Core OS

#WWDC14

Adopting AirPrint

Session 718

Howard Miller
Printing Engineering

Agenda

AirPrint technologies update

iOS Printing

Printing without showing UI

Demo

AirPrint Technologies Update

What is AirPrint?

What is AirPrint?

Great user experience

- No driver, no software, no configuration
- Full output quality

What is AirPrint?

Great user experience

- No driver, no software, no configuration
- Full output quality

Protocol

- Standards based plus Apple technology
- Zero-cost license for printer and server manufacturers

What is AirPrint?

What is AirPrint?

Printers

- Ubiquitous—Supported by every major printer manufacturer

What is AirPrint?

Printers

- Ubiquitous—Supported by every major printer manufacturer

Printing System

- iOS 4.2 and later

What is AirPrint?

Printers

- Ubiquitous—Supported by every major printer manufacturer

Printing System

- iOS 4.2 and later

Your App

AirPrint Is Easy and Powerful

Simple user interaction

Efficient and focused APIs

High-quality output

AirPrint Advances

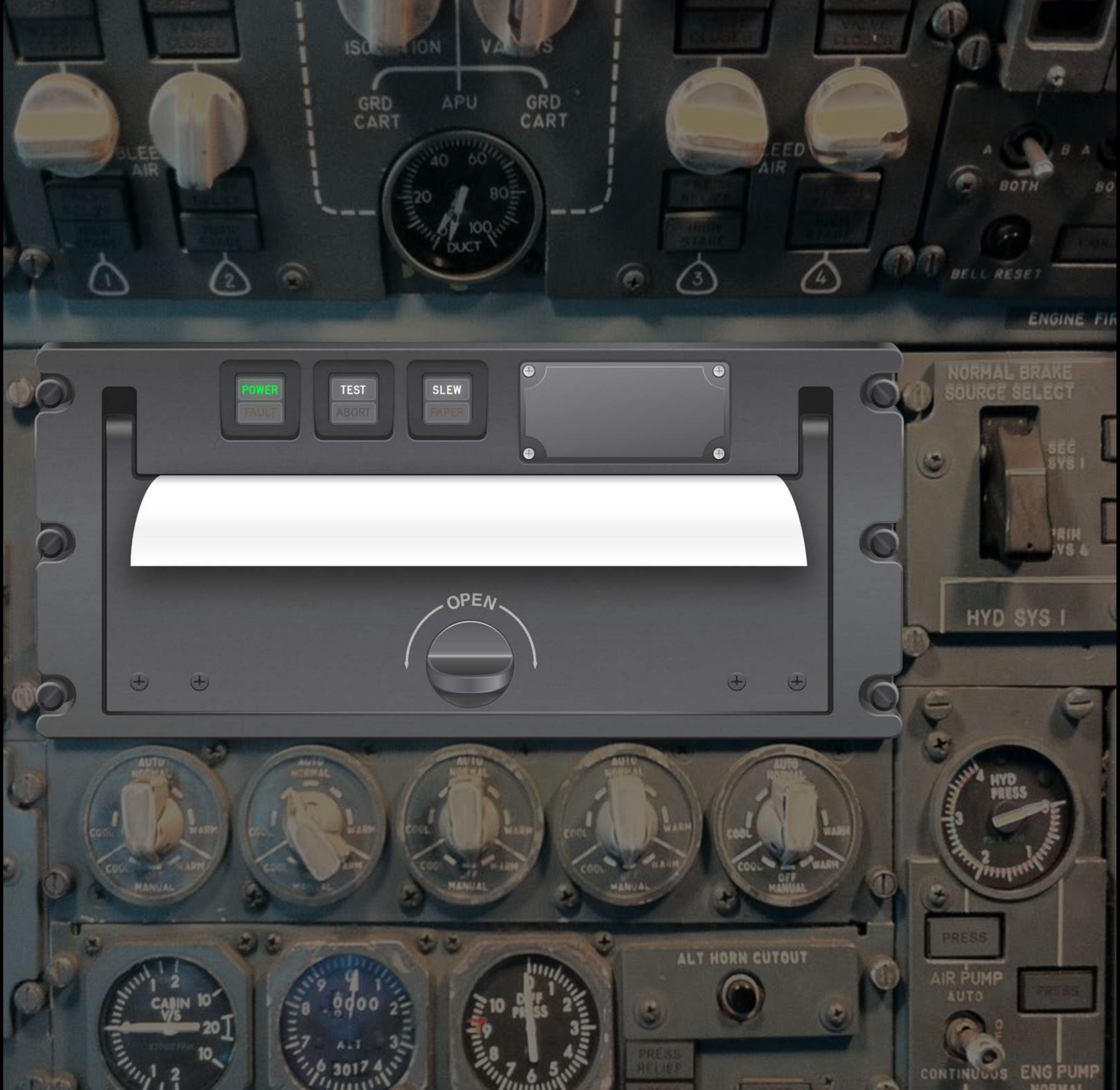
NEW

Expanded installed base

New classes of printers

AstroMed ToughWriter 5

POWER
FAULT
TEST
ABORT
SLEW
PAPER
OPEN


POWER
FAULT
TEST
ABORT
SLEW
PAPER
OPEN
GRD CART
APU
GRD CART
DUCT
BLEED AIR
BLEED AIR
BOTH
BELL RESET
ENGINE FIRE
NORMAL BRAKE SOURCE SELECT
HYD SYS 1
HYD PRESS
ALT HORN CUTOUT
CABIN V/S
DIFF PRESS
PRESS RELIEF
AIR PUMP
AUTO
CONTINUOUS
ENG PUMP
PRESS

Brother RuggedJet

AirPrint Advances

Continued

NEW

New APIs to support speciality printers and applications

Hints for Printing from Your App

Create your content for paper

- Use the space
- Provide higher quality artwork
- Consider readability

Work with the printing system

- Use just simplest APIs
- Do not introspect opaque objects
- Test with printer simulator

iOS Printing

Todd Ritland
AirPrint Engineer

iOS Printing Topics

Picking what to print

API overview

Printing UI options

iOS Printing Is Easy
But Powerful

Providing Good Content for Printing

Useful, attractive, and high quality

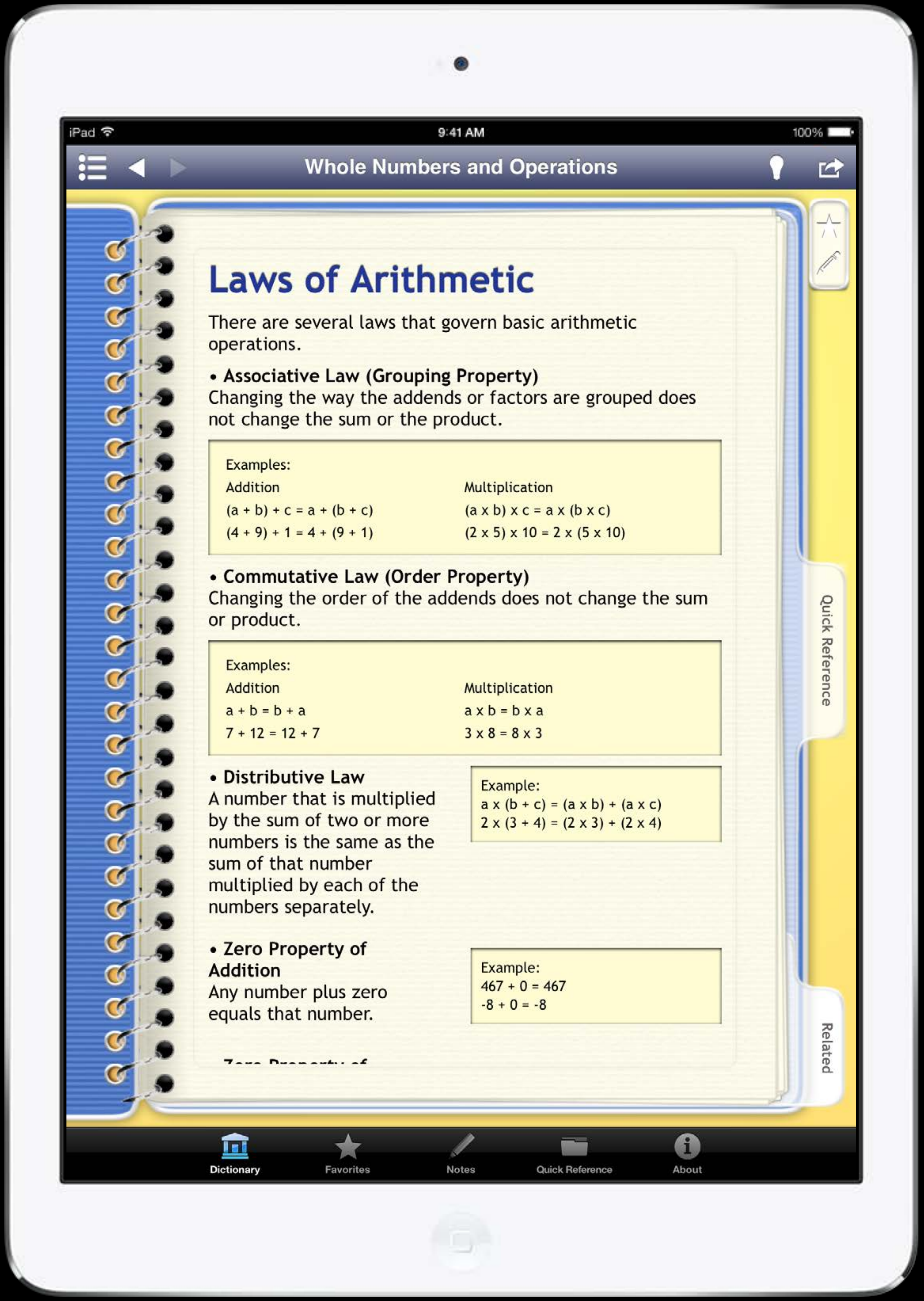
What looks good on screen doesn’t always look good on paper

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

Math dictionary for kids



Laws of Arithmetic

There are several laws that govern basic arithmetic operations.

- **Associative Law (Grouping Property)**
Changing the way the addends or factors are grouped does not change the sum or the product.

Examples:

Addition	Multiplication
(a + b) + c = a + (b + c)	(a x b) x c = a x (b x c)
(4 + 9) + 1 = 4 + (9 + 1)	(2 x 5) x 10 = 2 x (5 x 10)

- **Commutative Law (Order Property)**
Changing the order of the addends does not change the sum or product.

Examples:

Addition	Multiplication
a + b = b + a	a x b = b x a
7 + 12 = 12 + 7	3 x 8 = 8 x 3

- **Distributive Law**
A number that is multiplied by the sum of two or more numbers is the same as the sum of that number multiplied by each of the numbers separately.

Example:
a x (b + c) = (a x b) + (a x c)
2 x (3 + 4) = (2 x 3) + (2 x 4)

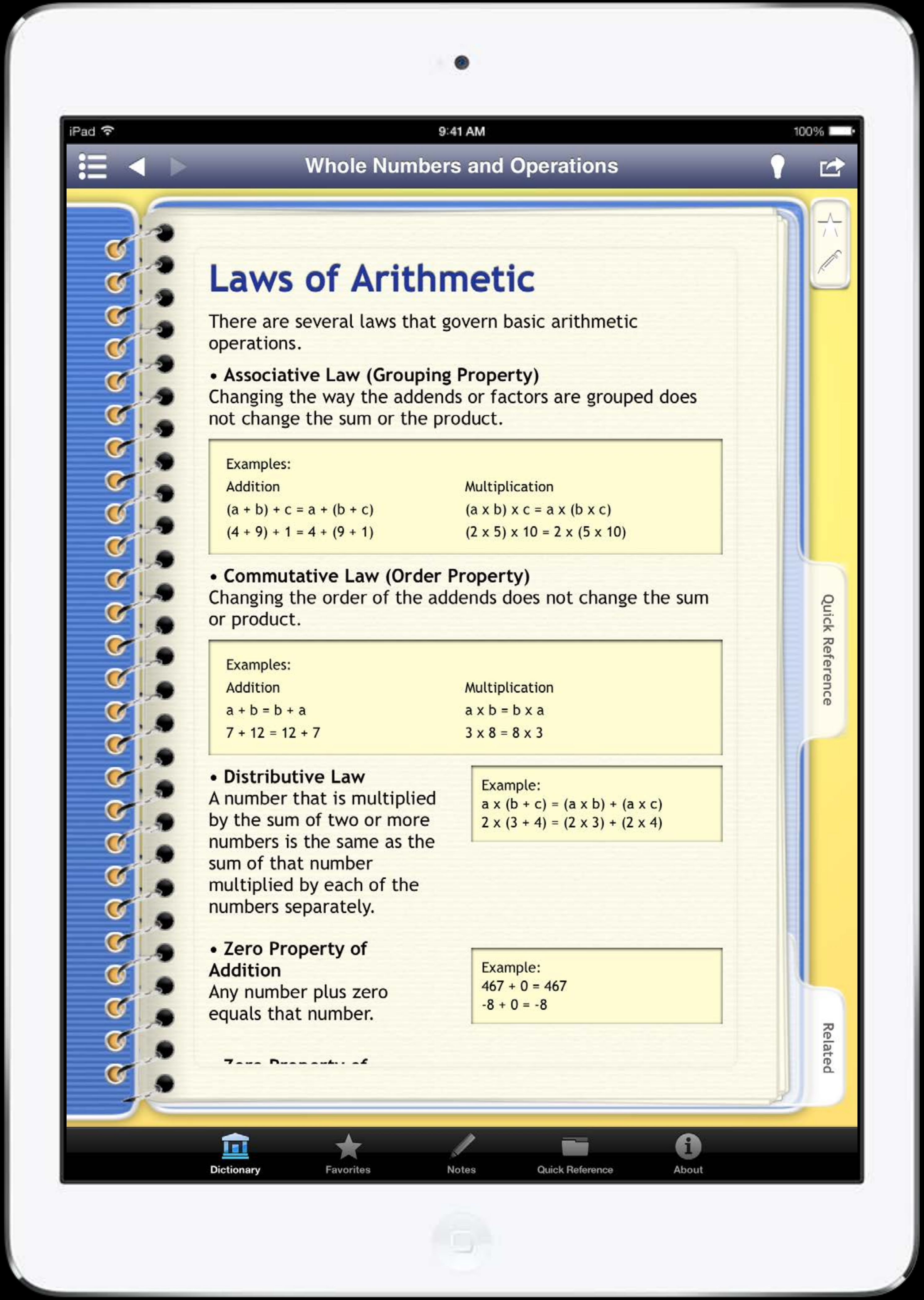
- **Zero Property of Addition**
Any number plus zero equals that number.

Example:
467 + 0 = 467
-8 + 0 = -8

- **Zero Property of**

Example

Math dictionary for kids



Laws of Arithmetic

There are several laws that govern basic arithmetic operations.

• **Associative Law (Grouping Property)**
Changing the way the addends or factors are grouped does not change the sum or the product.

Examples:

Addition	Multiplication
(a + b) + c = a + (b + c)	(a x b) x c = a x (b x c)
(4 + 9) + 1 = 4 + (9 + 1)	(2 x 5) x 10 = 2 x (5 x 10)

• **Commutative Law (Order Property)**
Changing the order of the addends does not change the sum or product.

Examples:

Addition	Multiplication
a + b = b + a	a x b = b x a
7 + 12 = 12 + 7	3 x 8 = 8 x 3

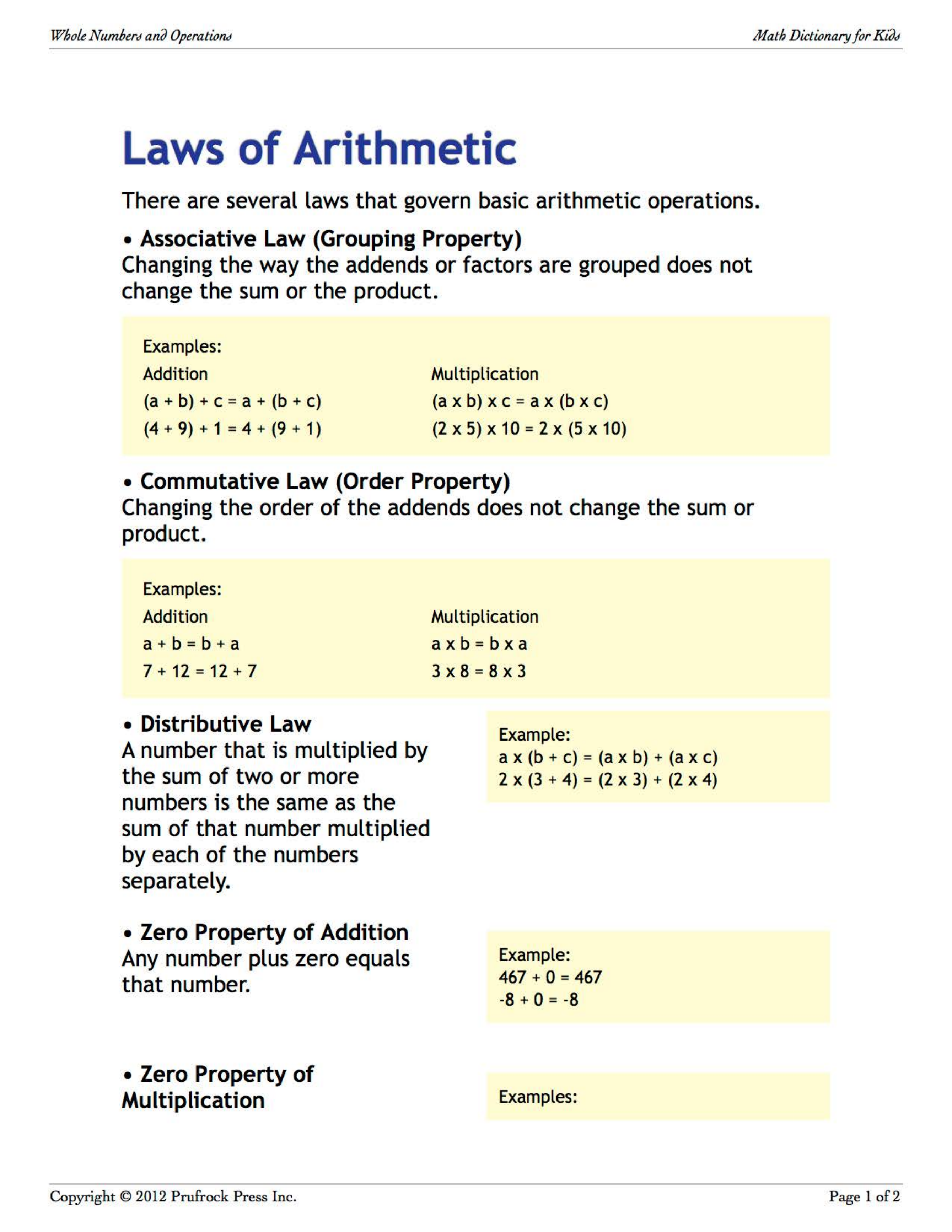
• **Distributive Law**
A number that is multiplied by the sum of two or more numbers is the same as the sum of that number multiplied by each of the numbers separately.

Example:
a x (b + c) = (a x b) + (a x c)
2 x (3 + 4) = (2 x 3) + (2 x 4)

• **Zero Property of Addition**
Any number plus zero equals that number.

Example:
467 + 0 = 467
-8 + 0 = -8

• **Zero Property of**



Whole Numbers and Operations *Math Dictionary for Kids*

Laws of Arithmetic

There are several laws that govern basic arithmetic operations.

• **Associative Law (Grouping Property)**
Changing the way the addends or factors are grouped does not change the sum or the product.

Examples:

Addition	Multiplication
(a + b) + c = a + (b + c)	(a x b) x c = a x (b x c)
(4 + 9) + 1 = 4 + (9 + 1)	(2 x 5) x 10 = 2 x (5 x 10)

• **Commutative Law (Order Property)**
Changing the order of the addends does not change the sum or product.

Examples:

Addition	Multiplication
a + b = b + a	a x b = b x a
7 + 12 = 12 + 7	3 x 8 = 8 x 3

• **Distributive Law**
A number that is multiplied by the sum of two or more numbers is the same as the sum of that number multiplied by each of the numbers separately.

Example:
a x (b + c) = (a x b) + (a x c)
2 x (3 + 4) = (2 x 3) + (2 x 4)

• **Zero Property of Addition**
Any number plus zero equals that number.

Example:
467 + 0 = 467
-8 + 0 = -8

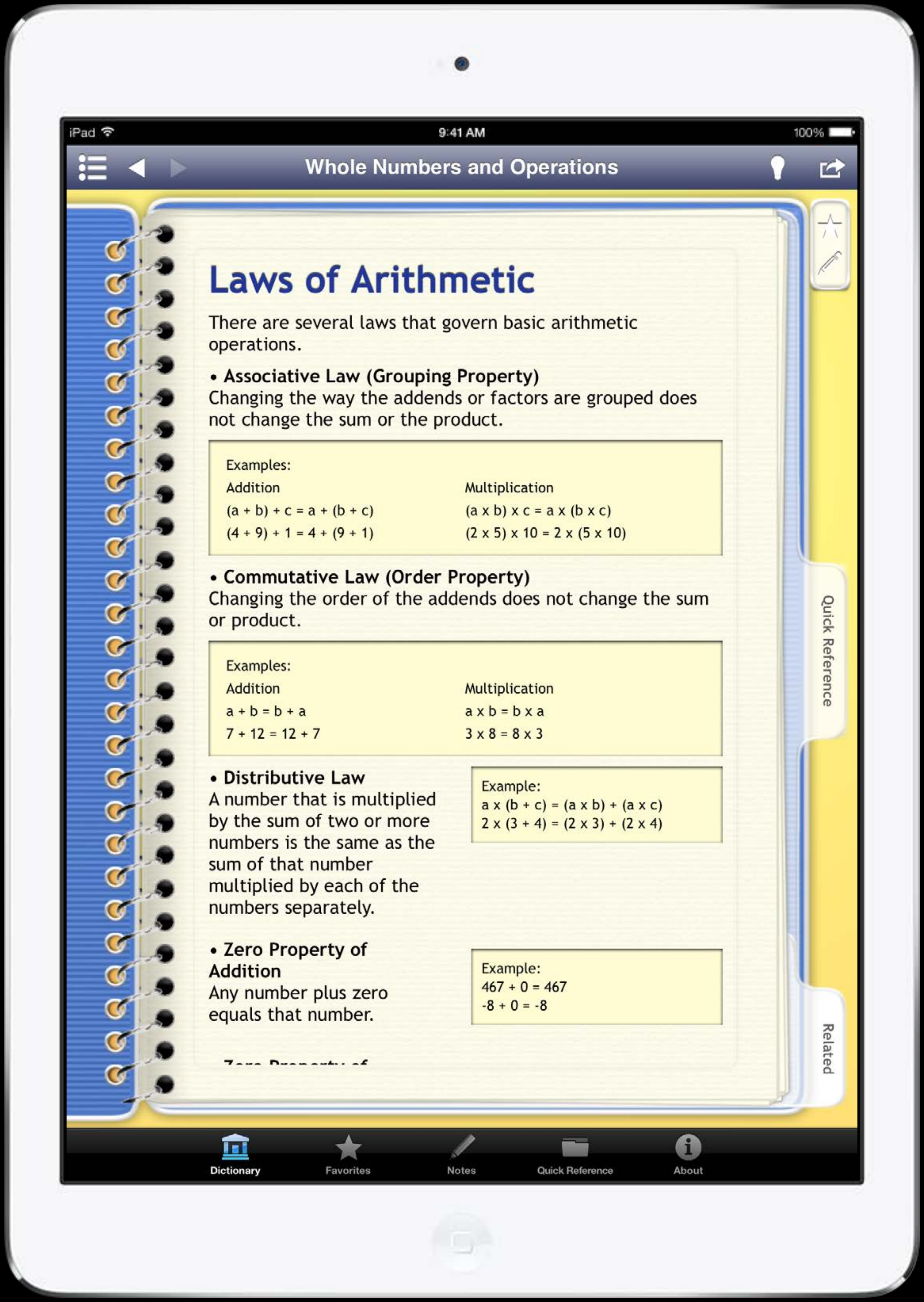
• **Zero Property of Multiplication**

Examples:

Copyright © 2012 Prufrock Press Inc. Page 1 of 2

Example

Math dictionary for kids



Laws of Arithmetic

There are several laws that govern basic arithmetic operations.

- **Associative Law (Grouping Property)**
Changing the way the addends or factors are grouped does not change the sum or the product.

Examples:

Addition	Multiplication
(a + b) + c = a + (b + c)	(a x b) x c = a x (b x c)
(4 + 9) + 1 = 4 + (9 + 1)	(2 x 5) x 10 = 2 x (5 x 10)

- **Commutative Law (Order Property)**
Changing the order of the addends does not change the sum or product.

Examples:

Addition	Multiplication
a + b = b + a	a x b = b x a
7 + 12 = 12 + 7	3 x 8 = 8 x 3

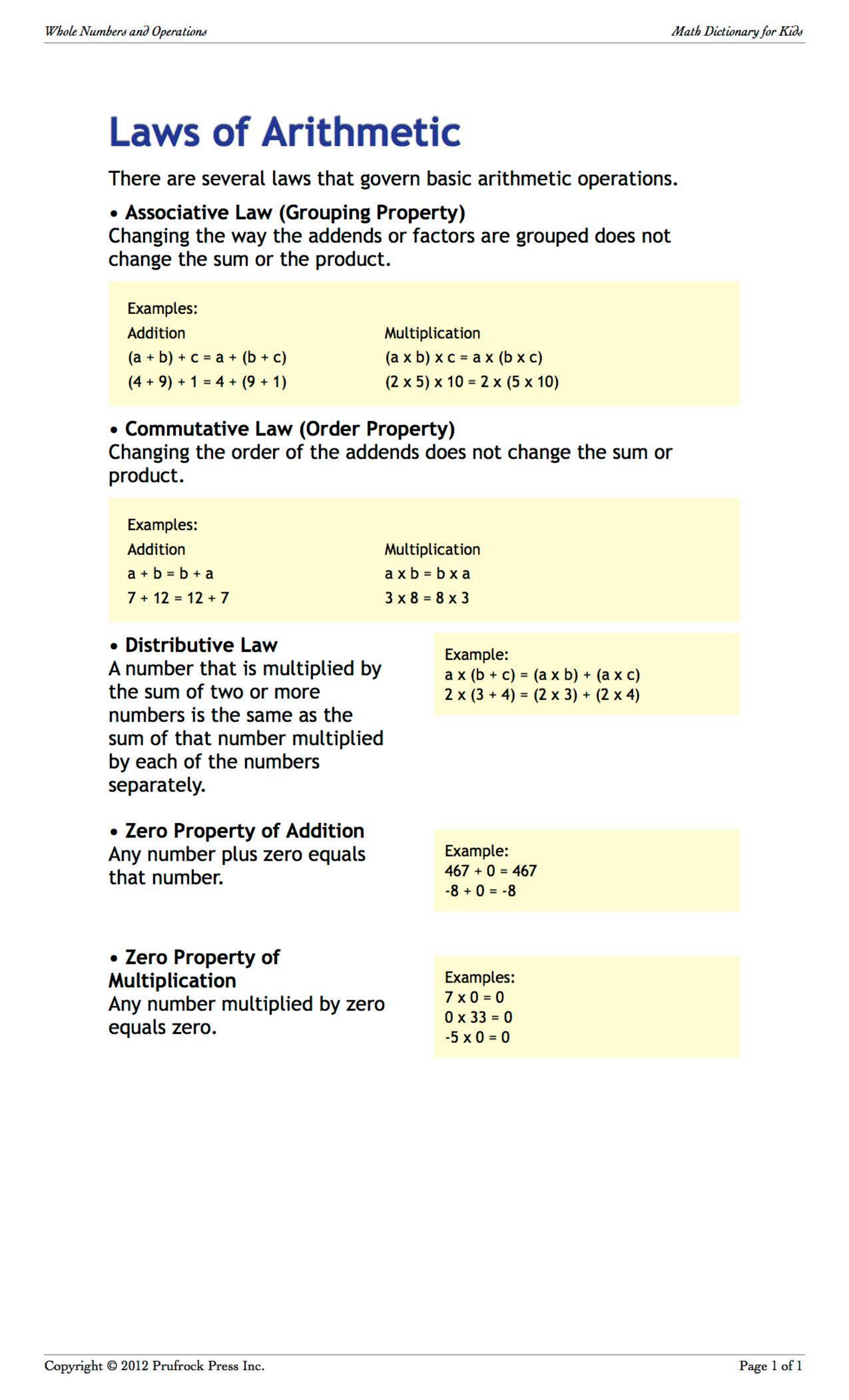
- **Distributive Law**
A number that is multiplied by the sum of two or more numbers is the same as the sum of that number multiplied by each of the numbers separately.

Example:
a x (b + c) = (a x b) + (a x c)
2 x (3 + 4) = (2 x 3) + (2 x 4)

- **Zero Property of Addition**
Any number plus zero equals that number.

Example:
467 + 0 = 467
-8 + 0 = -8

- **Zero Property of**



Laws of Arithmetic

There are several laws that govern basic arithmetic operations.

- **Associative Law (Grouping Property)**
Changing the way the addends or factors are grouped does not change the sum or the product.

Examples:

Addition	Multiplication
(a + b) + c = a + (b + c)	(a x b) x c = a x (b x c)
(4 + 9) + 1 = 4 + (9 + 1)	(2 x 5) x 10 = 2 x (5 x 10)

- **Commutative Law (Order Property)**
Changing the order of the addends does not change the sum or product.

Examples:

Addition	Multiplication
a + b = b + a	a x b = b x a
7 + 12 = 12 + 7	3 x 8 = 8 x 3

- **Distributive Law**
A number that is multiplied by the sum of two or more numbers is the same as the sum of that number multiplied by each of the numbers separately.

Example:
a x (b + c) = (a x b) + (a x c)
2 x (3 + 4) = (2 x 3) + (2 x 4)

- **Zero Property of Addition**
Any number plus zero equals that number.

Example:
467 + 0 = 467
-8 + 0 = -8

- **Zero Property of Multiplication**
Any number multiplied by zero equals zero.

Examples:
7 x 0 = 0
0 x 33 = 0
-5 x 0 = 0

API Overview

Basic Steps for Printing with UI

Your app will

- Get the print controller or activity controller
- Set up the attributes for the job
- Provide content to print
- Present the UI

Basic Steps for Printing with UI

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

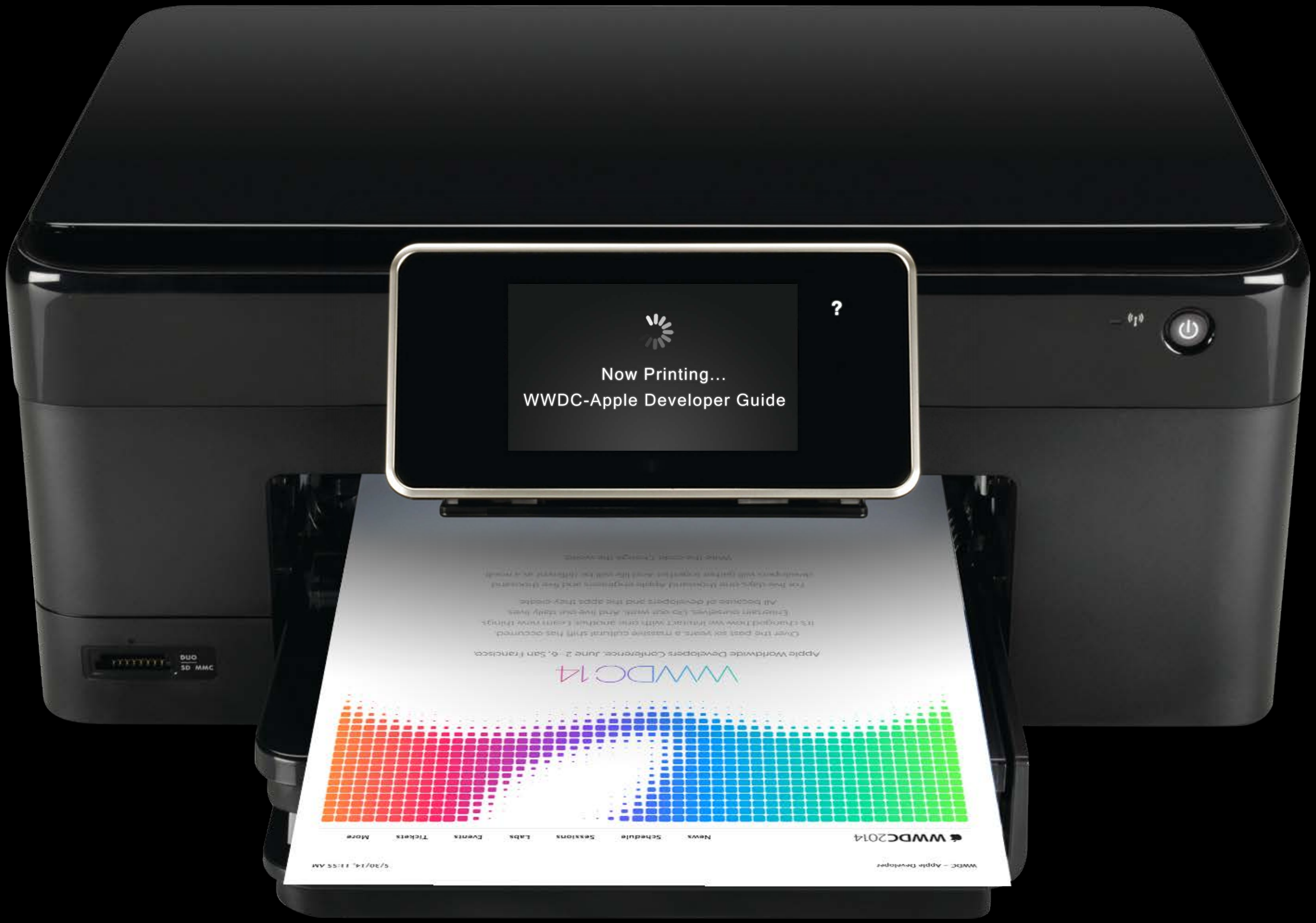
- UIPrintInteractionController
- UIActivityViewController

iOS Printing Classes

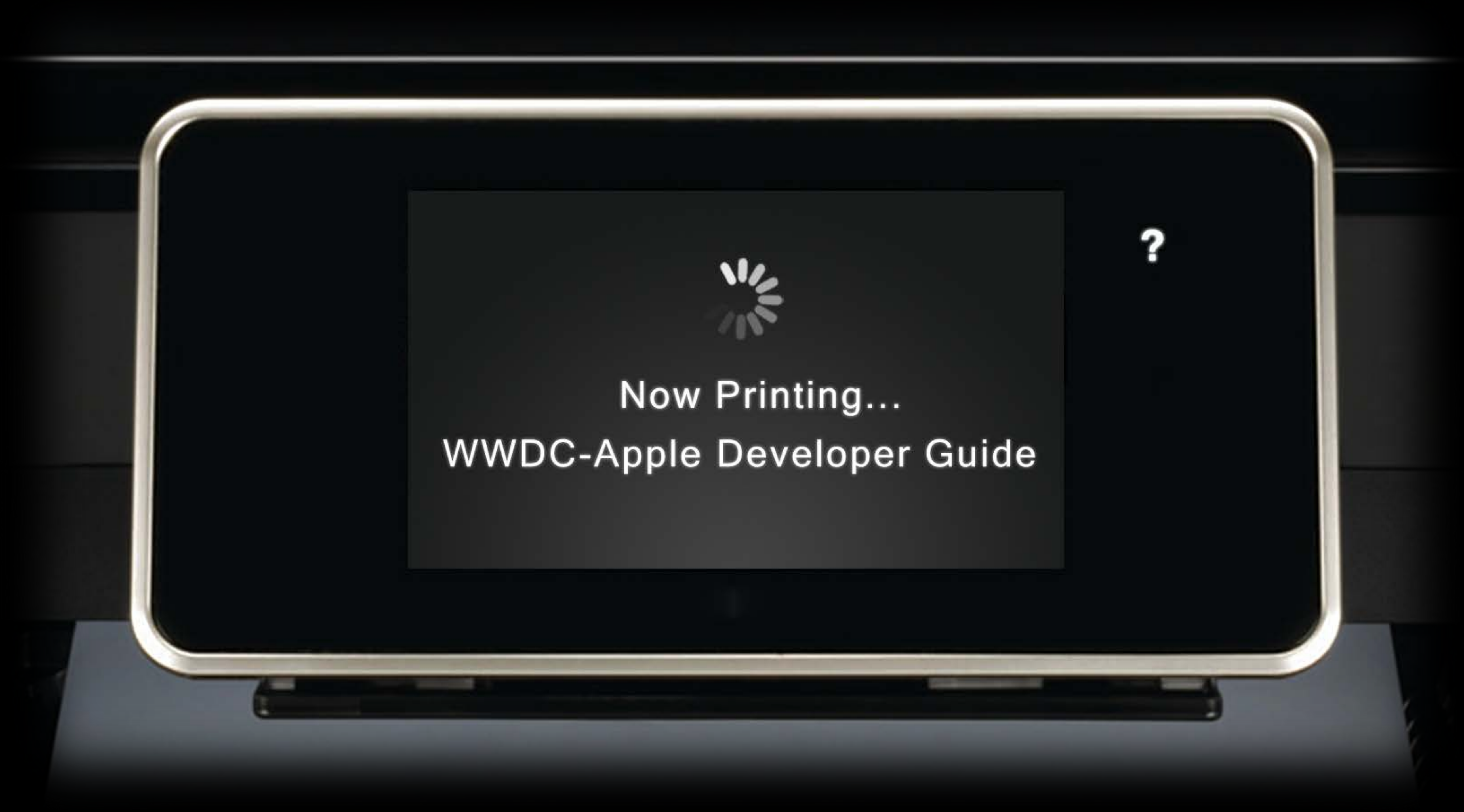
```
UIPrintInfo
UIPrintPaper
UIPrintFormatter
  UISimpleTextPrintFormatter
  UIMarkupTextPrintFormatter
  UIViewPrintFormatter
UIPrintPageRenderer

UIPrintInteractionController
UIActivityViewController
```

UIPrintInfo

Setting the Job Name



Setting the Job Name



Now Printing...

WWDC-Apple Developer Guide

Output Type

Tell the printing system about the type of content to be printed

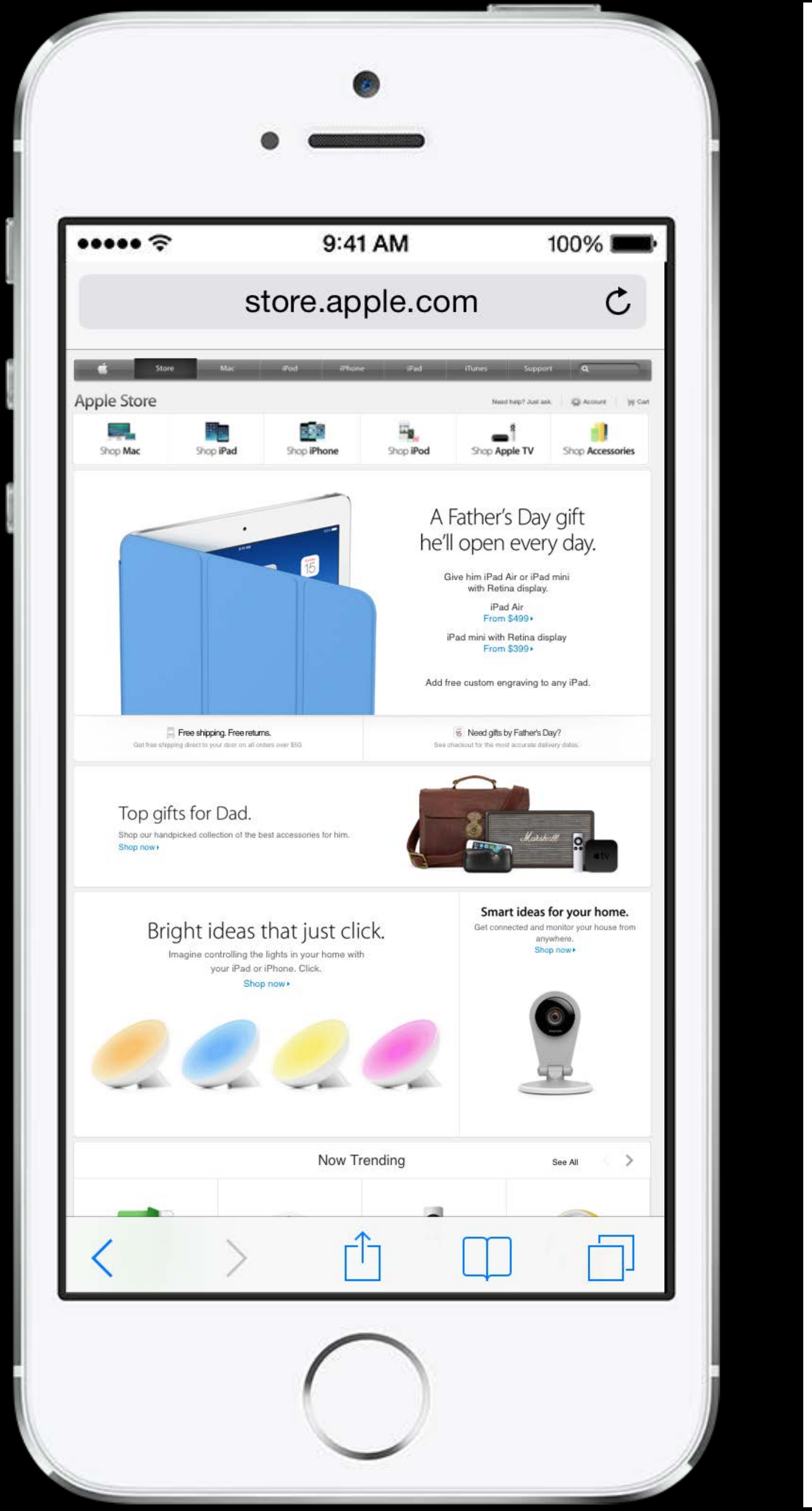
Allows the printing system to the choose appropriate

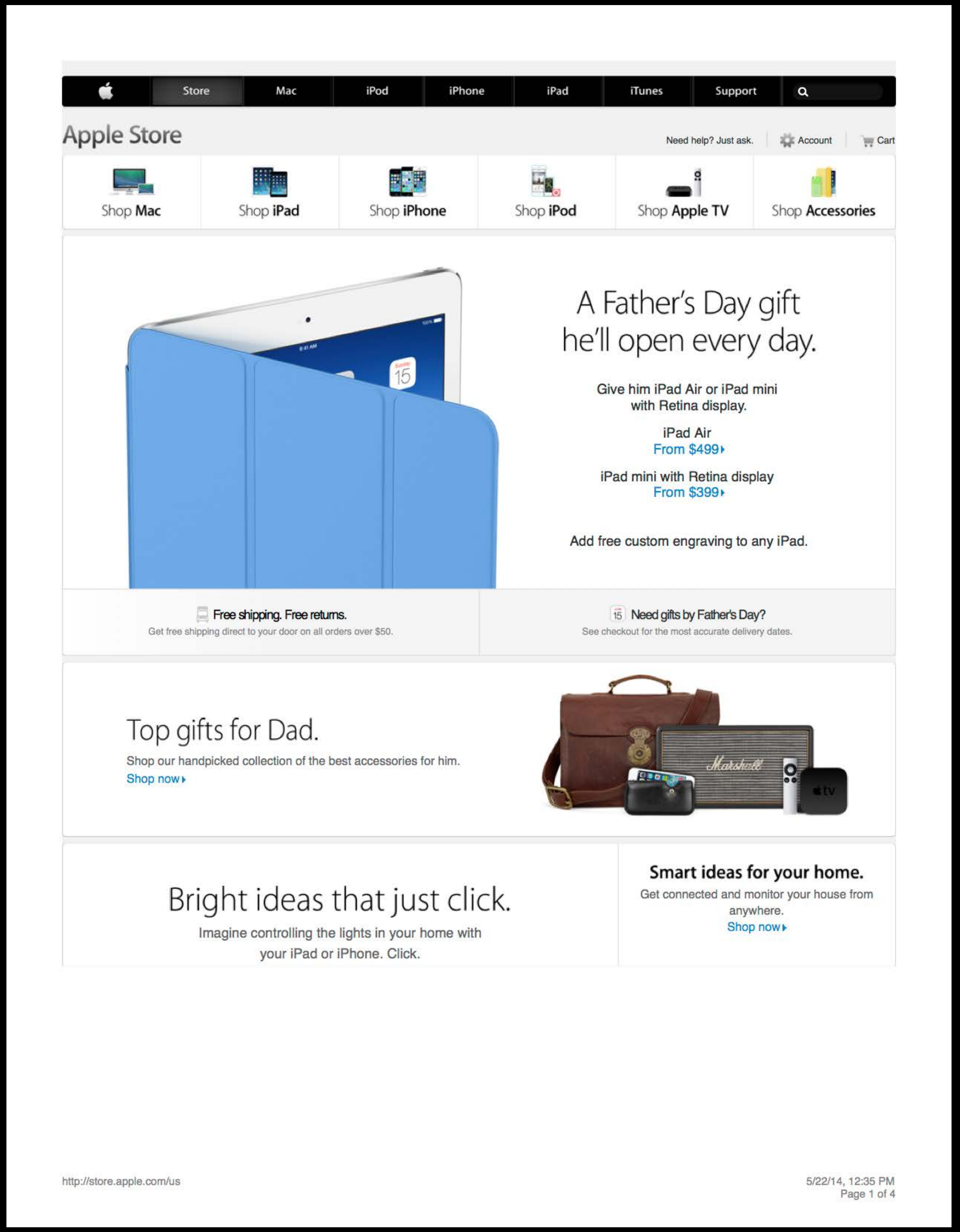
- Paper size
- Print quality mode
- Appropriate UI

Document

UIPrintInfoOutputGeneral

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





A Father's Day gift he'll open every day.

Give him iPad Air or iPad mini with Retina display.

iPad Air
From $499 ▸

iPad mini with Retina display
From $399 ▸

Add free custom engraving to any iPad.

Free shipping. Free returns.
Get free shipping direct to your door on all orders over $50.

Need gifts by Father's Day?
See checkout for the most accurate delivery dates.

Top gifts for Dad.
Shop our handpicked collection of the best accessories for him.
Shop now ▸

Bright ideas that just click.
Imagine controlling the lights in your home with your iPad or iPhone. Click.

Smart ideas for your home.
Get connected and monitor your house from anywhere.
Shop now ▸

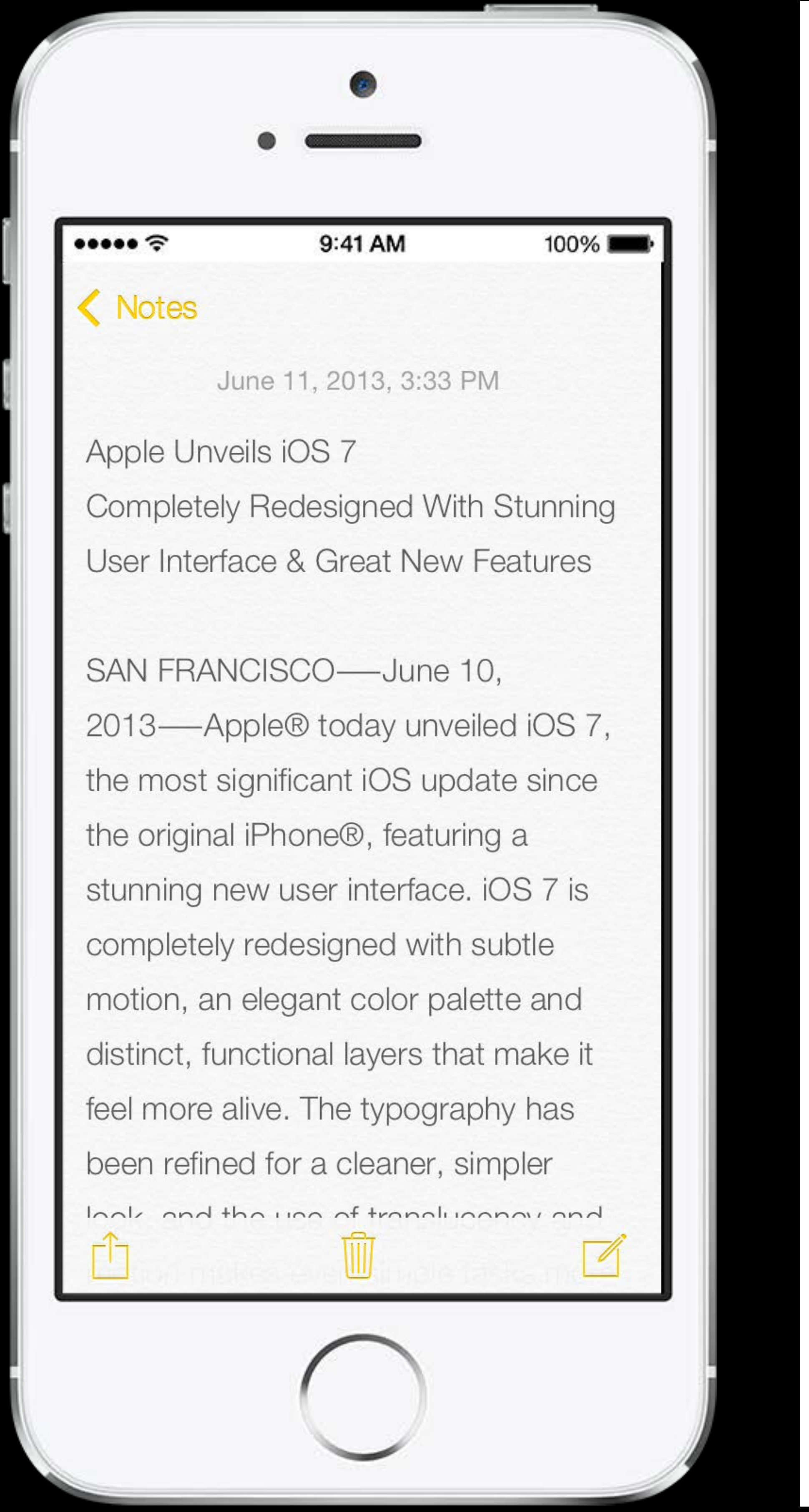
http://store.apple.com/us

5/22/14, 12:35 PM
Page 1 of 4

Document Grayscale

`UIPrintInfoOutputGrayscale`

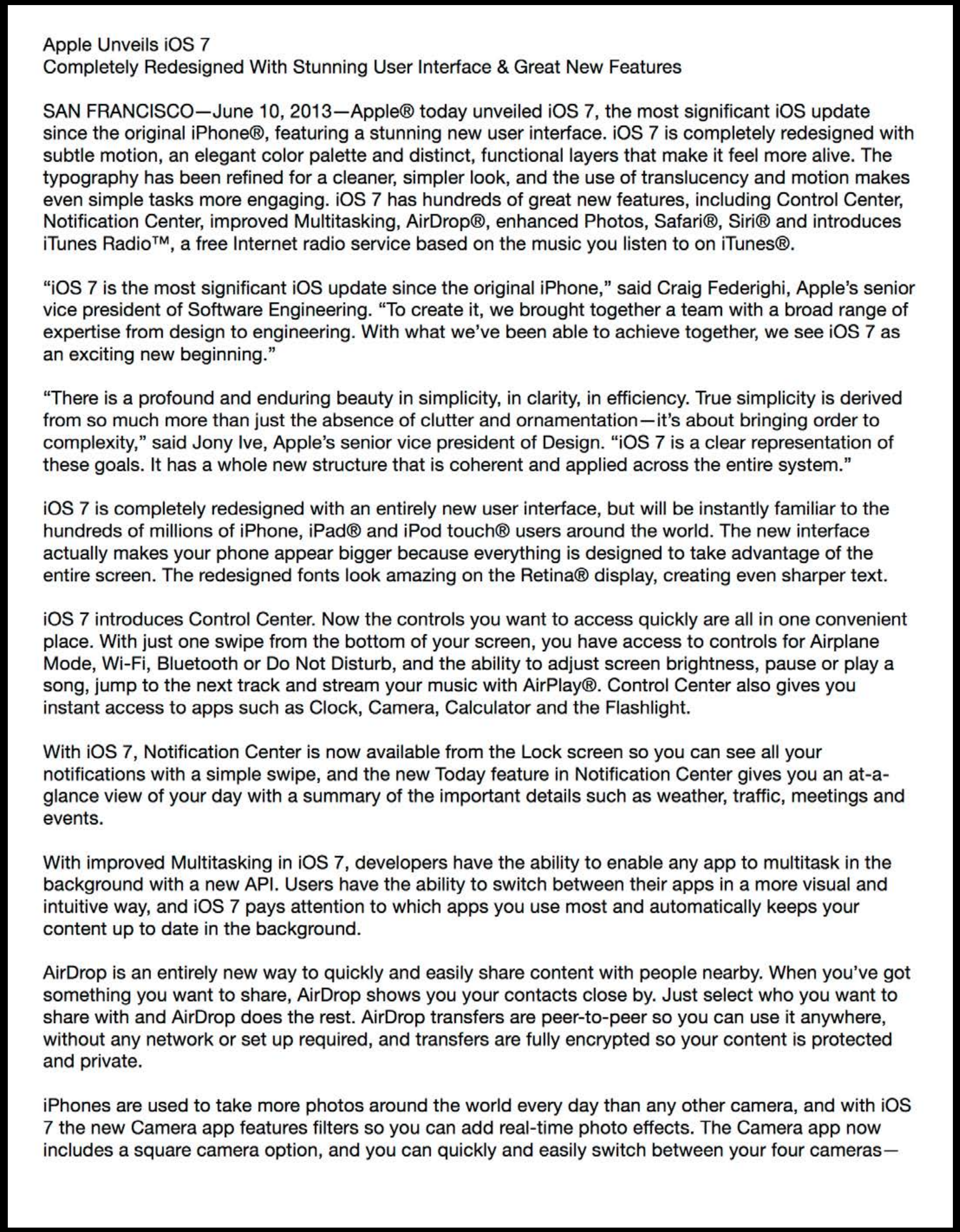
- Monochrome text and graphics
- Improved print speed
- Reduced ink usage
- Document paper size
- Duplex allowed
- Page range allowed



June 11, 2013, 3:33 PM

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



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

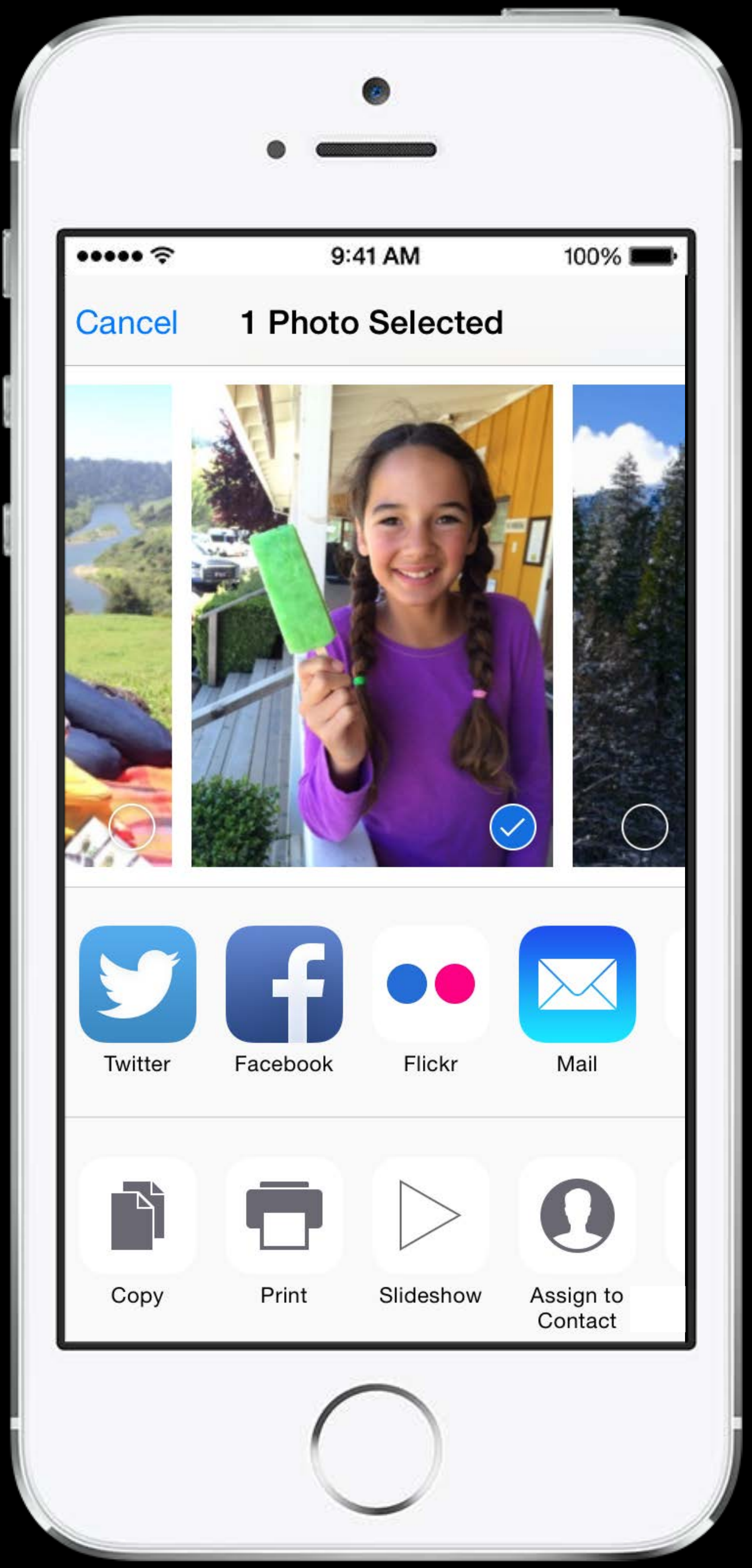


Photo High Quality Grayscale

`UIPrintInfoOutputPhotoGrayscale`

Photo paper size

Borderless if available

No duplex mode

No page range

iOS Printing Classes

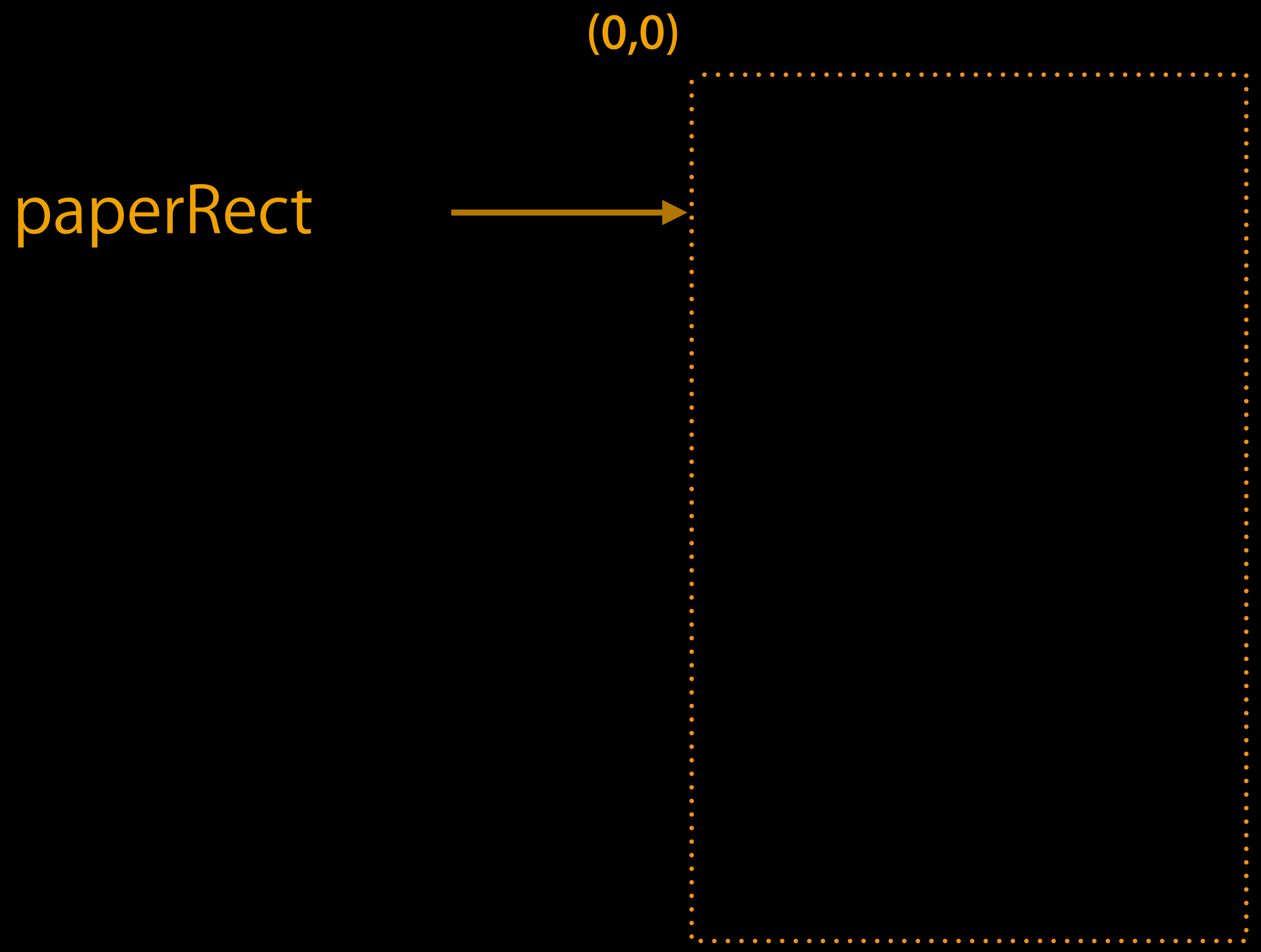
```
UIPrintInfo
UIPrintPaper
UIPrintFormatter
  UISimpleTextPrintFormatter
  UIMarkupTextPrintFormatter
  UIViewPrintFormatter
UIPrintPageRenderer

UIPrintInteractionController
UIActivityViewController
```

iOS Printing Classes

```
UIPrintInfo
UIPrintPaper
UIPrintFormatter
  UISimpleTextPrintFormatter
  UIMarkupTextPrintFormatter
  UIViewPrintFormatter
UIPrintPageRenderer

UIPrintInteractionController
UIActivityViewController
```

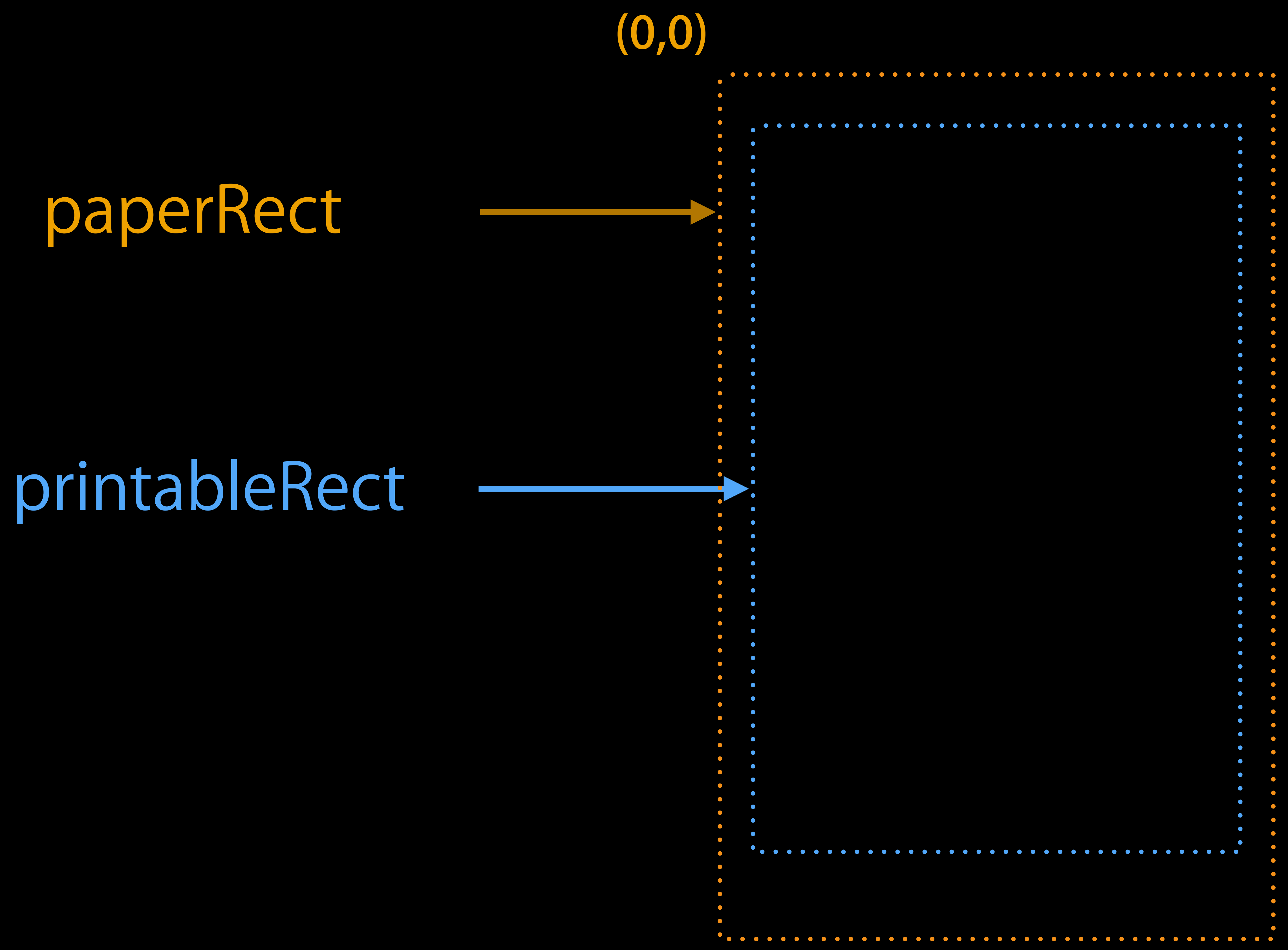
Paper

`UIPrintPaper`



Paper

UIPrintPaper



Providing Content

Printing items that already exist

Printing simple content

Fully custom drawn pages

Easy

Printing items that already exist

PDF files
Image files

Printing simple content

Fully custom drawn pages

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];
    }
}
```

Easy

Printing items that already exist

PDF files
Image files

Printing simple content

Fully custom drawn pages

Printing items that already exist

Basic

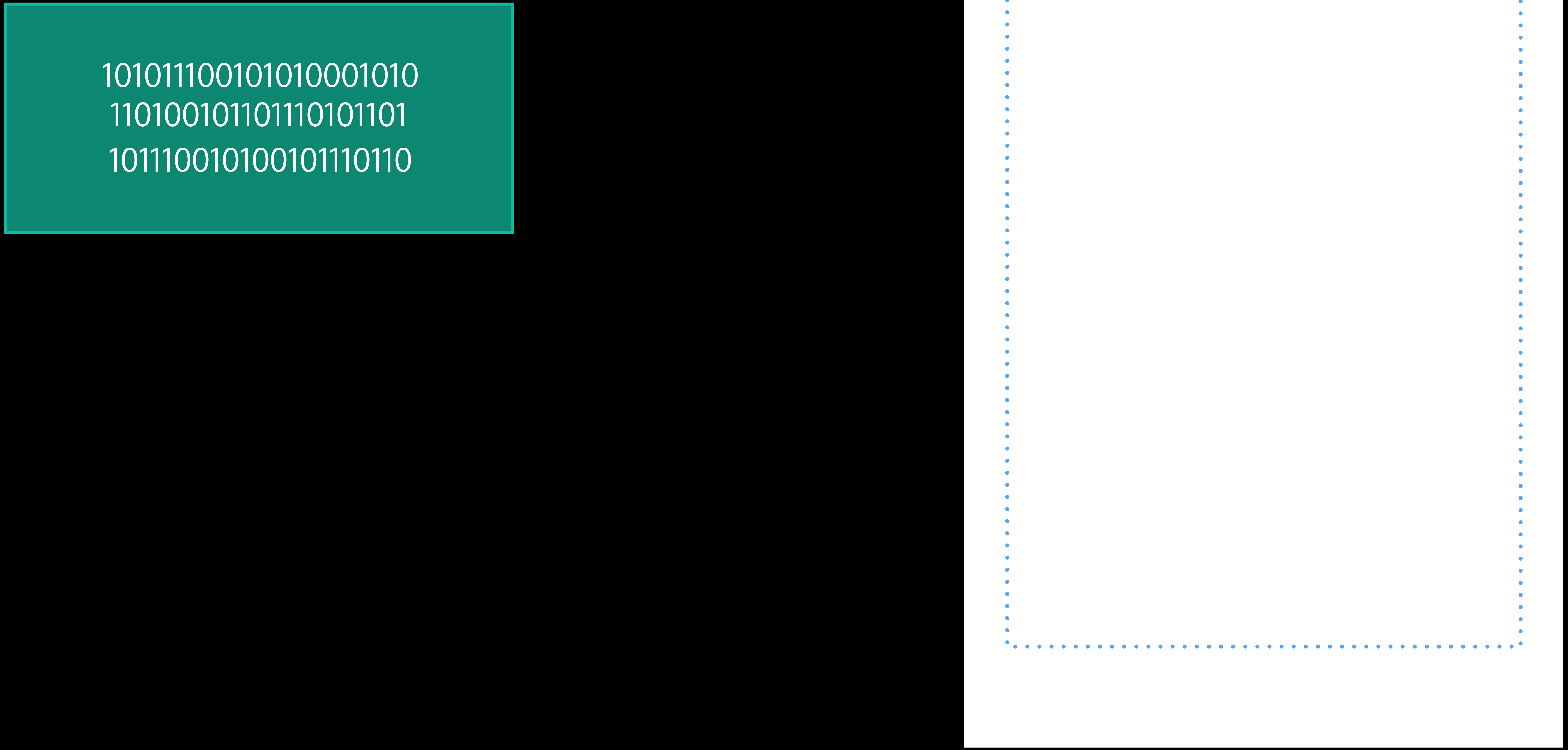
Printing simple content

Text, HTML, etc.
Use one of the provided Formatter classes

Fully custom drawn pages

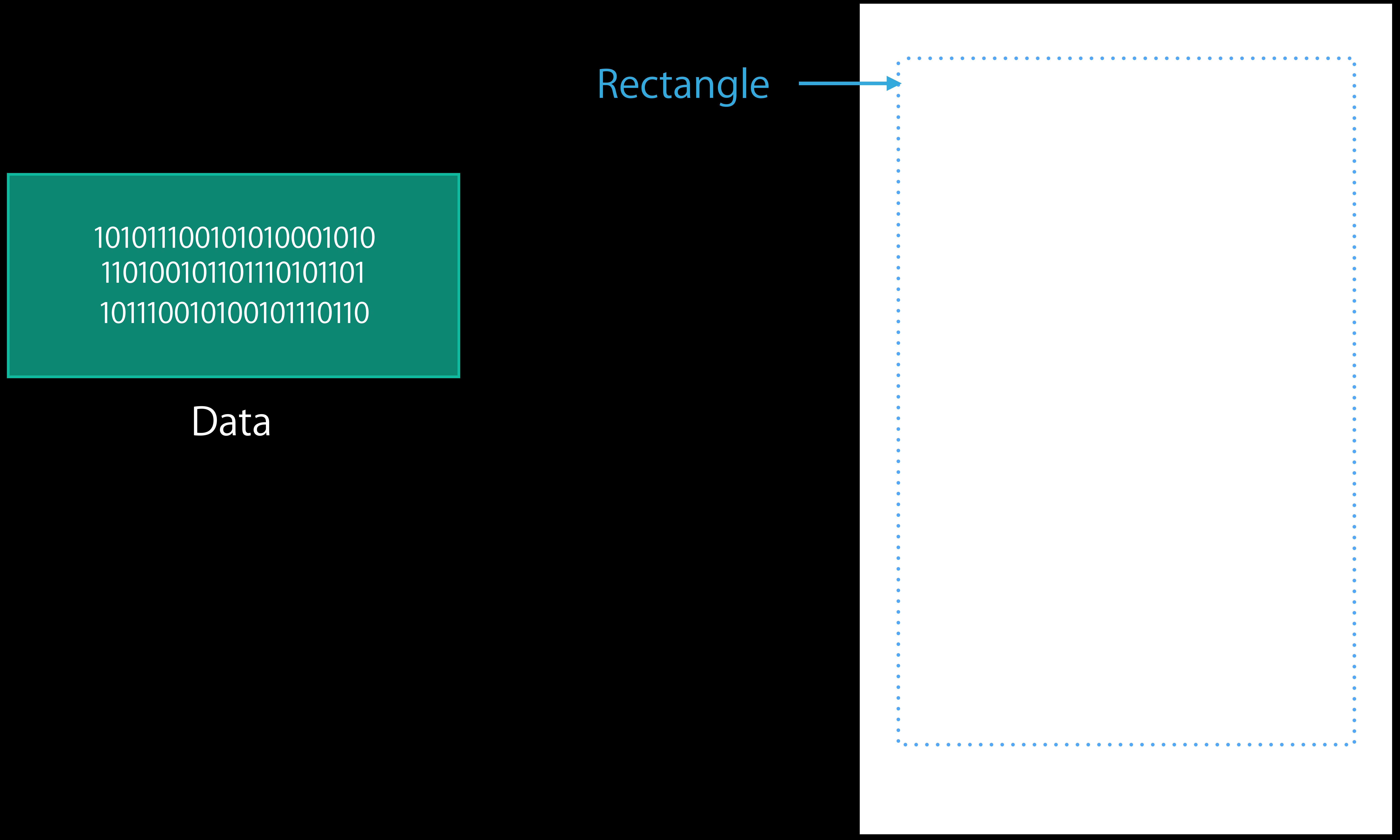
Formatters

What Is a Formatter?



101011100101010001010
110100101101110101101
101110010100101110110

What Is a Formatter?

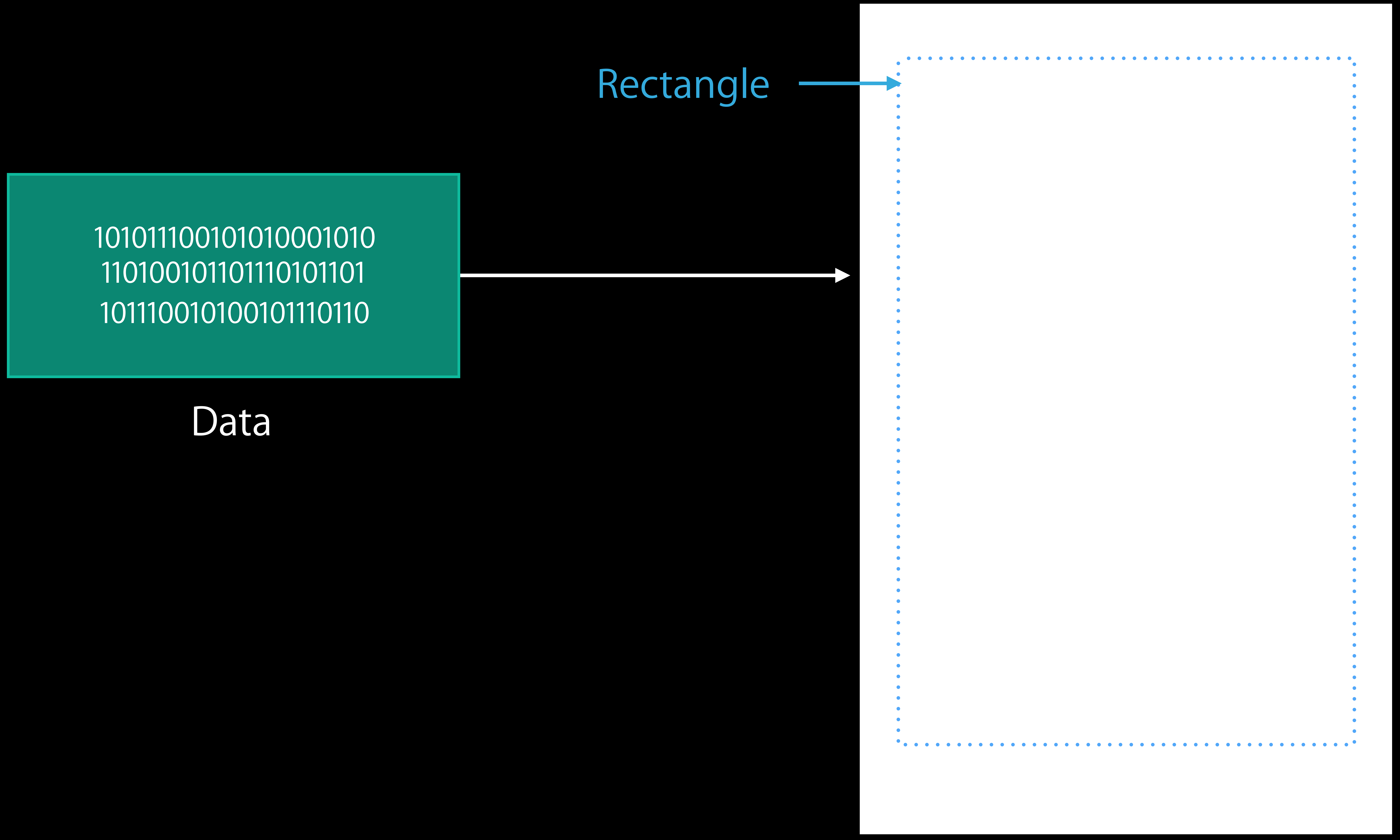


101011100101010001010
110100101101110101101
101110010100101110110

Data

Rectangle

What Is a Formatter?

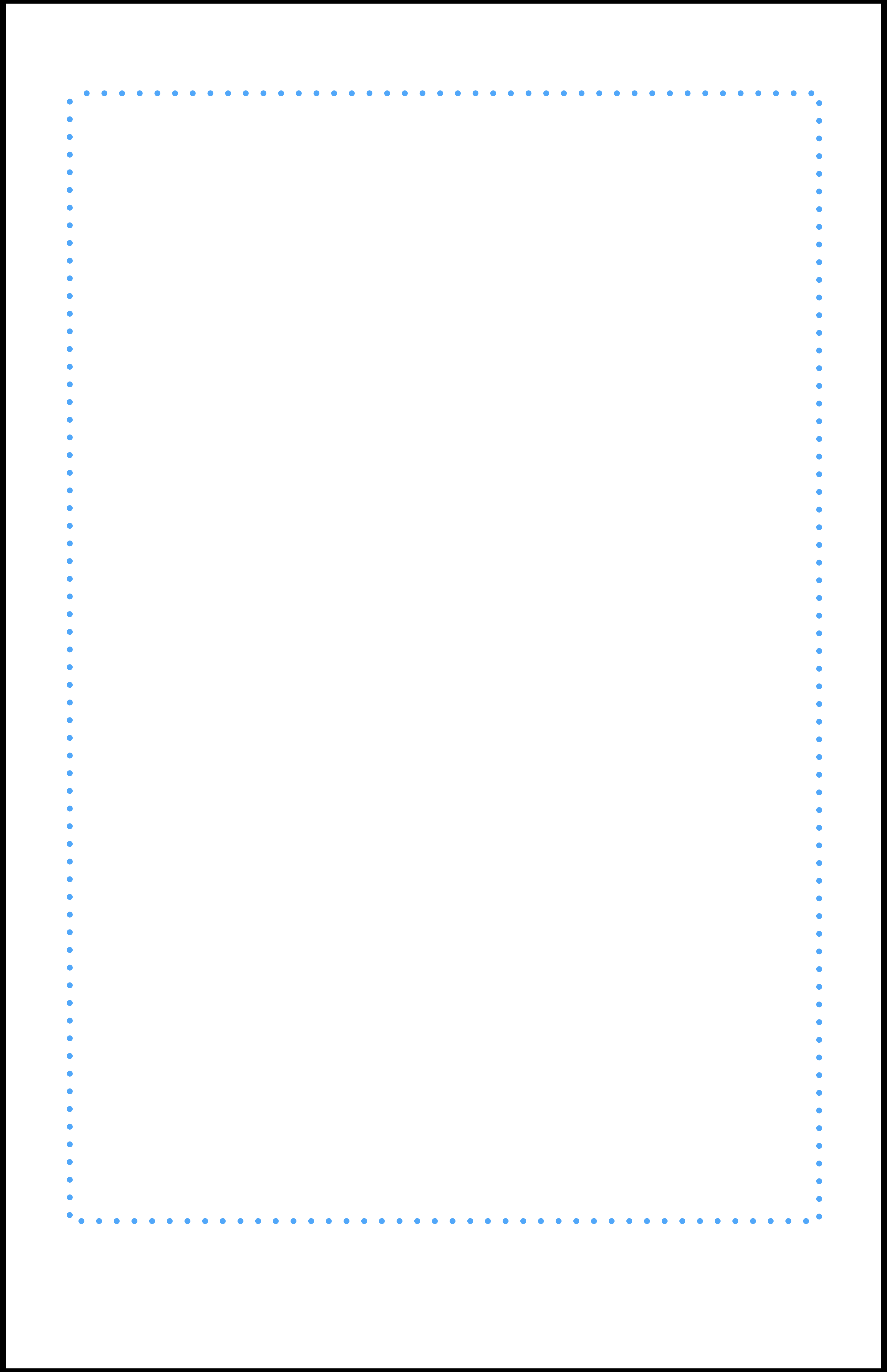


Rectangle

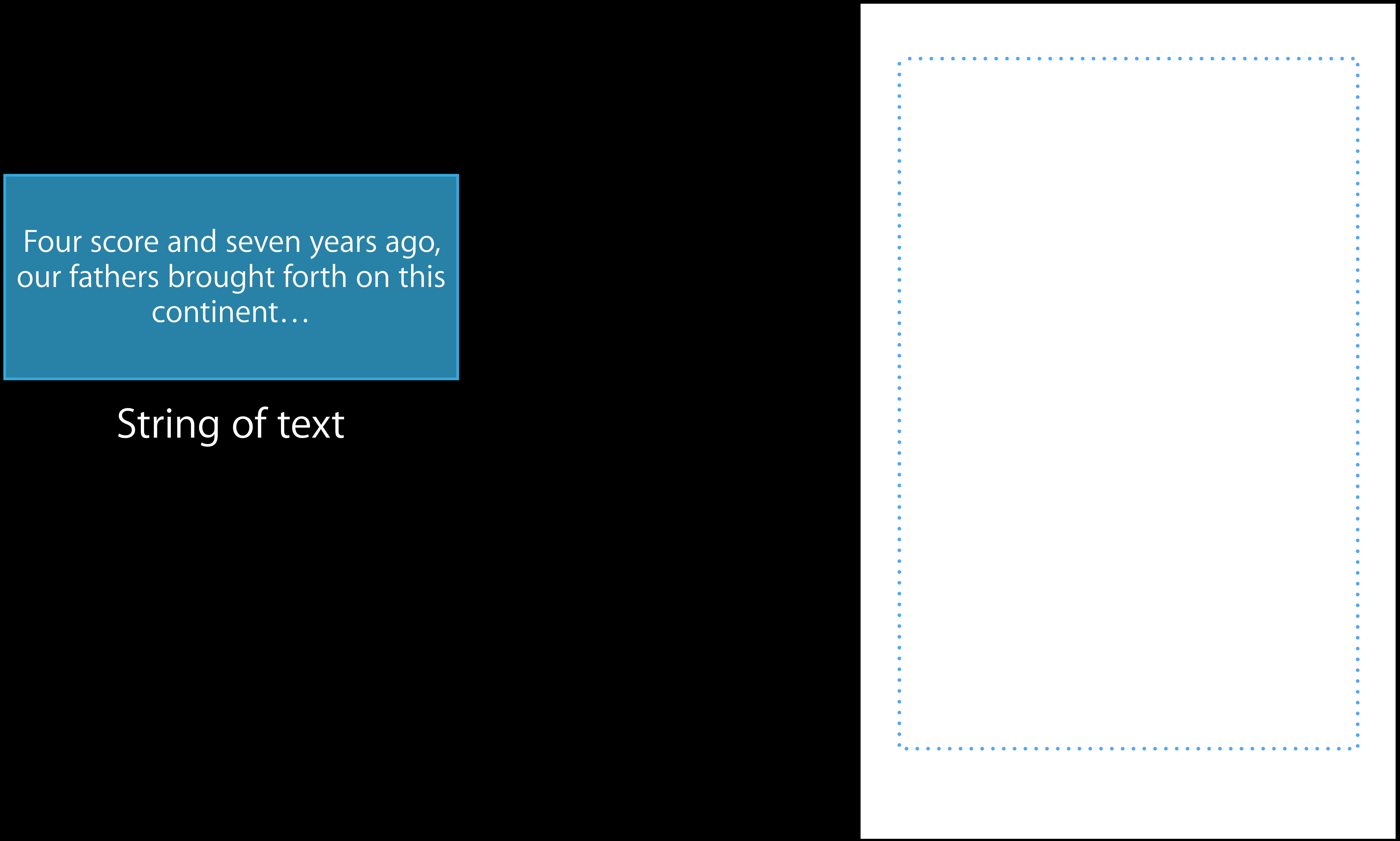
101011100101010001010
110100101101110101101
101110010100101110110

Data

What Is a Formatter?



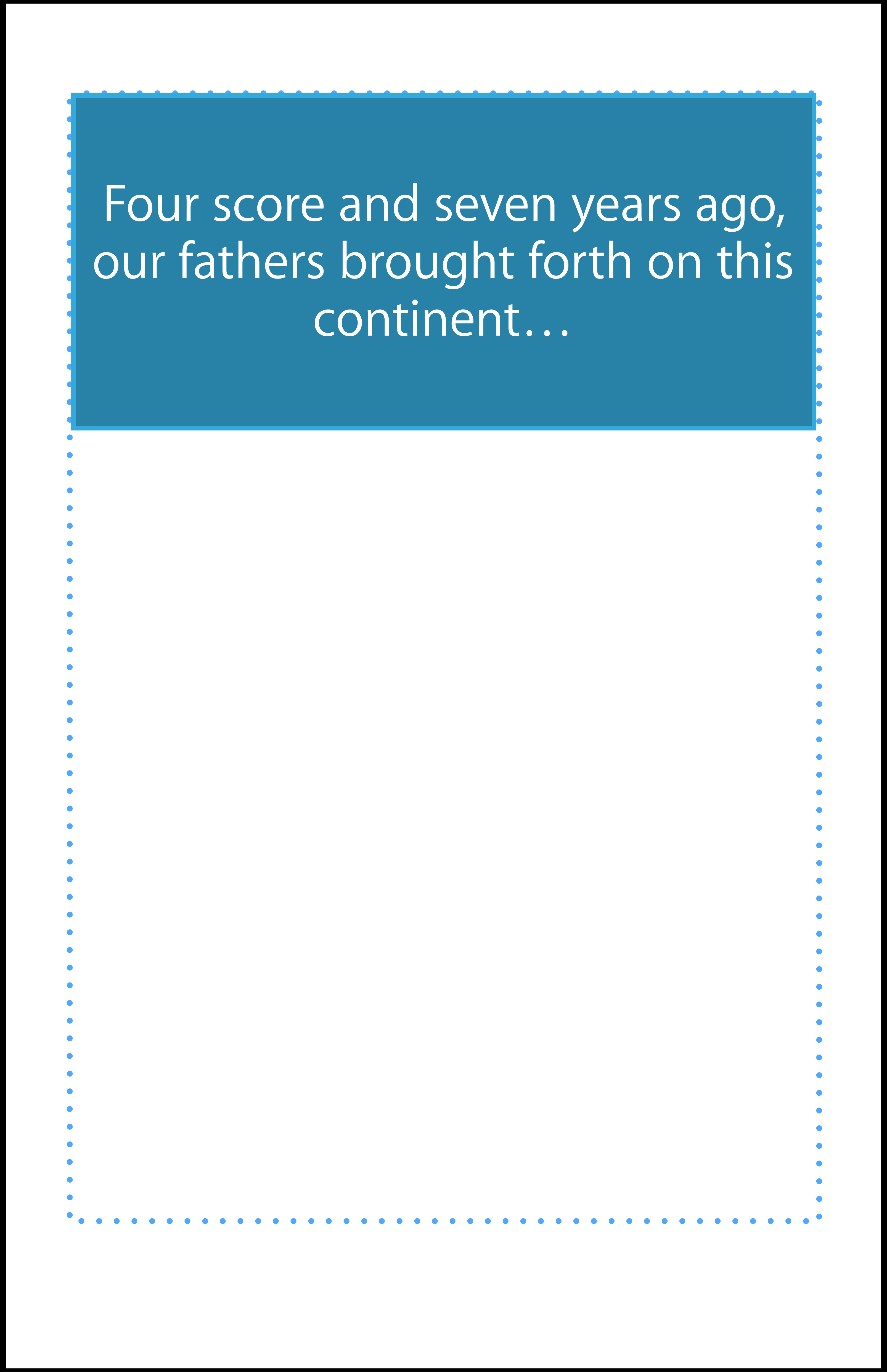
What Is a Formatter?



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

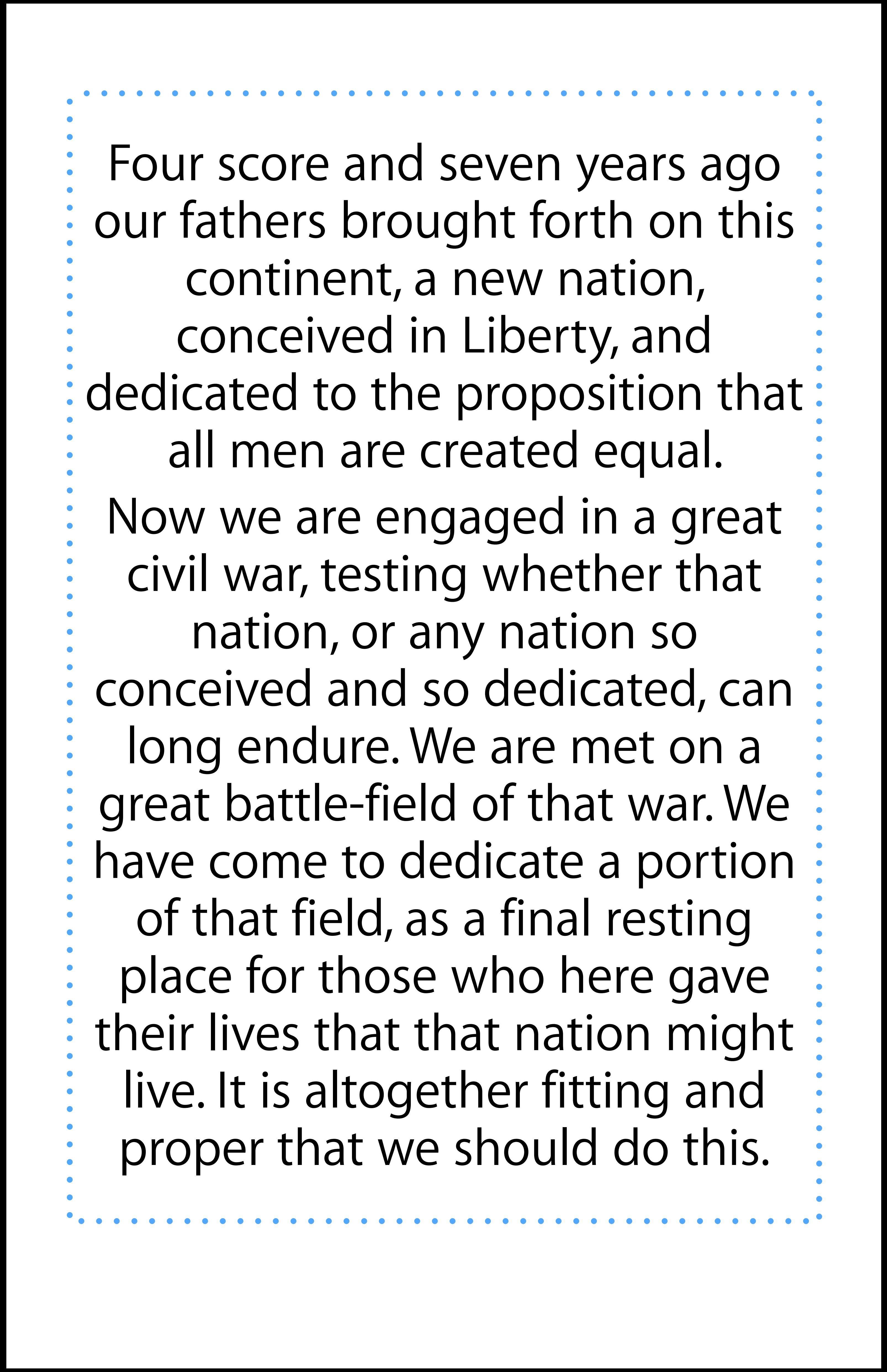
String of text

What Is a Formatter?



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

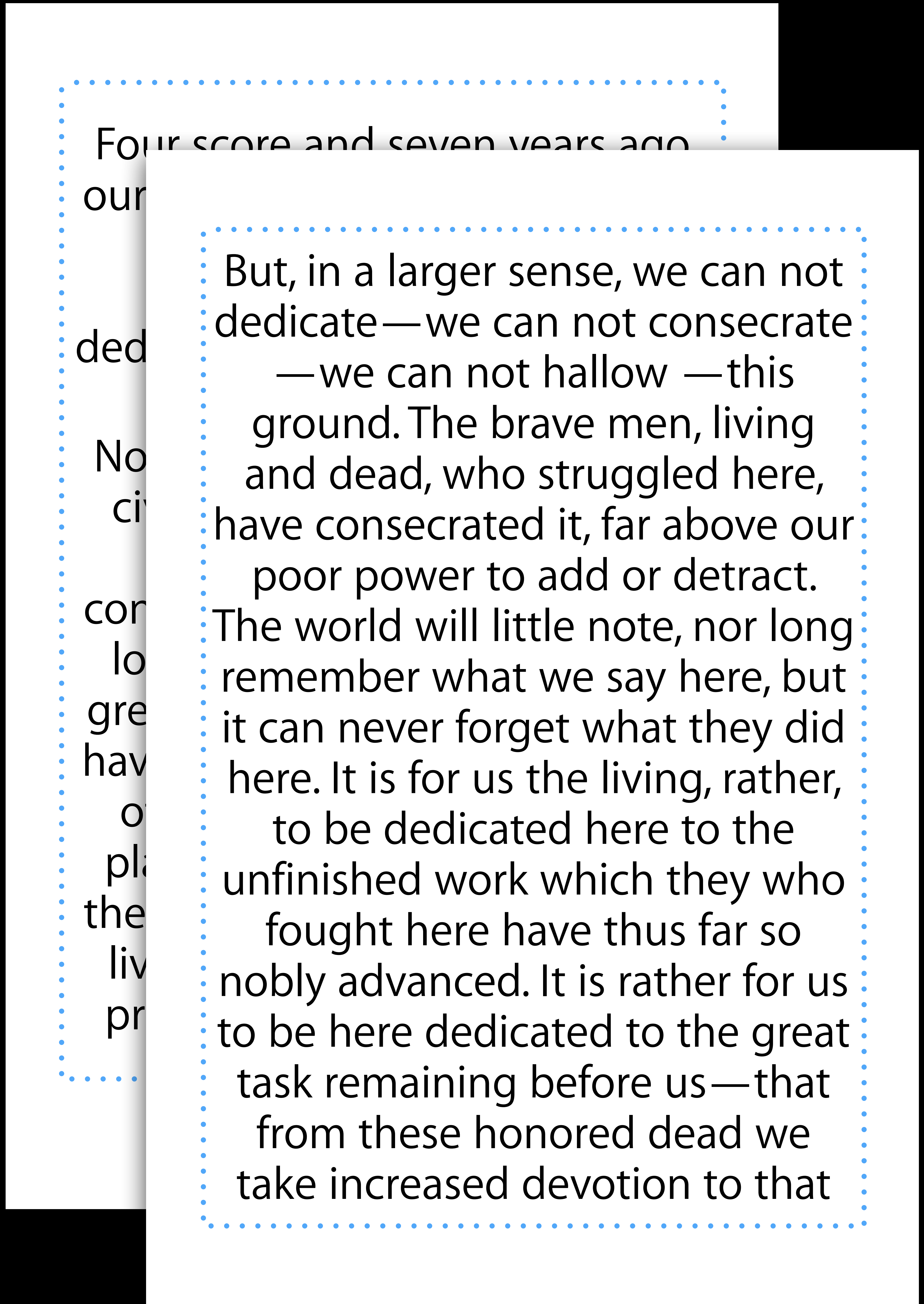
What Is a Formatter?



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?



Four score and seven years ago
our
ded
No
ci
co
lo
gre
hav
o
pl
the
liv
pr

But, in a larger sense, we can not
dedicate—we can not consecrate
—we can not hallow —this
ground. The brave men, living
and dead, who struggled here,
have consecrated it, far above our
poor power to add or detract.
The world will little note, nor long
remember what we say here, but
it can never forget what they did
here. It is for us the living, rather,
to be dedicated here to the
unfinished work which they who
fought here have thus far so
nobly advanced. It is rather for us
to be here dedicated to the great
task remaining before us—that
from these honored dead we
take increased devotion to that

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

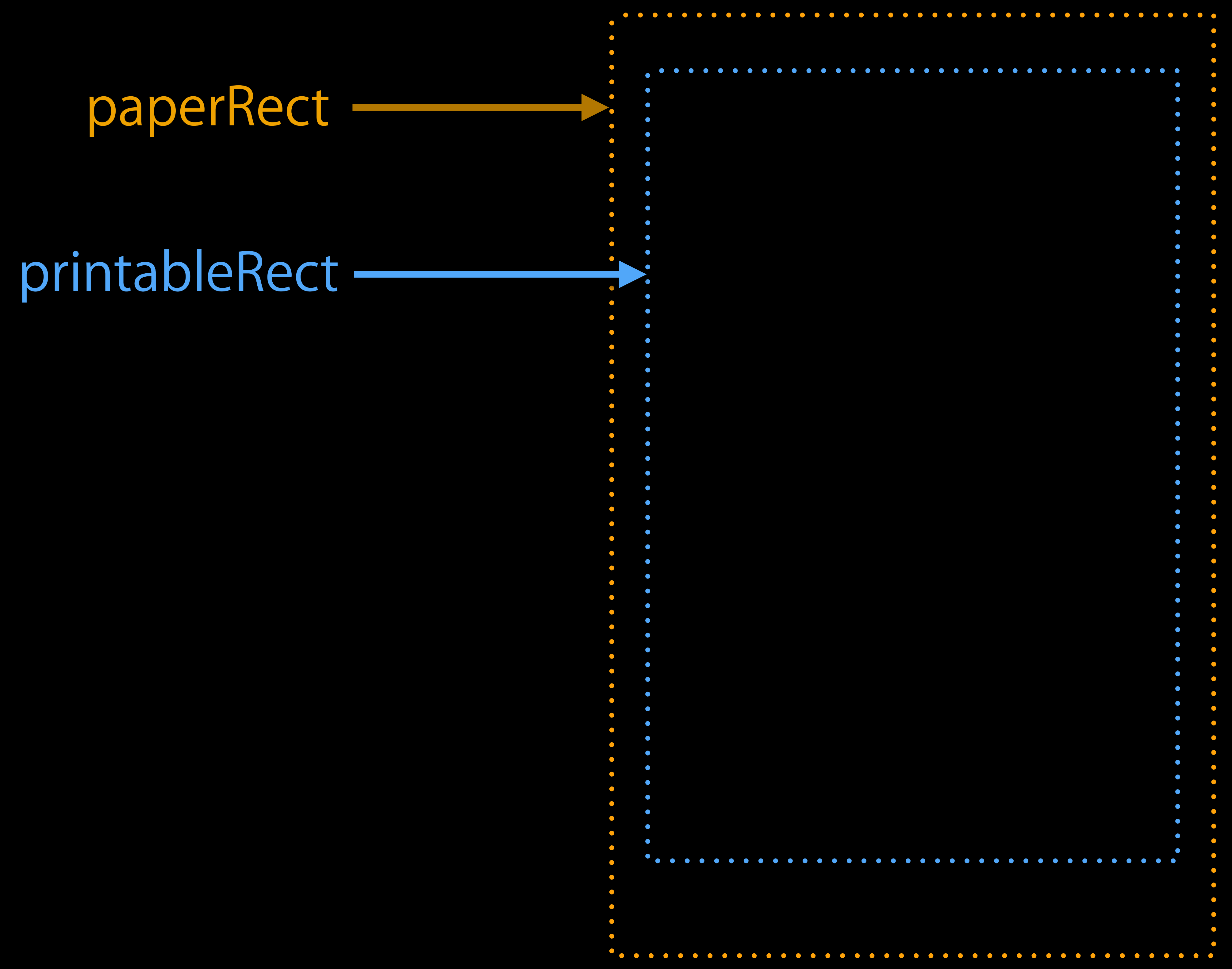
For HTML markup text use `UIMarkupTextFormatter`

Layout

NEW

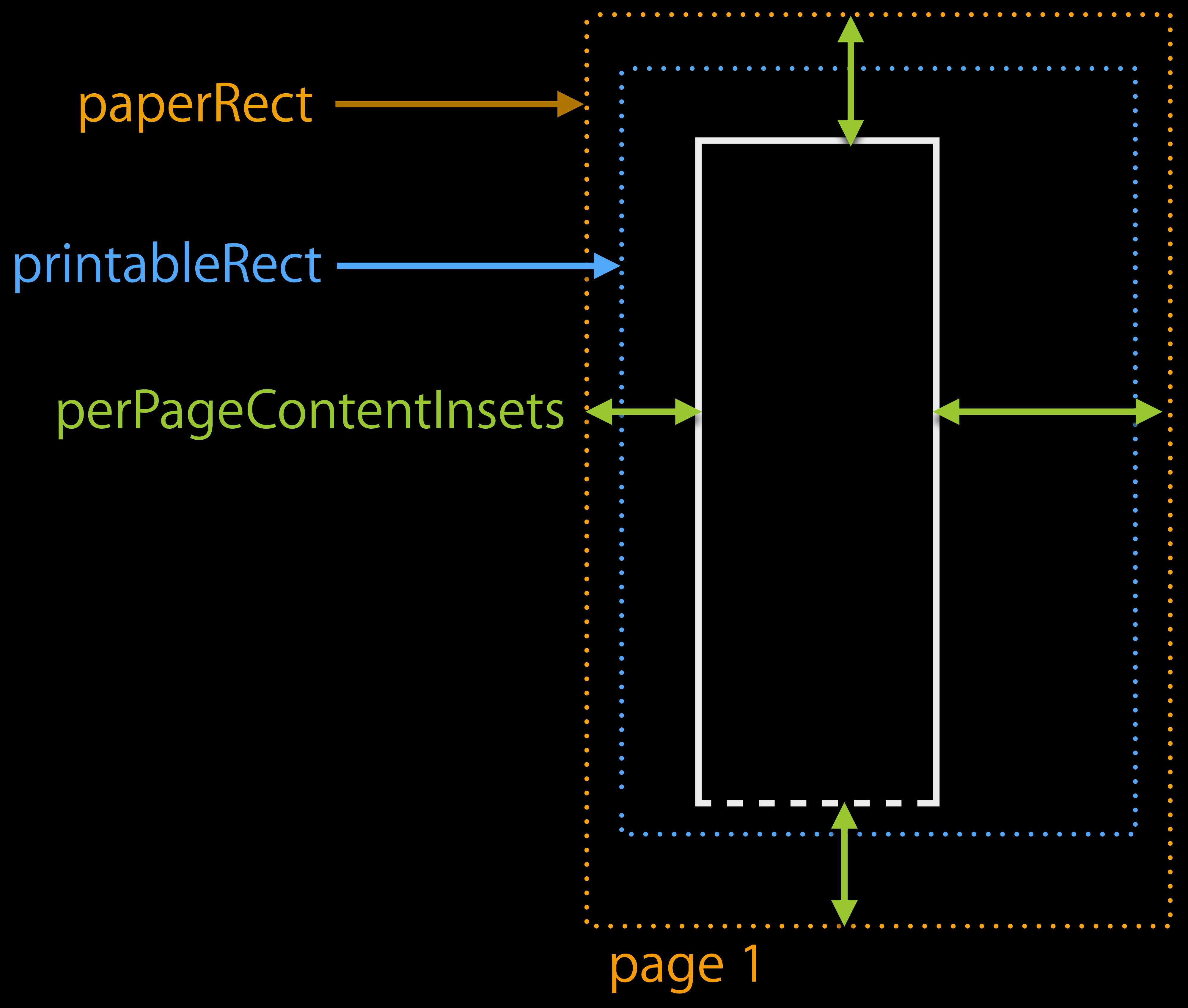
Layout

NEW



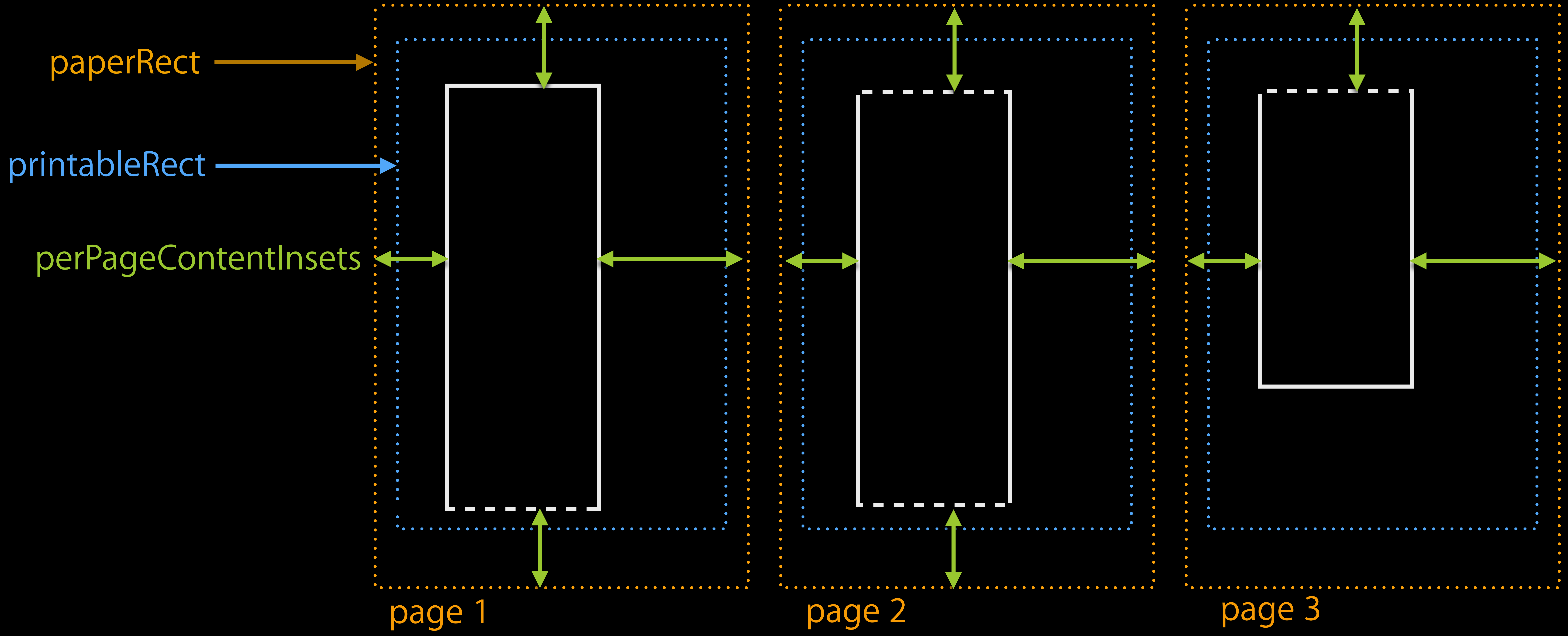
Layout

NEW



Layout

NEW


paperRect
printableRect
perPageContentInsets
page 1
page 2
page 3

Formatters

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

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIMarkupTextFormatter *formatter
                      = [[UIMarkupTextFormatter alloc] initWithText:text];
    formatter.perPageContentInsets =
      UIEdgeInsetsMake(POINTS_PER_INCH * 0.75f,  // 3/4 inch top margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch left margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch bottom margin
                       POINTS_PER_INCH * 0.75f); // 3/4 inch right margin
    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];
    formatter.perPageContentInsets =
      UIEdgeInsetsMake(POINTS_PER_INCH * 0.75f,  // 3/4 inch top margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch left margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch bottom margin
                       POINTS_PER_INCH * 0.75f); // 3/4 inch right margin
    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];
    formatter.perPageContentInsets =
      UIEdgeInsetsMake(POINTS_PER_INCH * 0.75f,  // 3/4 inch top margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch left margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch bottom margin
                       POINTS_PER_INCH * 0.75f); // 3/4 inch right margin
    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];
    formatter.perPageContentInsets =
      UIEdgeInsetsMake(POINTS_PER_INCH * 0.75f,  // 3/4 inch top margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch left margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch bottom margin
                       POINTS_PER_INCH * 0.75f); // 3/4 inch right margin
    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];
    formatter.perPageContentInsets =
      UIEdgeInsetsMake(POINTS_PER_INCH * 0.75f,  // 3/4 inch top margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch left margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch bottom margin
                       POINTS_PER_INCH * 0.75f); // 3/4 inch right margin
    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];
    formatter.perPageContentInsets =
      UIEdgeInsetsMake(POINTS_PER_INCH * 0.75f,  // 3/4 inch top margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch left margin
                       POINTS_PER_INCH * 0.75f,  // 3/4 inch bottom margin
                       POINTS_PER_INCH * 0.75f); // 3/4 inch right margin
    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)printMapView:(id) sender {

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIViewPrintFormatter *formatter
                      = [self.myMapView viewPrintFormatter];

    controller.printFormatter = formatter;

    UIPrintInfo *printInfo = [UIPrintInfo printInfo];
    printInfo.outputType = UIPrintInfoOutputGeneral;
    printInfo.jobName    = @“Map”;
    controller.printInfo = printInfo;

    [controller presentAnimated:YES completionHandler:NULL];

}
```

View Formatters

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

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIViewPrintFormatter *formatter
                      = [self.myMapView viewPrintFormatter];

    controller.printFormatter = formatter;

    UIPrintInfo *printInfo = [UIPrintInfo printInfo];
    printInfo.outputType = UIPrintInfoOutputGeneral;
    printInfo.jobName    = @“Map”;
    controller.printInfo = printInfo;

    [controller presentAnimated:YES completionHandler:NULL];

}
```

View Formatters

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

    UIPrintInteractionController *
           controller = [UIPrintInteractionController sharedPrintController];

    UIViewPrintFormatter *formatter
                      = [self.myMapView viewPrintFormatter];

    controller.printFormatter = formatter;

    UIPrintInfo *printInfo = [UIPrintInfo printInfo];
    printInfo.outputType = UIPrintInfoOutputGeneral;
    printInfo.jobName    = @“Map”;
    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
```

Printing items that already exist

Basic

Printing simple content

Text, HTML, etc.
Use one of the provided Formatter classes

Fully custom drawn pages

Printing items that already exist

Printing simple content

Advanced

Fully custom drawn pages

Render the pages from the content in your app
Use Formatter and Renderer classes

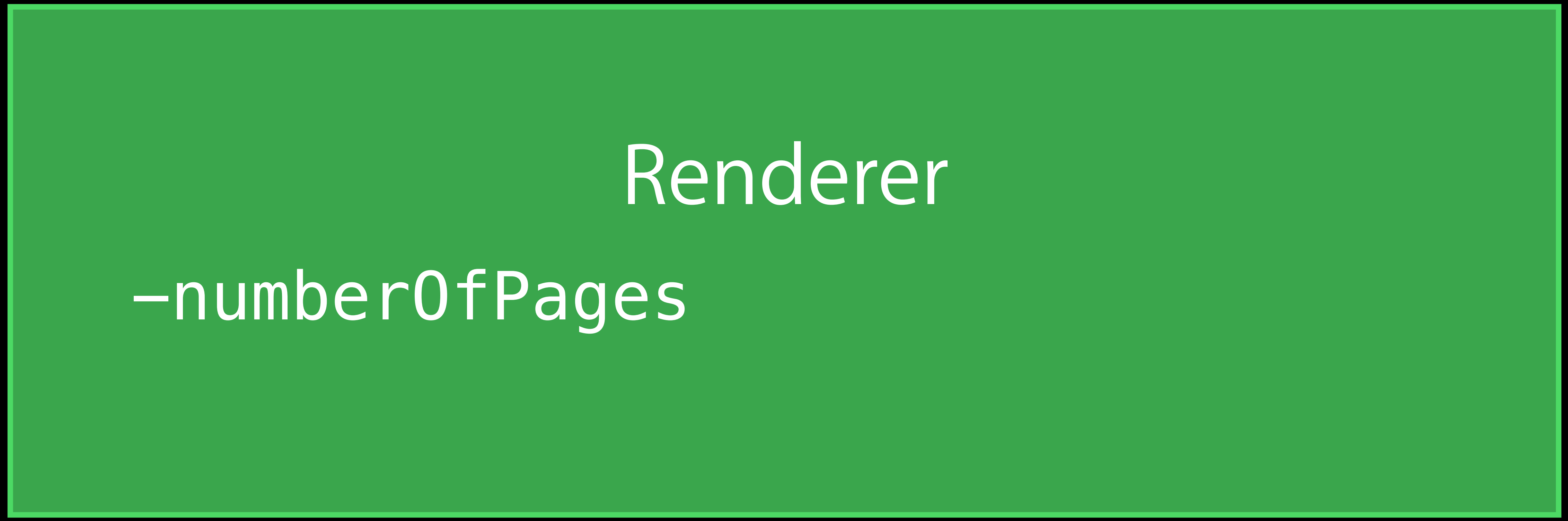
Renderers

What Is a Renderer?

What Is a Renderer?

Renderer

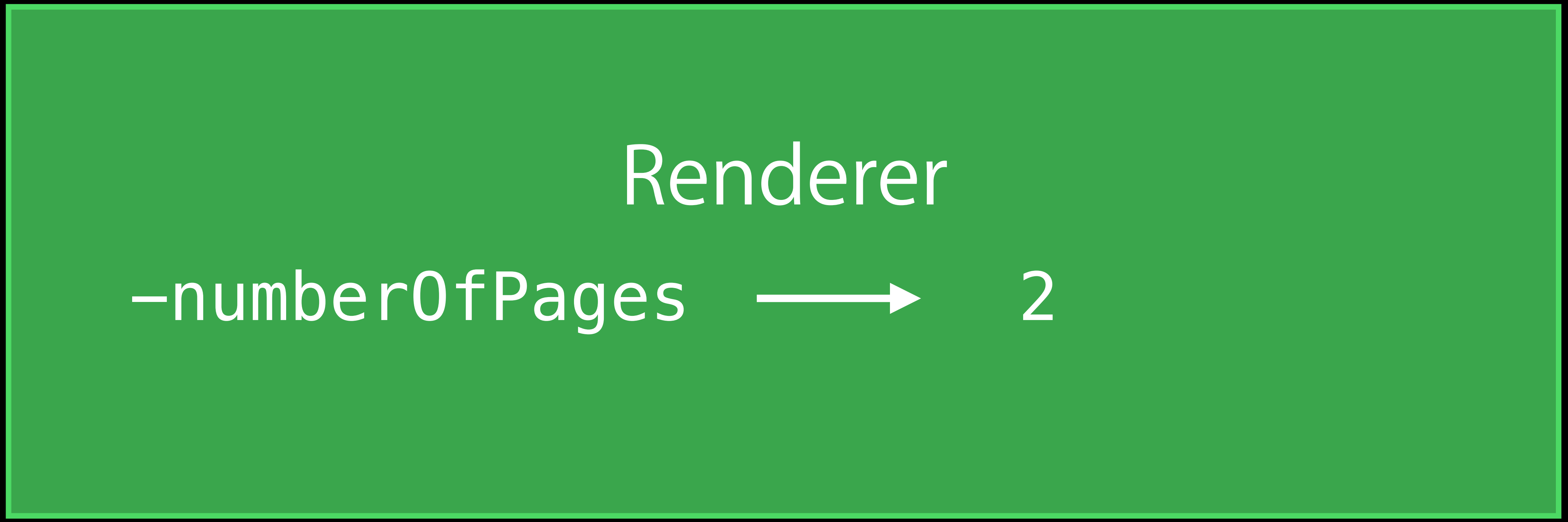
What Is a Renderer?



Renderer

`-numberOfPages`

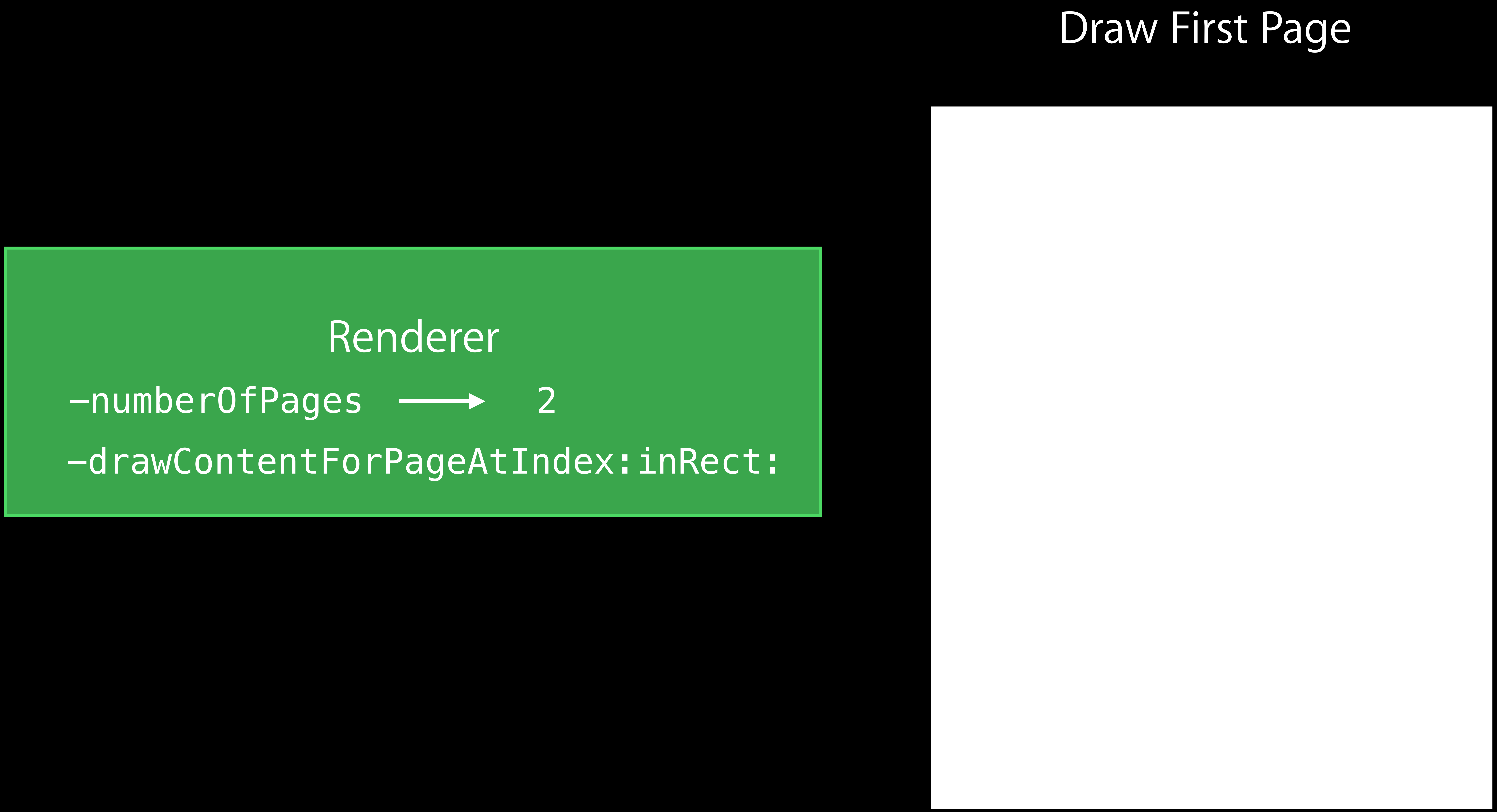
What Is a Renderer?



Renderer

-numberOfPages → 2

What Is a Renderer?



Draw First Page

Renderer

`-numberOfPages` → 2

`-drawContentForPageAtIndex:inRect:`

What Is a Renderer?

Renderer

-numberOfPages → 2

-drawContentForPageAtIndex:inRect:

Draw First Page

Impromptu in f-moll

Franz Schubert
(1797-1828)

D935 (Op. 142 No 4., 1827)

Allegro scherzando
p
7
14
f
p
21
f
p
f
p
29
f
p
cresc.
8va
f

Public Domain

What Is a Renderer?

Renderer

`-numberOfPages` → 2

`-drawContentForPageAtIndex:inRect:`

Impromptu in f-moll

Franz Schubert
(1797-1828)

D935 (Op. 142 No 4., 1827)

Allegro scherzando
p
7
14
f
p
21
f
p
f
p
29
f
p
cresc.
8va
f

Public Domain

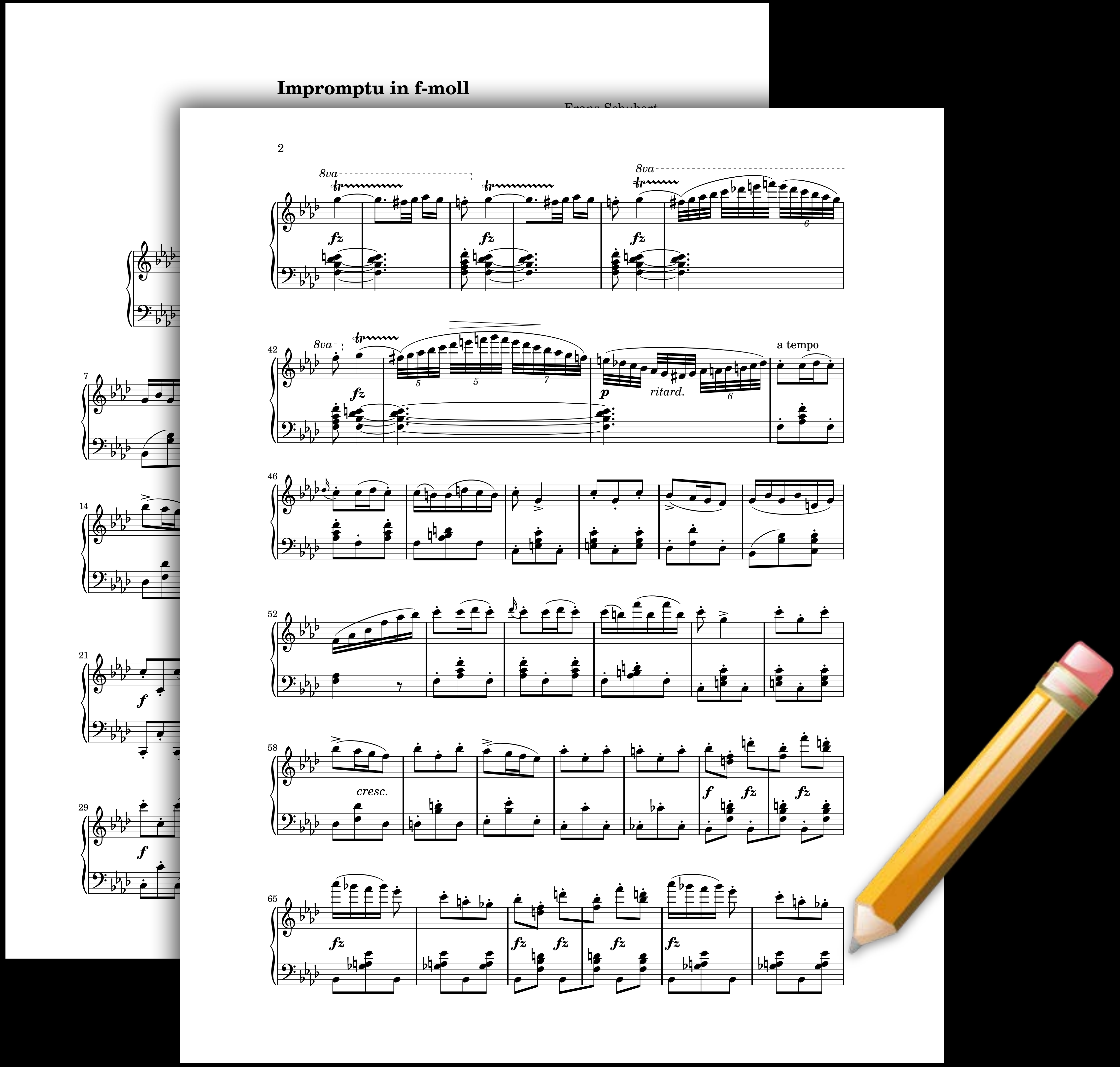
What Is a Renderer?

Renderer

`-numberOfPages` → 2

`-drawContentForPageAtIndex:inRect:`

Draw Second Page



Basic Rendering

Subclass `UIPrintPageRenderer`

Override

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

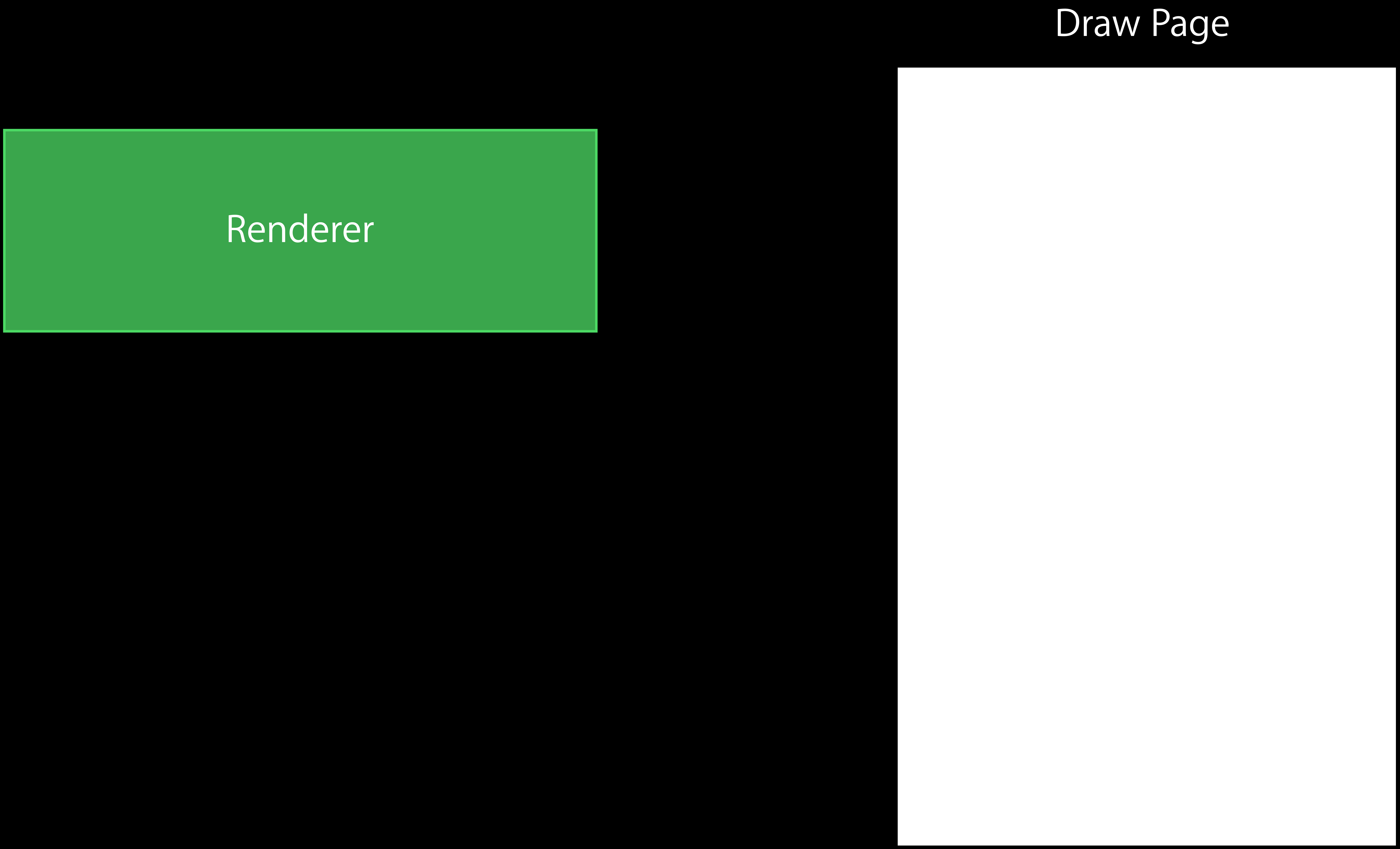
Set `UIPrintInteractionController.printPageRenderer`

or add to the array of activity items when creating the `UIActivityViewController` object

Can add a formatter object by calling `addPrintFormatter:startingAtPageIndex:`

Renderer with Formatter

Renderer with Formatter



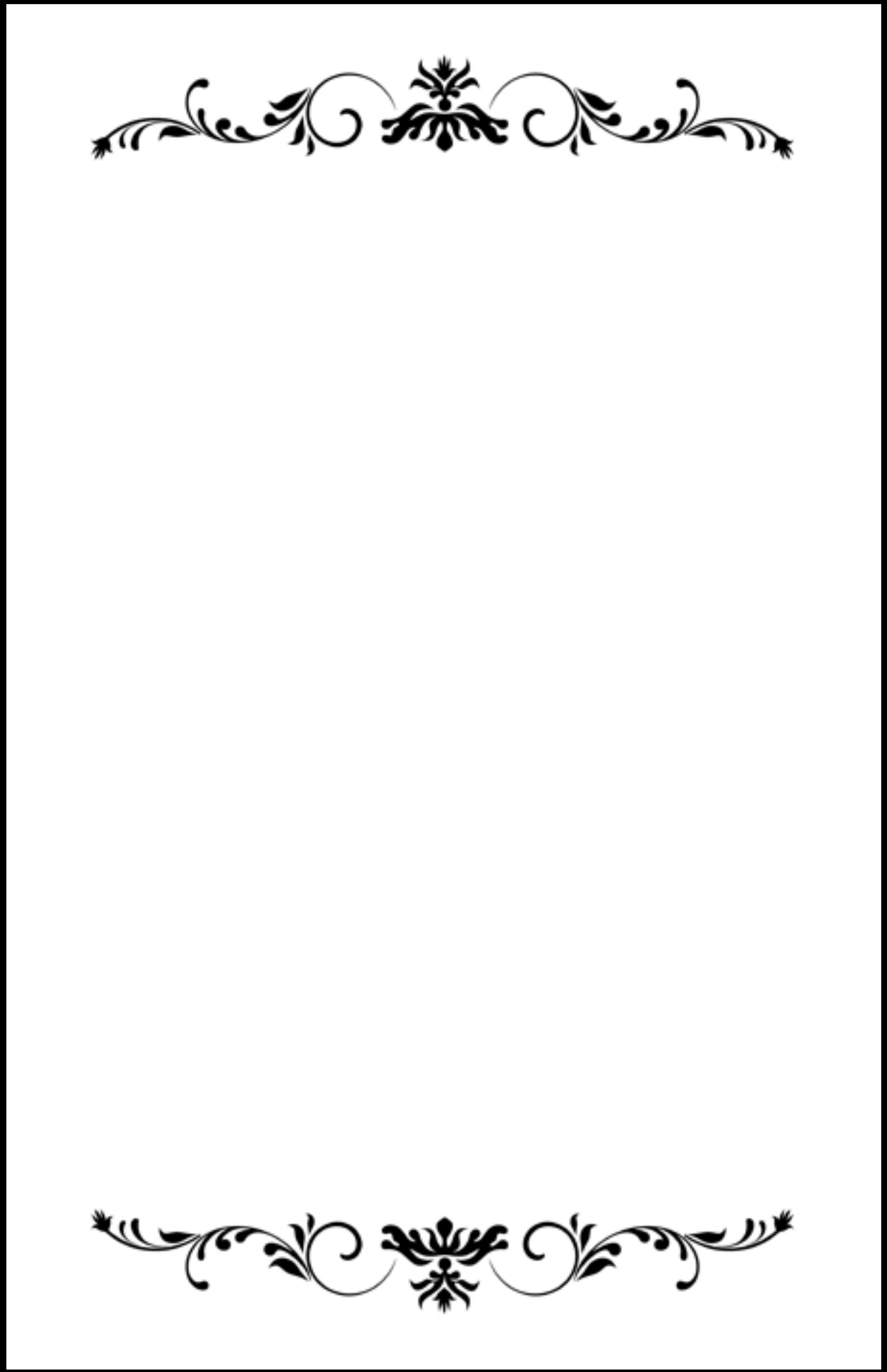
Renderer

Draw Page

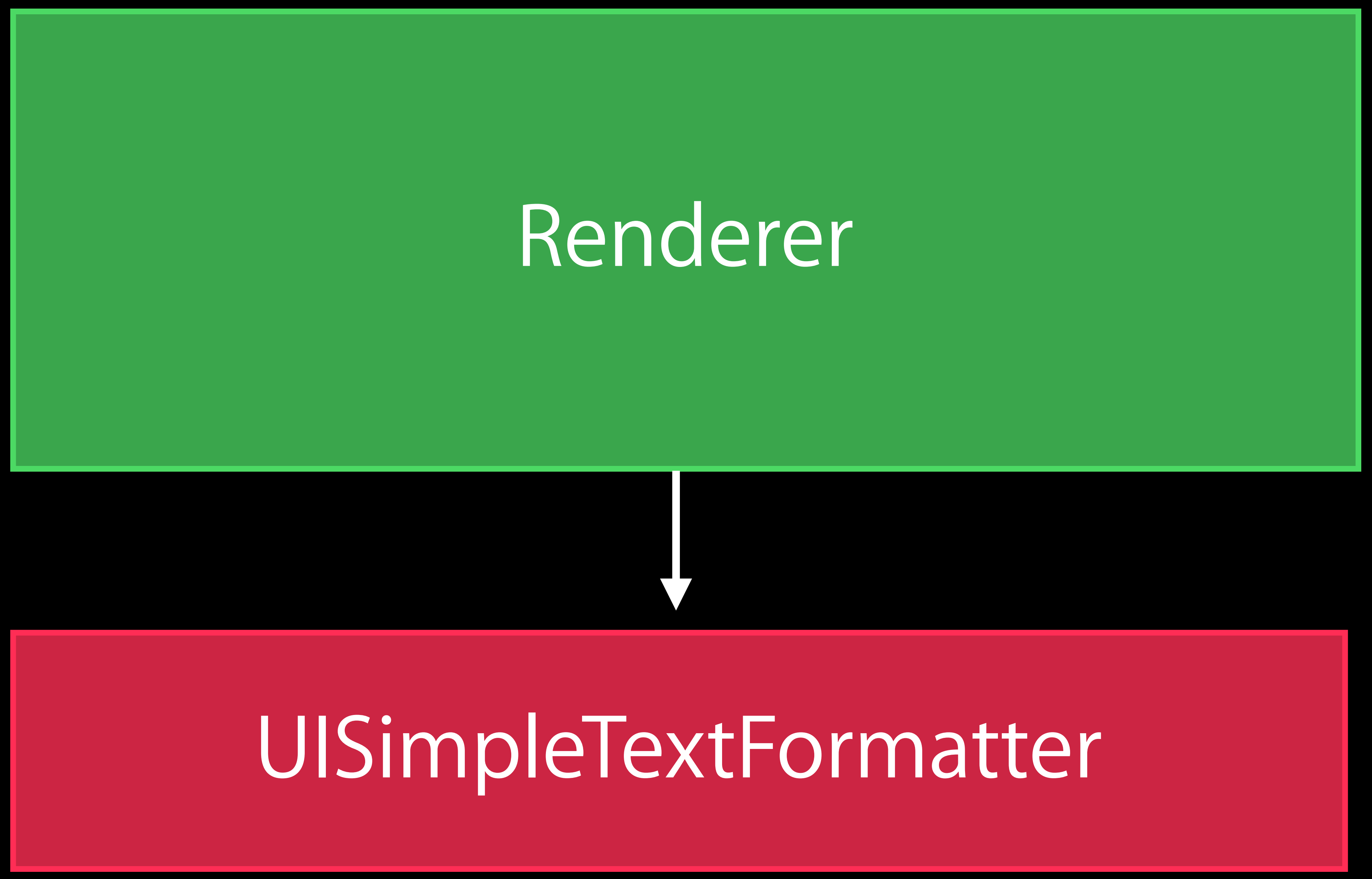
Renderer with Formatter

Renderer

Draw Page



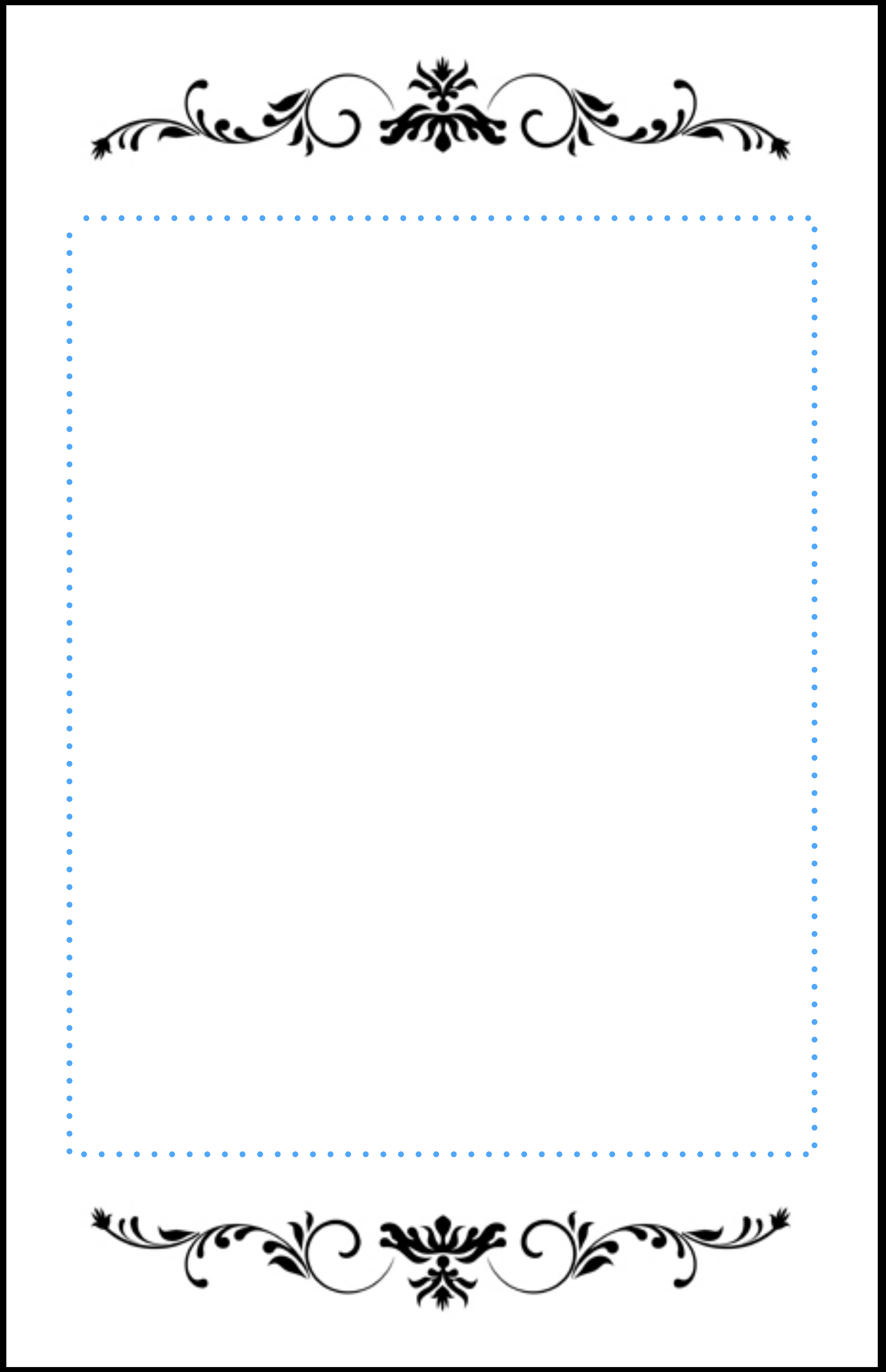
Renderer with Formatter



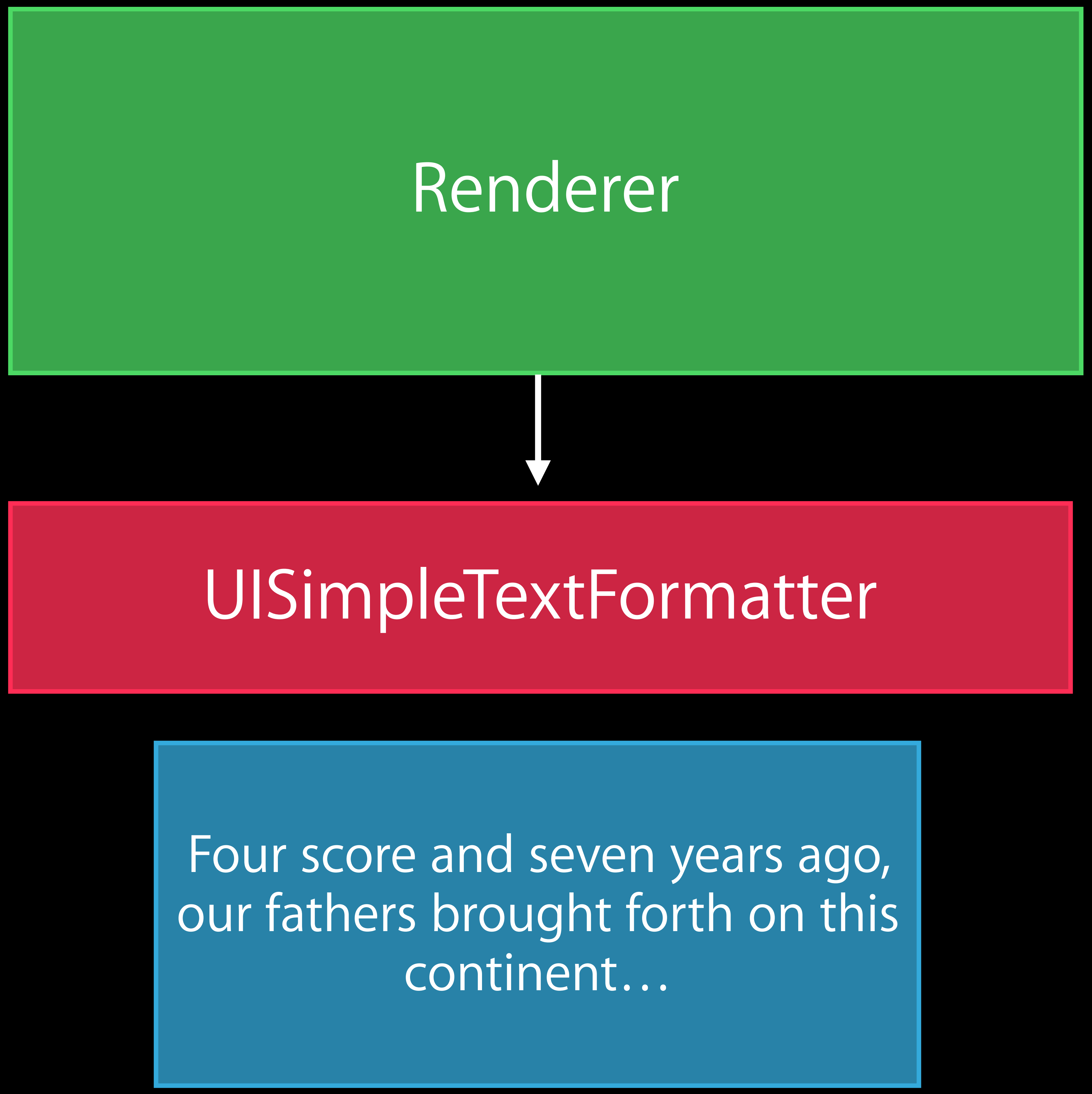
Renderer

UISimpleTextFormatter

Draw Page



Renderer with Formatter

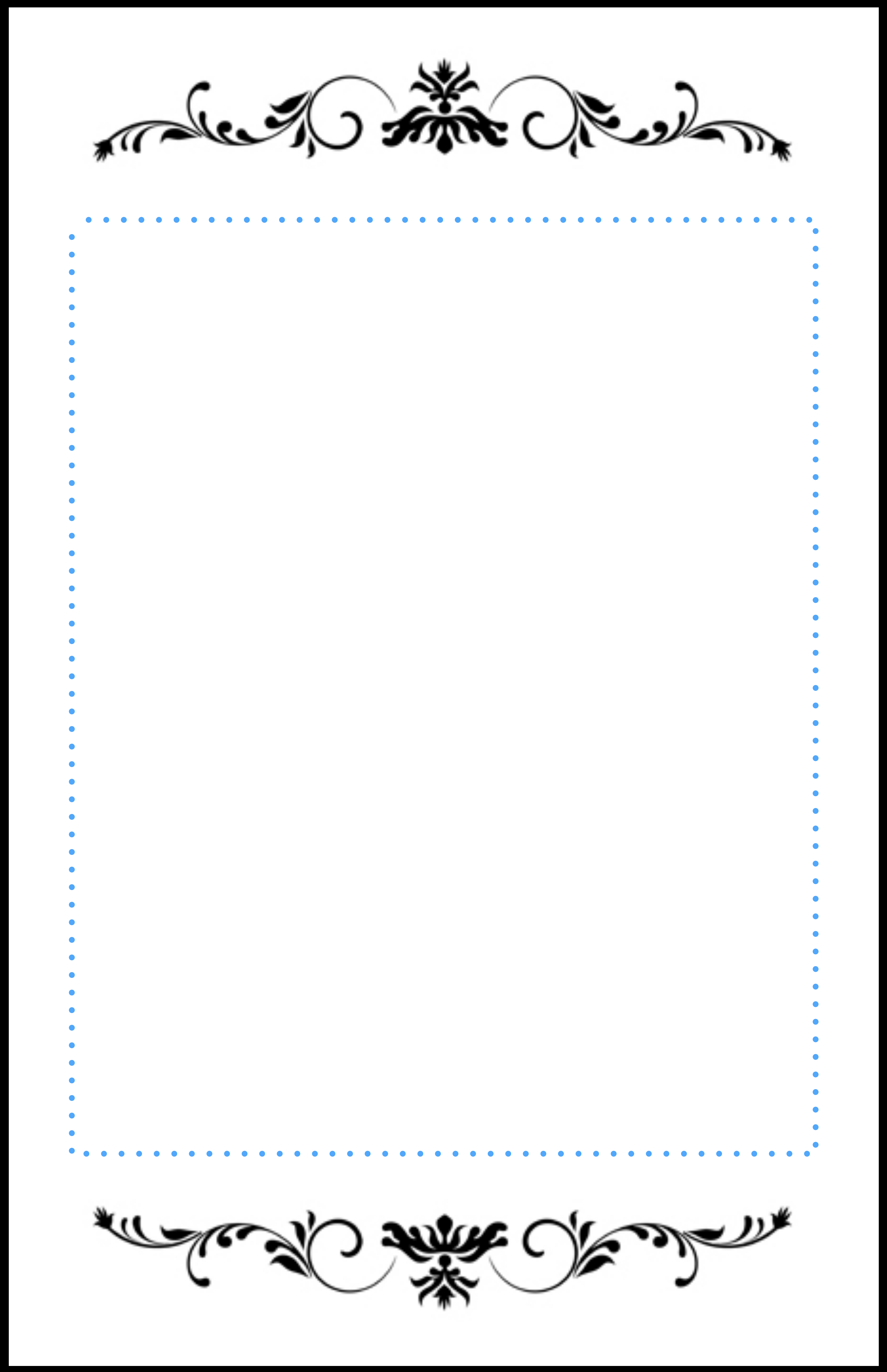


Renderer

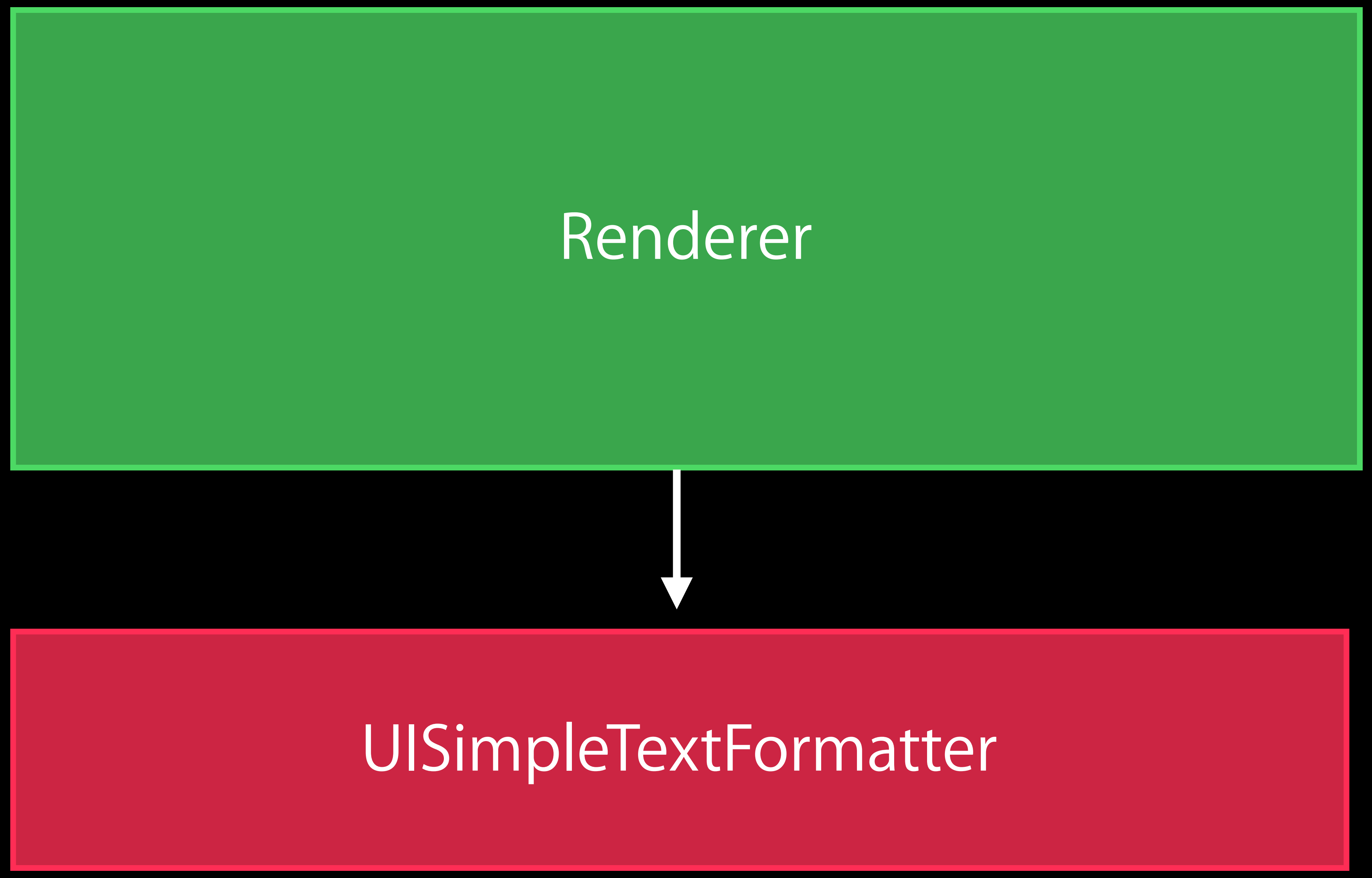
UISimpleTextFormatter

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

Draw Page

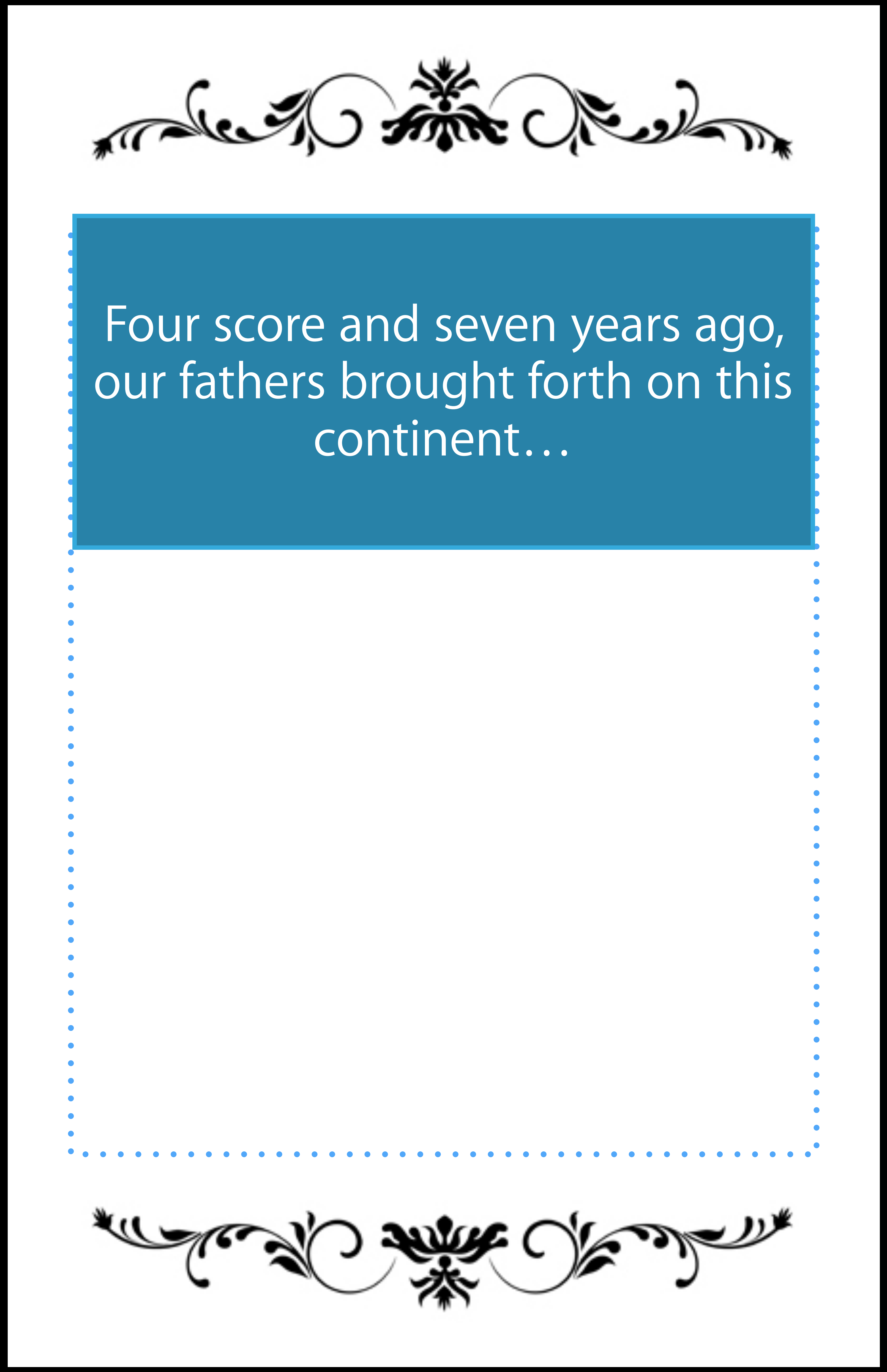


Renderer with Formatter



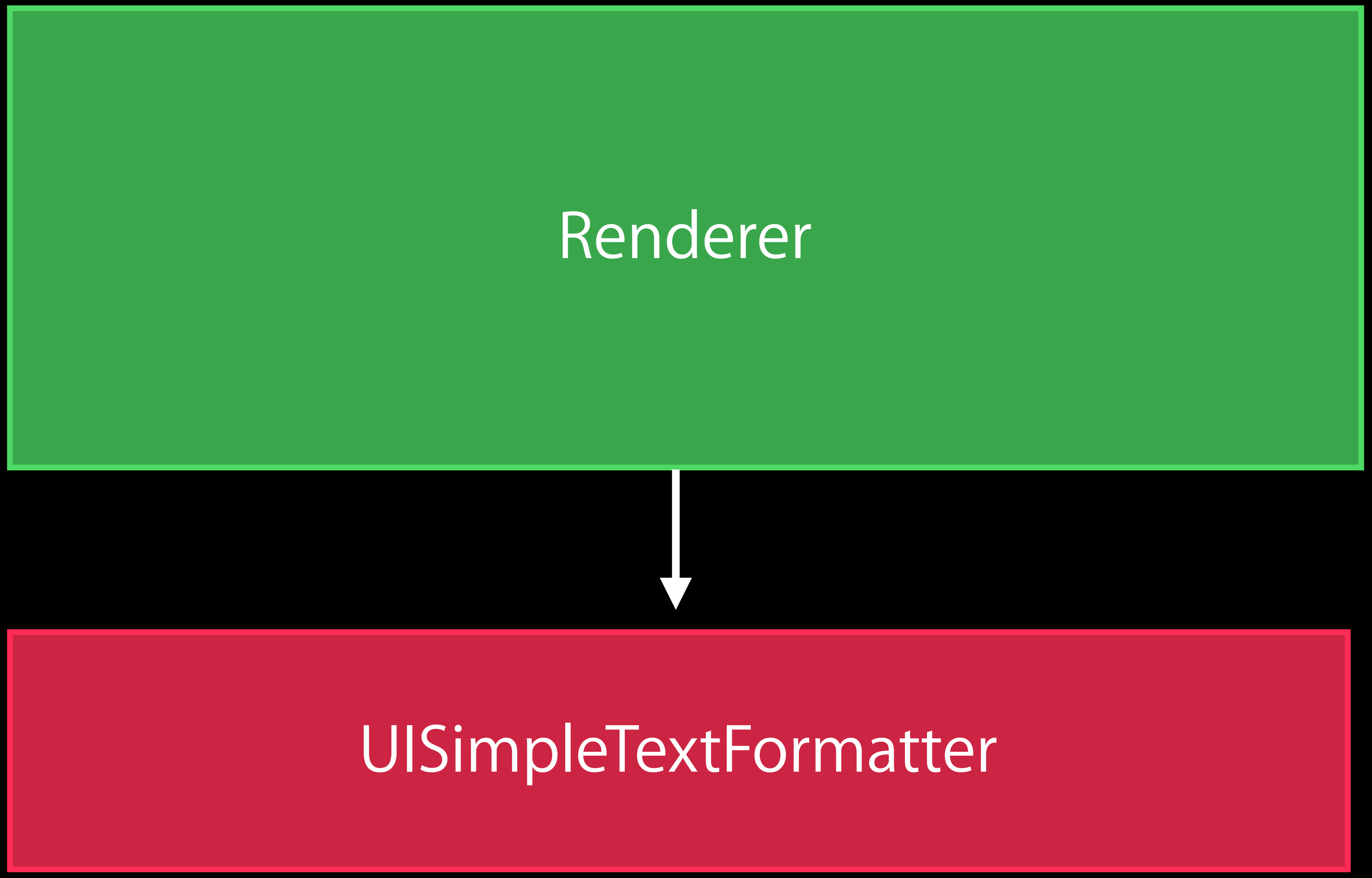
Renderer

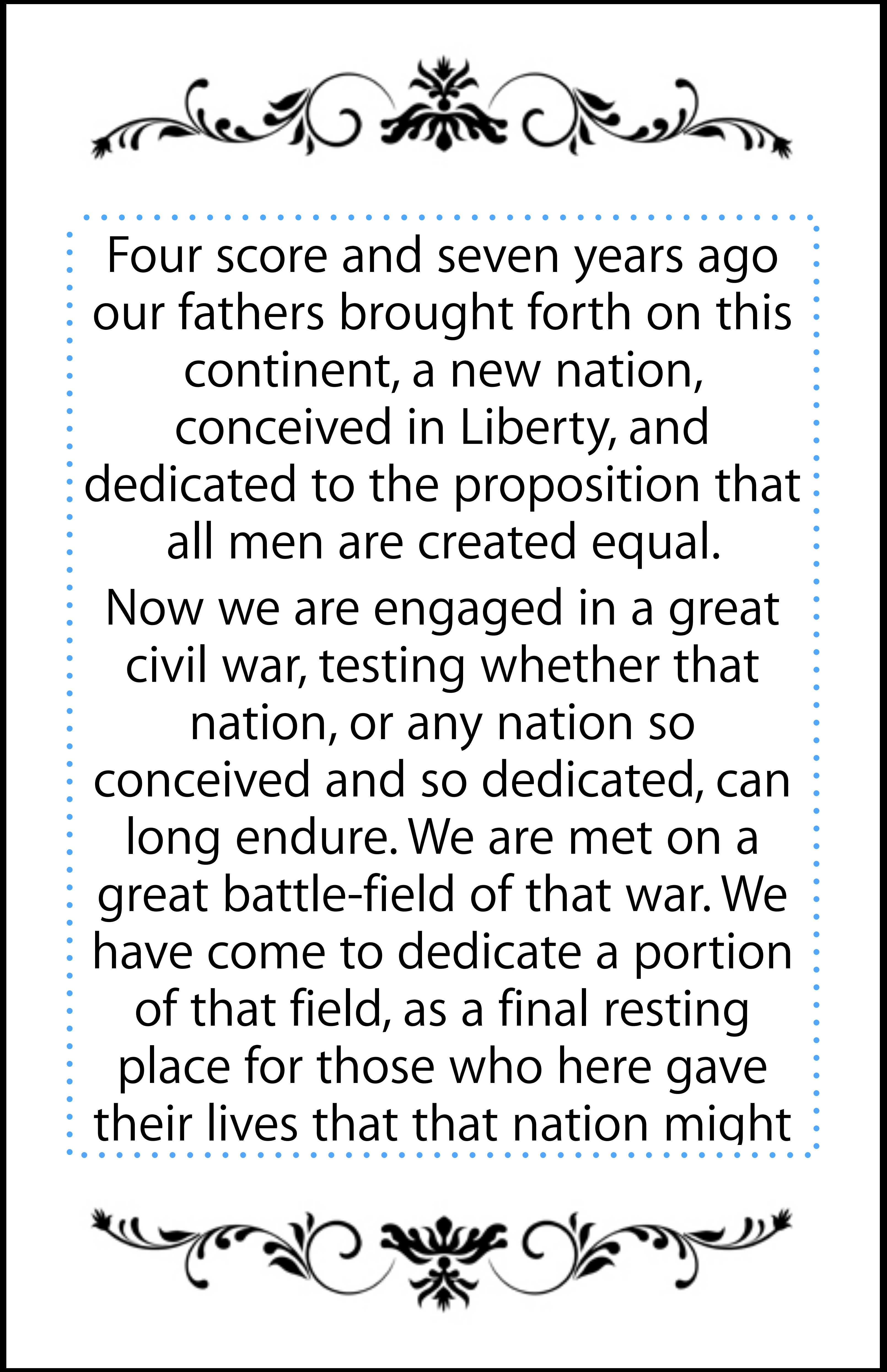
UISimpleTextFormatter



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

Renderer with Formatter



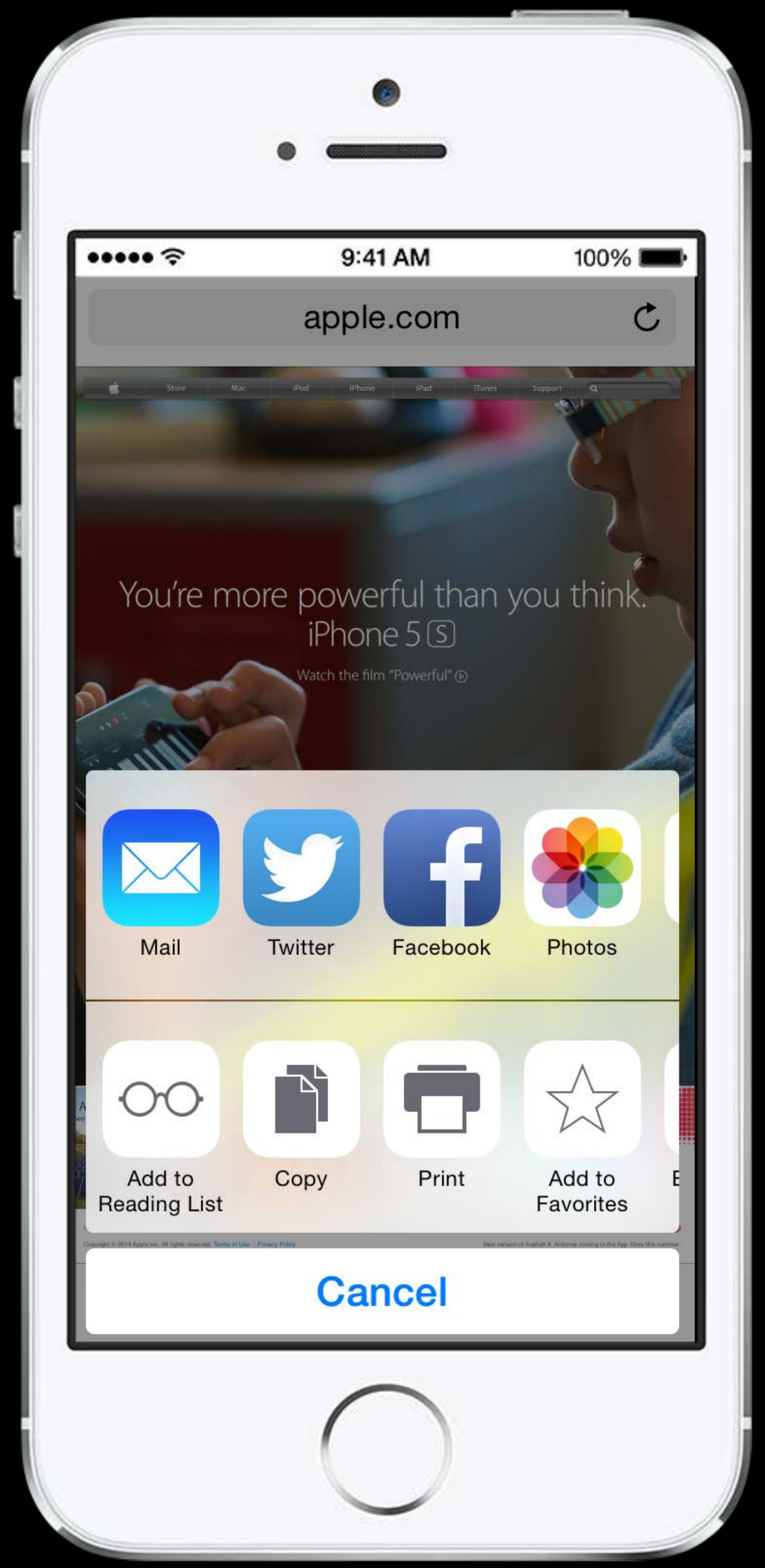


Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and dedicated to the proposition that all men are created equal.

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

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];

... 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];

... 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];
```

```
... 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];

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

Printing Using a “Print” Button

Create and setup the `UIPrintInteractionController`

Standard presentation

```
    -presentAnimated:completionHandler:
```

Popover presentation

```
    -presentFromRect:inView:animated:completionHandler:
    -presentFromBarButtonItem:animated:completionHandler:
```

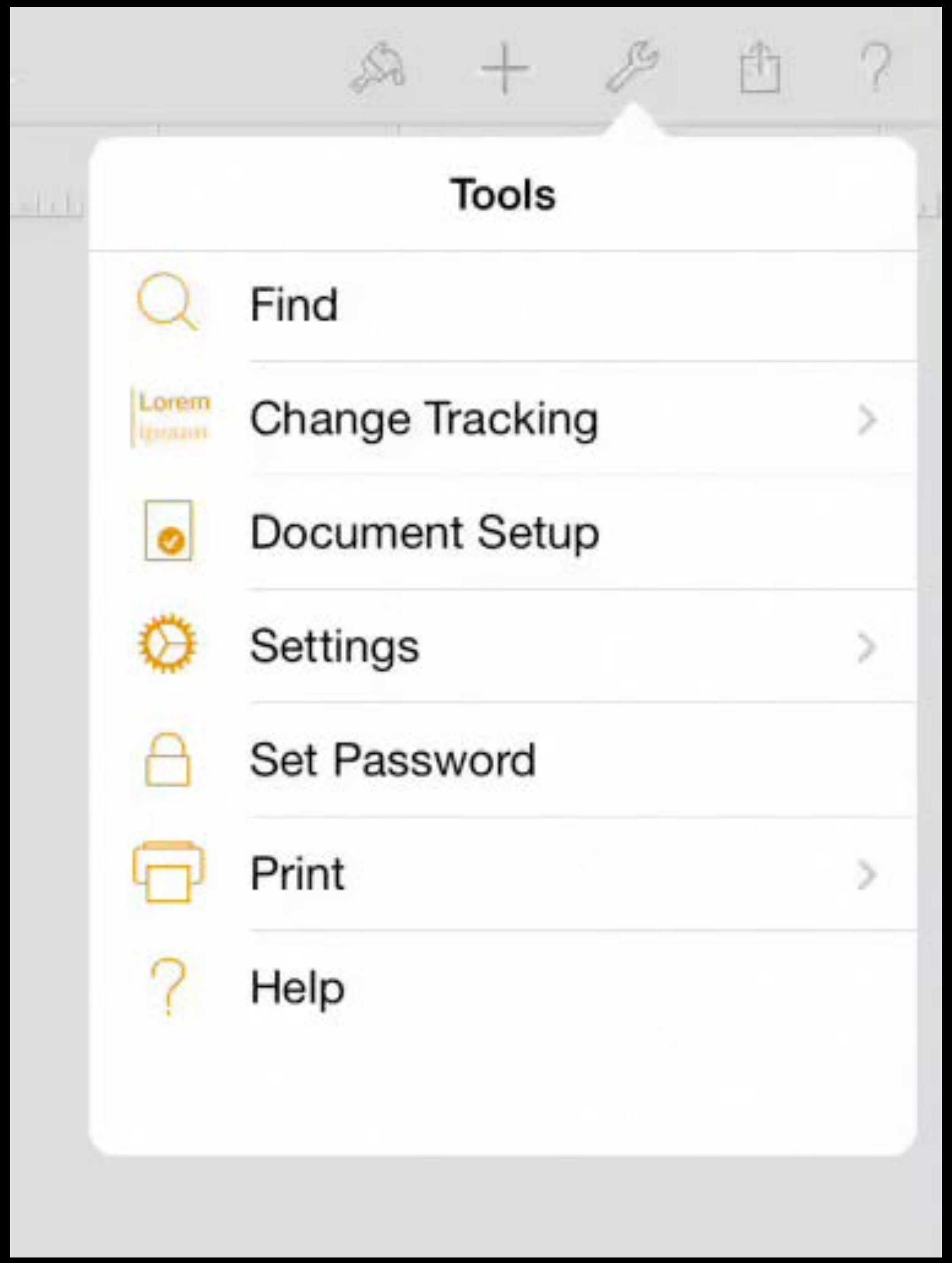
Printing as a Menu Item

Set your class as the delegate for the shared `UIPrintInteractionController`

Implement `printInteractionControllerParentViewController:`

When the user taps “Print” call `presentAnimated:CompletionHandler:`

- `UINavigationController`—push
- `UIViewController`—modal



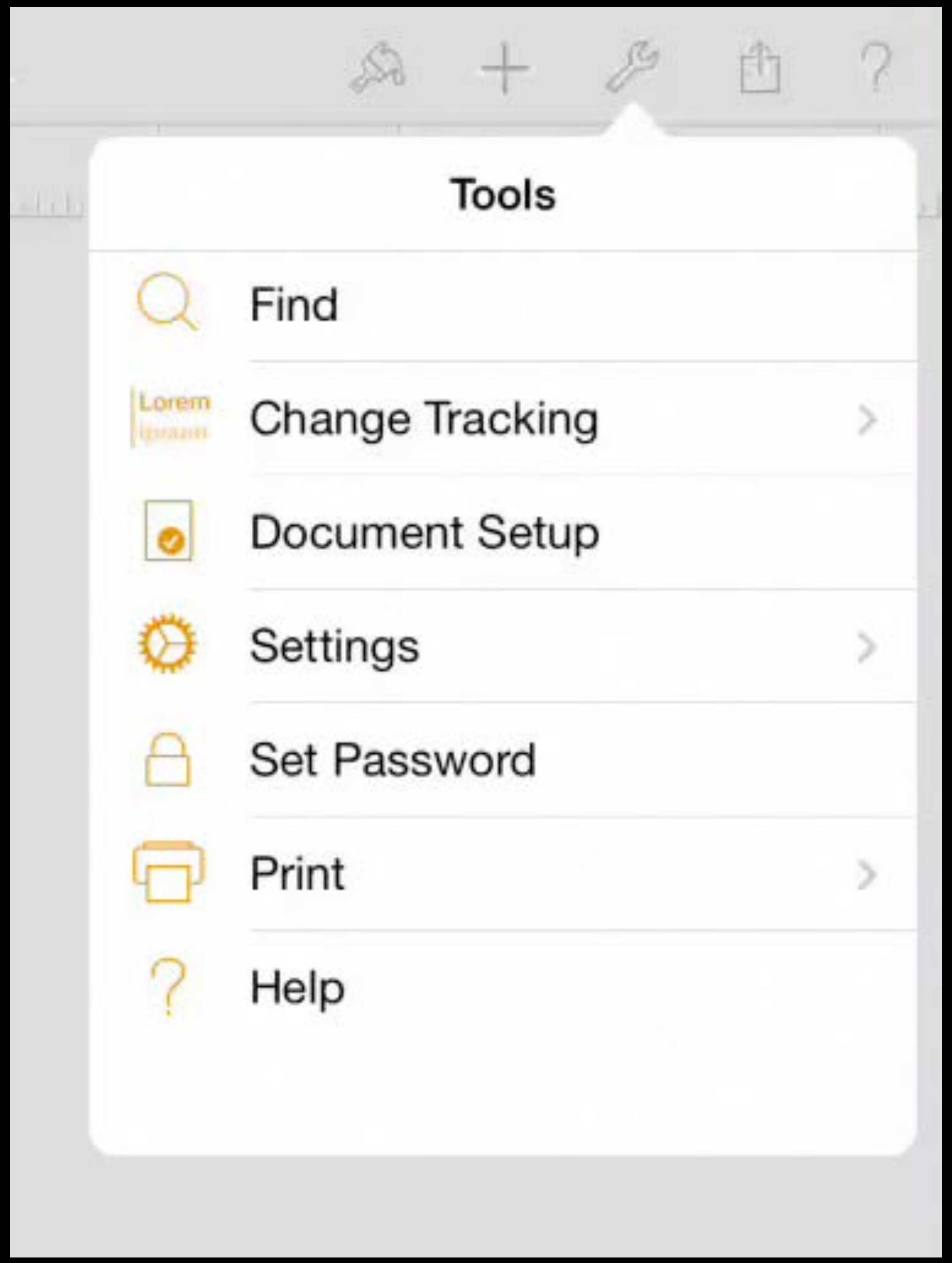
Printing as a Menu Item

Set your class as the delegate for the shared `UIPrintInteractionController`

Implement `printInteractionControllerParentViewController:`

When the user taps “Print” call `presentAnimated:CompletionHandler:`

- `UINavigationController`—push
- `UIViewController`—modal



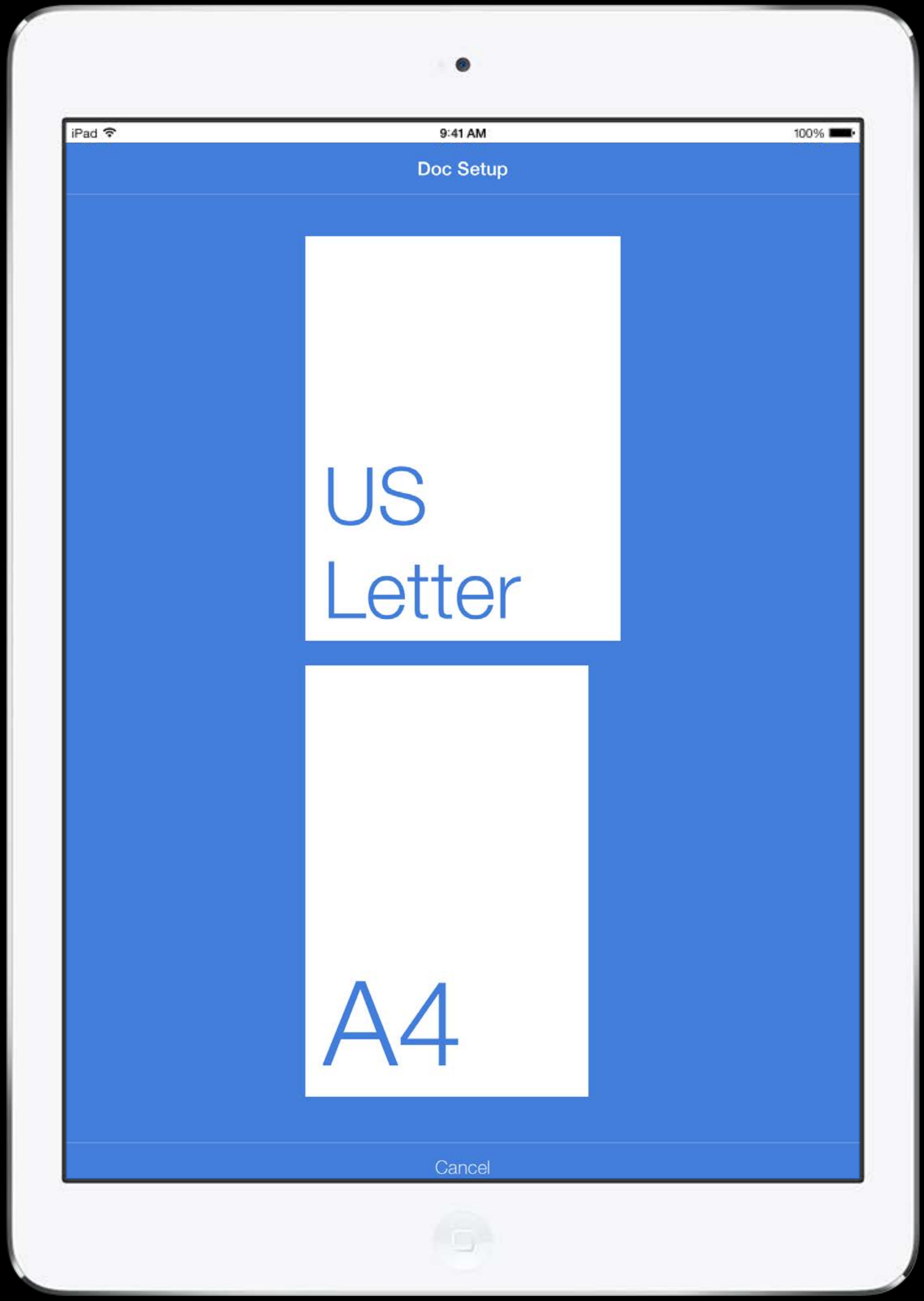
Paper Size

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 pageSize = CGSizeMake(POINTS_PER_INCH * 8.5f, POINTS_PER_INCH * 11.0f);

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

Paper Size

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

    CGSize pageSize = CGSizeMake(POINTS_PER_INCH * 8.5f, POINTS_PER_INCH * 11.0f);

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

Paper Size

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

    CGSize pageSize = CGSizeMake(POINTS_PER_INCH * 8.5f, POINTS_PER_INCH * 11.0f);

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

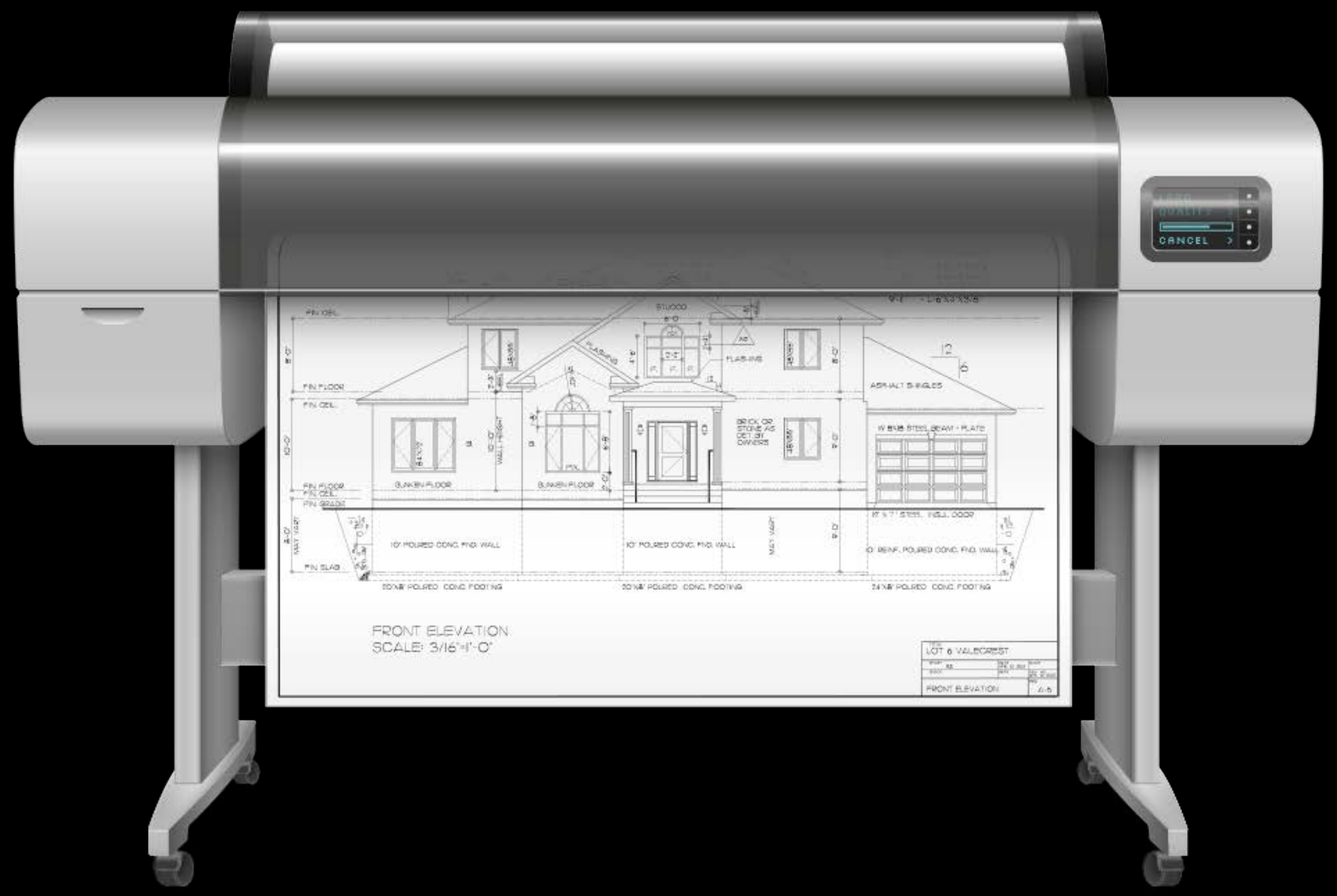
Roll Paper

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



Printing Without Showing UI

Submitting Jobs with No UI

NEW

Submitting Jobs with No UI

NEW

✓ Opens up opportunity for new types of printing applications

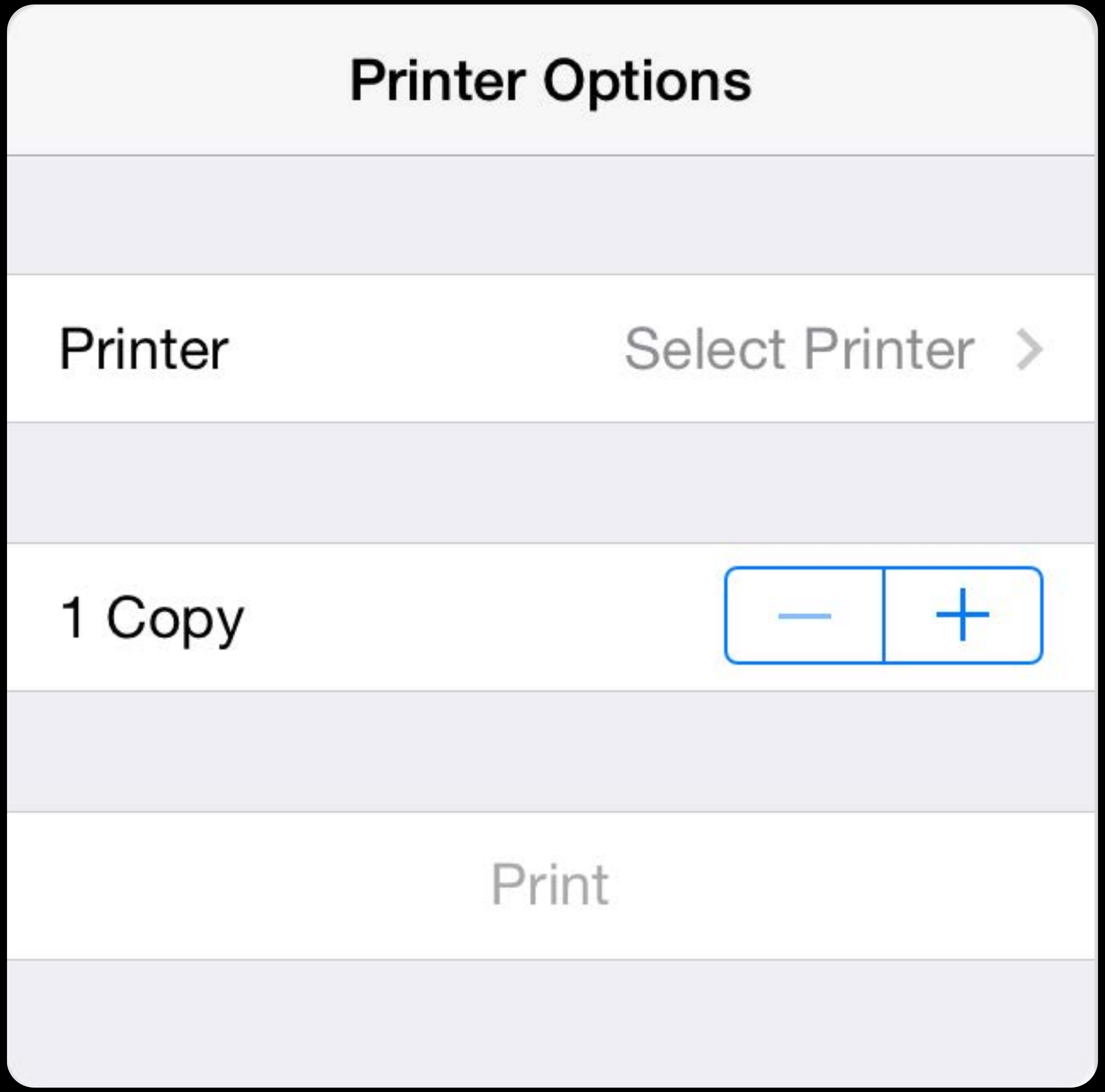
Submitting Jobs with No UI

NEW

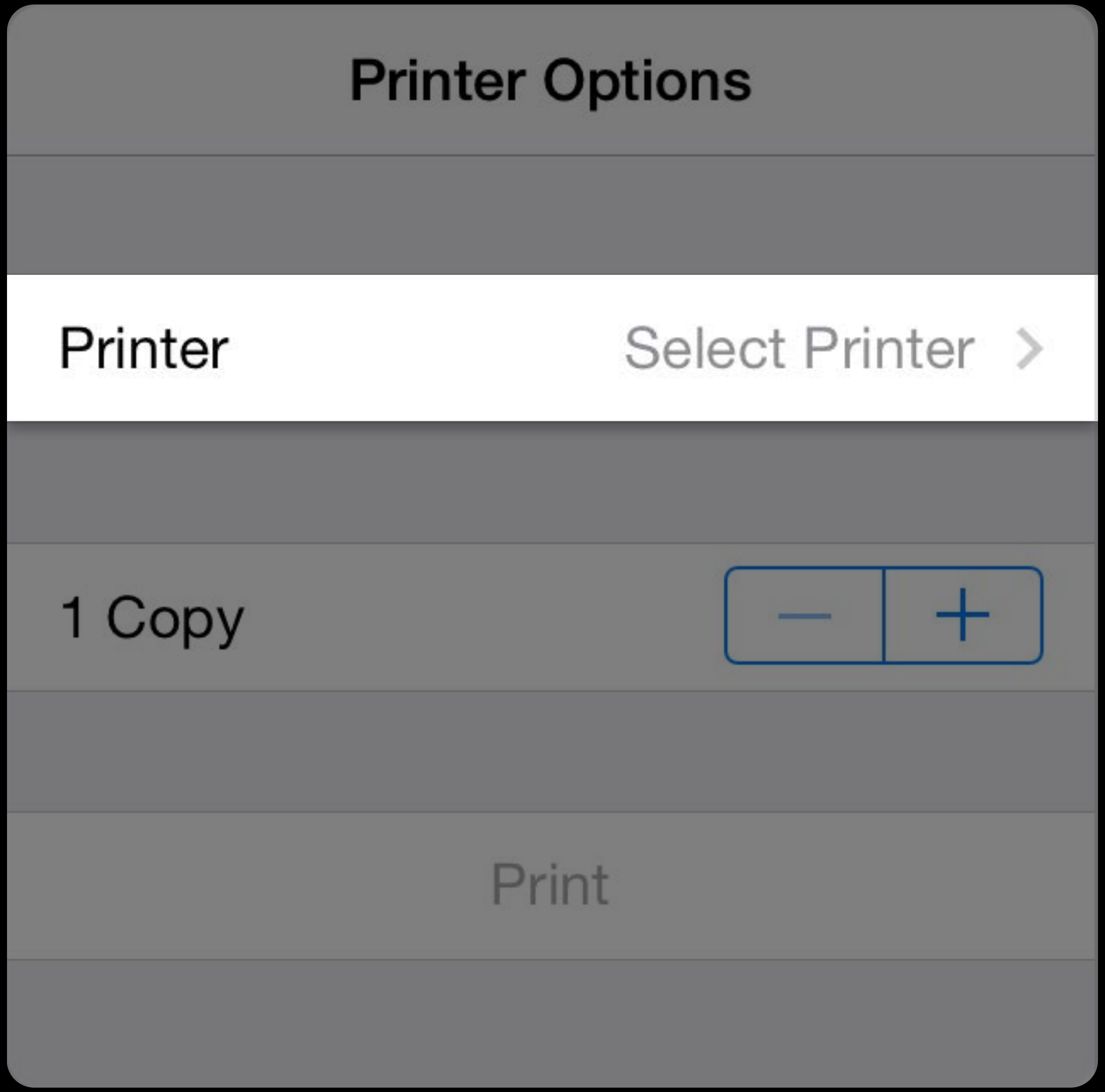
✓ Opens up opportunity for new types of printing applications

✗ Not for apps to provide custom print panel

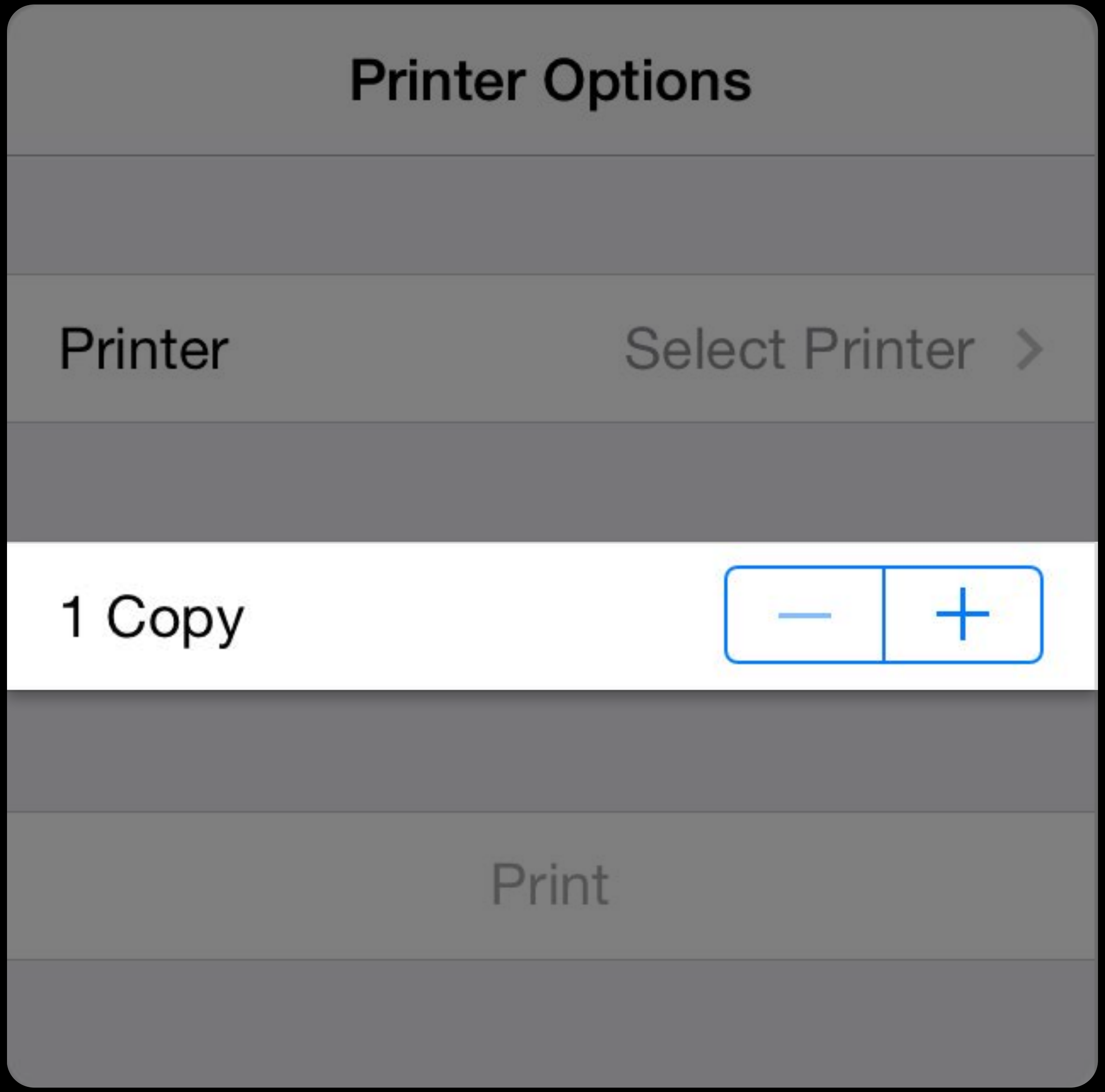
Steps of Printing



Steps of Printing



Steps of Printing

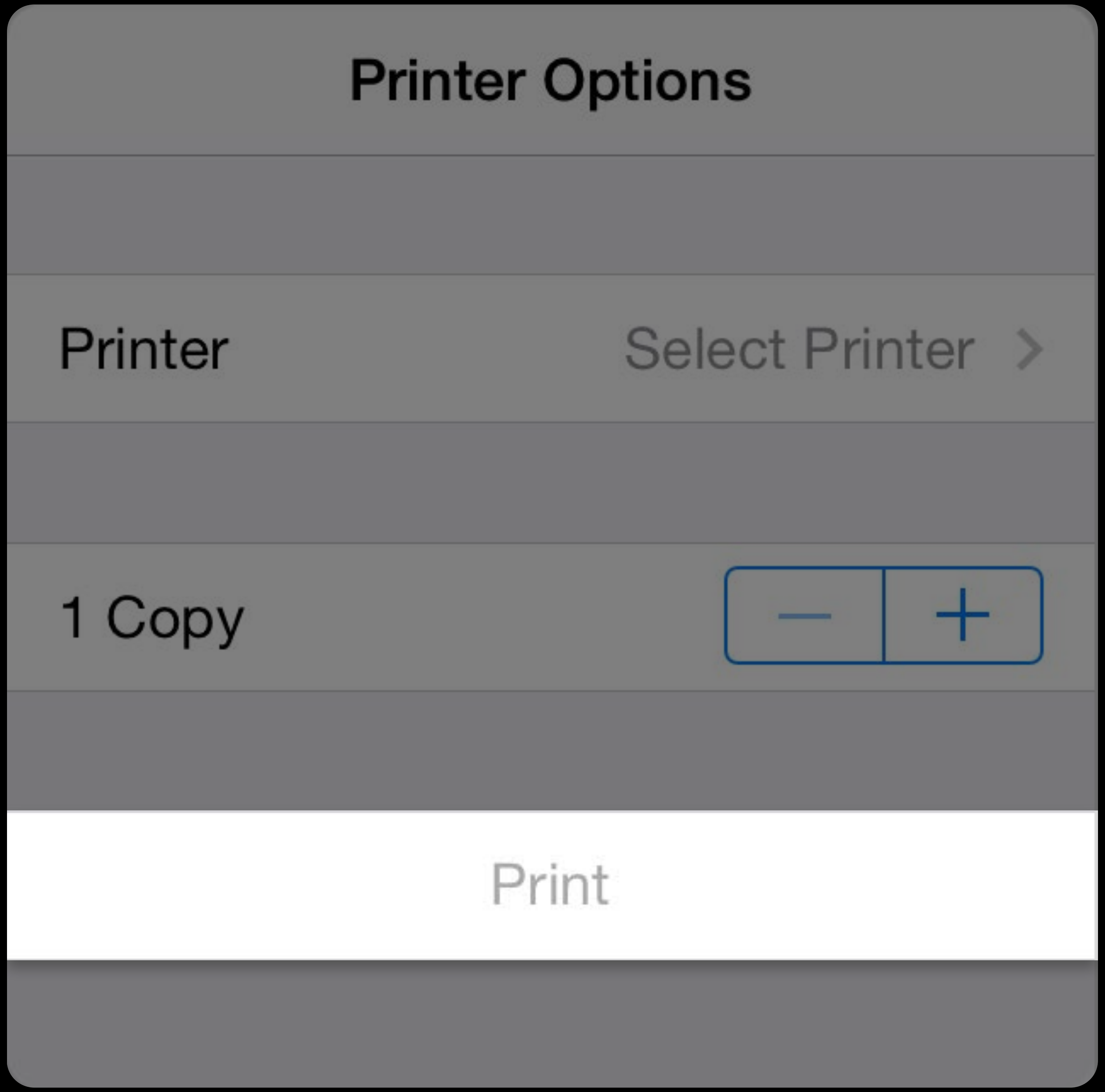


Printer Options

Printer Select Printer >

1 Copy

Steps of Printing

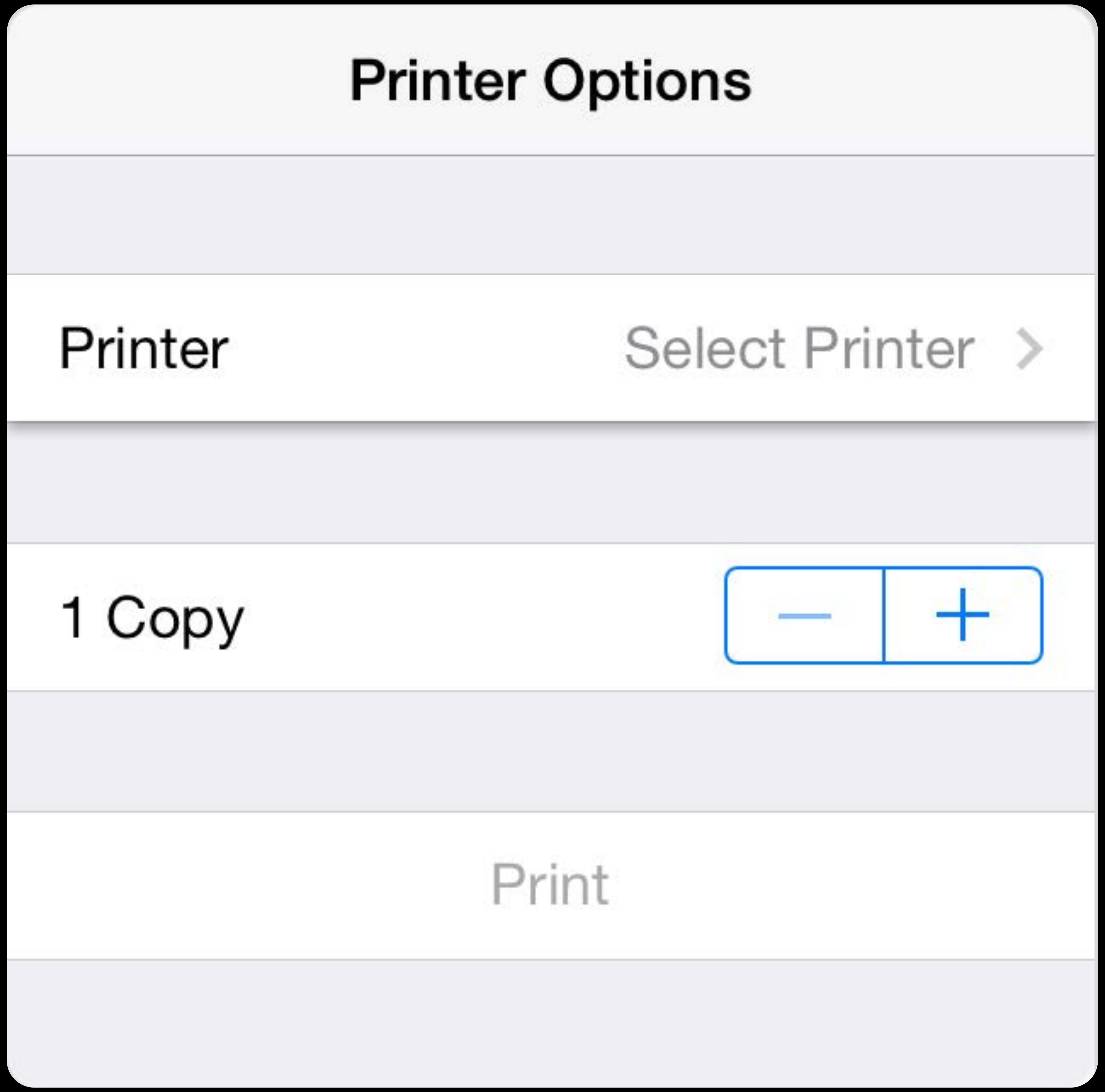


Printer Options

Printer Select Printer >

1 Copy

Steps of Printing



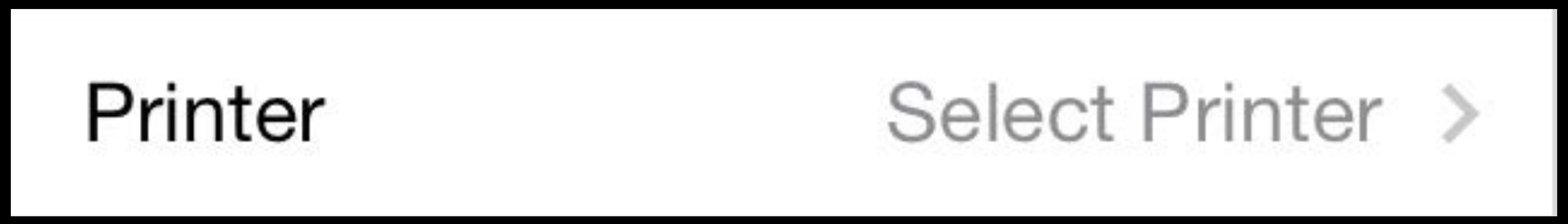
Printer Options

Printer: Select Printer >

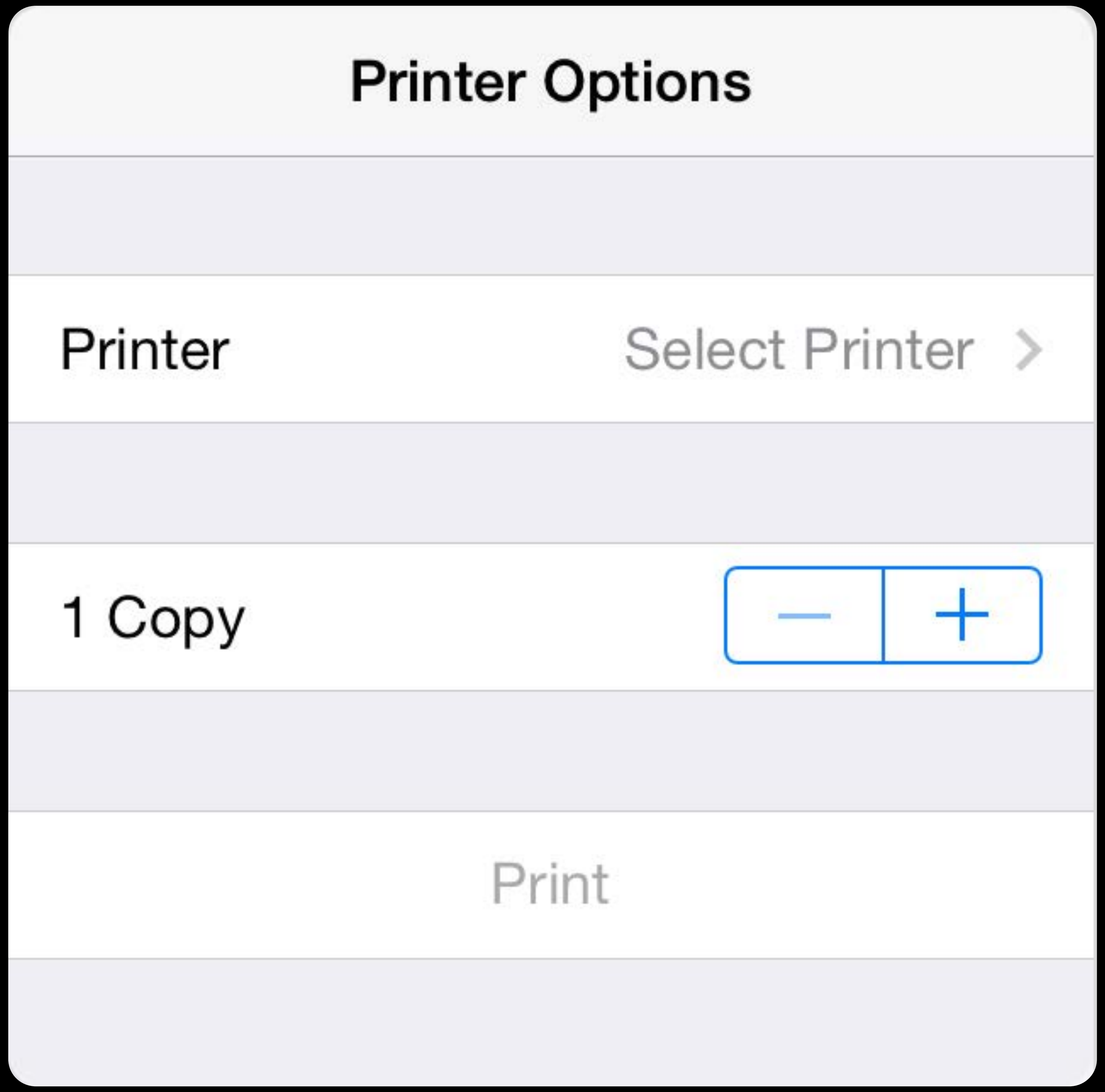
1 Copy − +

Print

Steps of Printing



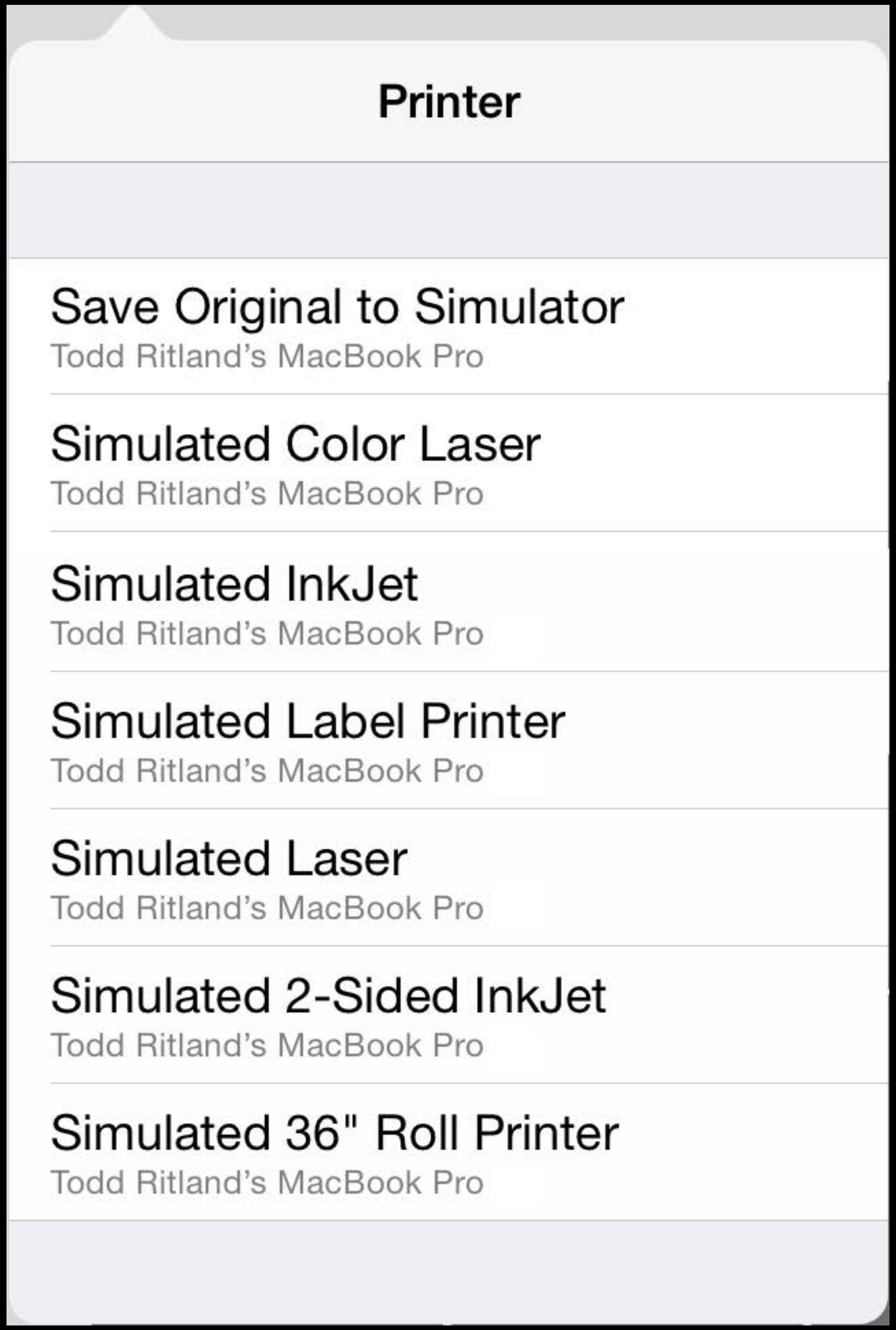
Printer Select Printer >



Printer Options

Printer Select Printer >

1 Copy



Printer

Save Original to Simulator
Todd Ritland's MacBook Pro

Simulated Color Laser
Todd Ritland's MacBook Pro

Simulated InkJet
Todd Ritland's MacBook Pro

Simulated Label Printer
Todd Ritland's MacBook Pro

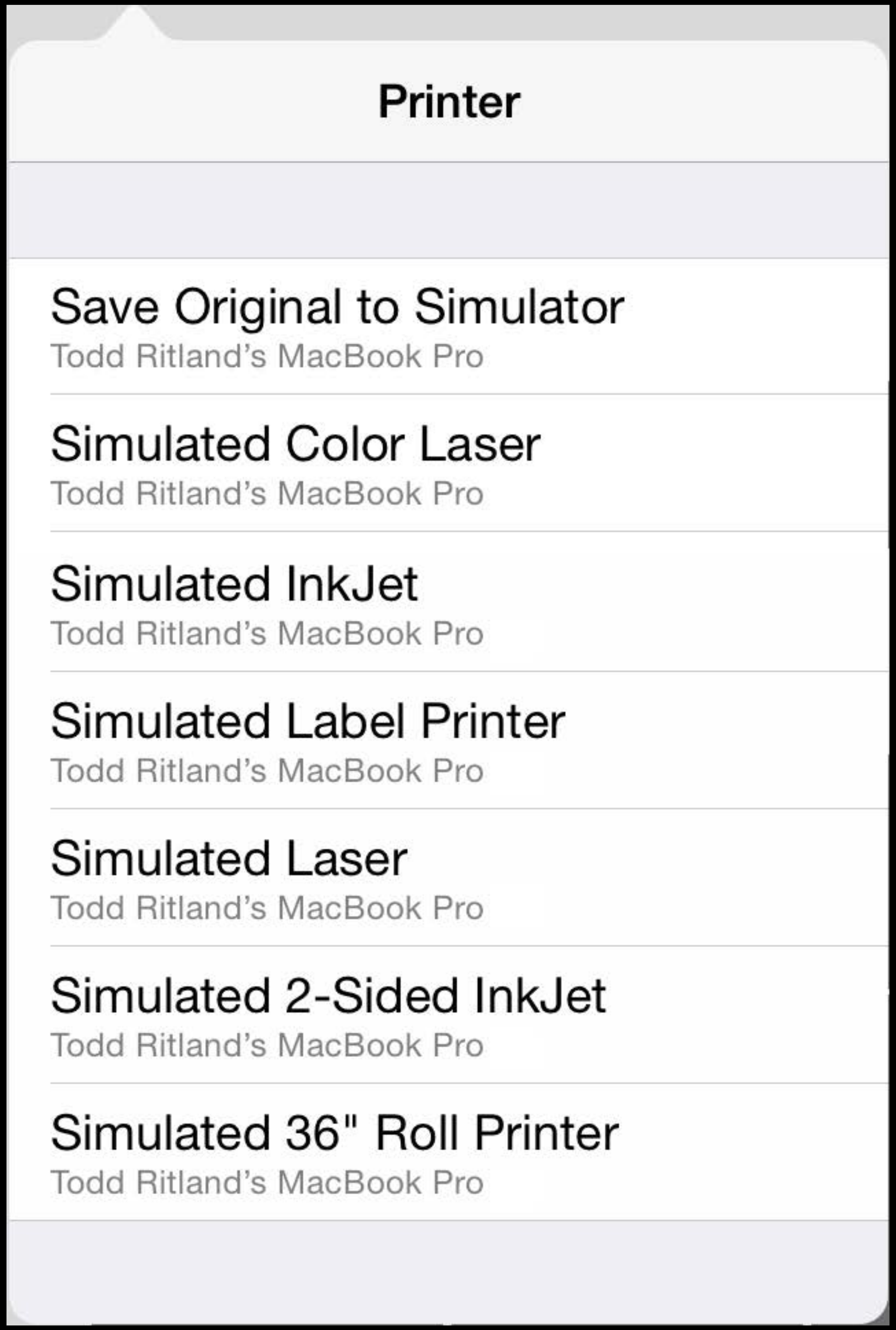
Simulated Laser
Todd Ritland's MacBook Pro

Simulated 2-Sided InkJet
Todd Ritland's MacBook Pro

Simulated 36" Roll Printer
Todd Ritland's MacBook Pro

Printer Picker

NEW



Printer

Save Original to Simulator
Todd Ritland's MacBook Pro

Simulated Color Laser
Todd Ritland's MacBook Pro

Simulated InkJet
Todd Ritland's MacBook Pro

Simulated Label Printer
Todd Ritland's MacBook Pro

Simulated Laser
Todd Ritland's MacBook Pro

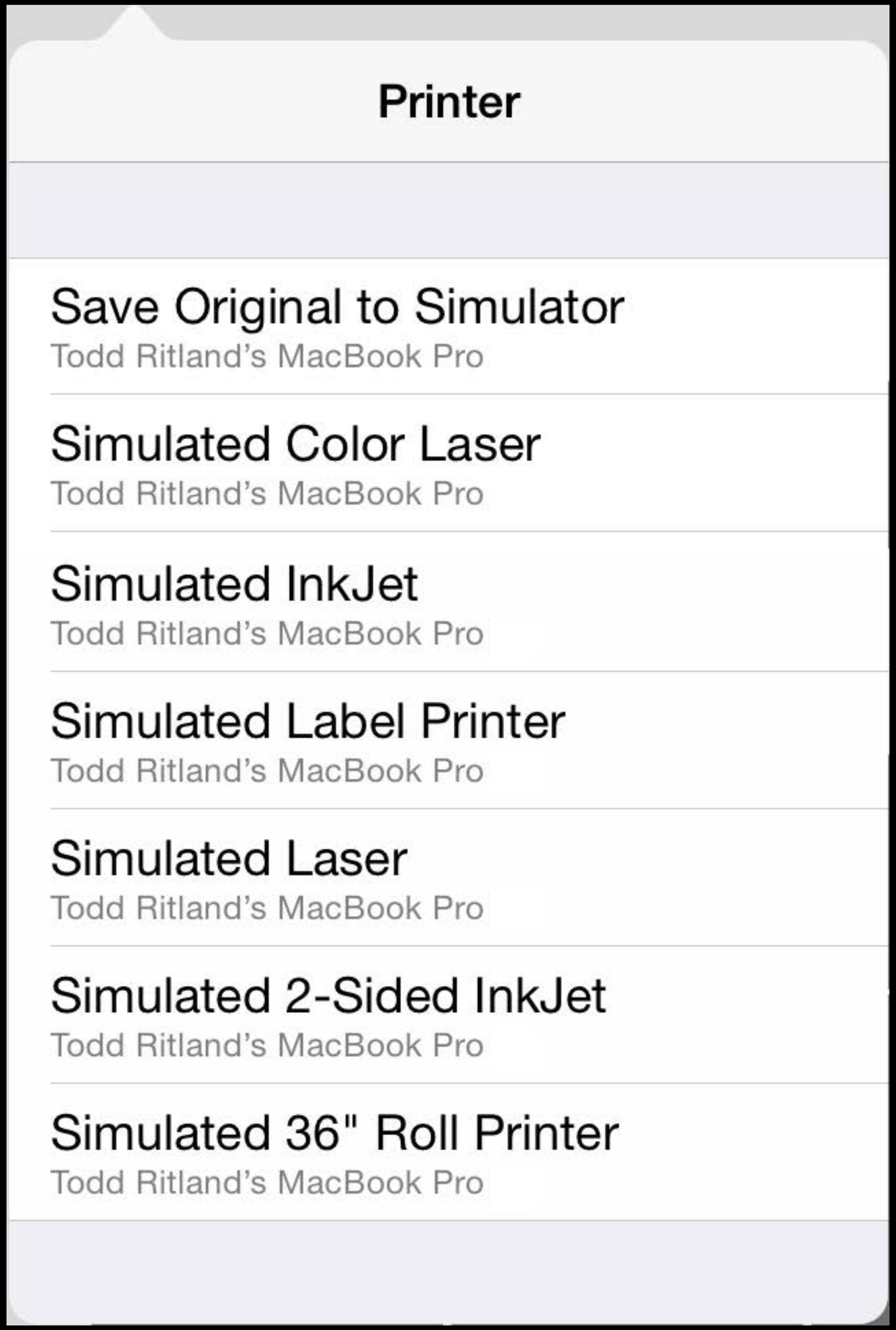
Simulated 2-Sided InkJet
Todd Ritland's MacBook Pro

Simulated 36" Roll Printer
Todd Ritland's MacBook Pro

Printer Picker

App Controlled Settings

API



Printer

Save Original to Simulator
Todd Ritland's MacBook Pro

Simulated Color Laser
Todd Ritland's MacBook Pro

Simulated InkJet
Todd Ritland's MacBook Pro

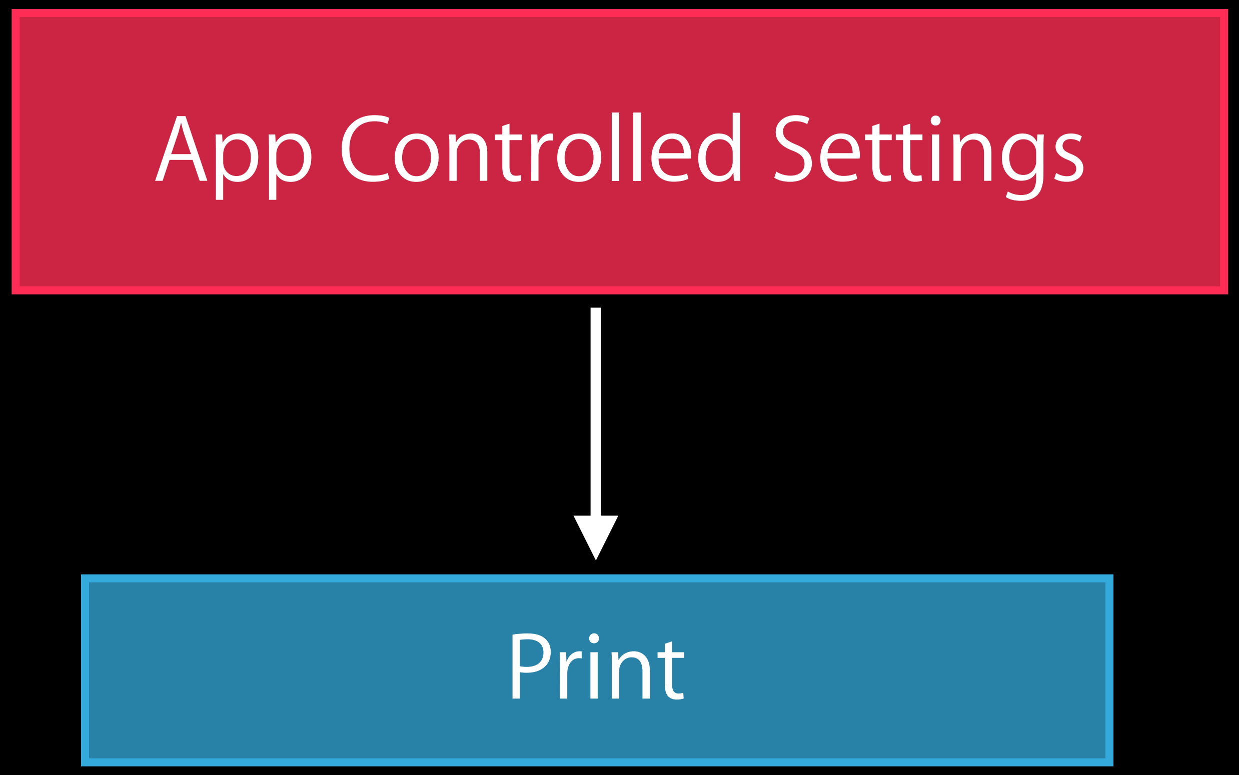
Simulated Label Printer
Todd Ritland's MacBook Pro

Simulated Laser
Todd Ritland's MacBook Pro

Simulated 2-Sided InkJet
Todd Ritland's MacBook Pro

Simulated 36" Roll Printer
Todd Ritland's MacBook Pro

Printer Picker



App Controlled Settings

Print

API

Basic Steps for Printing Without UI

NEW

Your app will include a setting for setting a printer

- Get the printer selector
- Present the UI
- Save the printer chosen by user

To print your app will

- Get the print controller
- Set up the attributes for the job
- Provide content to print
- Send the job with the new print controller method

Basic Steps for Printing Without UI

NEW

iOS will

- Communicate with AirPrint printer
- Daemon takes over and manages the job

iOS Background Printing Classes

NEW

`UIPrinterPickerController`

`UIPrinter`

`UIPrintInteractionController`

iOS Background Printing Classes

NEW

`UIPrinterPickerController`

`UIPrinter`

`UIPrintInteractionController`

iOS Background Printing Classes

NEW

`UIPrinterPickerController`

`UIPrinter`

`UIPrintInteractionController`

iOS Background Printing Classes

NEW

```
UIPrinterPickerController
UIPrinter

UIPrintInteractionController
```

UIPrinterPickerController

NEW

Your app must use this to choose the printer for printing

Same presentation options as `UIPrintInteractionContoller`

Present as a popover, sheet, or embedded in a navigation controller

Your app will be responsible for saving the URL

Allows filtering out printers

UIPrinterPickerController

NEW

Create and setup the `UIPrinterPickerController`

Standard presentation

`-presentAnimated:completionHandler:`

Popover presentation

`-presentFromRect:inView:animated:completionHandler:`

`-presentFromBarButtonItem:animated:completionHandler:`

In completion handler, check if user selected a printer, then save printer selected

Printing Without UI

NEW

When ready to print, use the `UIPrintInteractionController` and call

    `-printToPrinter:completionHandler:`

The `UIPrinter` passed in can be the object obtained by `UIPrinterPickerController`, or it can be a `UIPrinter` object instantiated with the URL

iOS Classes for Printing Without UI

NEW

```
UIPrinterPickerController
UIPrinter

UIPrintInteractionController
```

iOS Classes for Printing Without UI

NEW

```
UIPrinterPickerController
UIPrinter


UIPrintInteractionController
```

UIPrinter Methods and Properties

NEW

```
-contactPrinter:

-URL

-displayName
-displayLocation
-supportedJobTypes
-makeAndModel
-supportsColor
-supportsDuplex
```

UIPrinter Methods and Properties

NEW

```
-contactPrinter:

-URL

-displayName
-displayLocation
-supportedJobTypes
-makeAndModel
-supportsColor
-supportsDuplex
```

UIPrinter Methods and Properties

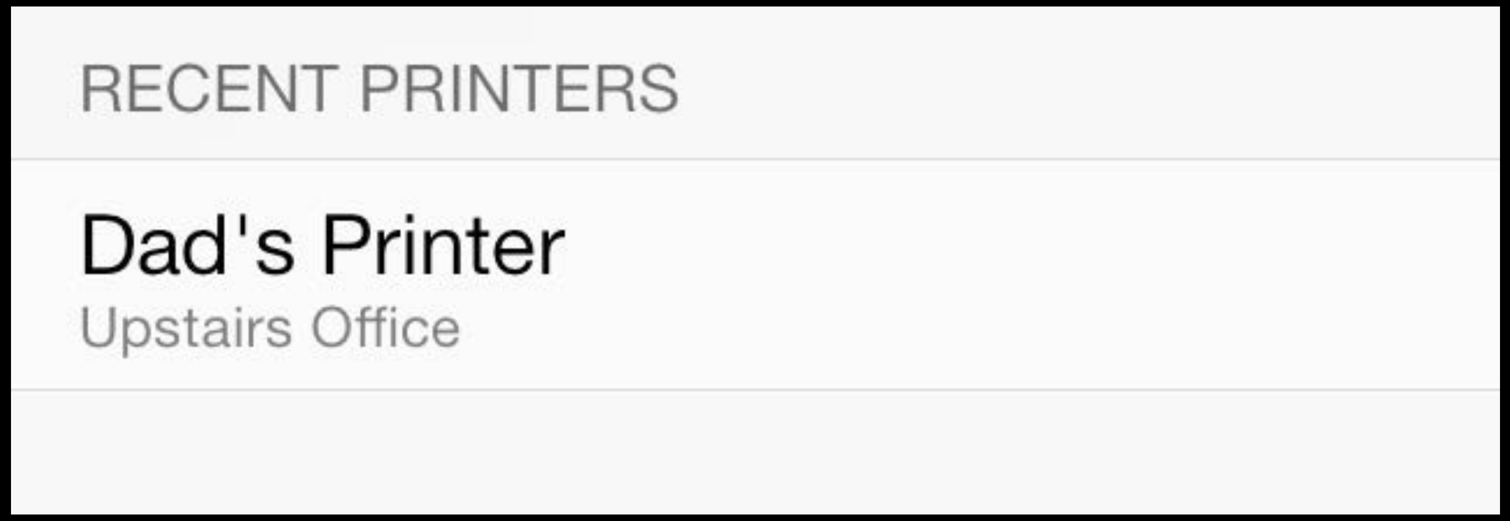
NEW

```
–contactPrinter:

–URL

–displayName
–displayLocation
–supportedJobTypes
–makeAndModel
–supportsColor
–supportsDuplex
```

UIPrinter Methods and Properties

NEW

```
-contactPrinter:

-URL

-displayName
-displayLocation
-supportedJobTypes
-makeAndModel
-supportsColor
-supportsDuplex
```



RECENT PRINTERS

Dad's Printer
Upstairs Office

UIPrinter Methods and Properties

NEW

-contactPrinter:

-URL

-displayName

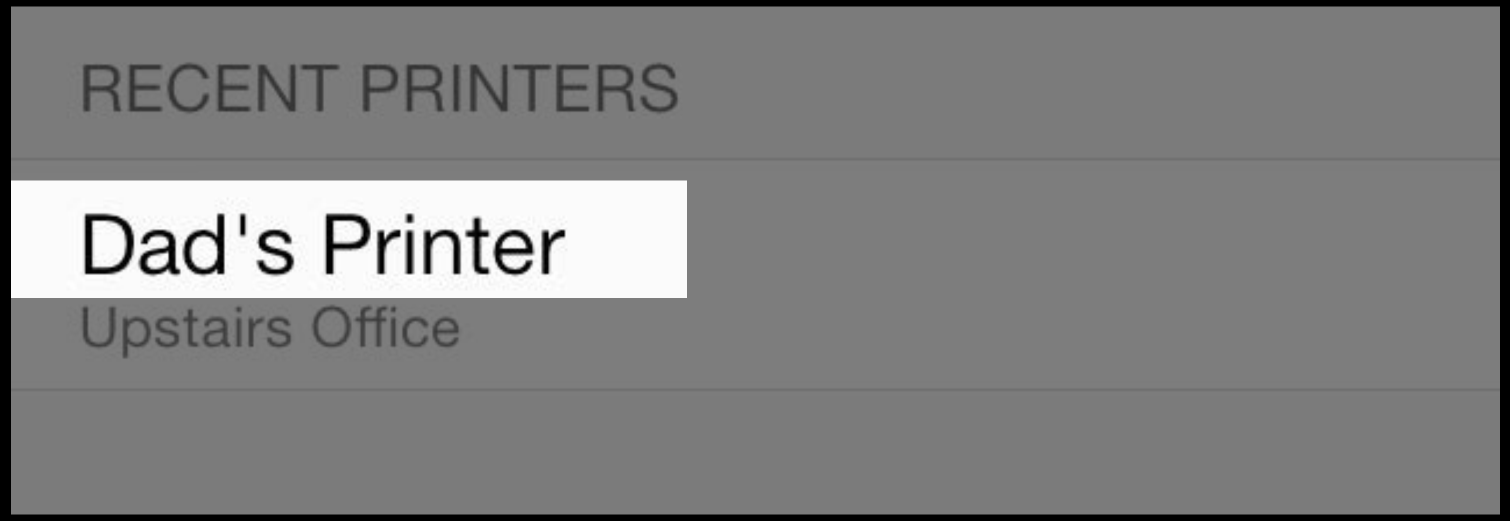
-displayLocation

-supportedJobTypes

-makeAndModel

-supportsColor

-supportsDuplex



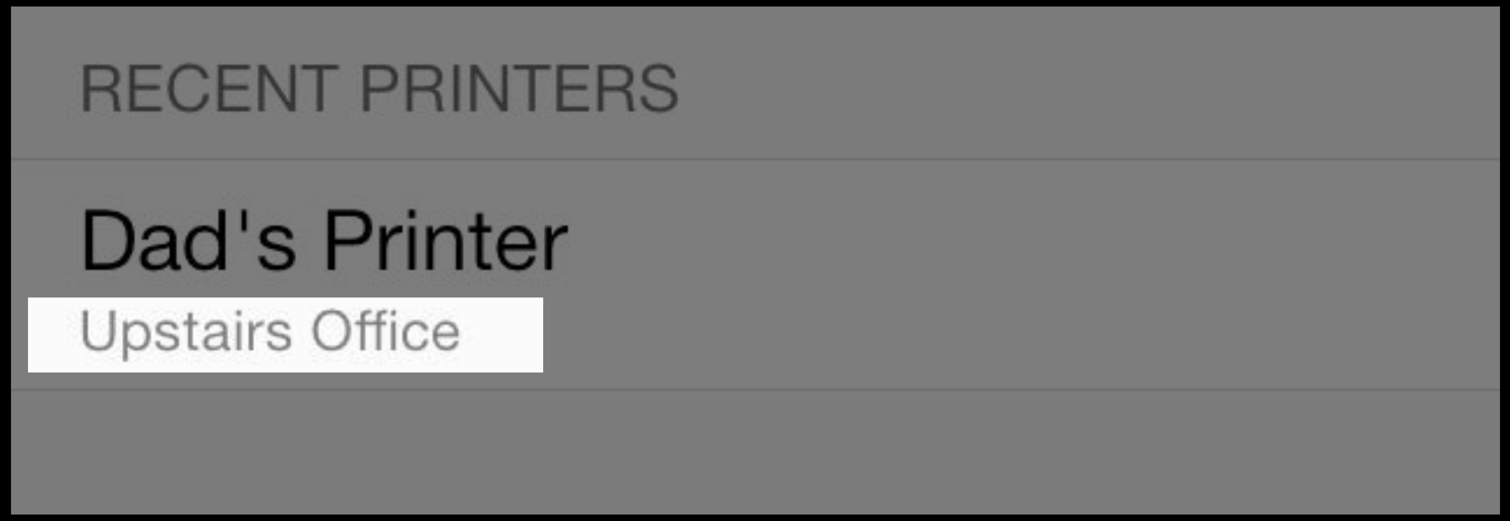
RECENT PRINTERS

Dad's Printer

Upstairs Office

UIPrinter Methods and Properties

```
-contactPrinter:

-URL

-displayName
-displayLocation
-supportedJobTypes
-makeAndModel
-supportsColor
-supportsDuplex
```



RECENT PRINTERS

Dad's Printer

Upstairs Office

UIPrinter Methods and Properties

NEW

```
-contactPrinter:

-URL

-displayName
-displayLocation
-supportedJobTypes
-makeAndModel
-supportsColor
-supportsDuplex
```

UIPrinter Methods and Properties

NEW

```
-contactPrinter:

-URL

-displayName
-displayLocation
-supportedJobTypes
-makeAndModel
-supportsColor
-supportsDuplex
```

Using UIPrinter for Your UI

```
NSURL *savedPrinterURL = [[NSUserDefaults standardUserDefaults]
                          URLForKey:@“savedPrinter”];

if (savedPrinterURL) {
    self.savedPrinter = [UIPrinter printerWithURL:savedPrinterURL];

    [self.savedPrinter contactPrinter:^(BOOL available) {
        if (available) {
            self.printerNameLabel.text = self.savedPrinter.displayName;
            self.printerLocationLabel.text = self.savedPrinter.displayLocation;
        }
        else {
            self.printerConnectionGoneIndicator.hidden = NO;
        }
    }];
}
```

Using UIPrinter for Your UI

```
NSURL *savedPrinterURL = [[NSUserDefaults standardUserDefaults]
                           URLForKey:@“savedPrinter”];
if (savedPrinterURL) {
    self.savedPrinter = [UIPrinter printerWithURL:savedPrinterURL];

    [self.savedPrinter contactPrinter:^(BOOL available) {
        if (available) {
            self.printerNameLabel.text = self.savedPrinter.displayName;
            self.printerLocationLabel.text = self.savedPrinter.displayLocation;
        }
        else {
            self.printerConnectionGoneIndicator.hidden = NO;
        }
    }];
}
```

Using UIPrinter for Your UI

```
NSURL *savedPrinterURL = [[NSUserDefaults standardUserDefaults]
                          URLForKey:@“savedPrinter”];
if (savedPrinterURL) {
    self.savedPrinter = [UIPrinter printerWithURL:savedPrinterURL];

    [self.savedPrinter contactPrinter:^(BOOL available) {
        if (available) {
            self.printerNameLabel.text = self.savedPrinter.displayName;
            self.printerLocationLabel.text = self.savedPrinter.displayLocation;
        }
        else {
            self.printerConnectionGoneIndicator.hidden = NO;
        }
    }];
}
```

Using UIPrinter for Your UI

```
NSURL *savedPrinterURL = [[NSUserDefaults standardUserDefaults]
                          URLForKey:@“savedPrinter”];

if (savedPrinterURL) {
    self.savedPrinter = [UIPrinter printerWithURL:savedPrinterURL];

    [self.savedPrinter contactPrinter:^(BOOL available) {
        if (available) {
            self.printerNameLabel.text = self.savedPrinter.displayName;
            self.printerLocationLabel.text = self.savedPrinter.displayLocation;
        }
        else {
            self.printerConnectionGoneIndicator.hidden = NO;
        }
    }];
}
```

Using UIPrinter for Your UI

```
NSURL *savedPrinterURL = [[NSUserDefaults standardUserDefaults]
                           URLForKey:@“savedPrinter”];
if (savedPrinterURL) {
    self.savedPrinter = [UIPrinter printerWithURL:savedPrinterURL];

    [self.savedPrinter contactPrinter:^(BOOL available) {
        if (available) {
            self.printerNameLabel.text = self.savedPrinter.displayName;
            self.printerLocationLabel.text = self.savedPrinter.displayLocation;
        }
        else {
            self.printerConnectionGoneIndicator.hidden = NO;
        }
    }];
}
```

Using UIPrinter for Your UI

```
NSURL *savedPrinterURL = [[NSUserDefaults standardUserDefaults]
                            URLForKey:@“savedPrinter”];

if (savedPrinterURL) {
    self.savedPrinter = [UIPrinter printerWithURL:savedPrinterURL];

    [self.savedPrinter contactPrinter:^(BOOL available) {
        if (available) {
            self.printerNameLabel.text = self.savedPrinter.displayName;
            self.printerLocationLabel.text = self.savedPrinter.displayLocation;

        }
        else {
            self.printerConnectionGoneIndicator.hidden = NO;
        }
    }];
}
```

Using UIPrinter for Your UI

```
NSURL *savedPrinterURL = [[NSUserDefaults standardUserDefaults]
                           URLForKey:@“savedPrinter”];

if (savedPrinterURL) {
    self.savedPrinter = [UIPrinter printerWithURL:savedPrinterURL];

    [self.savedPrinter contactPrinter:^(BOOL available) {
        if (available) {
            self.printerNameLabel.text = self.savedPrinter.displayName;
            self.printerLocationLabel.text = self.savedPrinter.displayLocation;

        }

        else {
            self.printerConnectionGoneIndicator.hidden = NO;
        }

    }];

}
```

UIPrinter and Filtering Printers

NEW

For apps designed for specific printers

Apps can control which printers are shown

Any property of the printer can be used to filter.

When presenting `UIPrinterPickerController`, use delegate method

```
    -shouldShowPrinter:
```

Called for every printer that is discovered

New Class of
Printing Applications

UI-less Printing Demo

Claudia Roberts
Printing System Engineer

Printing with No UI

Opens up door for new class of printing applications

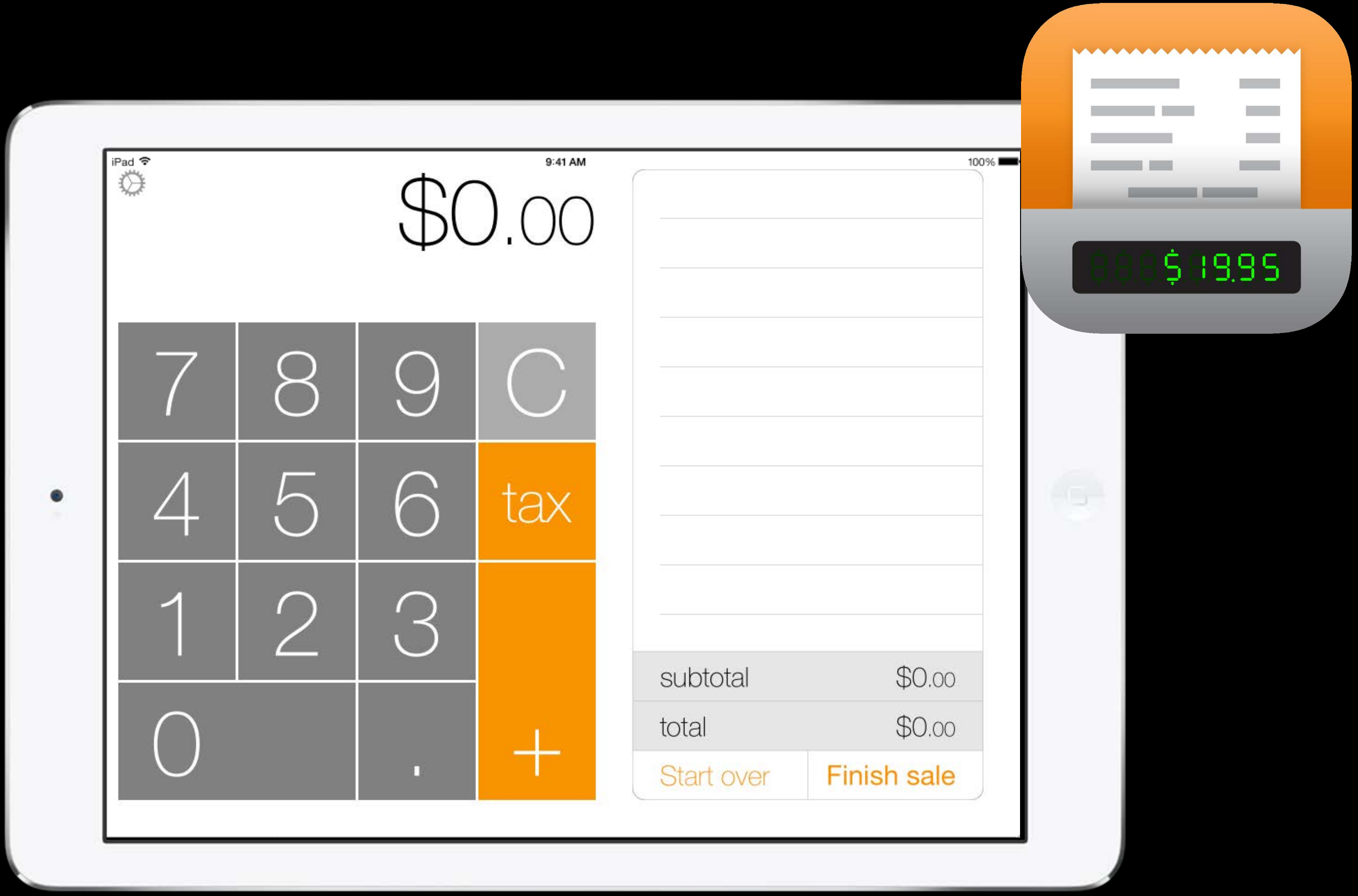
- Endless possibilities
- Application can save multiple printers

Printing with No UI

Case Study—cash register App

- Unnecessary interaction with print dialog in current printing paradigm
- New API affords the developer the ability to enhance and simplify the user printing experience
- Easy to refactor code to adopt UI-less printing

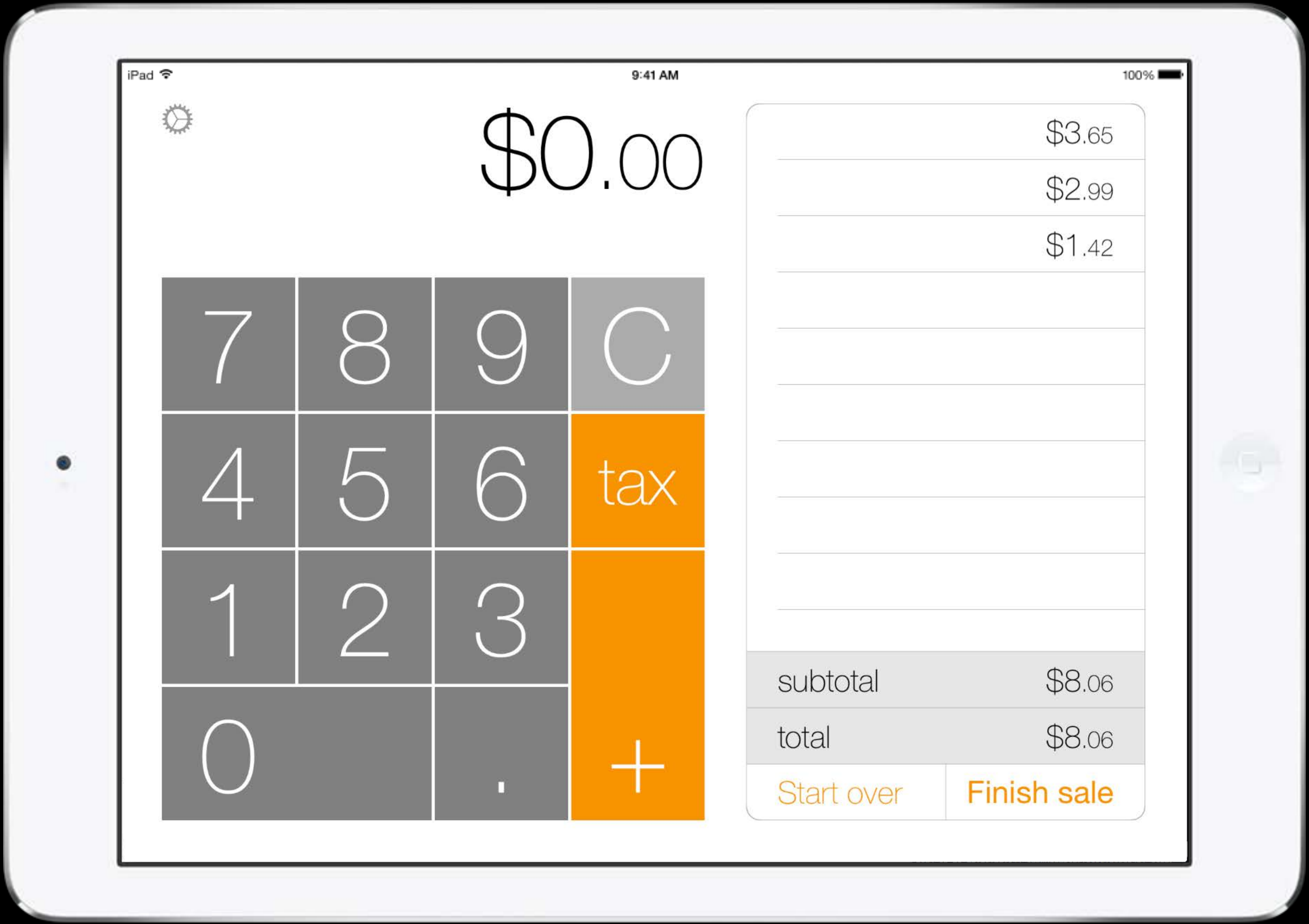
Case Study—Cash Register App



$0.00

subtotal $0.00

total $0.00

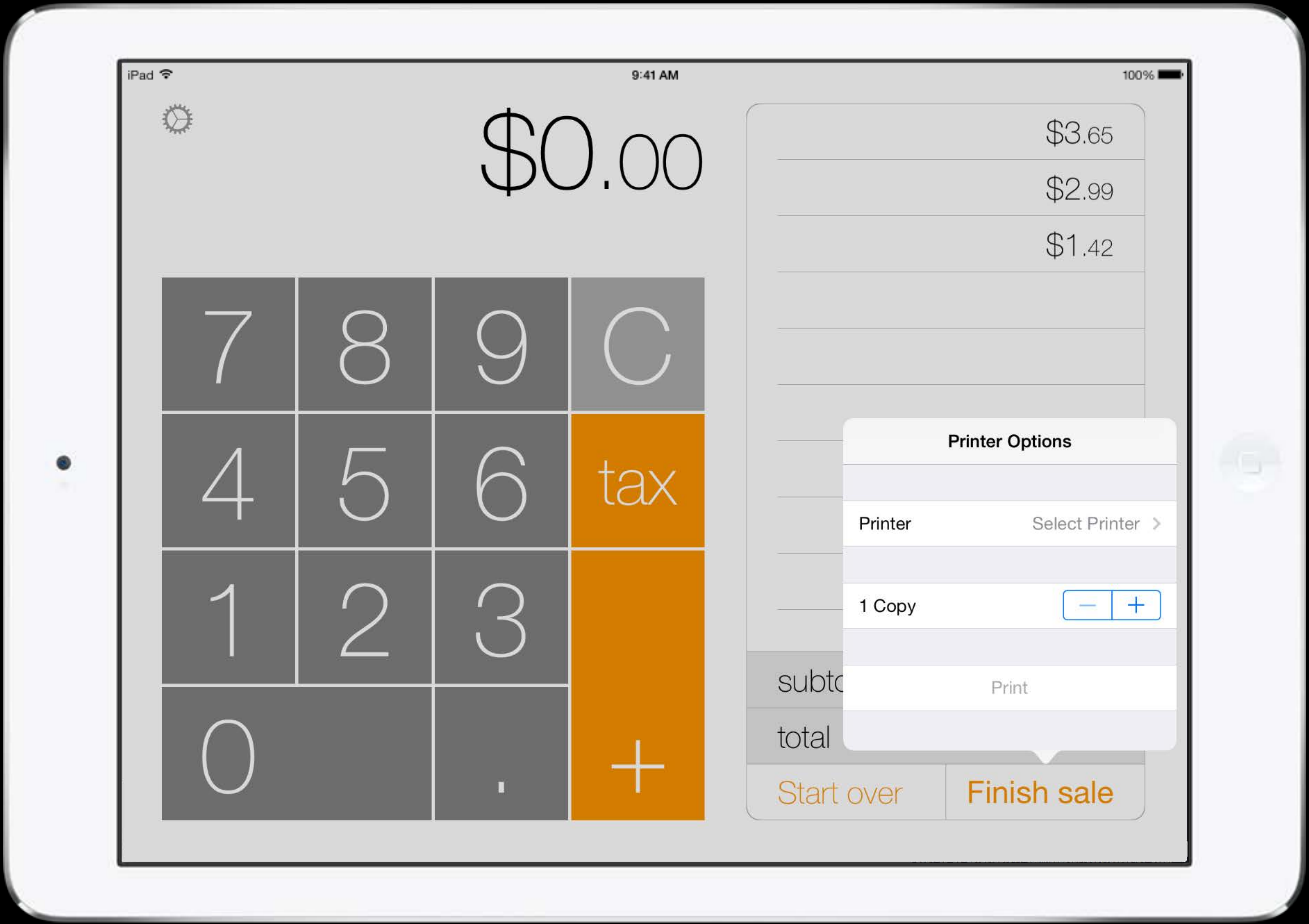


$3.65

$2.99

$1.42

subtotal	$8.06
total	$8.06

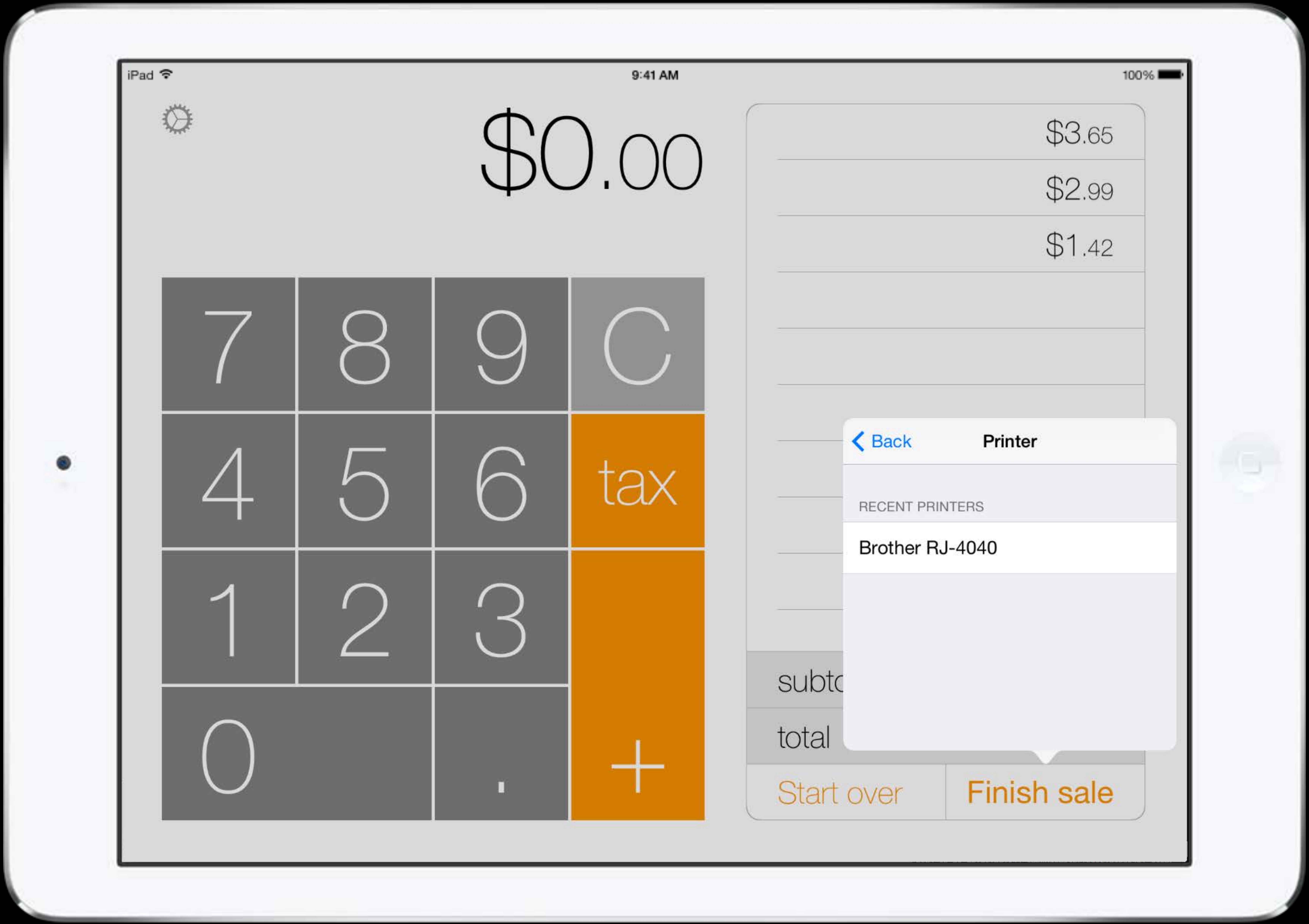


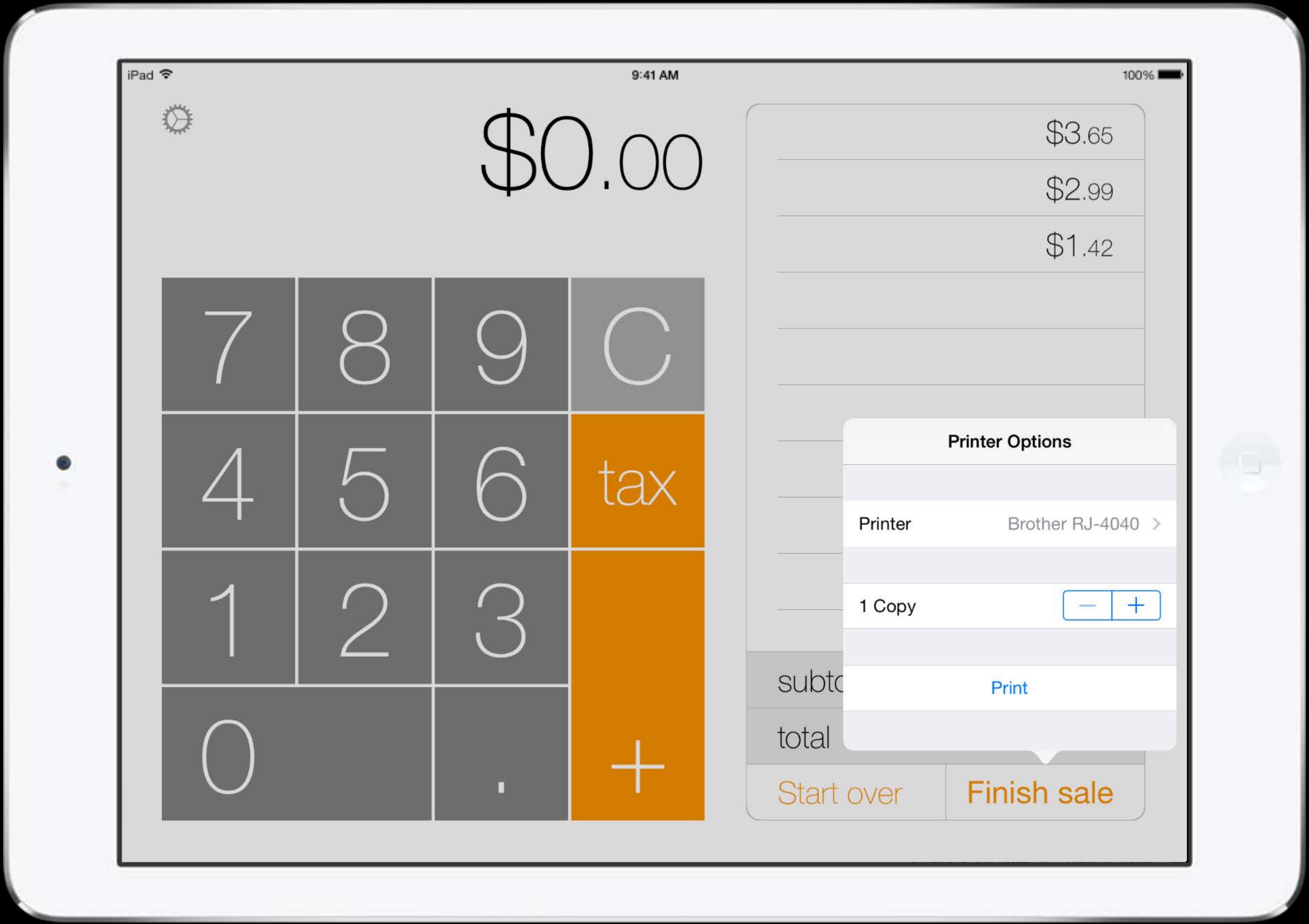
Printer Options

Printer Select Printer

1 Copy

Print


iPad
9:41 AM
100%
$0.00
$3.65
$2.99
$1.42
7
8
9
C
4
5
6
tax
1
2
3
0
.
+
Back
Printer
RECENT PRINTERS
Brother RJ-4040
subto
total
Start over
Finish sale



$0.00

$3.65

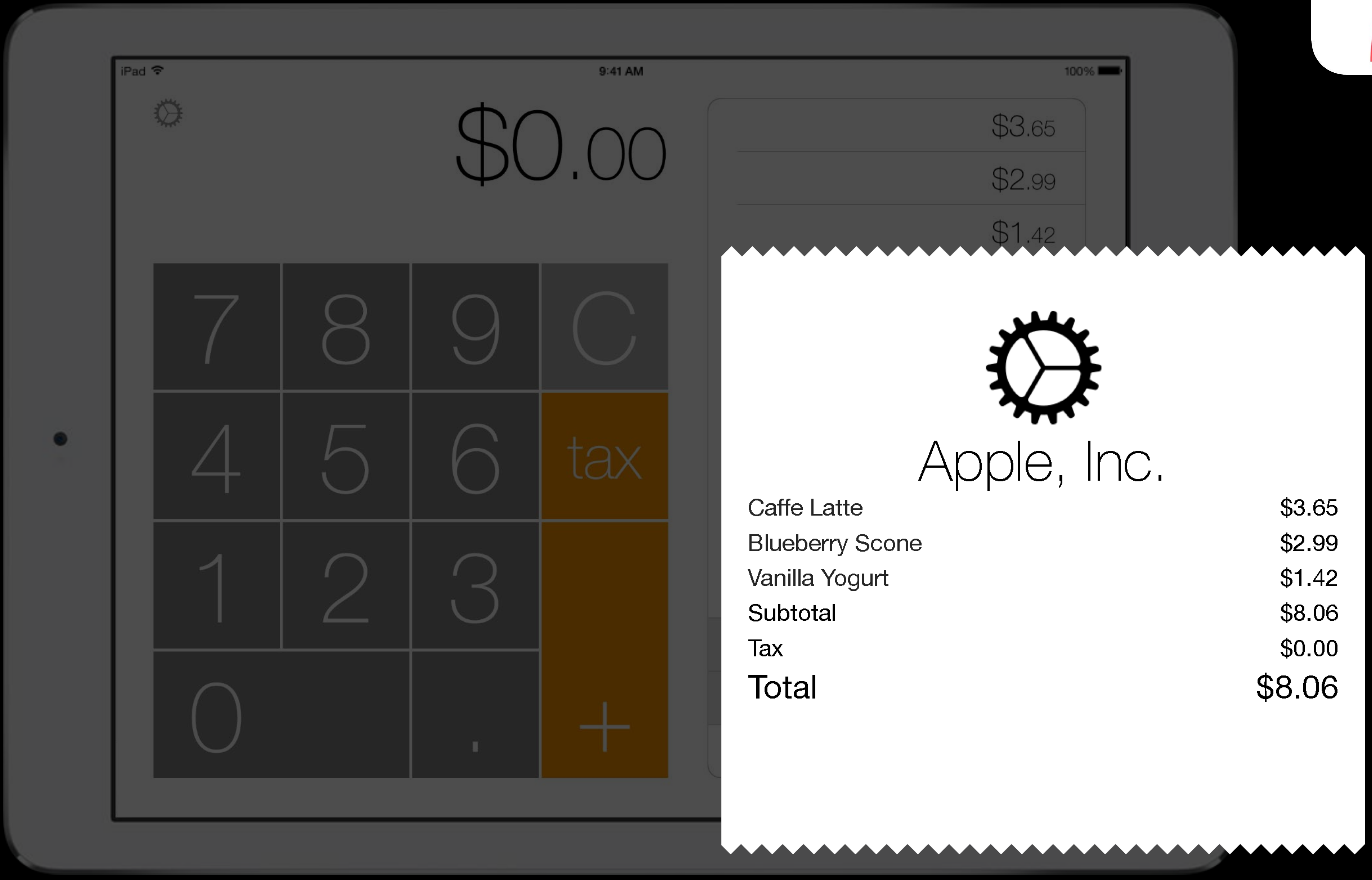
$2.99

$1.42

Printer Options

Printer Brother RJ-4040

1 Copy



Apple, Inc.

Caffe Latte	$3.65
Blueberry Scone	$2.99
Vanilla Yogurt	$1.42
Subtotal	$8.06
Tax	$0.00
Total	**$8.06**

Demo

Charles Duyk
Printing System Engineer

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

Apps can print without showing UI every time starting in iOS 8

More Information

Paul Danbold
Core OS Technologies Evangelist
danbold@apple.com

iOS Printing Documentation and Sample Code
http://developer.apple.com/airprint

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 10:15AM

Labs

●	AirPrint Lab	Core OS Lab A	Friday 10:15AM

WWDC14