Implementing Local and Push Notifications

Maybe you don't need multitasking after all...

Jacob Farkas
iPhone Software Engineer

The iPhone App Dilemma

- App isn’t always running
- App has an important message

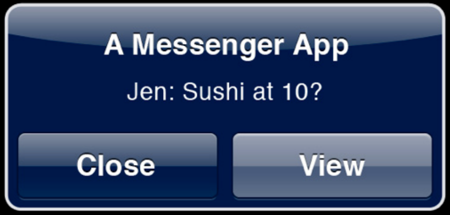
Three new crops are ready on your farm

The baseball game is about to start

John has sent you a message

What are Notifications?

Ways to notify your users


A Messenger App

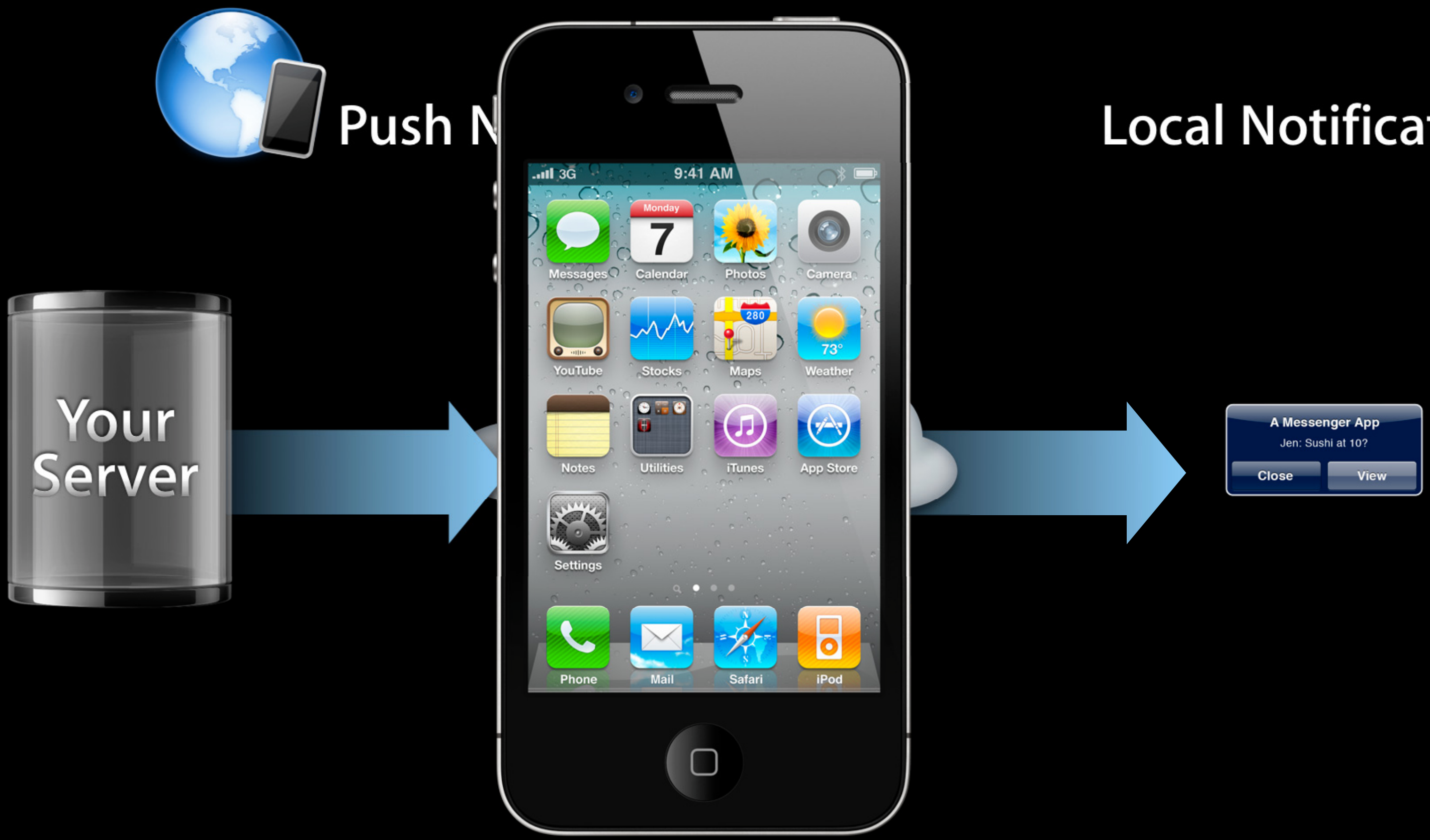
Jen: Sushi at 10?

Badges

Alerts

Sounds

Notification Methods



Local Notifications vs. Push Notifications

How they are similar

Push Notifications

Local Notifications

Appearance

iOS acts on behalf of your app

Local Notifications vs. Push Notifications

How they are different

Push Notifications	Local Notifications
Originate from server	Originate from your app
Connect with a network service	Scheduled
Single shot	Repeatable

Why Use Notifications?

- Alert the user
- Saves battery
- Connect with a network service

Push Notifications

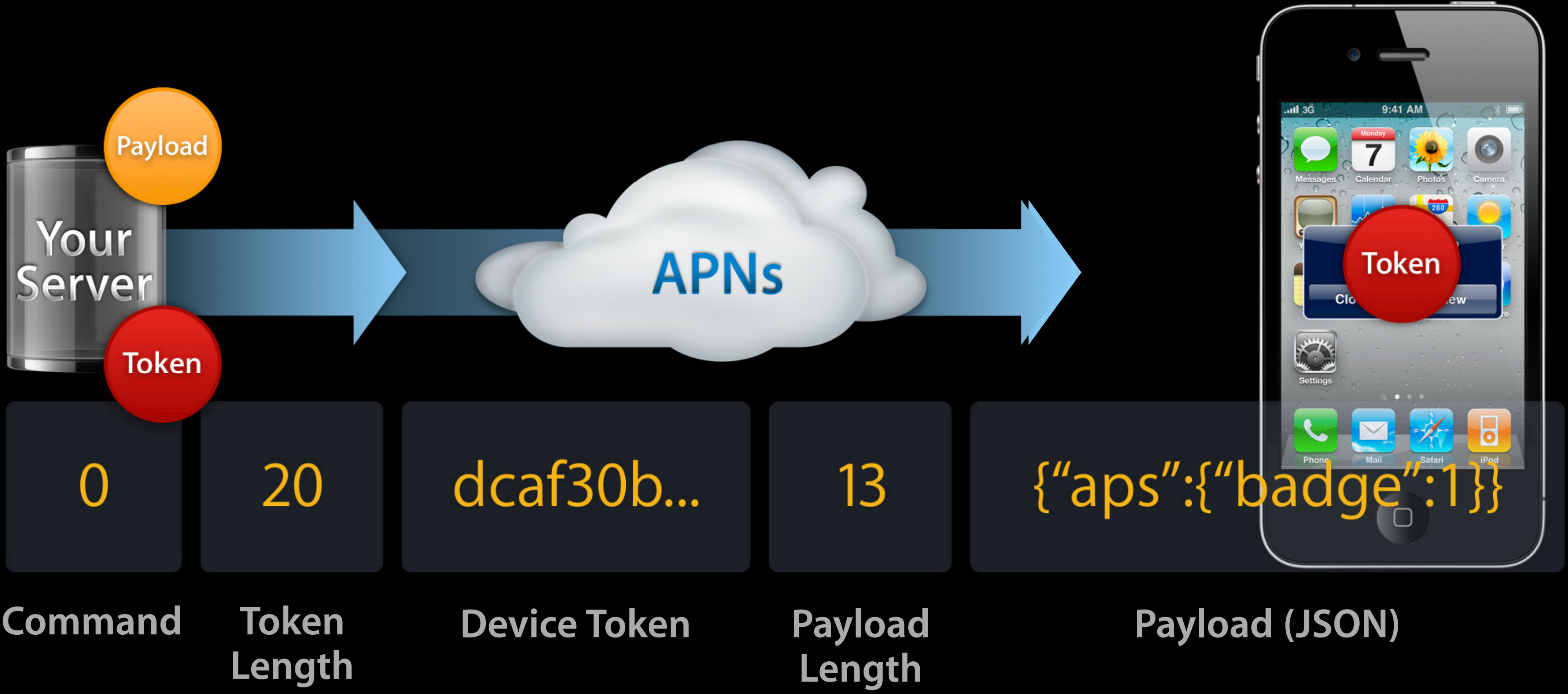
Service review, enhancements

Darryl Bleau
APNs Server Engineer

Push Notifications
Overview

- Notification architecture review
- Service enhancements introduction

Push Notification Service Architecture



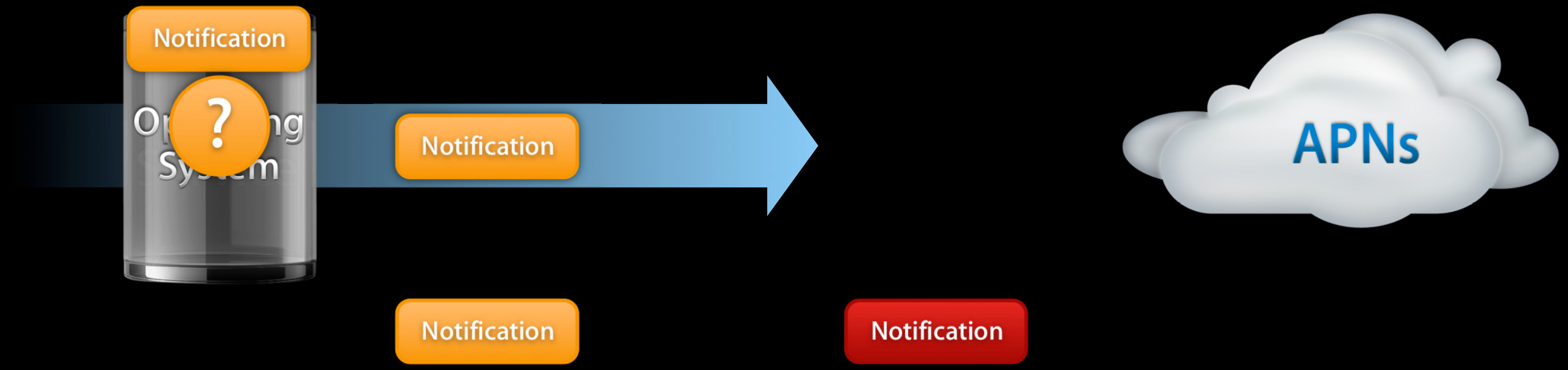
0	20	dcaf30b...	13	{"aps":{"badge":1}}
Command	Token Length	Device Token	Payload Length	Payload (JSON)

Sending Notifications

Considerations

- Designed for performance
  - Unidirectional Communication
  - Streaming Protocol
- Features store and forward
  - Reconnecting devices receive most recent offline notification
- Debugging challenges
  - Breaks connection in response to unintelligible or invalid input

Binary Interface

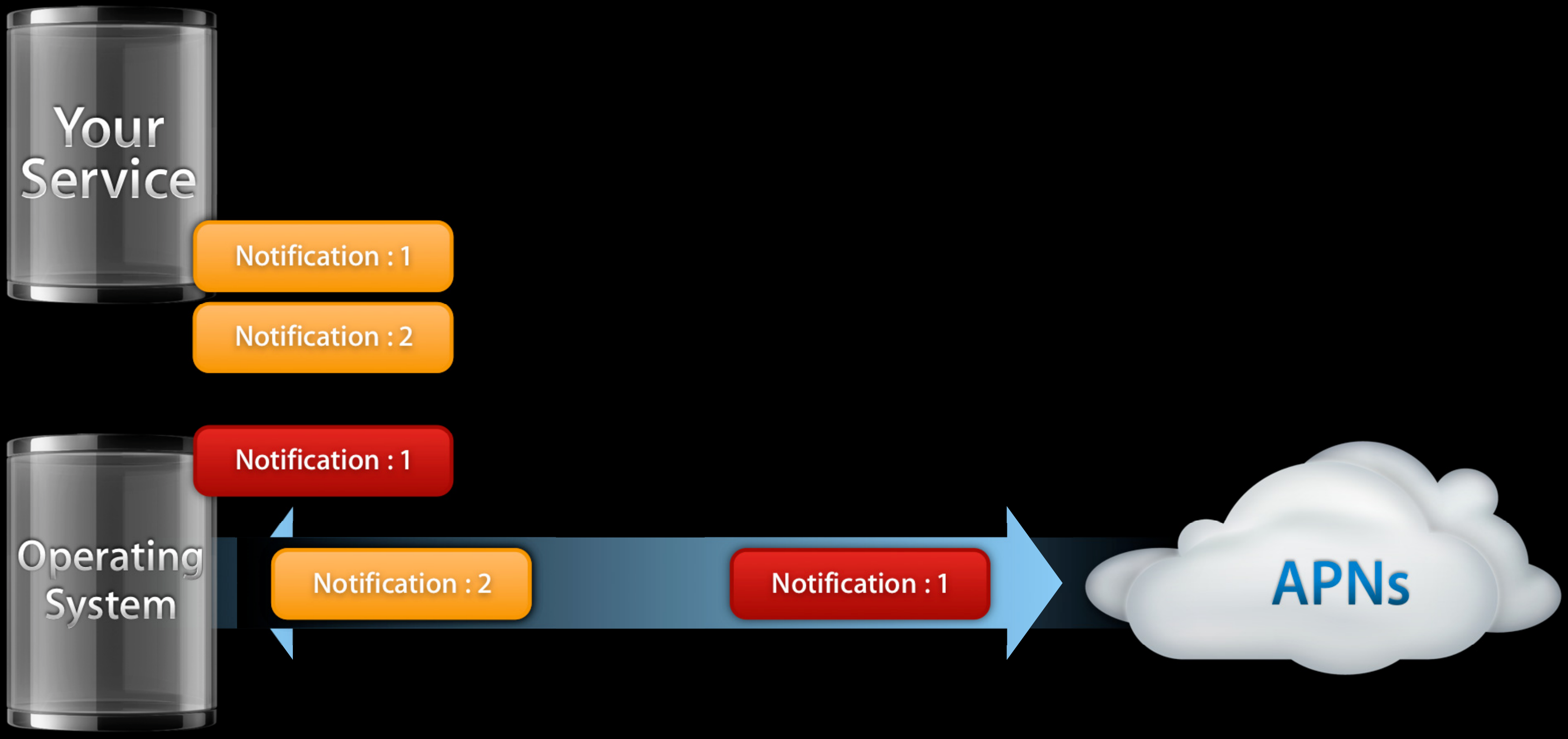


Enhanced Binary Interface

Increased feedback

- Preserves what’s best
  - High performance
  - Streaming capabilities
- Concise feedback
  - A request/response system would impair performance
  - Designed to provide response on error

Enhanced Binary Interface



Binary Interface

Store and forward

- Accepts notifications for offline devices
  - Stored for a limited time
  - Delivered when device is back online
- Notification information may be time sensitive
  - May be delivered after too much time has passed
  - Dutifully delivered potentially days after being useful

Enhanced Binary Interface

Notification expiration

- Specify a maximum useful lifetime for a notification
- Won’t deliver notifications if offline device reconnects after expiry
- Attempts to deliver at least one time
- Specified lifetime still cannot exceed the default expiry

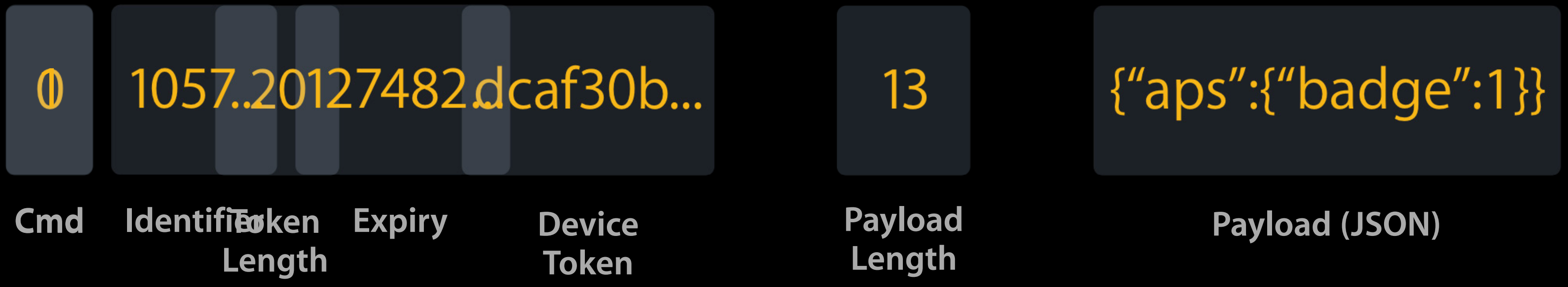
Enhanced Binary Interface

Notification expiry considerations

- Expiry is specified as seconds since epoch, in UTC
- Existing message command essentially means ‘as long as you can’
- Expects time coordination
  - Network Time Protocol

Enhanced Binary Interface

Putting it all together



Receiving Notifications on Wi-Fi

Push or poll?

- Prior to iOS 4, sleeping devices on Wi-Fi networks resorted to polling behavior to receive notifications
- iOS 4 fully supports push notifications over Wi-Fi connections
    - Requires iPod, iPad, or iPhone 3GS or newer

Push Notifications

Summary

- Enhanced Binary Interface
  - Alleviates unknown disconnection questions for developers
  - Allows control over message lifetime
  - Available now
- Enhanced Device Support
  - Support Push over Wi-Fi with iOS 4 on iPod, iPad, and iPhone 3GS or newer

Implementing Local Notifications

What Are Local Notifications?

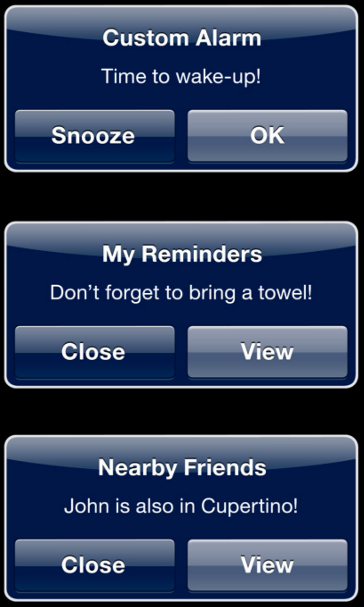
iOS 4

- New to iOS 4
- Scheduled by your app
- Repeatable
- Cancelable

Using Local Notifications

When to use them

- Alarm Clock
- Reminder
- Location



Using Local Notifications

When not to use them

- UIAlertView for errors
- EventKit if possible

Creating Local Notifications

Show us your code

UILocalNotification

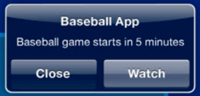
Badge

```
NSInteger applicationIconBadgeNumber
```

Sound

```
NSString *soundName
```

Alert



```
NSString *alertBody
BOOL hasAction
NSString *alertAction
NSString *alertLaunchImage
```

Scheduling

```
NSDate *fireDate
NSTimeZone *timeZone
```

Repeating

```
NSCalendarUnit repeatInterval
NSCalendar *repeatCalendar
```

Metadata

```
NSDictionary *userInfo
```

Creating Local Notifications

Badges

- Use the `applicationIconBadgeNumber` property

```
UILocalNotification *note =
  [[UILocalNotification alloc] init];
note.applicationIconBadgeNumber = 3;
```

- Setting to zero will **not** clear the badge

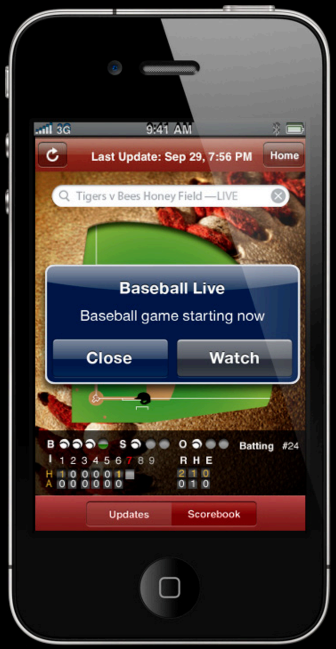
```
-[UIApplication applicationIconBadgeNumber]
```

Creating Local Notifications

Alerts

- Set the `alertBody` and `action` properties

```
UILocalNotification *note =
  [[UILocalNotification alloc] init];
note.alertBody = @”Baseball game starting now”;
note.hasAction = YES;
note.alertAction = @”Watch”;
```


Last Update: Sep 29, 7:56 PM
Home
Tigers v Bees Honey Field —LIVE
Baseball Live
Baseball game starting now
Close
Watch
Batting #24
Updates
Scorebook

Creating Local Notifications

A word on localization…

- Localize!

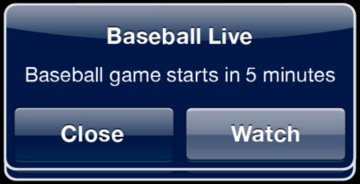
English.lproj/Localizable.strings

```
“MSG_POSTED” = “Baseball game starts in 5 minutes”
“SHOW_KEY” = “Watch”
```

French.lproj/Localizable.strings

```
“MSG_POSTED” = “Le match de baseball commence dans 5 minutes”
“SHOW_KEY” = “Regarder”
```

```
UILocalNotification *note =
  [[UILocalNotification alloc] init];
note.alertBody = @”MSG_POSTED”;
note.hasAction = YES;
note.alertAction = @”SHOW_KEY”;
```

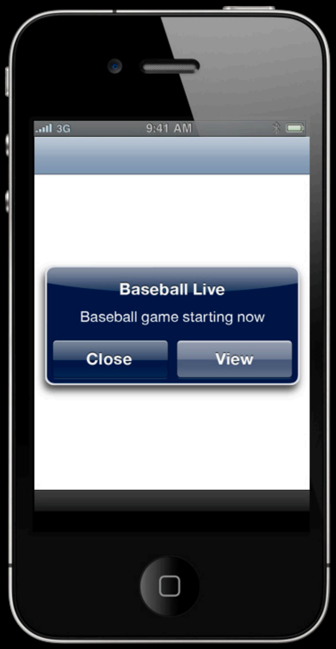

Baseball Live

Baseball game starts in 5 minutes

Creating Local Notifications

Launch images

- `alertLaunchImage` will change your app's launch image

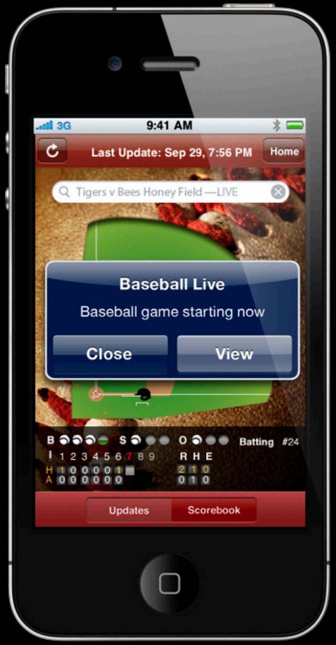


Creating Local Notifications

Launch images

- `alertLaunchImage` will change your app's launch image

```
note.alertLaunchImage = @"Default-watch.png";
```



3G
9:41 AM
Last Update: Sep 29, 7:56 PM
Home
Tigers v Bees Honey Field —LIVE
Baseball Live
Baseball game starting now
Close
View
Batting #24
Updates
Scorebook

Creating Local Notifications

Sounds

- soundName: a sound file in your app's bundle

```
UILocalNotification *note = [[UILocalNotification alloc] init];
note.soundName = @"MyAlert.aiff";
```

- UILocalNotificationDefaultSoundName plays a default sound

```
UILocalNotification *note = [[UILocalNotification alloc] init];
note.soundName = UILocalNotificationDefaultSoundName;
```

Creating Local Notifications

Scheduling local notifications

UILocalNotification

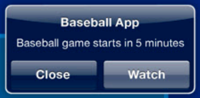
Badge

```
NSInteger applicationIconBadgeNumber
```

Sound

```
NSString *soundName
```

Alert


Baseball App

Baseball game starts in 5 minutes

```
NSString *alertBody
BOOL hasAction
NSString *alertAction
NSString *alertLaunchImage
```

Scheduling

```
NSDate *fireDate
NSTimeZone *timeZone
```

Repeating

```
NSCalendarUnit repeatInterval
NSCalendar *repeatCalendar
```

Metadata

```
NSDictionary *userInfo
```

Creating Local Notifications

Scheduling local notifications

- Use the `fireDate` and `timeZone` properties

```
UILocalNotification *note = [[UILocalNotification alloc] init];
NSCalendar *calendar = [NSCalenda[illegible]urrentC[illegible]ndar];
NSDateComponents *dateComps = [[NS[illegible]eCom[illegible]ents alloc] init];

[dateComps setDay:10];
[dateComps setMonth:6];
[dateComps setYear:2010];
[dateComps setHour:14];

note.fireDate = [calendar dateFromComponents:dateComps];
note.timeZone = [calendar timeZone];
```

A Quick Word About Dates…

A Word About Dates

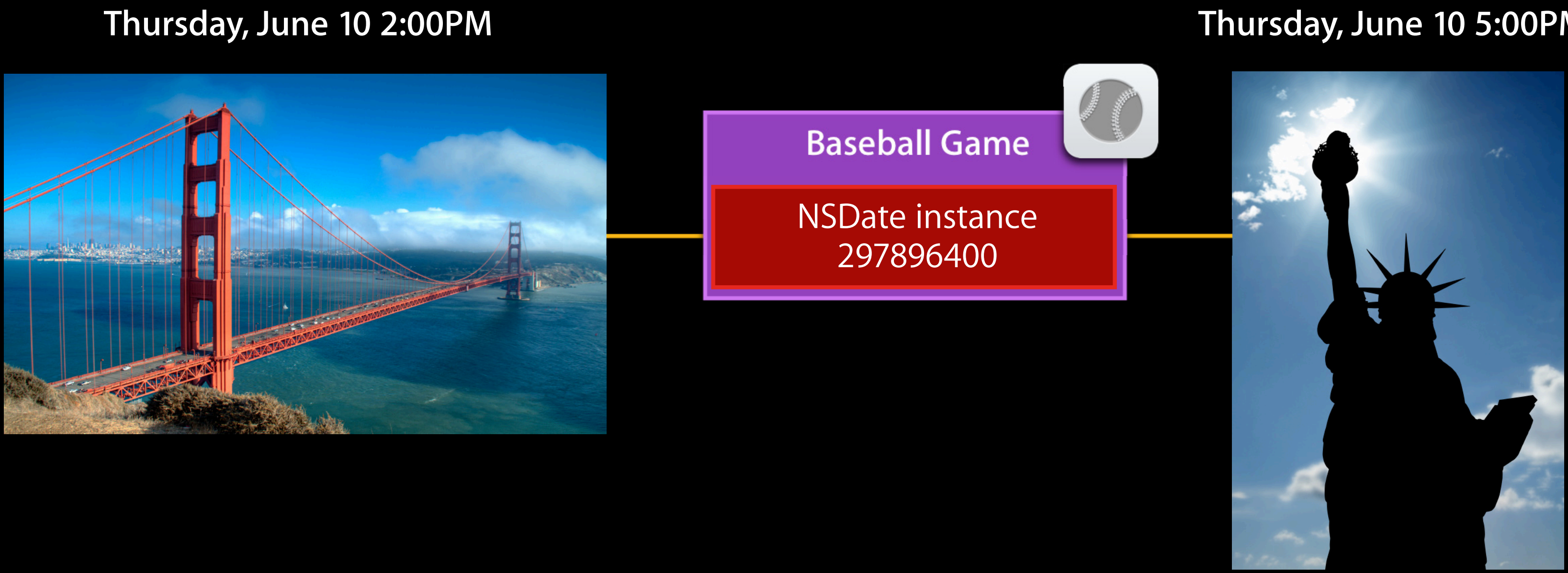
Time is all we need

- “Universal” time
- “Wall” time

A Word About Dates

Universal times

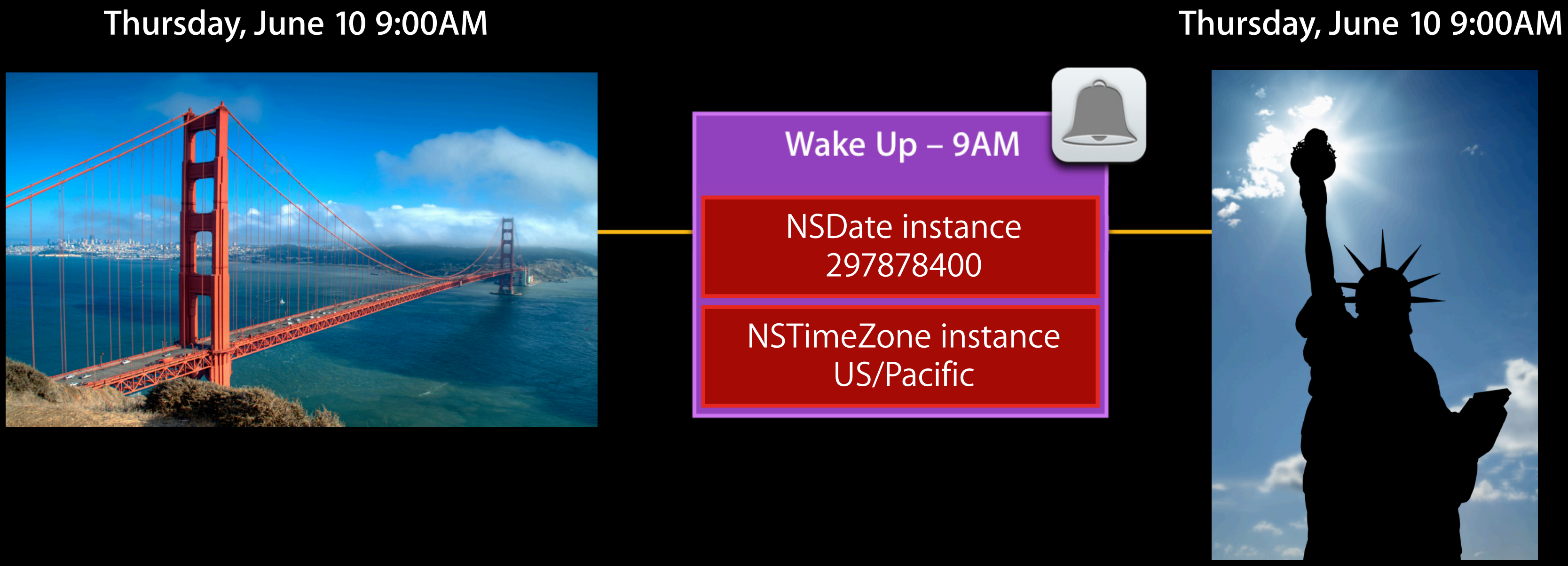
- Lonely NSDate = Universal Time



A Word About Dates

Wall time

- NSTimeZone + NSDate = wall time
- Adjusted based on current time zone



A Word About Dates

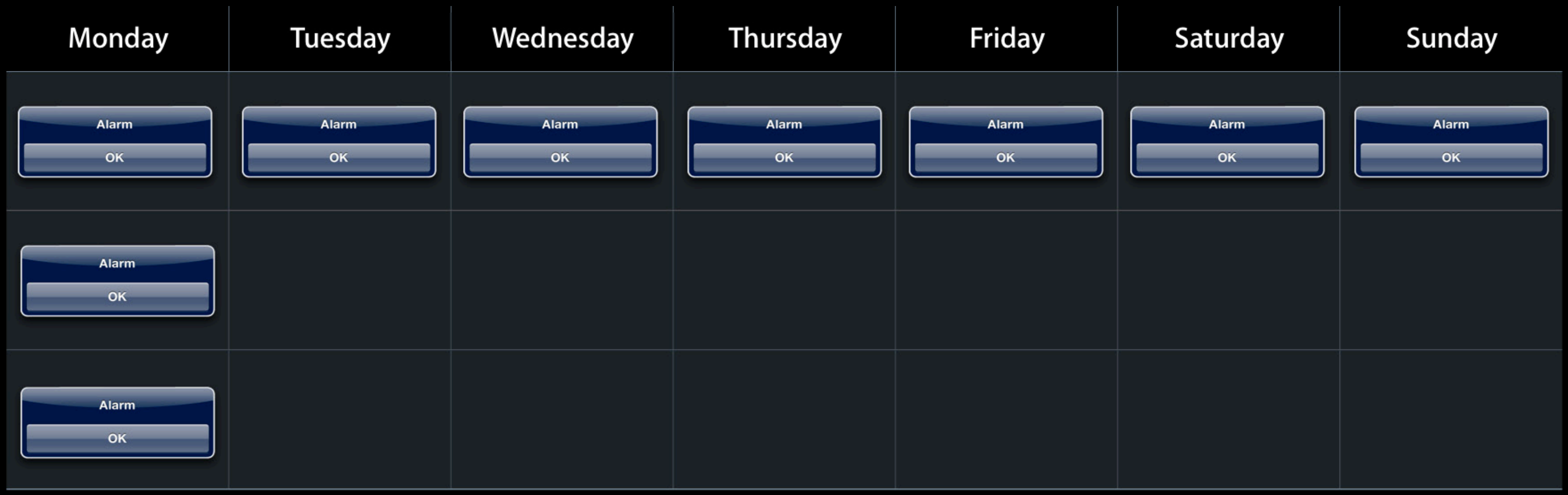
- No time zone: Universal Time
  - Baseball game (2:00PM PST/5:00PM EST)
  - Conference call
  - Stock Market Close
- With a time zone: Wall Time
  - 9:00AM alarm
  - New Years Eve
  - TV show (Thursdays at 8:00PM)

Creating Local Notifications

Scheduling repeating local notifications

- Use `repeatInterval` and `repeatCalendar`

```
note.repeatInterval = NSWeekCalendarUnit;
note.repeatCalendar = [NSCalendar currentCalendar];
```



Creating Local Notifications

Scheduling local notifications

UILocalNotification

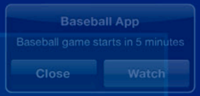
Badge

```
NSInteger applicationIconBadgeNumber
```

Sound

```
NSString *soundName
```

Alert



Baseball App

Baseball game starts in 5 minutes

```
NSString *alertBody
BOOL hasAction
NSString *alertAction
NSString *alertLaunchImage
```

Scheduling

```
NSDate *fireDate
NSTimeZone *timeZone
```

Repeating

```
NSCalendarUnit repeatInterval
NSCalendar *repeatCalendar
```

Metadata

```
NSDictionary *userInfo
```

Creating Local Notifications

Scheduling local notifications

UILocalNotification

Badge

```
NSInteger applicationIconBadgeNumber
```

Sound

```
NSString *soundName
```

Ale

```
NSString *al
BOOL hasActi
NSString *al
NSString *al
```

Scheduling

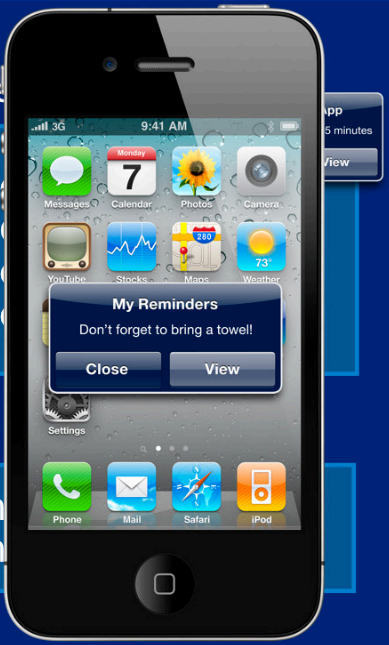
```
NSDate *fireDate
NSTimeZone *timeZone
```

Repeating

```
NSCalendarUnit repeatIn
NSCalendar *repeatCalen
```

Metadata

```
NSDictionary *userInfo
```



My Reminders

Don't forget to bring a towel!

Creating Local Notifications

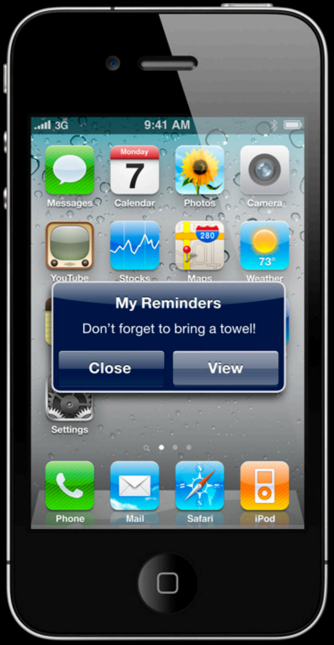
Scheduling local notifications

- Schedule with UIApplication

```
– (void)scheduleLocalNotification:
    (UILocalNotification *)notification;
```

- Cancel a notification

```
– (void)cancelLocalNotification:
    (UILocalNotification *)notification;
```



My Reminders

Don't forget to bring a towel!

Creating Local Notifications

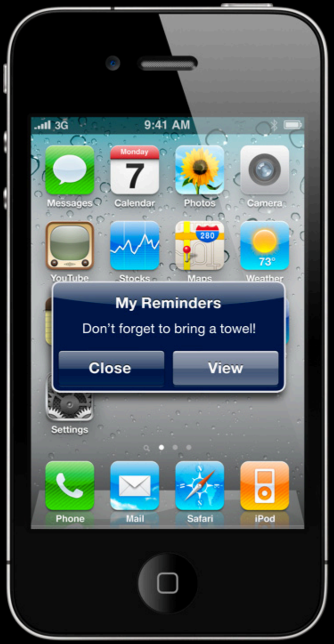
Scheduling local notifications

- All notifications

```
– (NSArray *)scheduledLocalNotifications;
```

- Cancel all notifications

```
– (void)cancelAllLocalNotifications;
```



Creating Local Notifications

Scheduling local notifications

- NSCoding for serialization

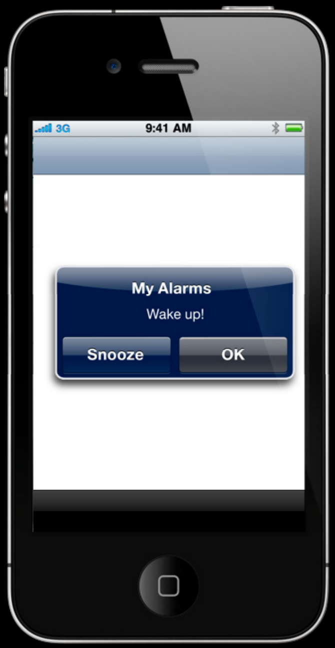
UILocalNotification
Database

Handling Local Notifications

Application not running

- iPhone OS handles the notification

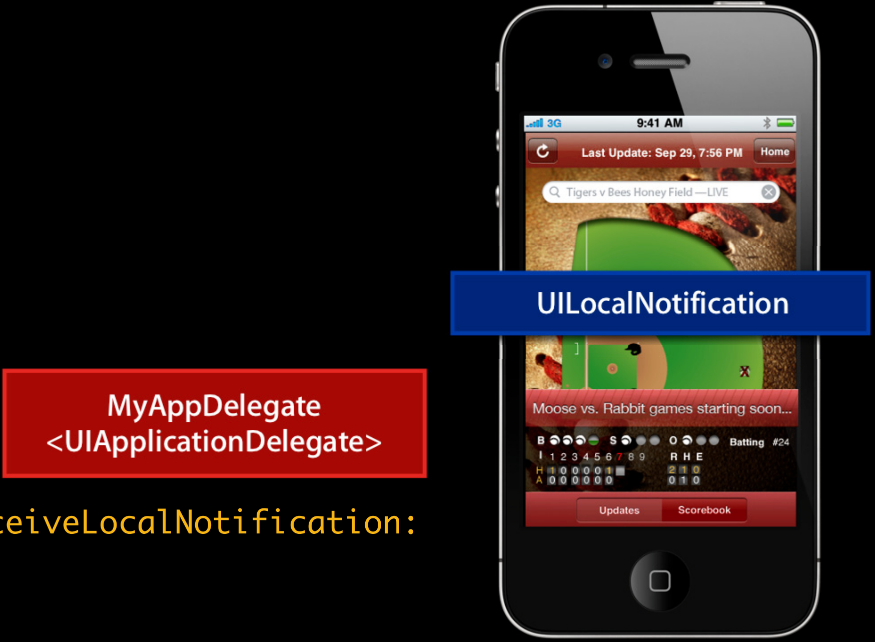
MyAppDelegate
<UIApplicationDelegate>



Handling Local Notifications

Foreground application

- Show custom UI


UILocalNotification
MyAppDelegate <UIApplicationDelegate>
Last Update: Sep 29, 7:56 PM
Home
Tigers v Bees Honey Field —LIVE
Moose vs. Rabbit games starting soon...
Updates
Scorebook

`application:didReceiveLocalNotification:`

Handling Local Notifications
Running in the background

- Special cases for background apps
  - Location
  - VoIP
  - Audio
- Schedule a notification now

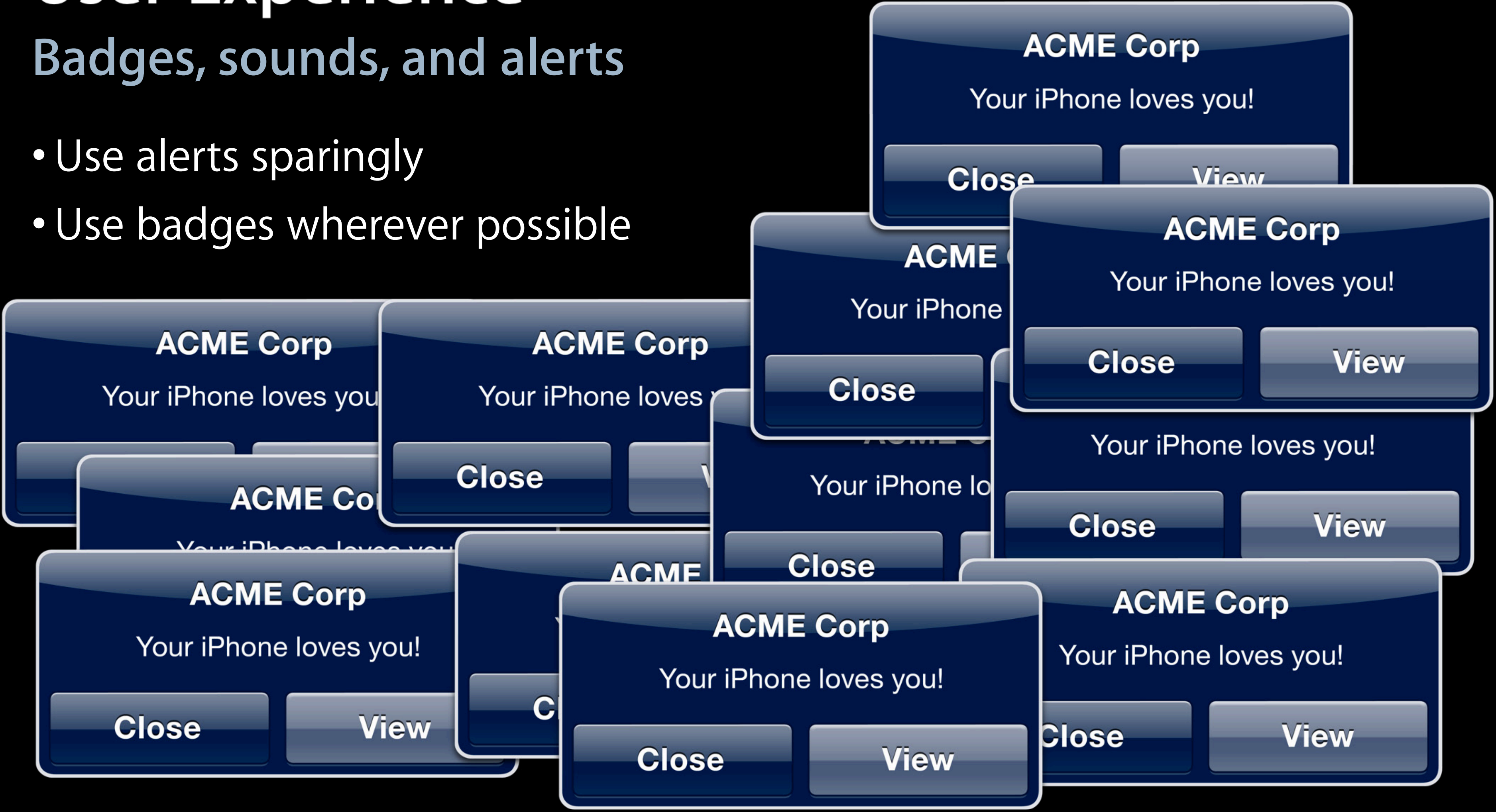
```
– (void)scheduleLocalNotificationNow:(UILocalNotification *)notification;
```

- User can launch your app
- Not for most apps

User Experience

Badges, sounds, and alerts

- Use alerts sparingly
- Use badges wherever possible



Demo

Chris Marcellino
iPhone Software Engineer

Summary

- Background app functionality
- Push: network
- Local: scheduled
- Localization, localization, localization
- Universal versus wall time
- Foreground apps display their own UI

More Information

Mark Malone
Integration Technologies Evangelist
mgm@apple.com

Apple Developer Forums
http://devforums.apple.com

Related Sessions

Adopting Multitasking on iPhone OS, Part 1 (Repeat)	Marina Friday 9:00AM
Calendar Integration with Event Kit	Mission Thursday 4:30PM
Using Core Location in iOS 4 (Repeat)	Pacific Heights Thursday 10:15AM

Labs

Local and Push Notifications Lab	Application Frameworks Lab A Thursday 4:30PM–6:00PM

Q&A

WWDC10

[redacted]