Advances in AirPrint

Session 711

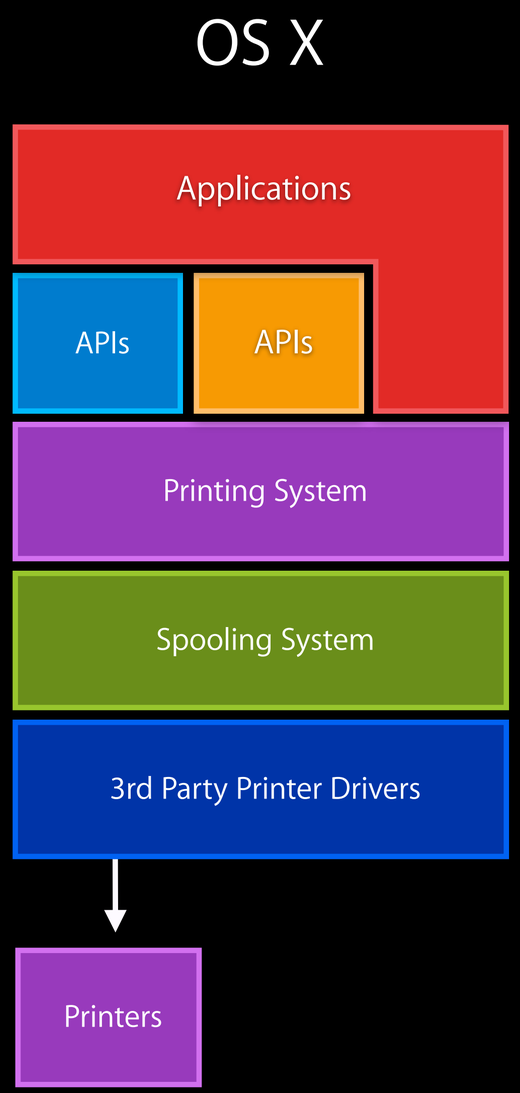
Howard Miller
Printing Engineering

Agenda

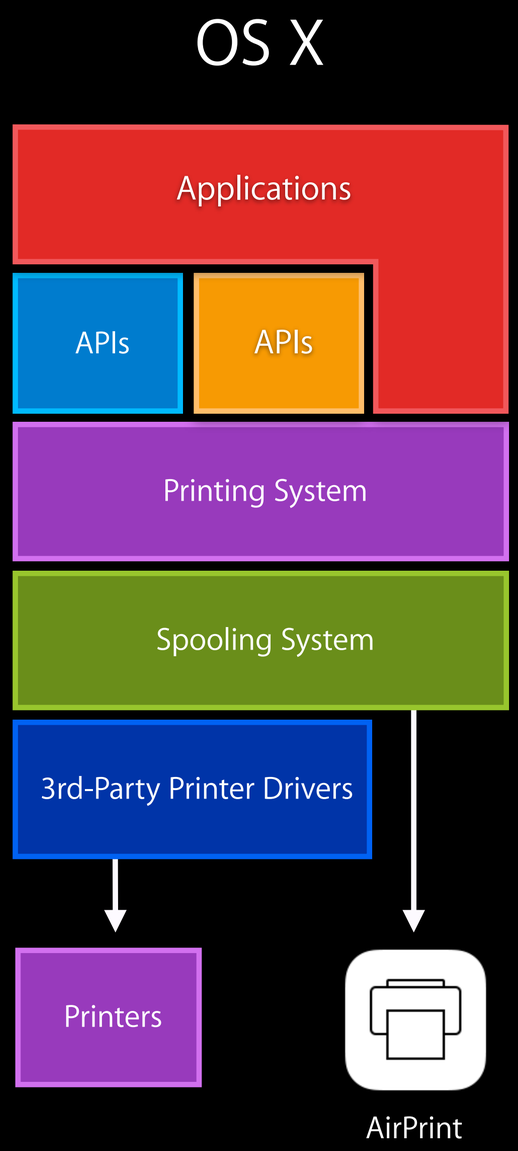
- AirPrint overview
- AirPrint in enterprise environment
- Adopting printing in your iOS App

Printing System Architecture

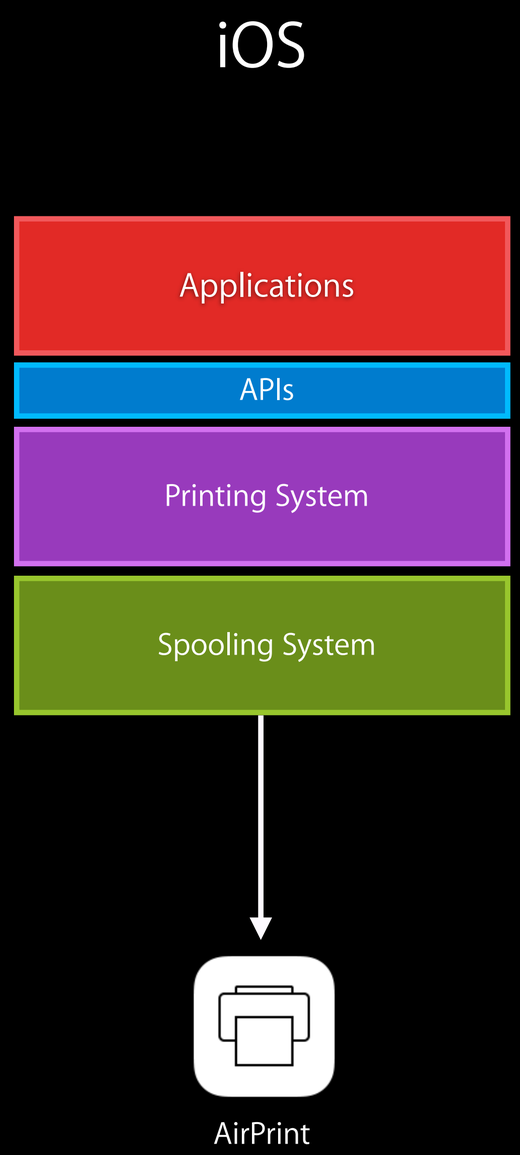
Printing System Architecture



Printing System Architecture



Printing System Architecture



AirPrint Technology

- Great user experience
  - No driver, no software to install
  - Full output quality
- Supported on all Apple platforms
  - iOS 4.2 and later
  - OS X 10.7, default on OS X 10.8 and beyond
- Standards based
- Zero-cost license for printer and server manufacturers

AirPrint Manufacturers

xerox

hp

lenovo

EPSON

Canon

RICOH

LEXMARK

OKI

NRG
RICOH COMPANY

LANIER

Gestetner

SAMSUNG

DELL

savin

infotec

KYOCERA

brother

FUJI xerox

AirPrint Licensees

csr

Sagemcom

PRINICS

MARVELL

TOSHIBA

zink
ZERO INK

hiti

SINDOH

Panasonic
Document Imaging Company

Zebra

Astro-Med, Inc
TEST & MEASUREMENT PRODUCT GROUP

ALPS

SHARP

AirPrint Printers

150 Million+

AirPrint Printers Sold

AirPrint’s Expanding Footprint

AirPrint Basics
http://support.apple.com/kb/ht4356

New AirPrint Product Categories in 2013

New AirPrint Product Categories in 2013

- Roll fed printers
- Enterprise class laser printers

New AirPrint Product Categories in 2013

- Roll fed printers
- Enterprise class laser printers
- Label printers
- Dye-sub photo printers

New AirPrint Product Categories in 2013

- Roll fed printers
- Enterprise class laser printers
- Label printers
- Dye-sub photo printers
- Consumer servers
- Enterprise class servers

AirPrint Enterprise Features

- Discovery
  - Bonjour for local network
  - DNS support for managed networks
  - iOS Profile support (new in iOS 7)
- Security
  - IPPS (Secure IPP with TLS) Standard
  - User/password support

AirPrint Printer and Server Enterprise Features

- High-end laser and servers
  - Finishing options
  - Print and release
  - Quota

OS X Printing

OS X 10.9 Printing

- No changes required
- No new printing APIs

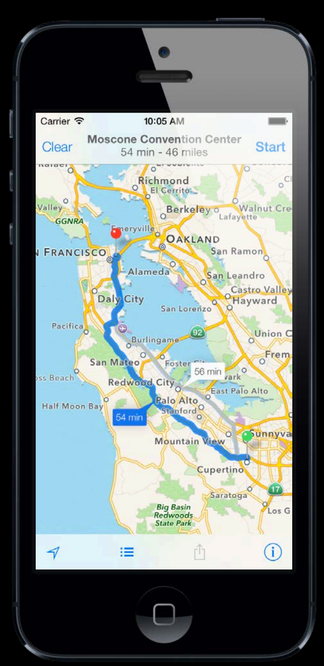
iOS Printing

Format for Print Not Screen

- Format for paper, not display
- Enhanced content
- High-quality drawing

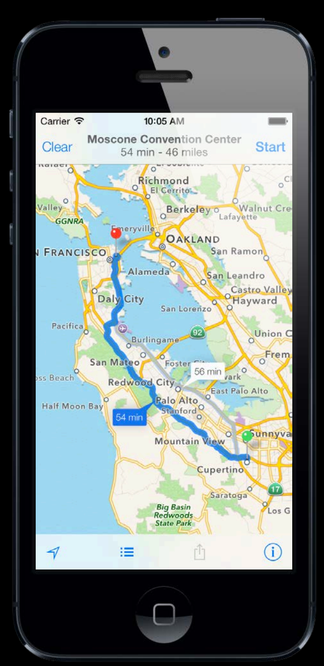
Format for Print Not Screen

- Format for paper, not display
- Enhanced content
- High-quality drawing



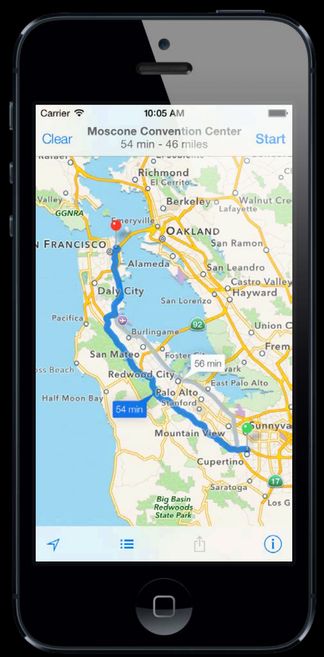
Format for Print Not Screen

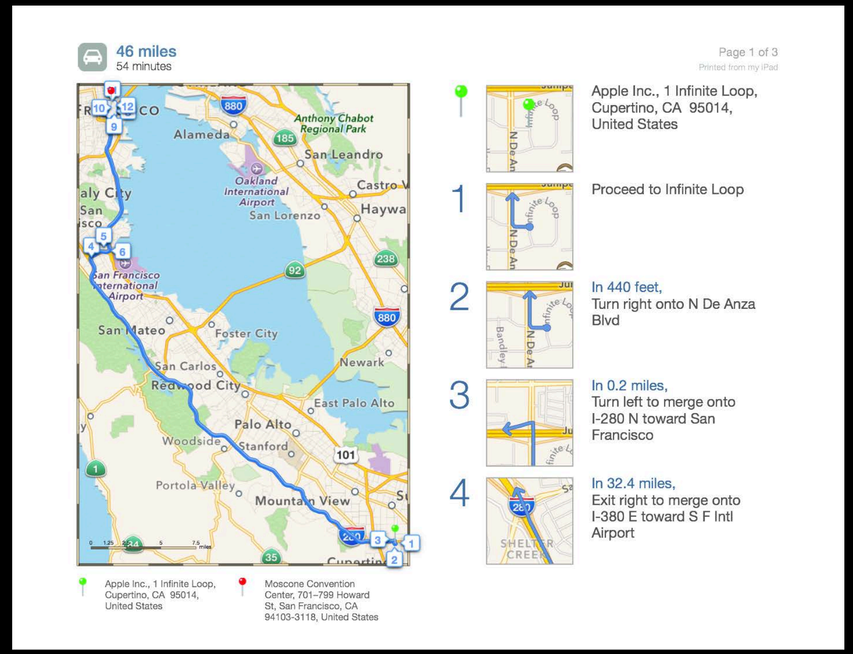
- Format for paper, not display
- Enhanced content
- High-quality drawing



Format for Print Not Screen

- Format for paper, not display
- Enhanced content
- High-quality drawing





Two Paths to Print

- Your app already has well-formed content
  - PDF, JPEG, etc.
  - Hand these to the printing system and we will do the rest
- Your app will draw the content at print time
  - Fine grained control of your output
  - Leverage existing iOS drawing methods

iOS Printing

Todd Ritland
AirPrint Engineer

Agenda

- Picking what to print
- API overview
- Printing UI options
- New types of printers
- Demo of roll fed sample app

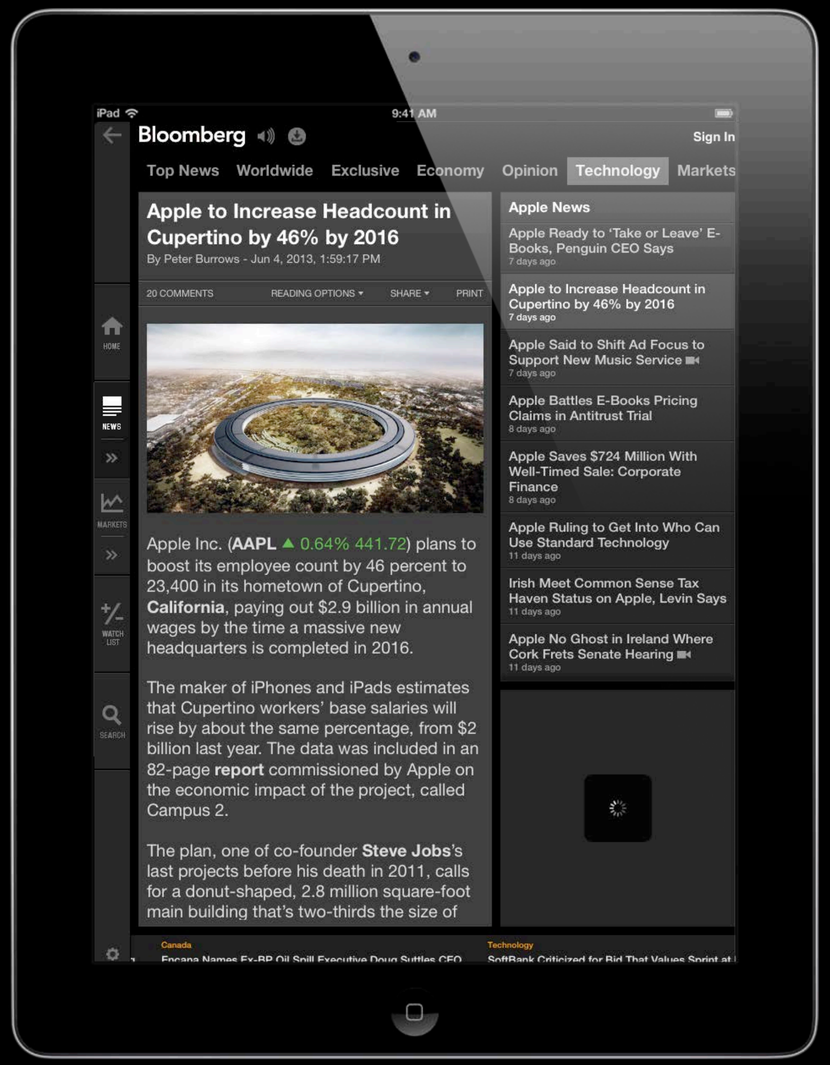
iOS Printing Is Easy

Providing Good Content for Printing

- Useful, attractive, and high quality
- What looks good on screen doesn't always look good on paper
- Make use of the dynamic printing system
  - Paper size can be anything
  - Printer hardware margins vary
  - Best not to produce a fixed size PDF

Example

Bloomberg for iPad



Example

Bloomberg for iPad

Bloomberg

Apple to Increase Headcount in Cupertino by 46% by 2016

Peter Burrows - Jun 4, 2013, 1:59:17 PM

Jobs's New Headquarters for Apple to Add 7,400 Jobs in Cupertino

Apple Inc. (AAPL) plans to boost its employee count by 46 percent to 23,400 in its hometown of Cupertino, **California**, paying out $2.9 billion in annual wages by the time a massive new headquarters is completed in 2016.

The maker of iPhones and iPads estimates that Cupertino workers' base salaries will rise by about the same percentage, from $2 billion last year. The data was included in an 82-page report commissioned by Apple on the economic impact of the project, called Campus 2.

The plan is for only six structures to be visible when it's all done: the headquarters, the lobby to an underground auditorium (no more schlepping to San Francisco for product demos), a four-story parking garage separating the campus from Interstate 280, a fitness center, and two research and development labs housing testing facilities such as an anechoic chamber for analyzing antenna signals.

"The overall feeling of the place is going to be a zillion times better than it is now," Jobs told the Cupertino City Council.

To achieve its goals of a "net-zero energy" campus, the roof of the spaceship will hold 700,000 square feet of solar panels, enough to generate 8 megawatts of power. (That's enough to power roughly 4,000 homes.) Apple says it's negotiating contracts for additional solar and wind power.

The main building will also be groundbreaking in how it's assembled. While the structural shell will be erected on site, the glass that forms the exterior walls will be bent and framed by Seele GmbH in its factory in Gersthofen, Germany.

Example

Bloomberg for iPad

Bloomberg

Apple to Increase Headcount in Cupertino by 46% by 2016

Peter Burrows - Jun 4, 2013, 1:59:17 PM

Jobs's New Headquarters for Apple to Add 7,400 Jobs in Cupertino

Apple Inc. (AAPL) plans to boost its employee count by 46 percent to 23,400 in its hometown of Cupertino, **California**, paying out $2.9 billion in annual wages by the time a massive new headquarters is completed in 2016.

The maker of iPhones and iPads estimates that Cupertino workers' base salaries will rise by about the same percentage, from $2 billion last year. The data was included in an 82-page report commissioned by Apple on the economic impact of the project, called Campus 2.

The plan is for only six structures to be visible when it's all done: the headquarters, the lobby to an underground auditorium (no more schlepping to San Francisco for product demos), a four-story parking garage separating the campus from Interstate 280, a fitness center, and two research and development labs housing testing facilities such as an anechoic chamber for analyzing antenna signals.

"The overall feeling of the place is going to be a zillion times better than it is now," Jobs told the Cupertino City Council.

To achieve its goals of a "net-zero energy" campus, the roof of the spaceship will hold 700,000 square feet of solar panels, enough to generate 8 megawatts of power. (That's enough to power roughly 4,000 homes.) Apple says it's negotiating contracts for additional solar and wind power.

The main building will also be groundbreaking in how it's assembled. While the structural shell will be erected on site, the glass that forms the exterior walls will be bent and framed by Seele GmbH in its factory in Gersthofen, Germany.

The plan, one of co-founder **Steve Jobs**'s last projects before his death in 2011, calls for a donut-shaped, 2.8 million square-foot main building that's two-thirds the size of the Pentagon, featuring curved 40-foot exterior walls made of concave glass from Germany. Apple would add 6,000 trees and hide almost all the roads and parking spaces underground.

The campus will also generate 9,200 construction jobs and $38.1 million in one-time construction taxes and fees for the city, according to the report. Taxes paid by Apple generated 18 percent of Cupertino's general fund in 2012, according to the report, which was prepared by consulting firm Keyser Marston Associates Inc.

To contact the reporter on this story: Peter Burrows in **San Francisco** at **pburrows@bloomberg.net**

To contact the editor responsible for this story: Tom Giles at **tgiles5@bloomberg.net**

API Overview

Basic Steps

- Your app will
  - Get the print controller or activity controller
  - Set up the attributes for the job
  - Provide content to print
  - Present the UI
- iOS will
  - Communicate with AirPrint printer
  - Daemon takes over and manages the job

iOS Printing Classes

```
UIPrintInfo
UIPrintPaper
UIPrintFormatter
   UISimpleTextPrintFormatter
   UIMarkupTextPrintFormatter
   UIViewPrintFormatter
UIPrintPageRenderer

UIPrintInteractionController
UIActivityViewController
```

iOS Printing Classes

UIPrintInfo

UIPrintPaper

UIPrintFormatter

- UISimpleTextPrintFormatter
- UIMarkupTextPrintFormatter
- UIViewPrintFormatter

UIPrintPageRenderer

UIPrintInteractionController

UIActivityViewController

iOS Printing Classes

UIPrintInfo

UIPrintPaper

UIPrintFormatter

- UISimpleTextPrintFormatter
- UIMarkupTextPrintFormatter
- UIViewPrintFormatter

UIPrintPageRenderer

UIPrintInteractionController

UIActivityViewController

iOS Printing Classes

UIPrintInfo

UIPrintPaper

UIPrintFormatter

- UISimpleTextPrintFormatter
- UIMarkupTextPrintFormatter
- UIViewPrintFormatter

UIPrintPageRenderer

UIPrintInteractionController

UIActivityViewController

iOS Printing Classes

UIPrintInfo

UIPrintPaper

UIPrintFormatter

- UISimpleTextPrintFormatter
- UIMarkupTextPrintFormatter
- UIViewPrintFormatter

UIPrintPageRenderer

UIPrintInteractionController

UIActivityViewController

iOS Printing Classes

UIPrintInfo

UIPrintPaper

UIPrintFormatter

- UISimpleTextPrintFormatter
- UIMarkupTextPrintFormatter
- UIViewPrintFormatter

UIPrintPageRenderer

UIPrintInteractionController

UIActivityViewController

iOS Printing Classes

UIPrintInfo

UIPrintPaper

UIPrintFormatter

- UISimpleTextPrintFormatter
- UIMarkupTextPrintFormatter
- UIViewPrintFormatter

UIPrintPageRenderer

UIPrintInteractionController

UIActivityViewController

iOS Printing Classes

`UIPrintInfo`

UIPrintInfo

Setting the Job Name

Setting the Job Name

Setting the Job Name

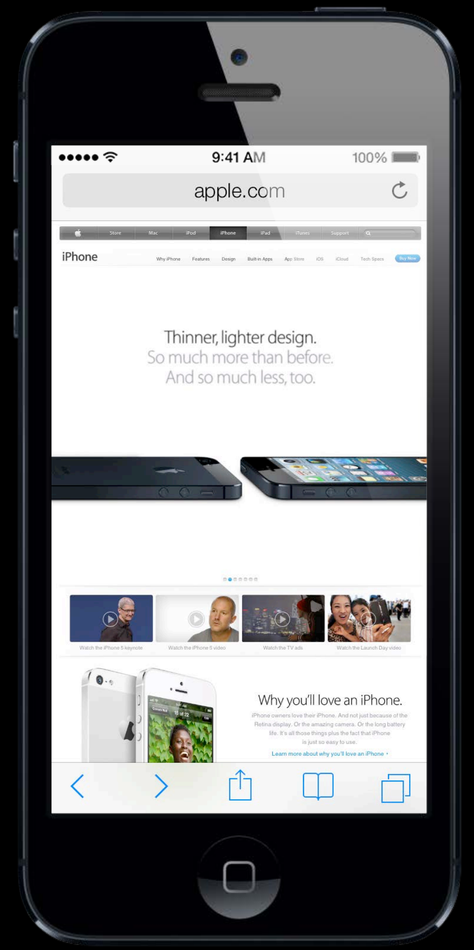
Output Type

- Tell the printing system about the type of content to be printed
- Allows the printing system to the choose appropriate
  - Paper size
  - Print quality mode
  - Appropriate UI

Document

`UIPrintInfoOutputGeneral`

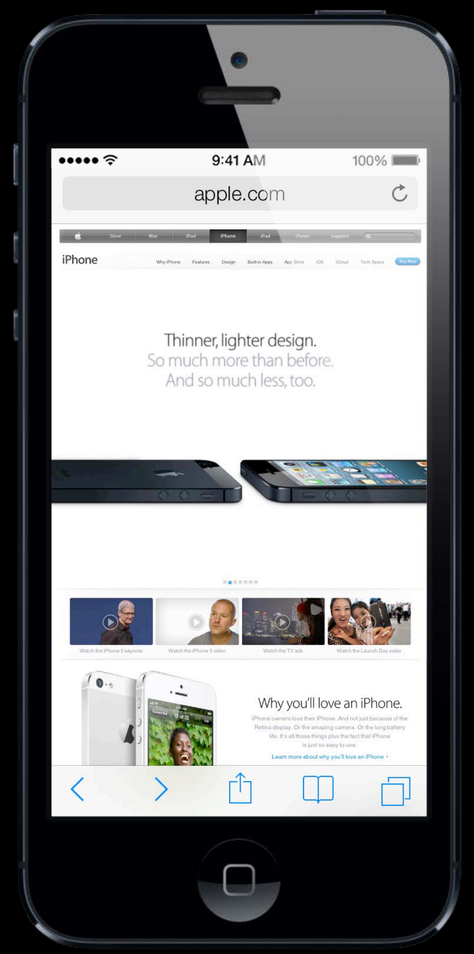
- Mixed text and graphics
- Normal quality
- Document paper size
- Duplex allowed
- Page range allowed

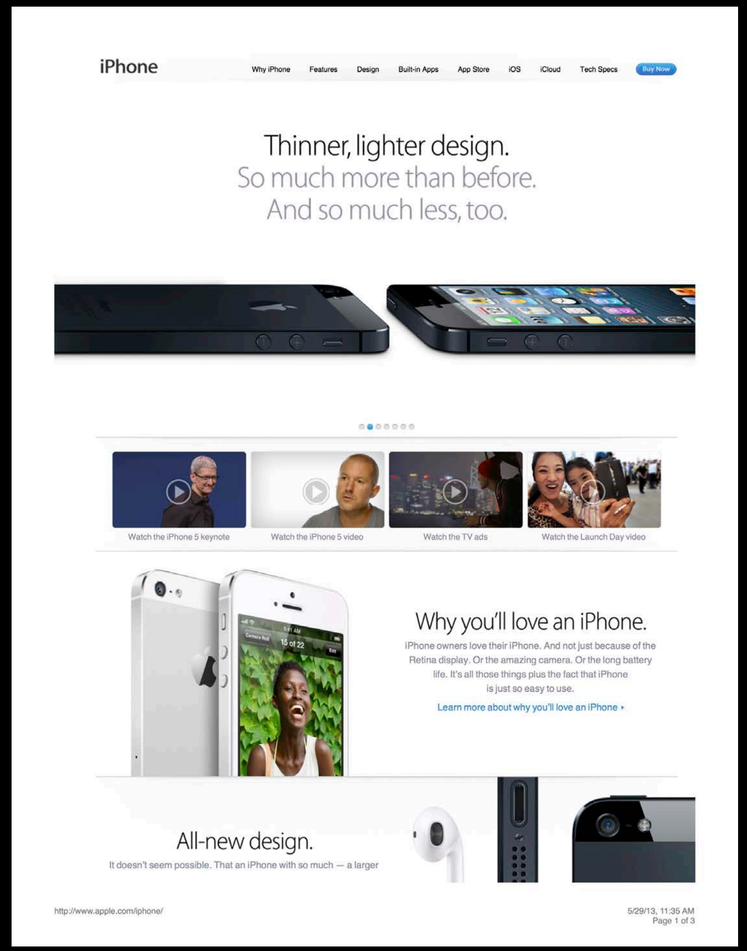


Document

`UIPrintInfoOutputGeneral`

- Mixed text and graphics
- Normal quality
- Document paper size
- Duplex allowed
- Page range allowed





Document Grayscale

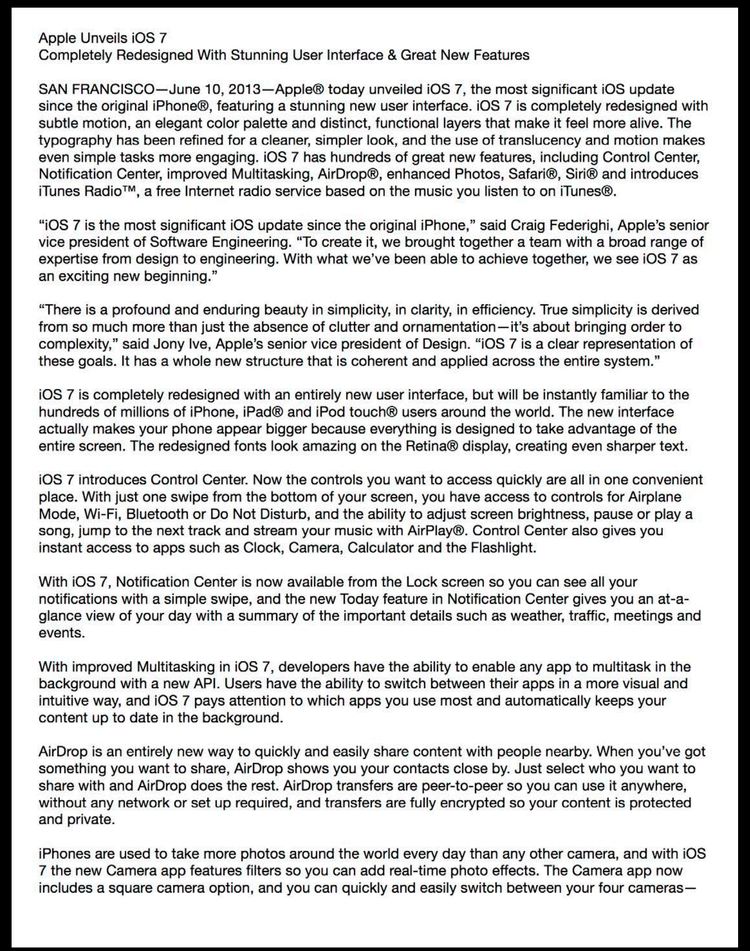
`UIPrintInfoOutputGrayscale`

- Monochrome text and graphics
- Improved print speed
- Reduced ink usage
- Document paper size
- Duplex allowed
- Page range allowed

Document Grayscale

`UIPrintInfoOutputGrayscale`

- Monochrome text and graphics
- Improved print speed
- Reduced ink usage
- Document paper size
- Duplex allowed
- Page range allowed



Apple Unveils iOS 7
Completely Redesigned With Stunning User Interface & Great New Features

SAN FRANCISCO—June 10, 2013—Apple® today unveiled iOS 7, the most significant iOS update since the original iPhone®, featuring a stunning new user interface. iOS 7 is completely redesigned with subtle motion, an elegant color palette and distinct, functional layers that make it feel more alive. The typography has been refined for a cleaner, simpler look, and the use of translucency and motion makes even simple tasks more engaging. iOS 7 has hundreds of great new features, including Control Center, Notification Center, improved Multitasking, AirDrop®, enhanced Photos, Safari®, Siri® and introduces iTunes Radio™, a free Internet radio service based on the music you listen to on iTunes®.

“iOS 7 is the most significant iOS update since the original iPhone,” said Craig Federighi, Apple’s senior vice president of Software Engineering. “To create it, we brought together a team with a broad range of expertise from design to engineering. With what we’ve been able to achieve together, we see iOS 7 as an exciting new beginning.”

“There is a profound and enduring beauty in simplicity, in clarity, in efficiency. True simplicity is derived from so much more than just the absence of clutter and ornamentation—it’s about bringing order to complexity,” said Jony Ive, Apple’s senior vice president of Design. “iOS 7 is a clear representation of these goals. It has a whole new structure that is coherent and applied across the entire system.”

iOS 7 is completely redesigned with an entirely new user interface, but will be instantly familiar to the hundreds of millions of iPhone, iPad® and iPod touch® users around the world. The new interface actually makes your phone appear bigger because everything is designed to take advantage of the entire screen. The redesigned fonts look amazing on the Retina® display, creating even sharper text.

iOS 7 introduces Control Center. Now the controls you want to access quickly are all in one convenient place. With just one swipe from the bottom of your screen, you have access to controls for Airplane Mode, Wi-Fi, Bluetooth or Do Not Disturb, and the ability to adjust screen brightness, pause or play a song, jump to the next track and stream your music with AirPlay®. Control Center also gives you instant access to apps such as Clock, Camera, Calculator and the Flashlight.

With iOS 7, Notification Center is now available from the Lock screen so you can see all your notifications with a simple swipe, and the new Today feature in Notification Center gives you an at-a-glance view of your day with a summary of the important details such as weather, traffic, meetings and events.

With improved Multitasking in iOS 7, developers have the ability to enable any app to multitask in the background with a new API. Users have the ability to switch between their apps in a more visual and intuitive way, and iOS 7 pays attention to which apps you use most and automatically keeps your content up to date in the background.

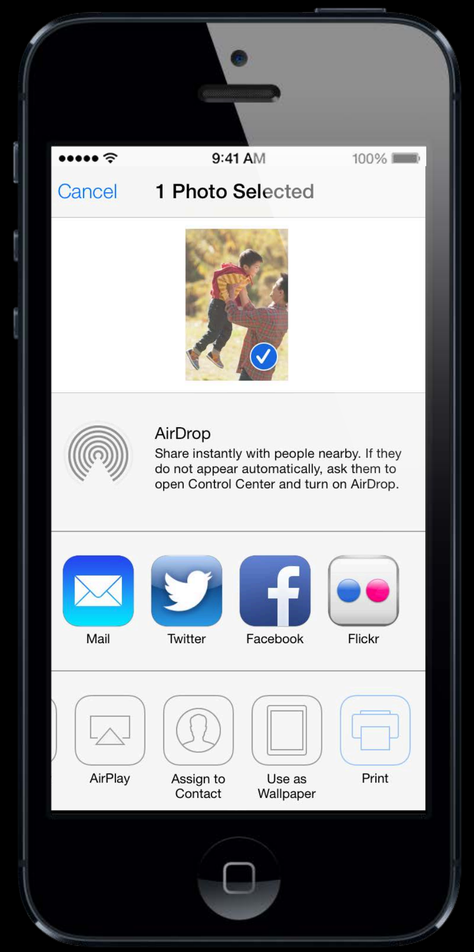
AirDrop is an entirely new way to quickly and easily share content with people nearby. When you’ve got something you want to share, AirDrop shows you your contacts close by. Just select who you want to share with and AirDrop does the rest. AirDrop transfers are peer-to-peer so you can use it anywhere, without any network or set up required, and transfers are fully encrypted so your content is protected and private.

iPhones are used to take more photos around the world every day than any other camera, and with iOS 7 the new Camera app features filters so you can add real-time photo effects. The Camera app now includes a square camera option, and you can quickly and easily switch between your four cameras—

Photo

`UIPrintInfoOutputPhoto`

- High quality
- Photo paper size
- Borderless if available
- No duplex mode
- No page range



Photo

`UIPrintInfoOutputPhoto`

- High quality
- Photo paper size
- Borderless if available
- No duplex mode
- No page range

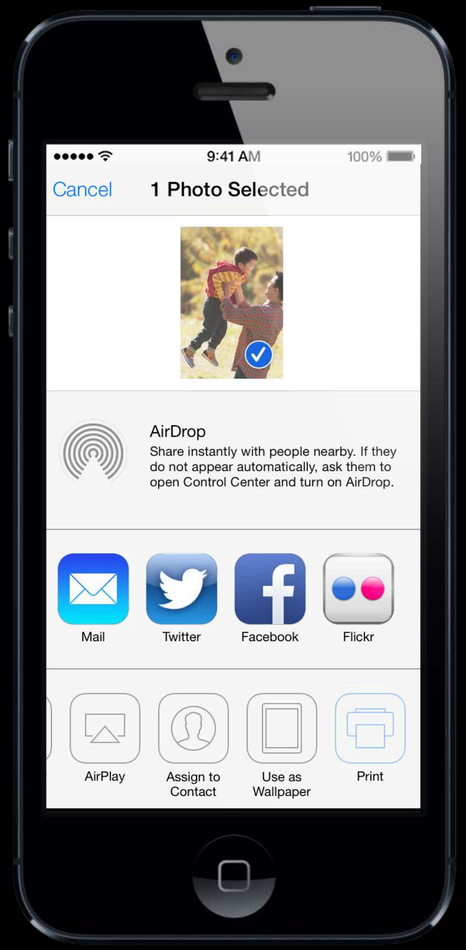


Photo High Quality Grayscale

`UIPrintInfoOutputPhotoGrayscale`

NEW

- Photo paper size
- Borderless if available
- No duplex mode
- No page range

iOS Printing Classes

```
UIPrintInfo
UIPrintPaper
UIPrintFormatter
   UISimpleTextPrintFormatter
   UIMarkupTextPrintFormatter
   UIViewPrintFormatter
UIPrintPageRenderer

UIPrintInteractionController
UIActivityViewController
```

iOS Printing Classes

UIPrintInfo

UIPrintPaper

UIPrintFormatter

- UISimpleTextPrintFormatter
- UIMarkupTextPrintFormatter
- UIViewPrintFormatter

UIPrintPageRenderer

UIPrintInteractionController

UIActivityViewController

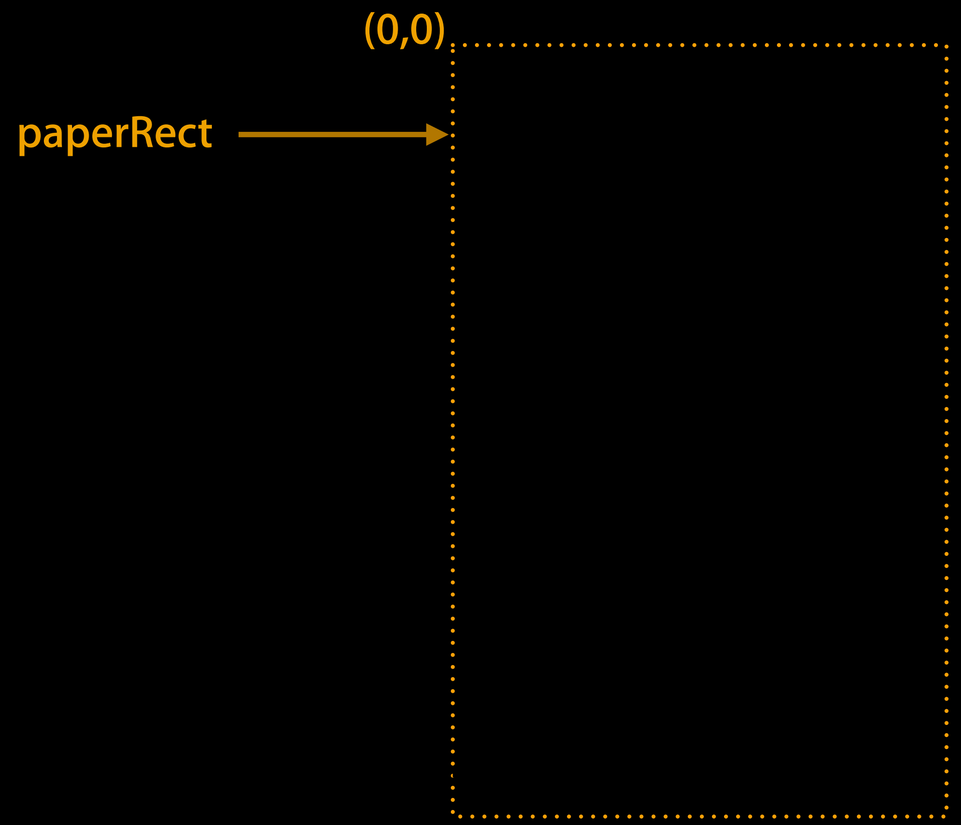
iOS Printing Classes

`UIPrintPaper`

UIPrintPaper

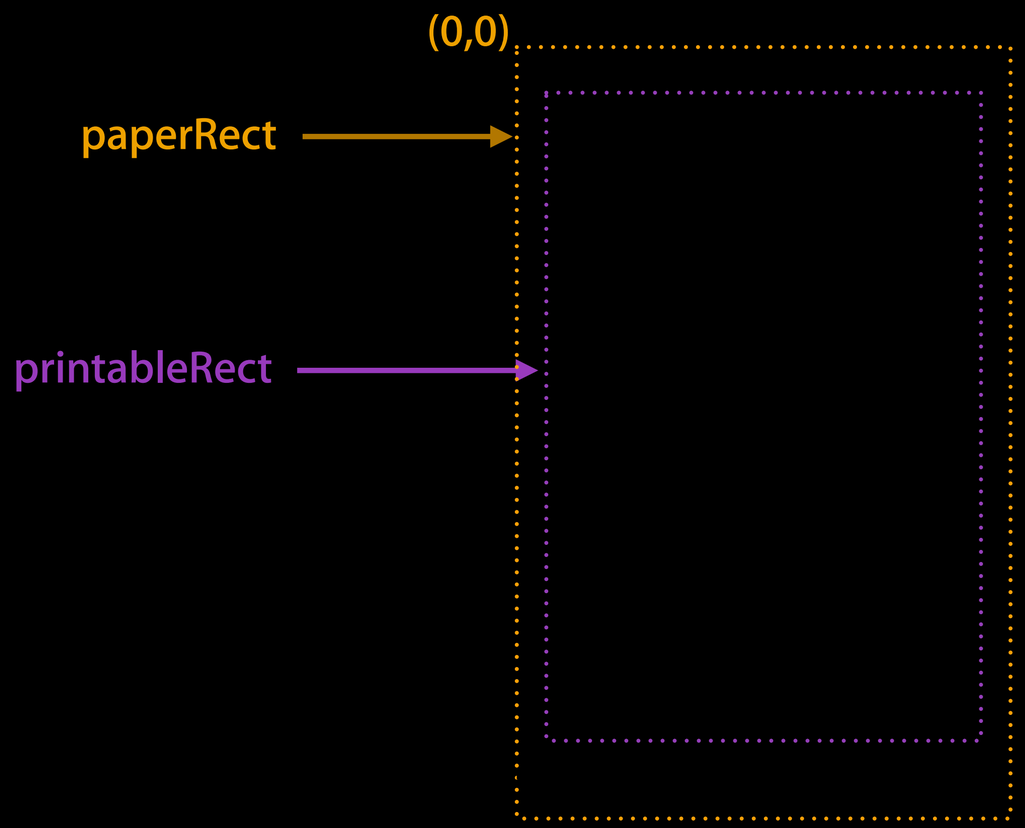
Paper

`UIPrintPaper`



Paper

UIPrintPaper



Providing Content

Two Types of Printing

- Printing items that already exist
  - PDF files
  - Image files
- Create the pages dynamically
  - Render the pages from the content in your app
  - Use Formatter and Renderer classes

Printing Items

- Single item or array of items
  - PDF, JPEG, other image types (PNG, etc.)

    `NSURL, NSData, UIImage, CIImage`
  - Asset library

    `ALAsset, ALAssetURL`
- Each item is a separate print job

Printing a PDF

```
- (void)printFile:(NSURL *)url {

    if ([UIPrintInteractionController canPrintURL:url]) {

        UIPrintInteractionController *
           controller = [UIPrintInteractionController
sharedPrintController];

        controller.printingItem = url;

        UIPrintInfo *printInfo = [UIPrintInfo printInfo];
        printInfo.outputType = UIPrintInfoOutputGeneral;
        printInfo.jobName    = [url lastPathComponent];
        controller.printInfo = printInfo;

        controller.showsPageRange = YES;

        [controller presentAnimated:YES completionHandler:NULL];
    }
}
```

Printing a PDF

```
- (void)printFile:(NSURL *)url {

    if ([UIPrintInteractionController canPrintURL:url]) {

        UIPrintInteractionController *
           controller = [UIPrintInteractionController
sharedPrintController];

        controller.printingItem = url;

        UIPrintInfo *printInfo = [UIPrintInfo printInfo];
        printInfo.outputType = UIPrintInfoOutputGeneral;
        printInfo.jobName    = [url lastPathComponent];
        controller.printInfo = printInfo;

        controller.showsPageRange = YES;

        [controller presentAnimated:YES completionHandler:NULL];
    }
}
```

Printing a PDF

```
- (void)printFile:(NSURL *)url {

    if ([UIPrintInteractionController canPrintURL:url]) {

        UIPrintInteractionController *
           controller = [UIPrintInteractionController
sharedPrintController];

        controller.printingItem = url;

        UIPrintInfo *printInfo = [UIPrintInfo printInfo];
        printInfo.outputType = UIPrintInfoOutputGeneral;
        printInfo.jobName    = [url lastPathComponent];
        controller.printInfo = printInfo;

        controller.showsPageRange = YES;

        [controller presentAnimated:YES completionHandler:NULL];
    }
}
```

Printing a PDF

```
- (void)printFile:(NSURL *)url {

    if ([UIPrintInteractionController canPrintURL:url]) {

        UIPrintInteractionController *
           controller = [UIPrintInteractionController
sharedPrintController];
        controller.printingItem = url;

        UIPrintInfo *printInfo = [UIPrintInfo printInfo];
        printInfo.outputType = UIPrintInfoOutputGeneral;
        printInfo.jobName    = [url lastPathComponent];
        controller.printInfo = printInfo;

        controller.showsPageRange = YES;

        [controller presentAnimated:YES completionHandler:NULL];
    }
}
```

Printing a PDF

```
- (void)printFile:(NSURL *)url {

    if ([UIPrintInteractionController canPrintURL:url]) {

        UIPrintInteractionController *
           controller = [UIPrintInteractionController
sharedPrintController];
        controller.printingItem = url;

        UIPrintInfo *printInfo = [UIPrintInfo printInfo];
        printInfo.outputType = UIPrintInfoOutputGeneral;
        printInfo.jobName    = [url lastPathComponent];
        controller.printInfo = printInfo;

        controller.showsPageRange = YES;

        [controller presentAnimated:YES completionHandler:NULL];
    }
}
```

Printing a PDF

```
- (void)printFile:(NSURL *)url {

    if ([UIPrintInteractionController canPrintURL:url]) {

        UIPrintInteractionController *
           controller = [UIPrintInteractionController
sharedPrintController];
           controller.printingItem = url;

           UIPrintInfo *printInfo = [UIPrintInfo printInfo];
           printInfo.outputType = UIPrintInfoOutputGeneral;
           printInfo.jobName    = [url lastPathComponent];
           controller.printInfo = printInfo;

           controller.showsPageRange = YES;

           [controller presentAnimated:YES completionHandler:NULL];
    }
 }
```

Printing a PDF

```
- (void)printFile:(NSURL *)url {

    if ([UIPrintInteractionController canPrintURL:url]) {

        UIPrintInteractionController *
           controller = [UIPrintInteractionController
sharedPrintController];

            controller.printingItem = url;

            UIPrintInfo *printInfo = [UIPrintInfo printInfo];
            printInfo.outputType = UIPrintInfoOutputGeneral;
            printInfo.jobName    = [url lastPathComponent];
            controller.printInfo = printInfo;

            controller.showsPageRange = YES;

            [controller presentAnimated:YES completionHandler:NULL];
    }
}
```

Printing a PDF

```
- (void)printFile:(NSURL *)url {

    if ([UIPrintInteractionController canPrintURL:url]) {

        UIPrintInteractionController *
           controller = [UIPrintInteractionController
sharedPrintController];

        controller.printingItem = url;

        UIPrintInfo *printInfo = [UIPrintInfo printInfo];
        printInfo.outputType = UIPrintInfoOutputGeneral;
        printInfo.jobName    = [url lastPathComponent];
        controller.printInfo = printInfo;

        controller.showsPageRange = YES;

        [controller presentAnimated:YES completionHandler:NULL];
    }
}
```

Formatters

What Is a Formatter?

What Is a Formatter?

```
10101110010101000101010
11010010110111010110100
10111001010010111011000
```

Data

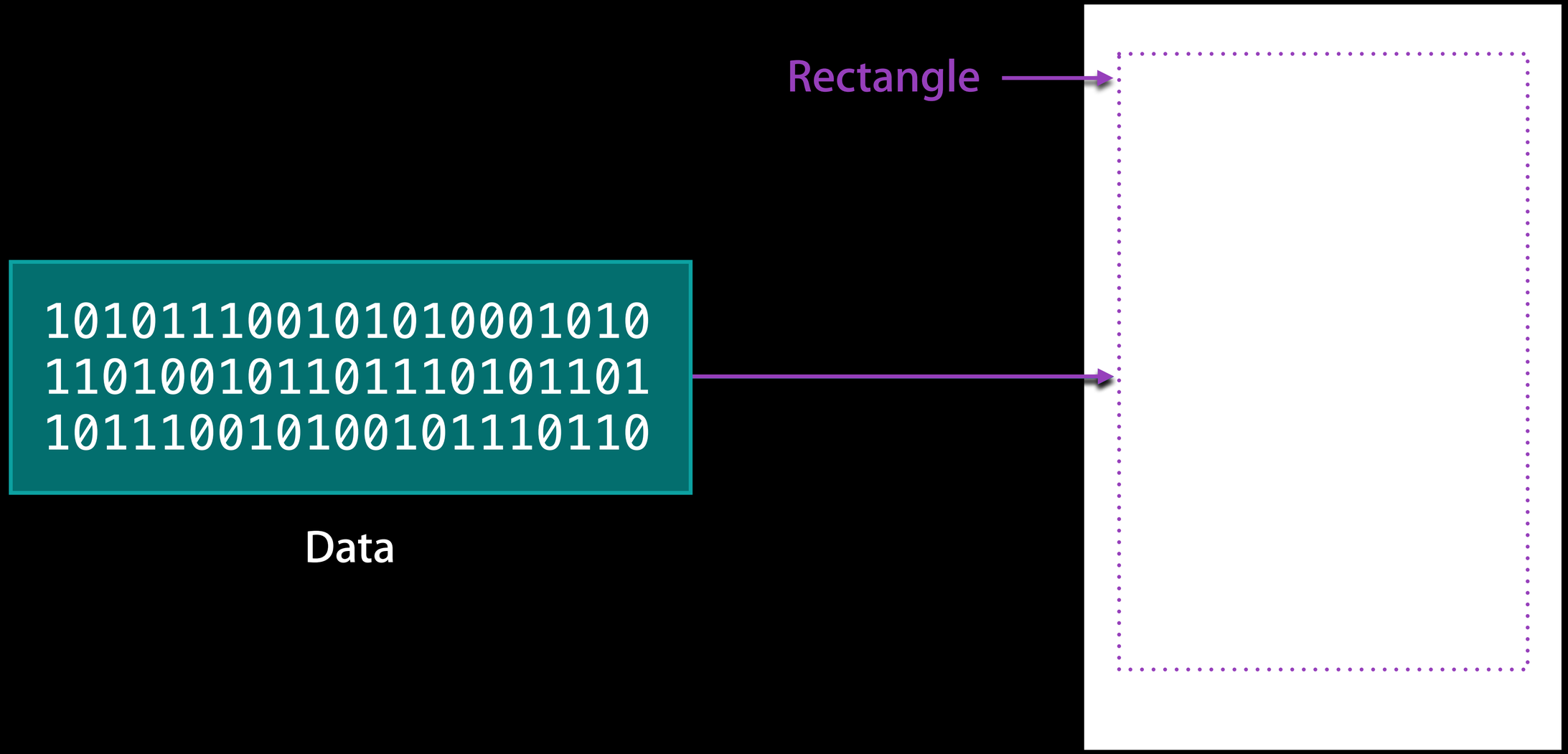
What Is a Formatter?

```
10101110010101000101010
11010010110111010110100
10111001010010111011000
```

Data

Rectangle

What Is a Formatter?



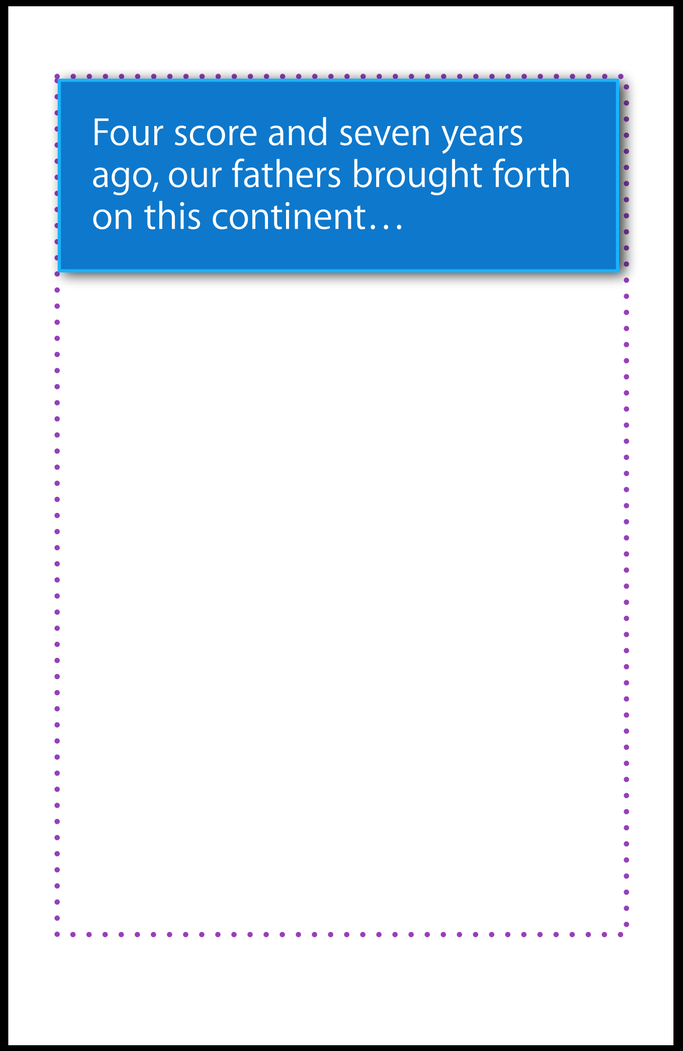
What Is a Formatter?

What Is a Formatter?

Four score and seven years ago, our fathers brought forth on this continent…

String of Text

What Is a Formatter?

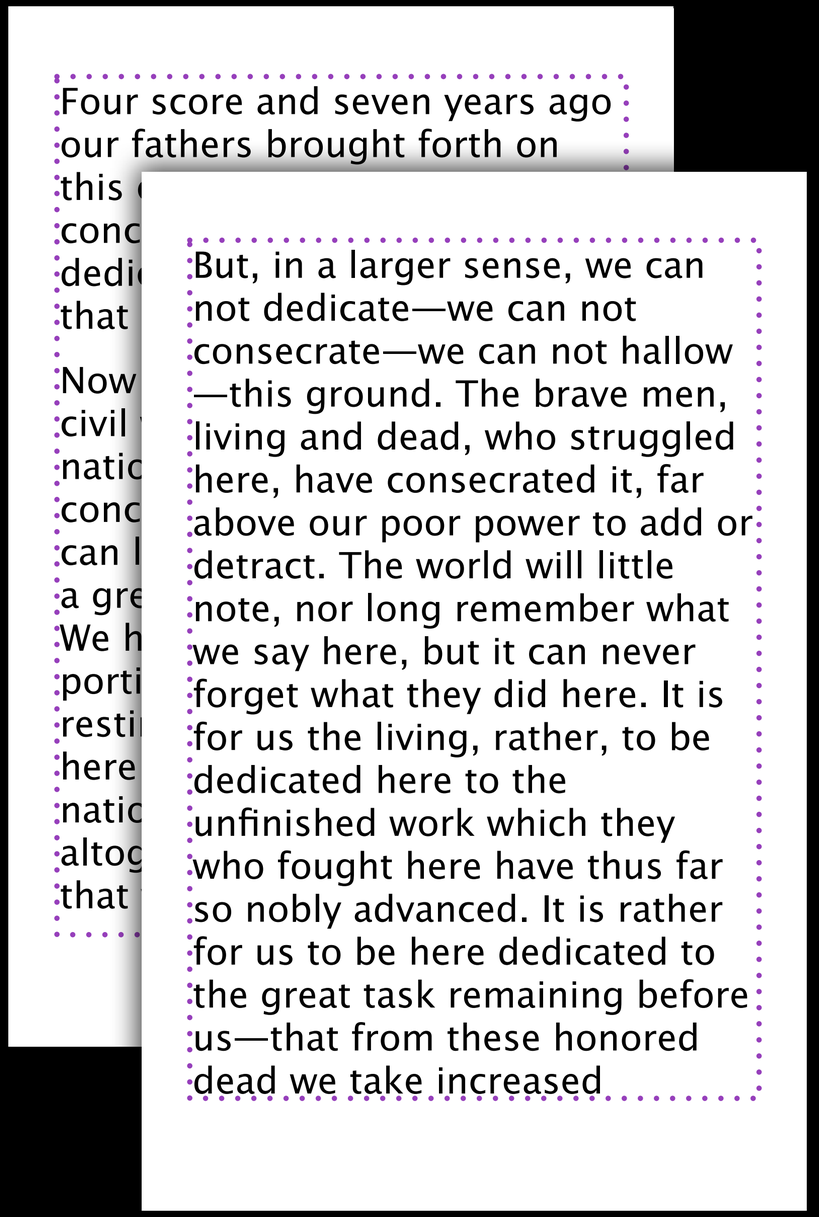


What Is a Formatter?

Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and dedicated to the proposition that all men are created equal.

Now we are engaged in a great civil war, testing whether that nation, or any nation so conceived and so dedicated, can long endure. We are met on a great battle-field of that war. We have come to dedicate a portion of that field, as a final resting place for those who here gave their lives that that nation might live. It is altogether fitting and proper that we should do this.

What Is a Formatter?

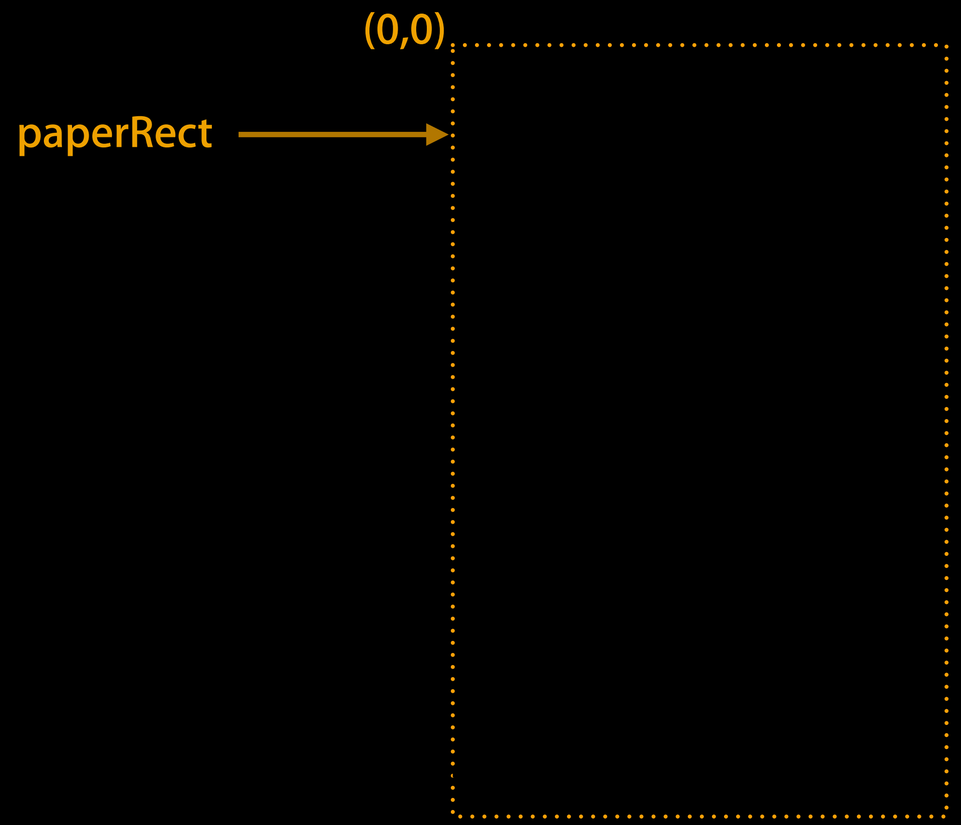


Formatters

- Use with `UIPrintInteractionController, UIActivityController` to format for the whole page
- Use as a helper to format data in a full renderer
- For plain text use `UISimpleTextFormatter` and specify the
  - Font
  - Color
  - Alignment
- For HTML markup text use `UIMarkupTextFormatter`

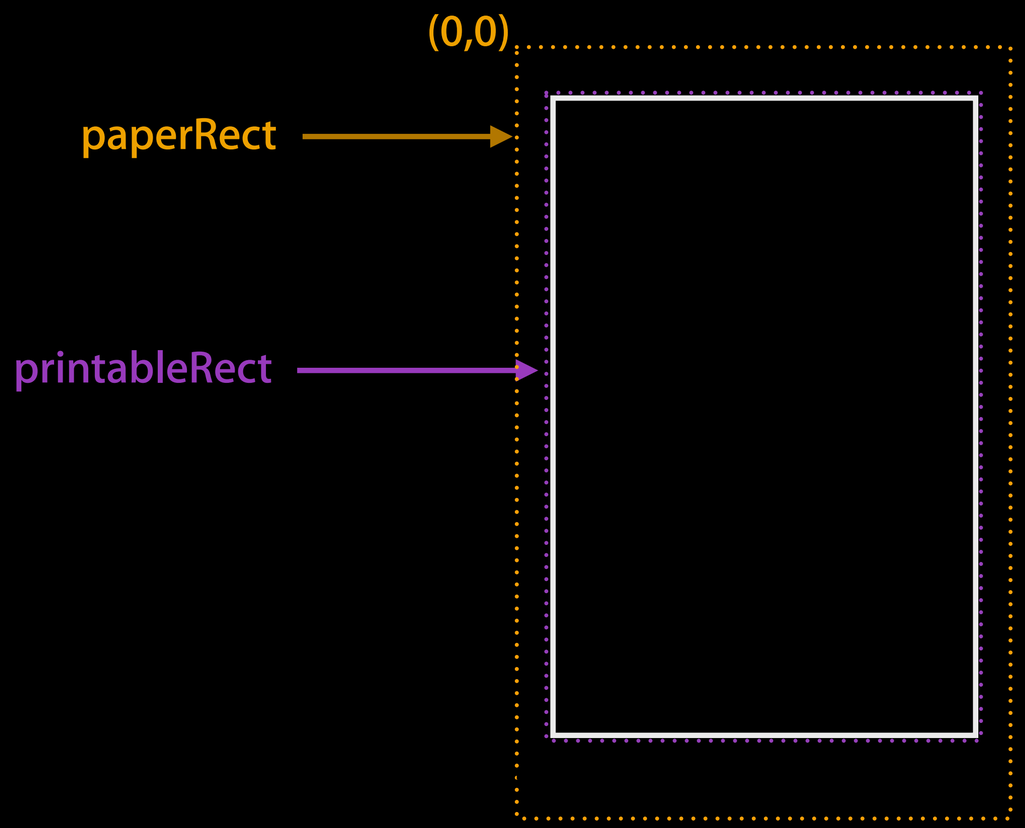
Formatter Used for Whole Sheet

Content is drawn inside the printable rect



Formatter Used for Whole Sheet

Content is drawn inside the printable rect

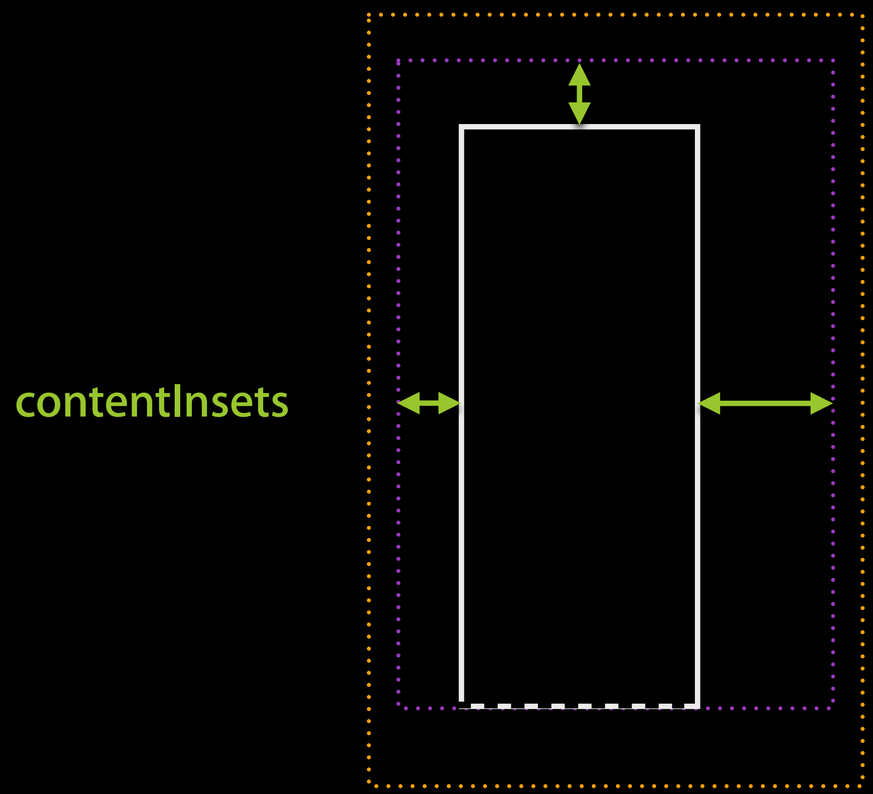


Layout

Content insets

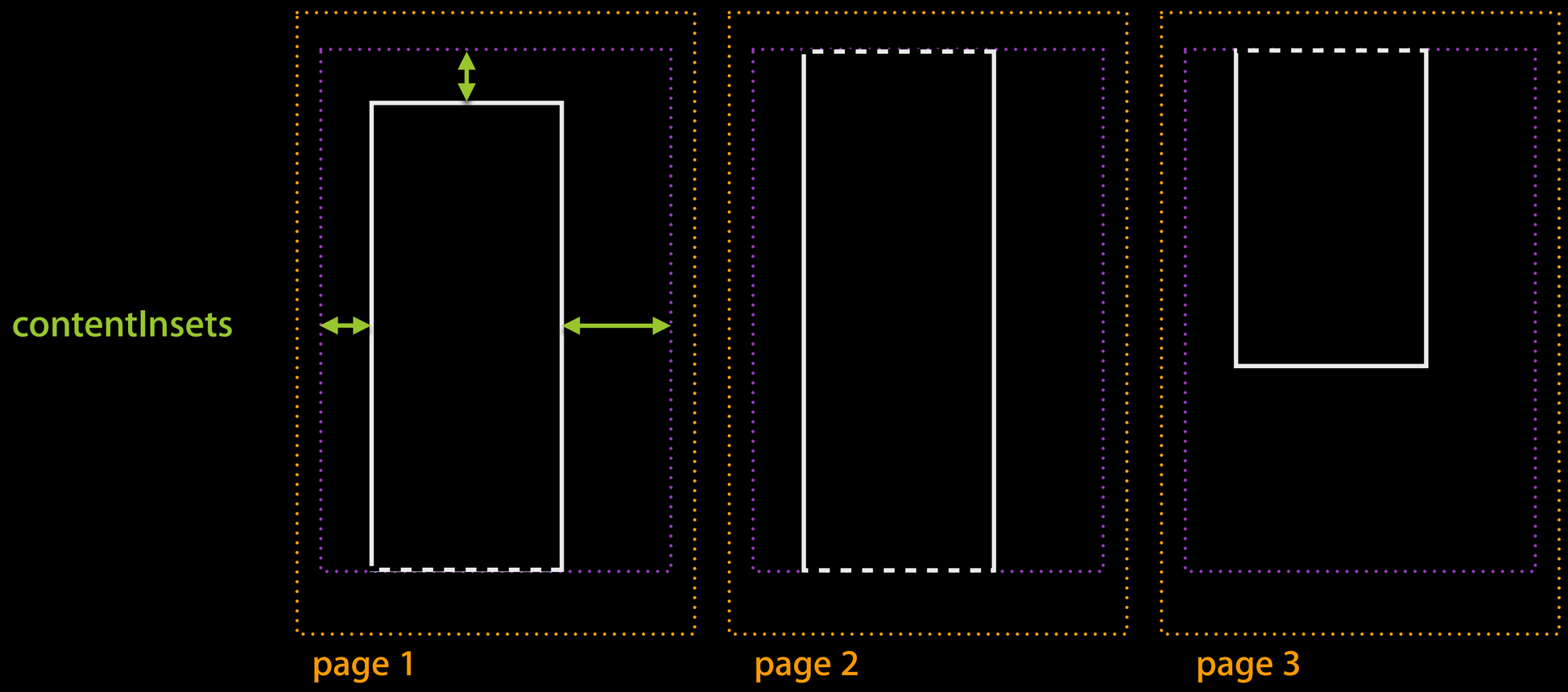
Layout

Content insets



Layout

Content insets



Formatters

```
- (void)printHTMLText:(NSString *)text {

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIMarkupTextFormatter *formatter
                      = [[UIMarkupTextFormatter alloc] initWithText:text];

    controller.printFormatter = formatter;

    UIPrintInfo *printInfo = [UIPrintInfo printInfo];
    printInfo.outputType = UIPrintInfoOutputGeneral;
    printInfo.jobName    = [urlField webPage];
    controller.printInfo = printInfo;

    [controller presentAnimated:YES completionHandler:NULL];

}
```

Formatters

```
- (void)printHTMLText:(NSString *)text {

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIMarkupTextFormatter *formatter
                      = [[UIMarkupTextFormatter alloc] initWithText:text];

    controller.printFormatter = formatter;

    UIPrintInfo *printInfo = [UIPrintInfo printInfo];
    printInfo.outputType = UIPrintInfoOutputGeneral;
    printInfo.jobName    = [urlField webPage];
    controller.printInfo = printInfo;

    [controller presentAnimated:YES completionHandler:NULL];

}
```

Formatters

```
- (void)printHTMLText:(NSString *)text {

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIMarkupTextFormatter *formatter
                      = [[UIMarkupTextFormatter alloc] initWithText:text];
    controller.printFormatter = formatter;

    UIPrintInfo *printInfo = [UIPrintInfo printInfo];
    printInfo.outputType = UIPrintInfoOutputGeneral;
    printInfo.jobName    = [urlField webPage];
    controller.printInfo = printInfo;

    [controller presentAnimated:YES completionHandler:NULL];

}
```

Formatters

```
- (void)printHTMLText:(NSString *)text {

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIMarkupTextFormatter *formatter
                      = [[UIMarkupTextFormatter alloc] initWithText:text];
    controller.printFormatter = formatter;

    UIPrintInfo *printInfo = [UIPrintInfo printInfo];
    printInfo.outputType = UIPrintInfoOutputGeneral;
    printInfo.jobName    = [urlField webPage];
    controller.printInfo = printInfo;

    [controller presentAnimated:YES completionHandler:NULL];

}
```

View Formatters

```
- (void)printWebView:(id) sender {

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIViewPrintFormatter *formatter
                       = [self.myWebView viewPrintFormatter];

    controller.printFormatter = formatter;

    UIPrintInfo *printInfo = [UIPrintInfo printInfo];
    printInfo.outputType = UIPrintInfoOutputGeneral;
    printInfo.jobName    = [urlField myWebPage];
    controller.printInfo = printInfo;

    [controller presentAnimated:YES completionHandler:NULL];

}
```

View Formatters

```
- (void)printWebView:(id) sender {

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIViewPrintFormatter *formatter
                       = [self.myWebView viewPrintFormatter];
    controller.printFormatter = formatter;

    UIPrintInfo *printInfo = [UIPrintInfo printInfo];
    printInfo.outputType = UIPrintInfoOutputGeneral;
    printInfo.jobName    = [urlField myWebPage];
    controller.printInfo = printInfo;

    [controller presentAnimated:YES completionHandler:NULL];

}
```

View Formatters

```
- (void)printWebView:(id) sender {

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIViewPrintFormatter *formatter
                       = [self.myWebView viewPrintFormatter];

    controller.printFormatter = formatter;

    UIPrintInfo *printInfo = [UIPrintInfo printInfo];
    printInfo.outputType = UIPrintInfoOutputGeneral;
    printInfo.jobName    = [urlField myWebPage];
    controller.printInfo = printInfo;

    [controller presentAnimated:YES completionHandler:NULL];

}
```

iOS Printing Classes

UIPrintInfo

UIPrintPaper

UIPrintFormatter

- UISimpleTextPrintFormatter
- UIMarkupTextPrintFormatter
- UIViewPrintFormatter

UIPrintPageRenderer

UIPrintInteractionController

UIActivityViewController

iOS Printing Classes

```
UIPrintInfo
UIPrintPaper
UIPrintFormatter
     UISimpleTextPrintFormatter
     UIMarkupTextPrintFormatter
     UIViewPrintFormatter
UIPrintPageRenderer

UIPrintInteractionController
UIActivityViewController
```

iOS Printing Classes

UIPrintPageRenderer

UIPrintPageRenderer

Renderers

What is a Renderer?

What is a Renderer?

Renderer

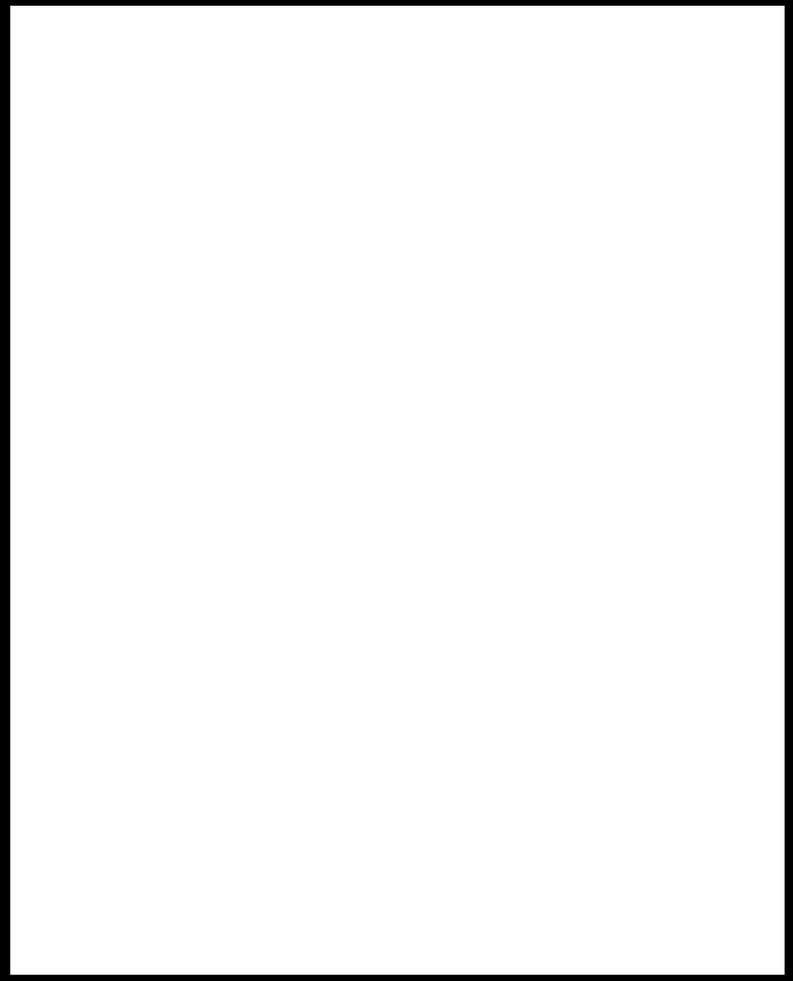
Number of Pages: 2

What is a Renderer?

Renderer

Number of Pages: 2

Draw First Page

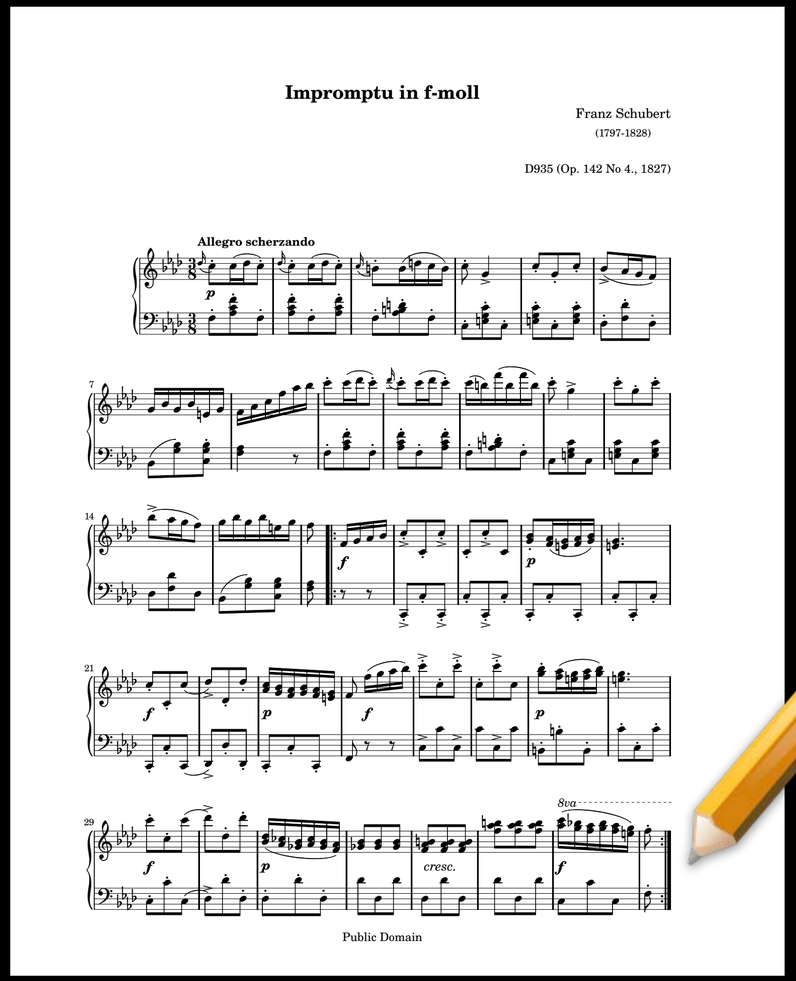


What is a Renderer?

Renderer

Number of Pages: 2

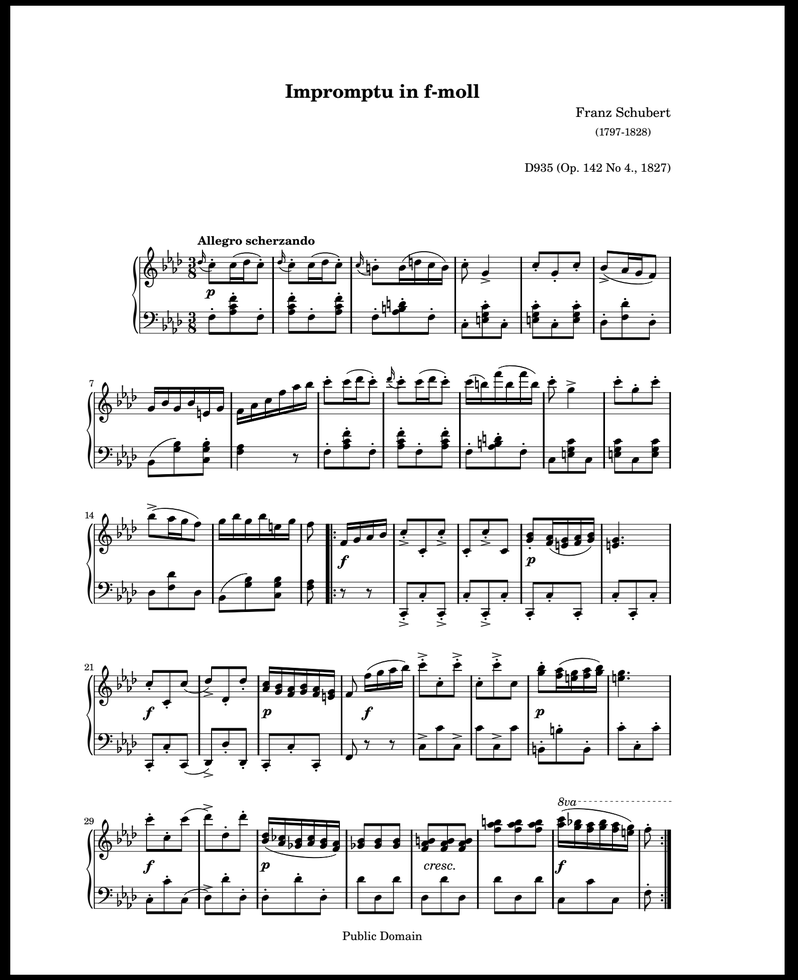
Draw First Page



What is a Renderer?

Renderer

Number of Pages: 2



Impromptu in f-moll

Franz Schubert
(1797-1828)

D935 (Op. 142 No 4., 1827)

Allegro scherzando

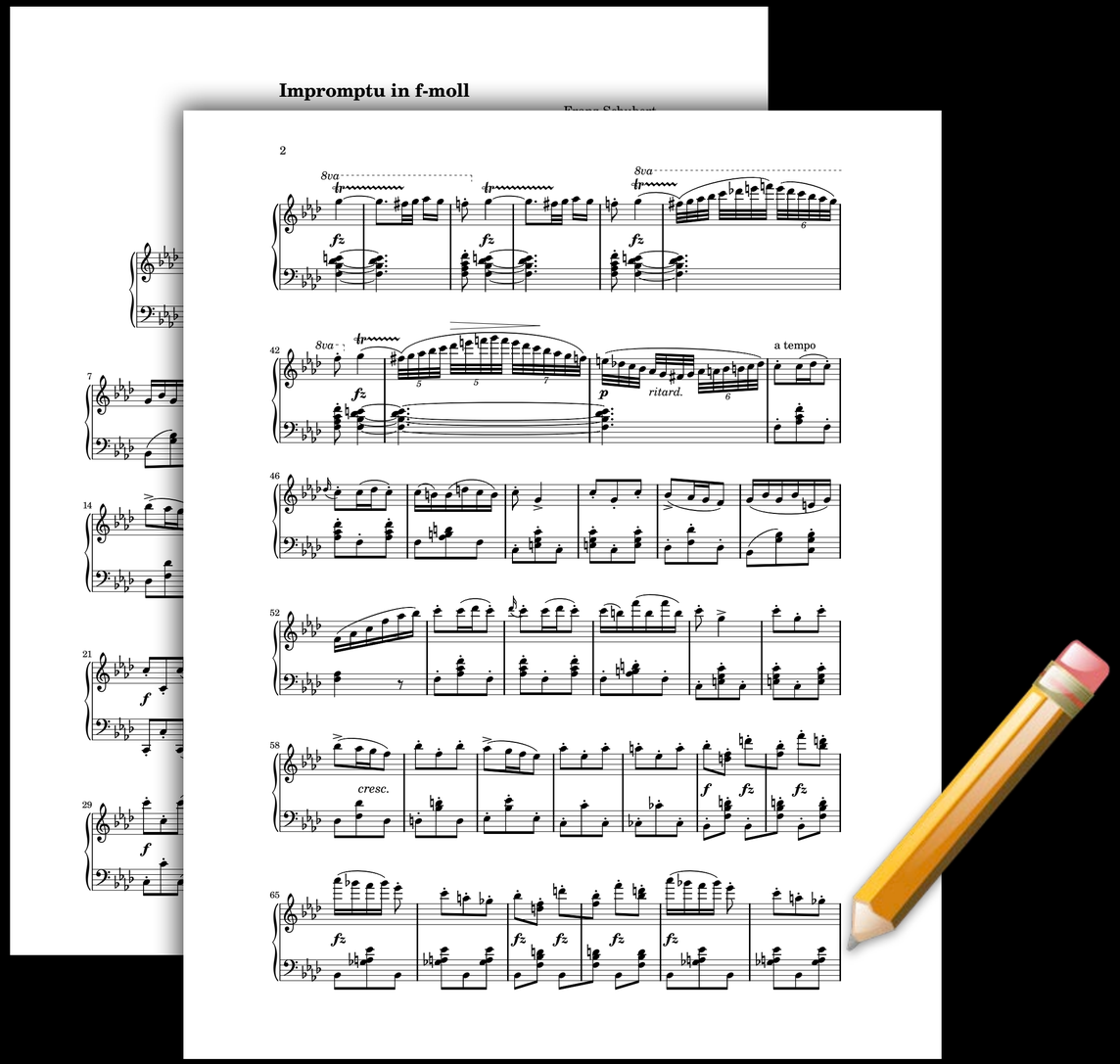
Public Domain

What is a Renderer?

Renderer

Number of Pages: 2

Draw Second Page



Print Page Renderer

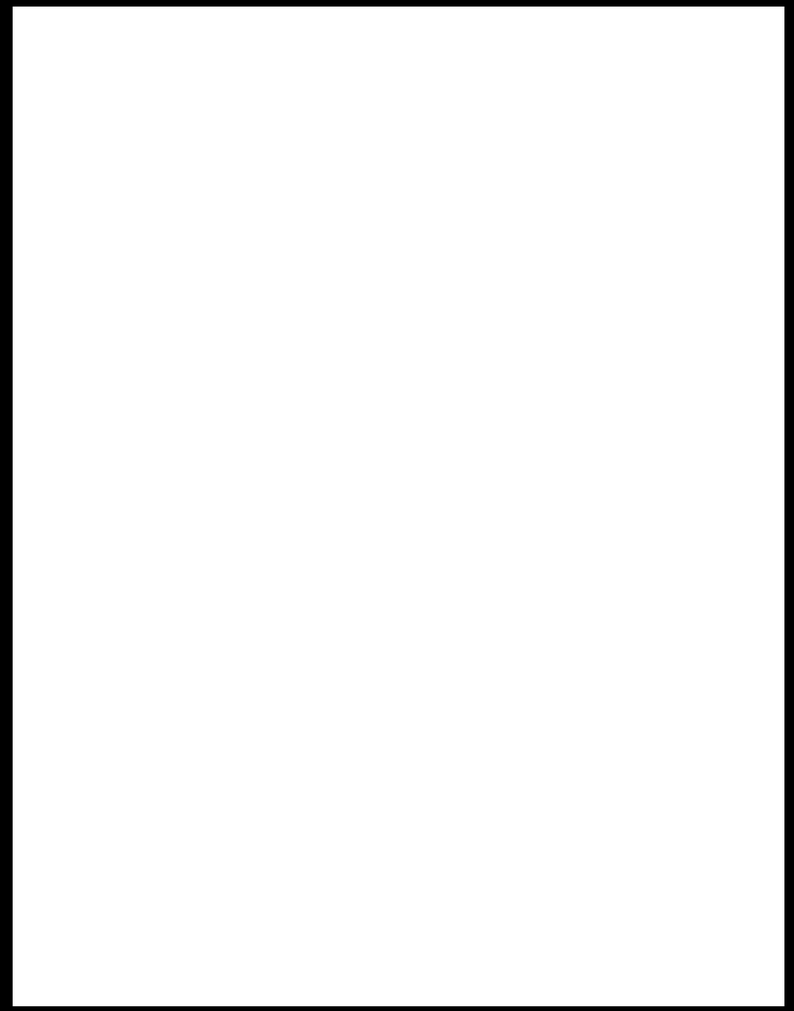
- Full drawing control
- Custom page-drawing object
  - Calculates page count
  - Draws page contents
- Add space for headers and footers
- Add formatters

Renderer with Formatter

Renderer with Formatter

Renderer

Draw Page

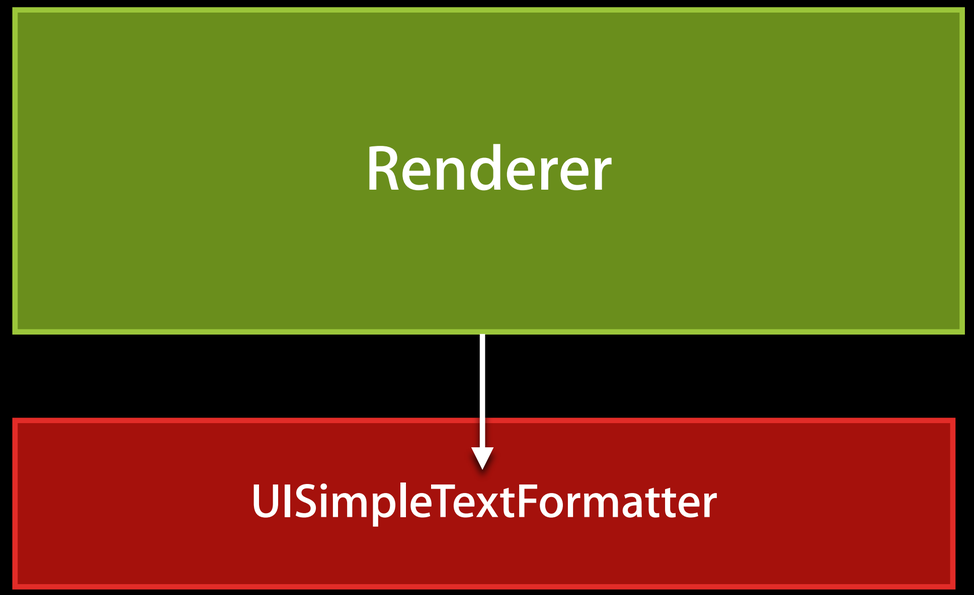


Renderer with Formatter

Renderer

Draw Page

Renderer with Formatter



Renderer with Formatter

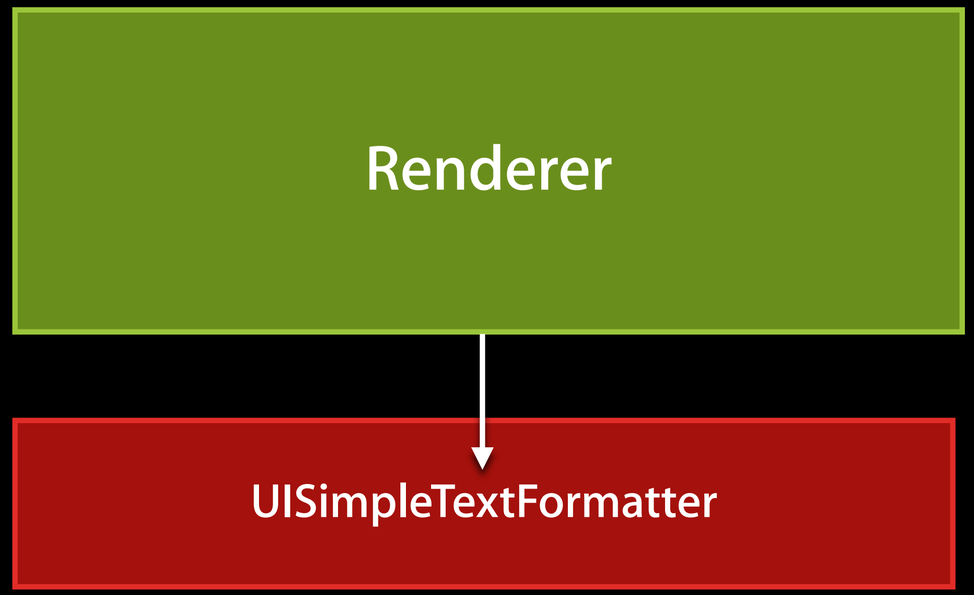
Renderer

UISimpleTextFormatter

Four score and seven years ago, our fathers brought forth on this continent…

Draw Page

Renderer with Formatter



Renderer with Formatter

Renderer

UISimpleTextFormatter

Draw Page

Four score and seven years ago our fathers brought forth on this continent, a new nation,conceived in Liberty, and dedicated to the proposition that all men are created equal. Now we are engaged in a great civil war, testing whether that nation, or any nation so conceived and so dedicated, can long endure. We are met on a great battle-field of that war. We have come to dedicate a portion of that field, as a final resting place for those who here gave their lives that that nation might live. It is altogether fitting and proper that we should do this.

Basic Rendering

- Subclass `UIPrintPageRenderer`
- Override

```
-numberOfPages
-drawContentForPageAtIndex:inRect:
```

- Set `UIPrintInteractionController.printPageRenderer` or add to the array of activity items when creating the `UIActivityViewController` object.
- To use formatters, create the formatter object and call `addPrintFormatter:startingAtPageIndex:`

Showing the UI

Printing from the Share Sheet

Printing from the Share Sheet

```
NSArray *activityItems = @ [ printInfo, myRenderer, [urlField text] ];

UIActivityViewController *viewController = [[UIActivityViewController alloc]
initWithActivityItems:activityItems applicationActivities:nil];

viewController.completionHandler = ^(NSString * activityType, BOOL completed)
{
    [viewController release];
    [myRenderer release];
 };

... present using standard view controller present methods
```

Printing from the Share Sheet

```
NSArray *activityItems = @ [ printInfo, myRenderer, [urlField text] ];

UIActivityViewController *viewController = [[UIActivityViewController alloc]
initWithActivityItems:activityItems applicationActivities:nil];

viewController.completionHandler = ^(NSString * activityType, BOOL completed)
{
    [viewController release];
    [myRenderer release];
 };

... present using standard view controller present methods
```

Printing from the Share Sheet

```
NSArray *activityItems = @ [ printInfo, myRenderer, [urlField text] ];

UIActivityViewController *viewController = [[UIActivityViewController alloc]
initWithActivityItems:activityItems applicationActivities:nil];

viewController.completionHandler = ^(NSString * activityType, BOOL completed)
{
    [viewController release];
    [myRenderer release];
 };

... present using standard view controller present methods
```

Printing from the Share Sheet

```
NSArray *activityItems = @ [ printInfo, myRenderer, [urlField text] ];

UIActivityViewController *viewController = [[UIActivityViewController alloc]
initWithActivityItems:activityItems applicationActivities:nil];

viewController.completionHandler = ^(NSString * activityType, BOOL completed)
{
    [viewController release];
    [myRenderer release];
};

... present using standard view controller present methods
```

Printing Using a “Print” Button

- Create and setup the UIPrintInteractionController
- iPhone

  `-presentAnimated:completionHandler:`
- iPad

  `-presentFromRect:inView:animated:completionHandler:`

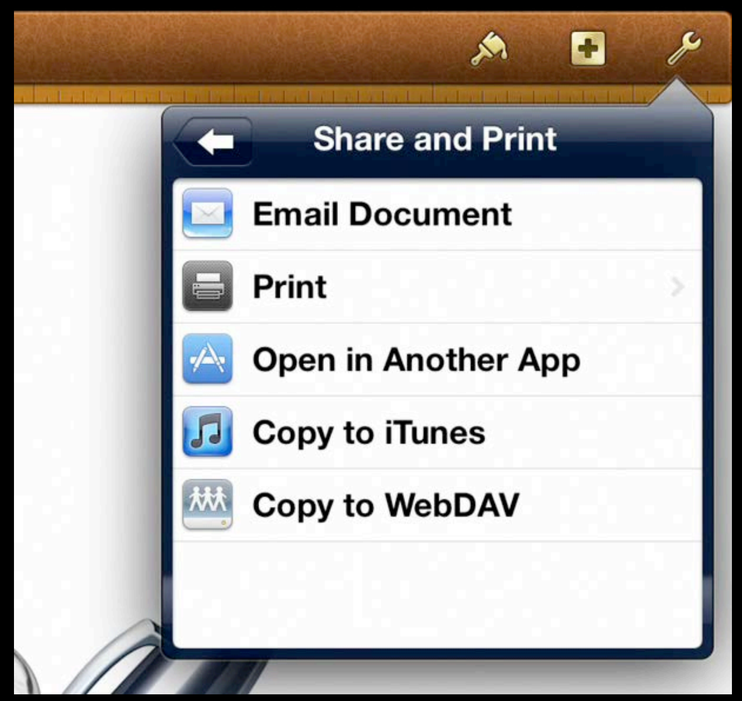
  `-presentFromBarButtonItem:animated:completionHandler:`

Printing as a Menu Item

- Set your class as the delegate for the shared UIPrintInteractionController
- Implement—`printInteractionControllerParentViewController:`
- When the user taps "Print" call `presentAnimated:CompletionHandler:`
  `UINavigationController`—push
  `UIViewController`—modal
- Do not peek

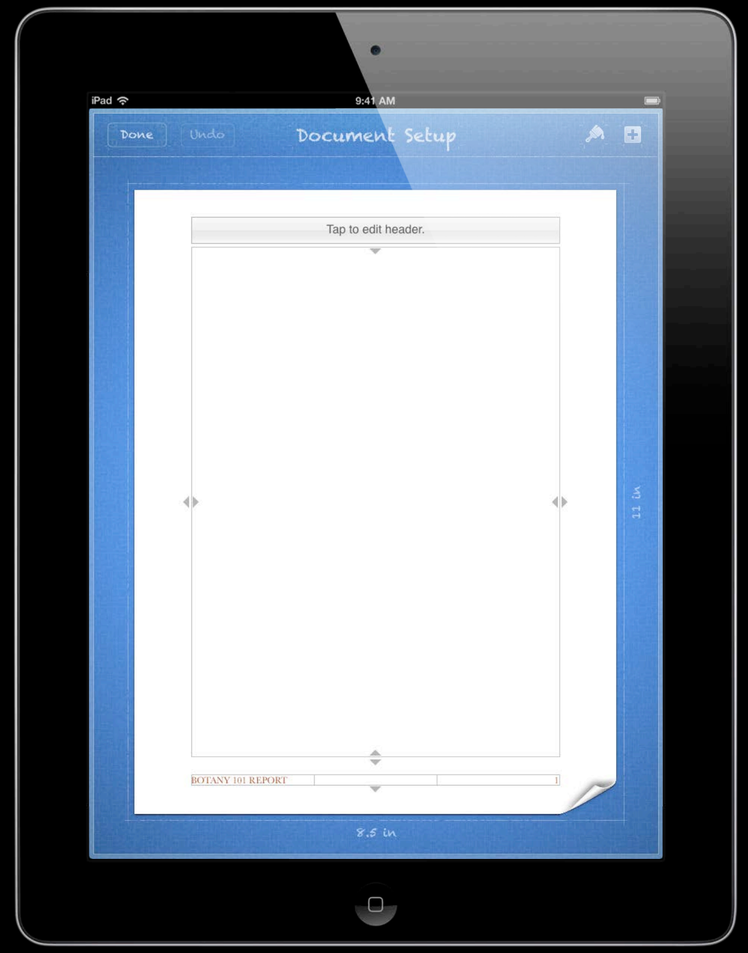
Printing as a Menu Item

- Set your class as the delegate for the shared UIPrintInteractionController
- Implement—`printInteractionControllerParentViewController:`
- When the user taps “Print” call `presentAnimated:CompletionHandler:`
  - `UINavigationController`—push
  - `UIViewController`—modal
- Do not peek



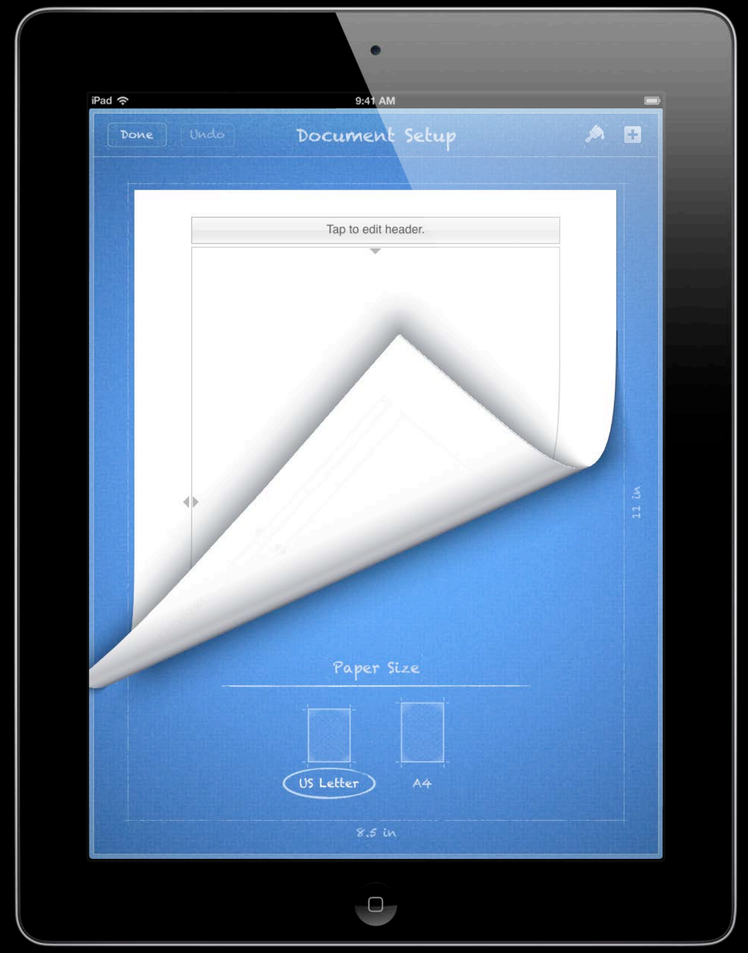
Paper Size

- For document-centric apps that require a paper size in order to format the content



Paper Size

- Provide your own paper selection UI

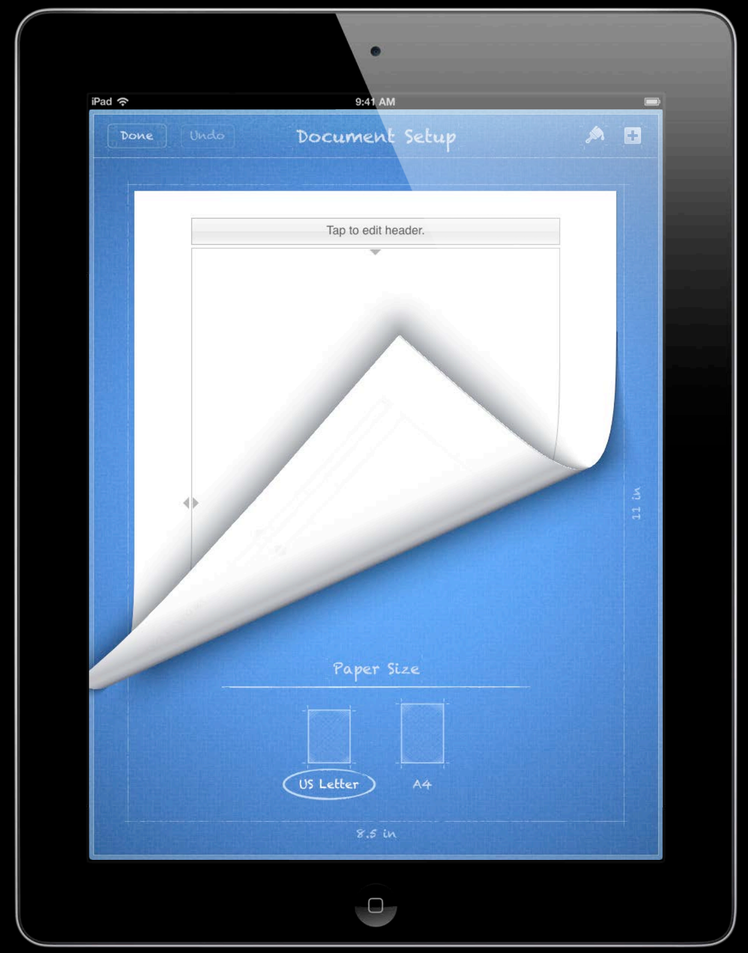


Paper Size

- Provide your own paper selection UI
- Use delegate method

  `–printInteractionController:choosePaper:`

  - Called after user selects a printer
  - You ask for a paper size that is a good match to the user-selected paper
  - If a printer has paper sensors, the array will only be detected papers



Paper Size

```
- (UIPrintPaper *)printInteractionController:(UIPrintInteractionController *)
                printInteractionController choosePaper:(NSArray *)paperList {

    CGSize paperSize = CGSizeMake(8.5 * 72.0, 11.0 * 72.0);

    return [UIPrintPaper bestPaperForPageSize:pageSize
                          withPapersFromArray:paperList];
}
```

Paper Size

```
- (UIPrintPaper *)printInteractionController:(UIPrintInteractionController *)
                printInteractionController choosePaper:(NSArray *)paperList {

    CGSize paperSize = CGSizeMake(8.5 * 72.0, 11.0 * 72.0);

    return [UIPrintPaper bestPaperForPageSize:pageSize
                          withPapersFromArray:paperList];
}
```

Paper Size

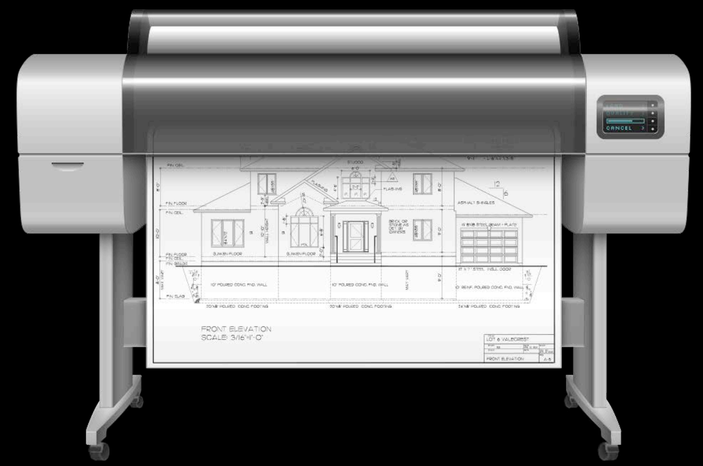
```
- (UIPrintPaper *)printInteractionController:(UIPrintInteractionController *)
                printInteractionController choosePaper:(NSArray *)paperList {

    CGSize paperSize = CGSizeMake(8.5 * 72.0, 11.0 * 72.0);

    return [UIPrintPaper bestPaperForPageSize:pageSize
                          withPapersFromArray:paperList];
}
```

Roll Paper

- AirPrint now supports roll paper
- Use delegate method

  `–printInteractionController:cutLengthForPaper:`

  - Called after user selects a printer that has a roll loaded
  - `UIPrintPaper` width of the roll and the maximum height
- By default, the cut length will be proportional to the default paper



Roll Paper Size


```
- (UIPrintPaper *)printInteractionController:(UIPrintInteractionController
*)printInteractionController cutLengthForPaper:(UIPrintPaper *)paper {
    return paper.printableRect.size.width * 2;
}
```

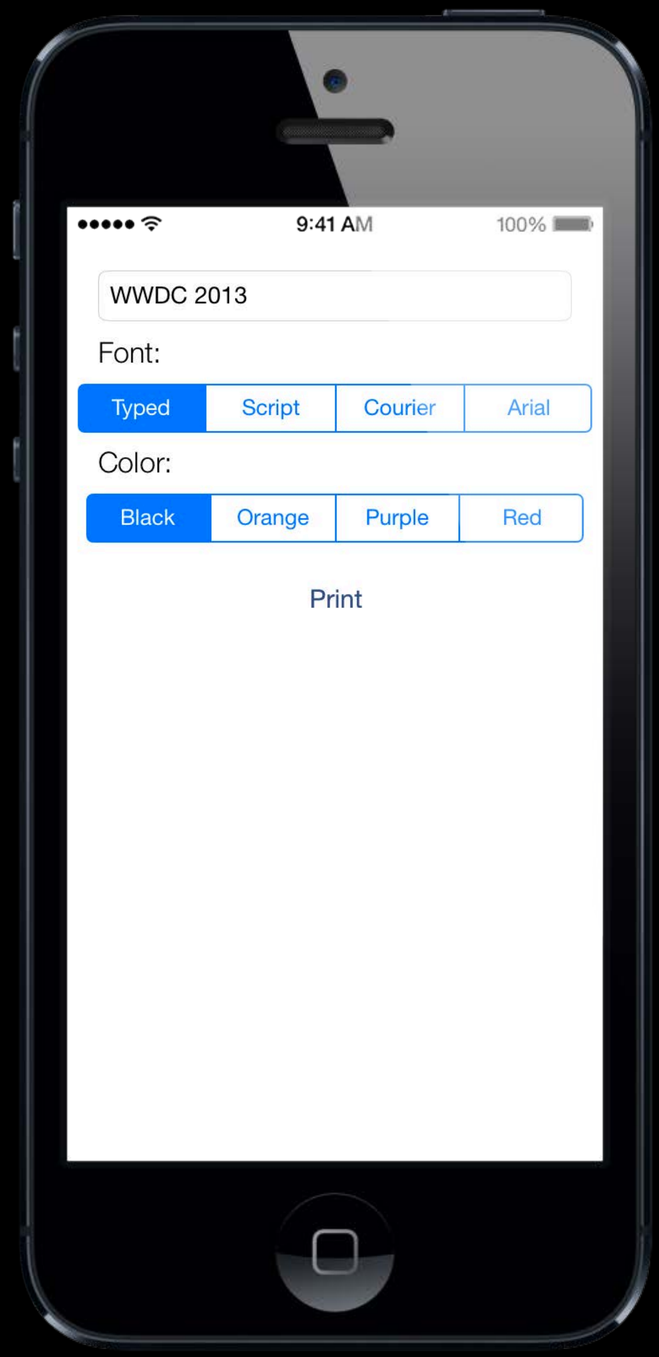
Roll Paper Size


```
- (UIPrintPaper *)printInteractionController:(UIPrintInteractionController
*)printInteractionController cutLengthForPaper:(UIPrintPaper *)paper {
    return paper.printableRect.size.width * 2;
}
```

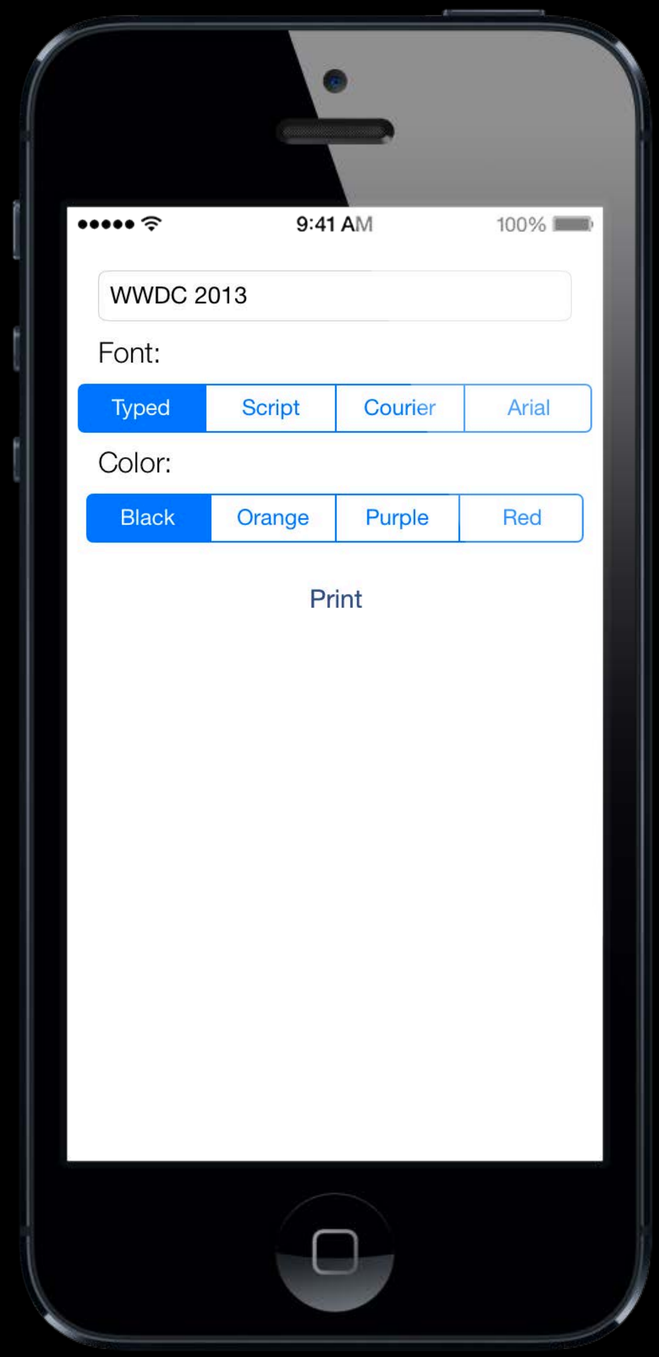
Roll Paper Size


```
- (UIPrintPaper *)printInteractionController:(UIPrintInteractionController
*)printInteractionController cutLengthForPaper:(UIPrintPaper *)paper {
    return paper.printableRect.size.width * 2;
}
```

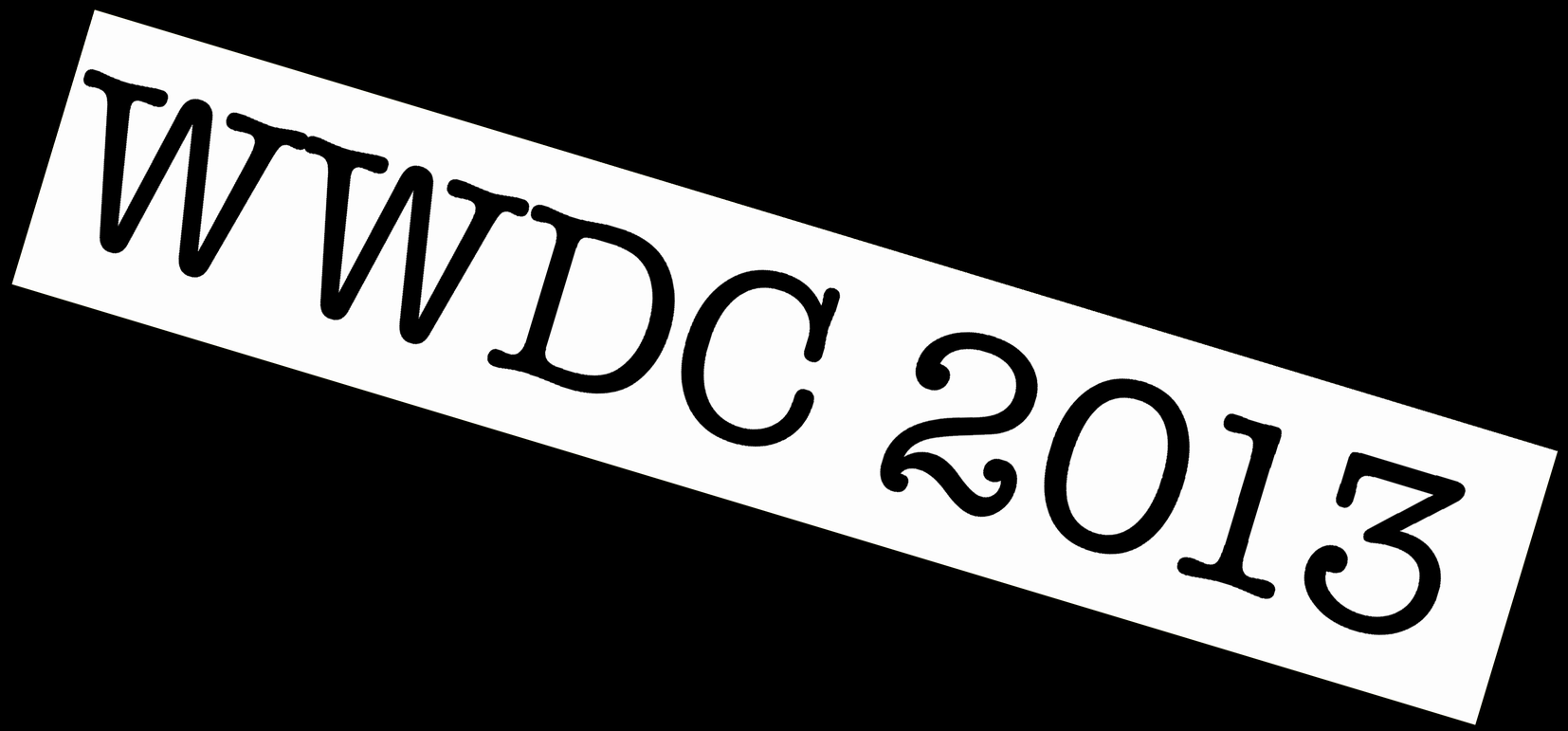
Demo “Print Banner”



Demo “Print Banner”



Demo “Print Banner”



Demo “Print Banner”

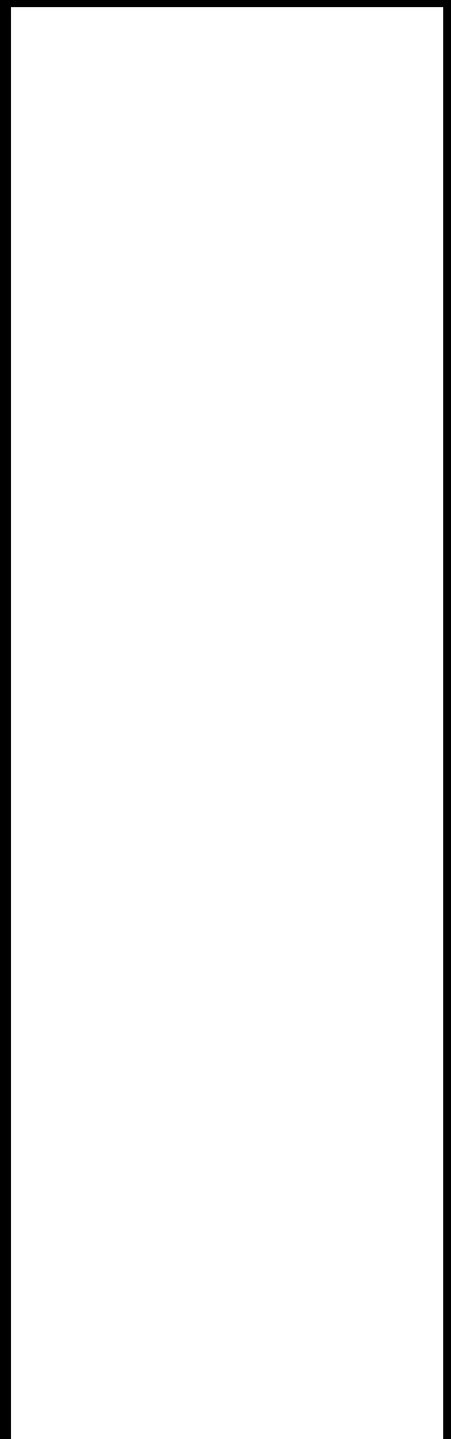
WWDC 2013

String of Text

Demo “Print Banner”

WWDC 2013

String of Text



Demo “Print Banner”

WWDC 2013

Demo “Print Banner”

WWDC 2013

Demo “Print Banner”

WWDC 2013

Demo “Print Banner”

WWDC 2013

Demo

Summary

- Adding printing in your app is easy
- Printing items such as PDFs or images is just 10 lines of code
- Formatters and Renderers give you full control over printing
- Print Banner sample app is available today

More Information

Paul Danbold
Core OS Technologies Evangelist
danbold@apple.com

iOS Printing Documentation
http://developer.apple.com/search/index.php?q=printing

iOS Printing Sample Code
http://developer.apple.com/library/ios/#samplecode/PrintWebView
http://developer.apple.com/library/ios/#samplecode/PrintPhoto
http://developer.apple.com/library/ios/#samplecode/Recipes_+_Printing
http://developer.apple.com/library/prerelease/ios/samplecode/PrintBanner

More Information

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

AirPrint Basics
http://support.apple.com/kb/ht4356

Related Sessions

What's New in Cocoa Touch	Presidio Tuesday 11:30AM	

Labs

Printing Lab	Core OS Lab B Thursday 11:30AM

WWDC2013