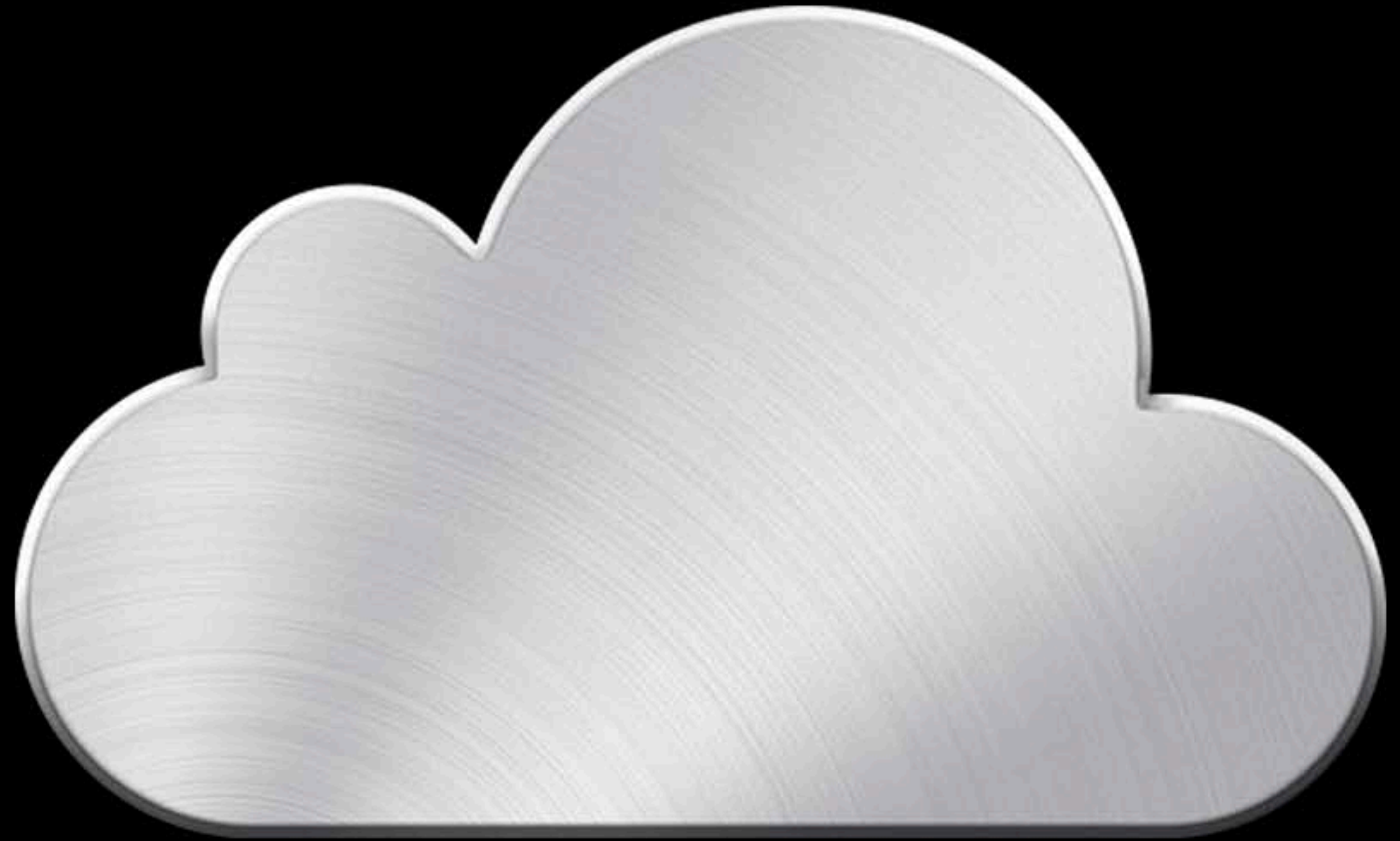
iCloud Storage

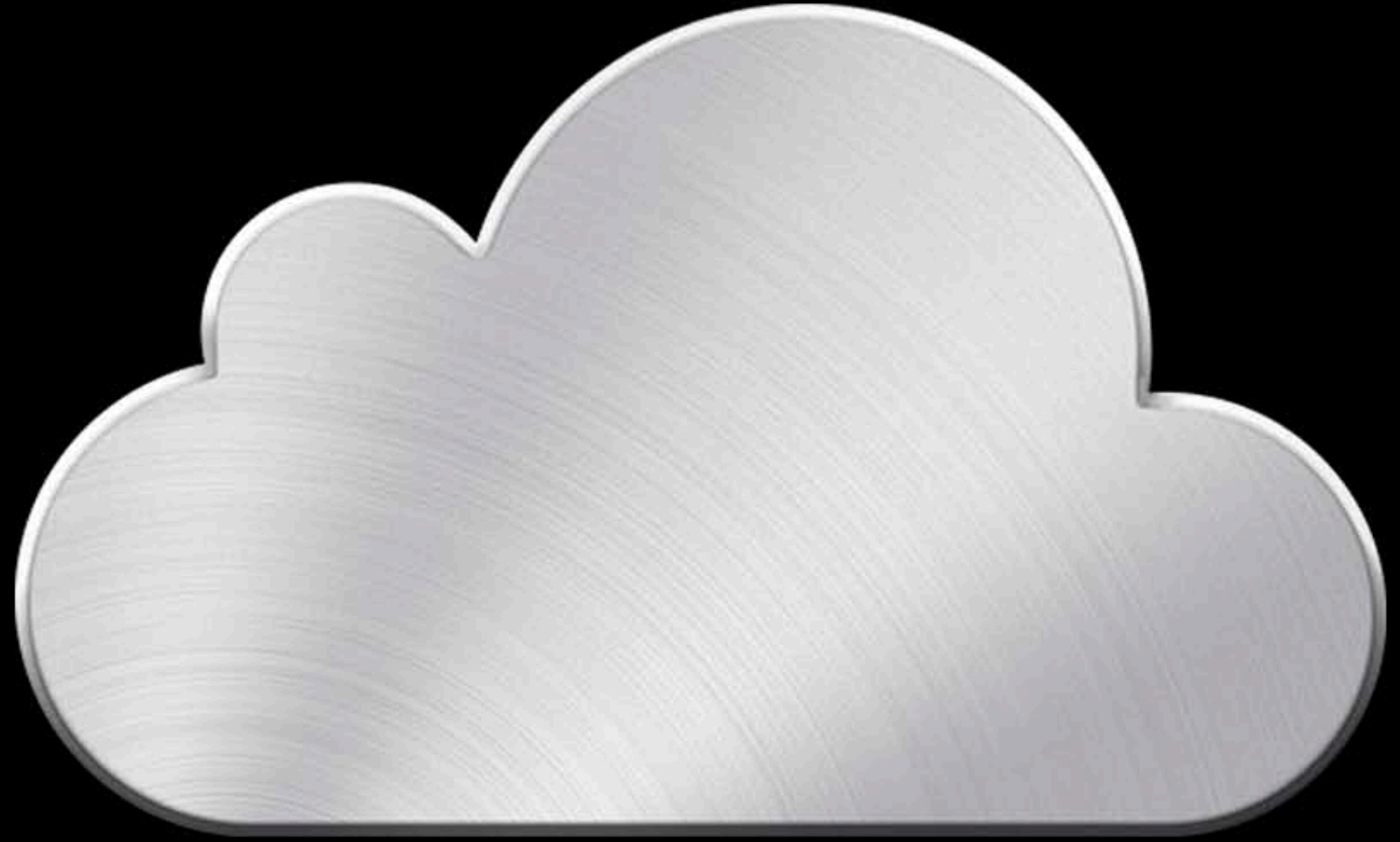


- Moves the data between each device
- Notifies your application of file changes
- Resolves conflicts automatically
- It just works
- Even if your application isn't running

How Does It Work?

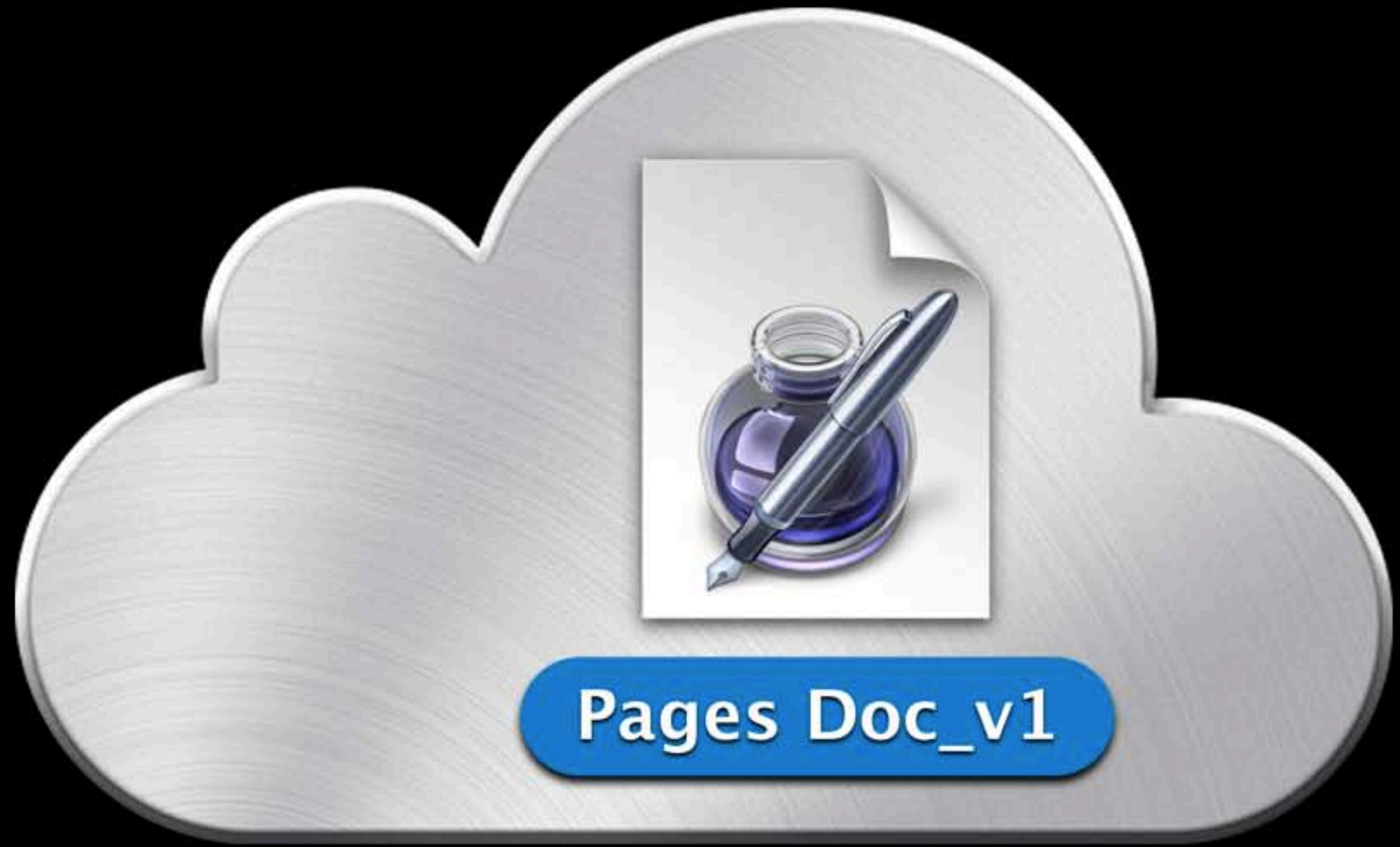




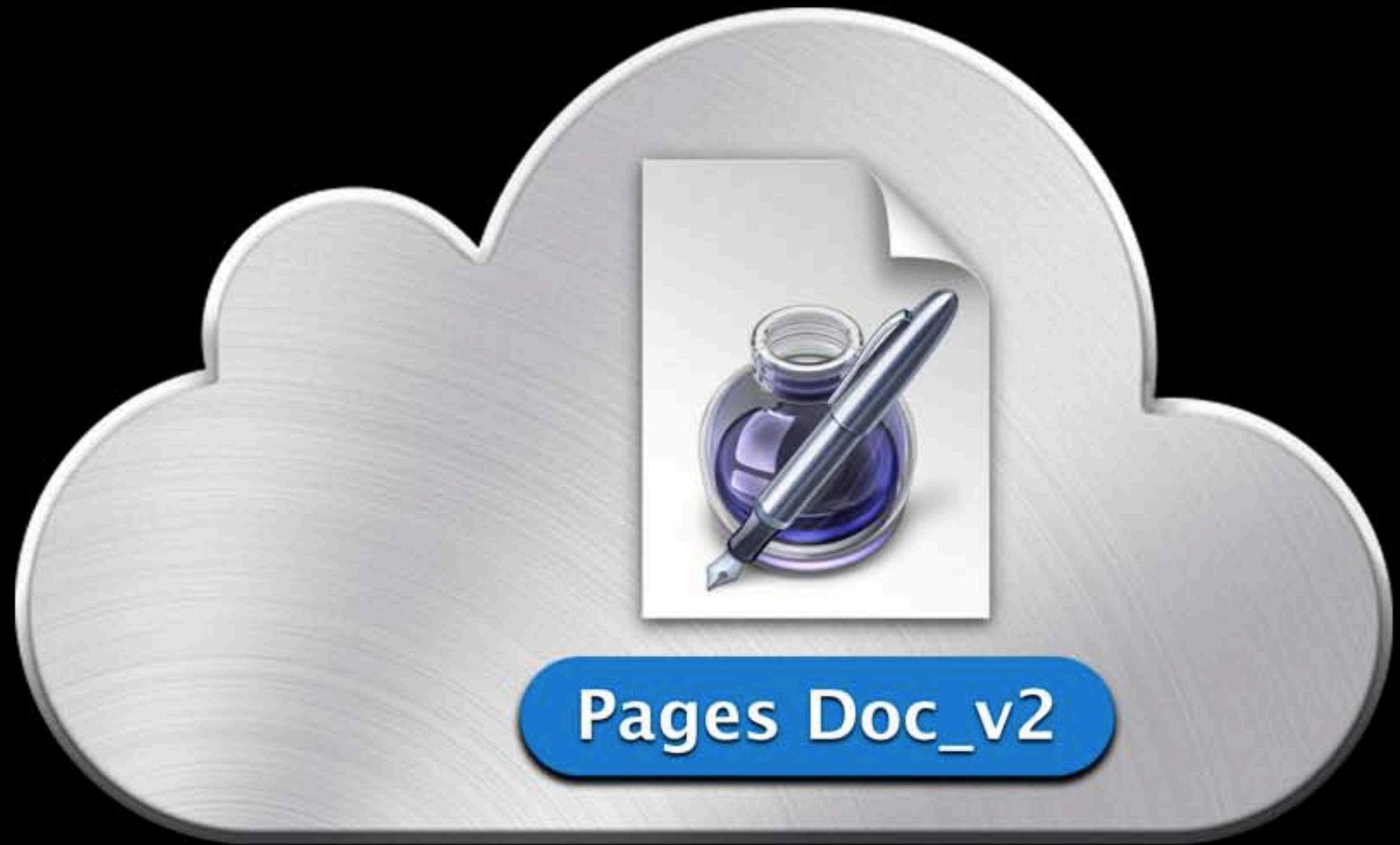








Pages Doc_v1



How Does It Work?

- It's atomic
- It also works with packages
- It goes peer to peer when possible

iCloud (Meta)data

- Metadata is always pushed to iCloud
- iCloud always pulls new files
- Devices pull files when appropriate
 - The OS may pull data automatically
- Your app will know of files that aren't yet local
 - The file exists, but the data hasn't arrived

iCloud

The service behind cloud storage

Eric Krugler
iCloud Storage



iCloud



iCloud stores your content,
and wirelessly pushes it to
all your devices



iCloud stores your content,
and wirelessly pushes it to
all your devices

iCloud integrates seamlessly
with your apps, so everything
happens automatically







iCloud Storage API

iCloud Is Free with iOS 5



- iCloud is part of iOS 5 setup
- Use Apple ID or create one
- Free 5GB storage
- iCloud Storage is automatically on

Building Cloud Storage

- Set up and configure servers
- Create your own protocol
- Write a lot of code
- Add push notifications
- Add replication for HA and DR
- Be on-call all the time
- Pay for storage and bandwidth

Use iCloud Storage APIs

Experience with Cloud Services



Apple Scale Infrastructure



- New Maiden, NC data center—
third facility
- \$500 million investment
- iCloud Storage APIs gives you
access to Apple's infrastructure



Your App Here

iCloud Storage APIs

Dallas De Atley

Core OS

iCloud Storage API

- NSFileManager
- NSMetadataQuery
- UIDocument
- NSDocument
- Native Windows API
- Key Value Store
 - Not available on Windows



NSFileManager

- Move a file into the iCloud sandbox
- Determine if a given file is in iCloud
- Locate the iCloud sandbox for your app
- Publish a URL for downloading

NSMetadataQuery

- Asynchronous discovery and notification
- New search scopes for iCloud Storage
 - `NSMetadataQueryLocalDocumentsScope`
 - `NSMetadataQueryUbiquitousDocumentsScope`
- New `NSMetadataItem` attributes
 - File attributes
 - iCloud attributes
- New scopes do not work with complex predicates

UIDocument

- A simple asynchronous reading and writing model
- A simple autosaving mechanism
 - Includes automatic support for suspended apps
- Includes new Cocoa document model
 - Provides iCloud Storage support

NSDocument

- Modernized document model
 - Includes iCloud Storage support
- NSFileCoordinator
 - Coordinate with other processes
- NSFilePresenter
 - Manage concurrent access
- NSFileVersion
 - Manage multiple versions and iCloud conflicts

Windows API

- Discover the iCloud sandbox
- Receive notifications when files change

iCloud and the Shoebox

- Not every document is a simple file
- Shoebox = database + files
- Core Data supports iCloud Storage

Key Value Data

- When you want to store a small amount of information
 - Stocks
 - Cities
 - Pages read in a book
- For storing tens of kilobytes

NSUbiquitousKeyValueStore

- Store simple plist values
- Reserved quota per application
- Change notifications
- No conflicts
 - Last change wins

Observing Changes

```
// get application's default store  
kvStore = [NSUbiquitousKeyValueStore defaultStore];
```

```
// observe changes  
[[NSNotificationCenter defaultCenter] addObserver:self selector:@selector  
(kvStoreDidChange:)  
name:NSUbiquitousKeyValueStoreDidChangeExternallyNotification object:nil];
```

```
// get any change since last launch  
[kvStore synchronize];
```

Making Changes

```
// store values  
[kvStore setObject:someObject forKey:@"someKey"];  
[kvStore setBool:YES forKey:@"someOtherKey"];
```

```
// write values on disk
```

```
[kvStore synchronize];
```


Getting the Notification

```
- (void)kvStoreDidChange:(NSNotification *)notification
{
    NSDictionary* userInfo = [notification userInfo];

    // get change reason (initial download, external change or quota
    violation change)
    int reason = [[userInfo
objectForKey:NSUbiquitousKeyValueStoreChangeReasonKey] intValue];

    // get the affected keys
    NSArray* changedKeys = [userInfo
objectForKey:NSUbiquitousKeyValueStoreChangedKeysKey];

    // use the new values
}
```


Building Your iCloud App

- Update your provisioning profile with Xcode 4
- Add the iCloud Storage entitlements to your application
 - `com.apple.developer.ubiquity-container-identifiers`
 - `com.apple.developer.ubiquity-kvstore-identifier`

Using iCloud with Other Apps

- You may have a Lite and Pro version of your app
- You may have a suite of apps that work together
- The container entitlement allows access between apps
- Other files available via the same iCloud Storage API
- You can manage how your quota is displayed
 - Use a display set

Quota



- 5GB free
- Manage quota through Settings
- Quota is shared across:
 - Mail
 - Backup
 - iCloud Storage documents
- Delete individual documents or an entire app's data

Things to Consider

- Avoid saving if nothing changed
- Avoid publishing private information
 - Undo stacks
 - Use the Keychain API for passwords and credentials
- Be generous in what you receive
- Respect other file formats
- Don't store autogenerated data in iCloud
- Use Key Value data where appropriate

Recap

- iCloud Storage
 - Let the OS move it for you
 - Available on iOS, Lion, and Windows later this year
- iCloud Storage APIs
 - UIDocument
 - NSDocument
 - NSFileCoordinator
 - NSFilePresenter
 - NSFileVersion
 - CoreData
- Key Value Data

Related Sessions

Auto Save and Versions in Mac OS X Lion

Pacific Heights
Tuesday 3:15PM

What's New in Core Data on iOS

Mission
Tuesday 3:15PM

Taking Advantage of File Coordination

Pacific Heights
Tuesday 4:30PM

Storing Documents in iCloud Using iOS 5

Presidio
Wednesday 3:15PM

Securing Application Data

Nob Hill
Thursday 9:00AM

What's New in Core Data on Mac OS X

Nob Hill
Thursday 11:30AM

Labs

Cocoa Lab	App Frameworks Lab A Tuesday 2:00PM
iCloud Storage Lab	Internet & Web Lab B Tuesday 2:00PM
Core Data Lab	Developer Tools Lab B Tuesday 4:30PM
Cocoa Touch Lab	App Frameworks Lab D Wednesday 2:00PM
Core Data Lab	Developer Tools Lab B Wednesday 4:30PM
UIDocument Lab	App Frameworks Lab D Thursday 9:00AM
Security Lab	Core OS Lab B Thursday 11:30AM
iCloud Storage Lab	Internet & Web Lab A Thursday 11:30AM
Cocoa, Autosave, File Coordination, and Resume Lab	App Frameworks Lab A Thursday 2:00PM
App Publishing with iTunes Connect Lab	Internet & Web Lab A Thursday 2:00PM
Security Lab	Core OS Lab B Friday 11:30AM

