



News.me

Brooklyn iOS Dev Meetup – March 28, 2012

Robert Tolar Haining

Background: me



CONDÉ NAST
DIGITAL



The News.me Team



Jon Ferrer



Me!



Jake Levine



Tim Ullrich



Josh Petri



Justin
Van Slembrouck



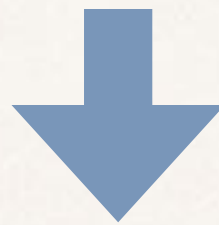
Mike Young



Day Man

Background: News.me

The New York Times



bitly



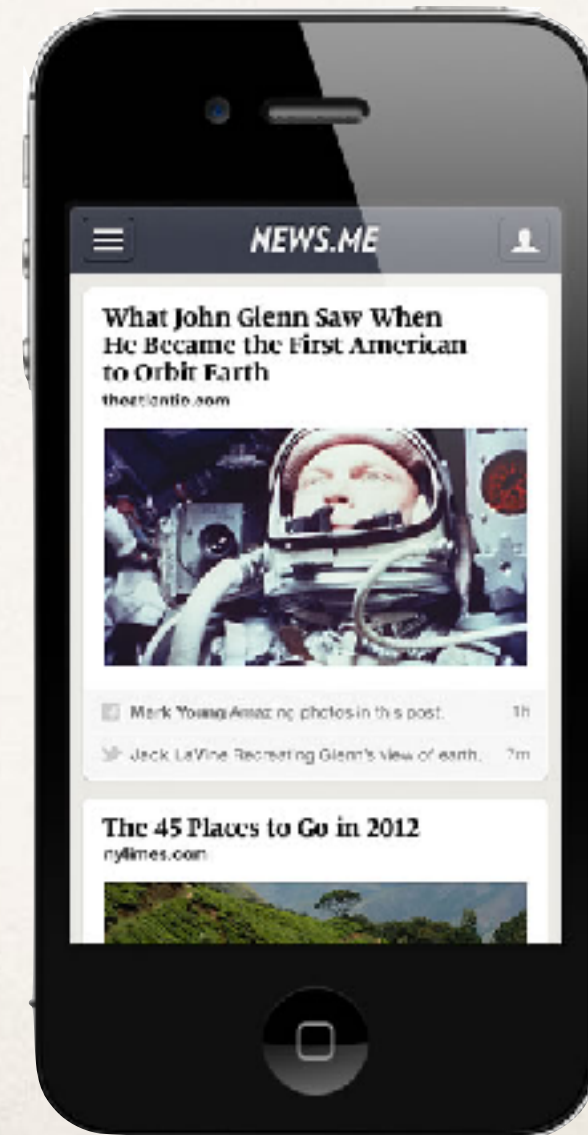
betaworks

From iPad to iPhone

April 2011



March 2012



Web soon!



Demo — News.me v2.0.1

Core Text

Core Text vs UILabel

- ❖ UILabel
 - ❖ Simple, basic text display
 - ❖ But limited:
 - ❖ font, color, alignment, line break mode

Core Text vs UILabel

- ✧ Core Text
 - ✧ Powerful, but tricky
 - ✧ We use it for:
 - ✧ Mixed formatting – bold, multiple fonts, different colors
 - ✧ Line height
 - ✧ Kerning [spacing between letters]

Core Text: Comparison

UILabels

**My Completely
Overwhelming And
Amazing SXSW
Adventure**



businessinsider.com



businessinsider My Completely Overwhelming
And Amazing SXSW Adventure by
@shontelaylay

1h

Core Text

**My Completely
Overwhelming And
Amazing SXSW
Adventure**



businessinsider.com



businessinsider My Completely
Overwhelming And Amazing SXSW Adventure
by @shontelaylay

56m

Core Text: Overview

- ❖ Create a NSAttributedString object
- ❖ Use string ranges to modify formatting
- ❖ Create a CTFramesetterRef object
- ❖ Draw the framesetter
- ❖ Or, for more control, draw line-by-line

NMCustomLabel : UILabel

- ❖ Subclassed UILabel
- ❖ Built in all the features we needed

Example: bold

```
int locOfTag = -1;
int totalTagLength = 0;
int eachTagLength = 7;

NSRange range0 = NSMakeRange(0, 0);
while(range0.location != NSNotFound){
    NSRange range1 = [self.text rangeOfString:@"<b>" options:NSLiteralSearch
range:NSMakeRange(range0.location+range0.length, self.text.length - range0.location- range0.length)];
    if(range1.location != NSNotFound){
        NSRange range2 = [self.text rangeOfString:@"</b>" options:NSLiteralSearch
range:NSMakeRange(range1.location+range1.length, self.text.length - range1.location- range1.length)];
        if(range2.location != NSNotFound){
            CFRange boldRange = CFRangeMake(range1.location, range2.location-range1.location-3);
            if(locOfTag >= 0 && locOfTag < boldRange.location){
                boldRange.location -= totalTagLength;
            }
            locOfTag = boldRange.location + boldRange.length;
            totalTagLength += eachTagLength;

            NSDictionary *attributes = [NSDictionary dictionaryWithObjectsAndKeys:(__bridge id)bodyFontBold,
kCTFontAttributeName, [textColorBold CGColor], kCTForegroundColorAttributeName, nil];
            CFAttributedStringSetAttributes(attrString, boldRange, (__bridge CFDictionaryRef)attributes, NO);
            range0 = range2;
        }else{
            break;
        }
    }else{
        break;
    }
}
```


Example: bold

```
int locOfTag = -1;
int totalTagLength = 0;
int eachTagLength = 7;

NSRange range0 = NSMakeRange(0, 0);
while(range0.location != NSNotFound){
    NSRange range1 = [self.text rangeOfString:@"<b>" options:NSLiteralSearch
range:NSMakeRange(range0.location+range0.length, self.text.length - range0.location- range0.length)];
    if(range1.location != NSNotFound){

}
    }else{
        break;
    }
}
```

Example: bold

```
int locOfTag = -1;
int totalTagLength = 0;
int eachTagLength = 7;

NSRange range0 = NSMakeRange(0, 0);
while(range0.location != NSNotFound){
    NSRange range1 = [self.text rangeOfString:@"<b>" options:NSLiteralSearch
range:NSMakeRange(range0.location+range0.length, self.text.length - range0.location- range0.length)];
    if(range1.location != NSNotFound){
        NSRange range2 = [self.text rangeOfString:@"</b>" options:NSLiteralSearch
range:NSMakeRange(range1.location+range1.length, self.text.length - range1.location- range1.length)];
        if(range2.location != NSNotFound){

        }else{
            break;
        }
    }else{
        break;
    }
}
```


Example: bold

```
int locOfTag = -1;
int totalTagLength = 0;
int eachTagLength = 7;

NSRange range0 = NSMakeRange(0, 0);
while(range0.location != NSNotFound){
    NSRange range1 = [self.text rangeOfString:@"<b>" options:NSLiteralSearch
range:NSMakeRange(range0.location+range0.length, self.text.length - range0.location- range0.length)];
    if(range1.location != NSNotFound){
        NSRange range2 = [self.text rangeOfString:@"</b>" options:NSLiteralSearch
range:NSMakeRange(range1.location+range1.length, self.text.length - range1.location- range1.length)];
        if(range2.location != NSNotFound){
            CFRange boldRange = CFRangeMake(range1.location, range2.location-range1.location-3);
            if(locOfTag >= 0 && locOfTag < boldRange.location){
                boldRange.location -= totalTagLength;
            }
            locOfTag = boldRange.location + boldRange.length;
            totalTagLength += eachTagLength;
        }
    }
}

}else{
    break;
}
}else{
    break;
}
}
```

Example: bold

```
int locOfTag = -1;
int totalTagLength = 0;
int eachTagLength = 7;

NSRange range0 = NSMakeRange(0, 0);
while(range0.location != NSNotFound){
    NSRange range1 = [self.text rangeOfString:@"<b>" options:NSLiteralSearch
range:NSMakeRange(range0.location+range0.length, self.text.length - range0.location- range0.length)];
    if(range1.location != NSNotFound){
        NSRange range2 = [self.text rangeOfString:@"</b>" options:NSLiteralSearch
range:NSMakeRange(range1.location+range1.length, self.text.length - range1.location- range1.length)];
        if(range2.location != NSNotFound){
            CFRange boldRange = CFRangeMake(range1.location, range2.location-range1.location-3);
            if(locOfTag >= 0 && locOfTag < boldRange.location){
                boldRange.location -= totalTagLength;
            }
            locOfTag = boldRange.location + boldRange.length;
            totalTagLength += eachTagLength;

            NSDictionary *attributes = [NSDictionary dictionaryWithObjectsAndKeys:(__bridge id)bodyFontBold,
kCTFontAttributeName, [textColorBold CGColor], kCTForegroundColorAttributeName, nil];
            CFAttributedStringSetAttributes(attrString, boldRange, (__bridge CFDictionaryRef)attributes, NO);
            range0 = range2;
        }else{
            break;
        }
    }else{
        break;
    }
}
```


Example: Paragraph Styles

```
int numParagraphSpecifiers = 3;
CTParagraphStyleSetting _settings[] = {
    {kCTParagraphStyleSpecifierAlignment, sizeof(ctTextAlignment), &ctTextAlignment},
    {kCTParagraphStyleSpecifierMaximumLineHeight, sizeof(CGFloat), &lineHeight},
    {kCTParagraphStyleSpecifierMinimumLineHeight, sizeof(CGFloat), & lineHeight},
    {kCTParagraphStyleSpecifierCount, sizeof(int), &numParagraphSpecifiers}
};
CTParagraphStyleRef paragraphStyle = CTParagraphStyleCreate(_settings, sizeof(_settings) /
sizeof(_settings[0]));
CFAttributedStringSetAttribute(attrString, CFRangeMake(0, CFAttributedStringGetLength(attrString)),
kCTParagraphStyleAttributeName, paragraphStyle);
```

Pro Tip: Estimating Text Size

```
- (CGSize)sizeThatFits:(CGSize)size{
    if(!self.text || self.text.length == 0){
        return CGSizeZero;
    }
    CGSize suggestedSize = CGSizeZero;
    if(!attrString){
        [self createAttributedString];
    }
    if(framesetter){
        if(size.width < 1){ size.width = 10000; }
        if(size.height < 1){ size.height = 10000; }
        suggestedSize = CTFramesetterSuggestFrameSizeWithConstraints(
            framesetter, /* Framesetter */
            CRangeMake(0, CFAttributedStringGetLength(attrString)), /* String range (entire
string) */
            NULL, /* Frame attributes */
            size, /* Constraints (CGFLOAT_MAX indicates unconstrained) */
            NULL /* Gives the range of string that fits into the constraints, doesn't matter
in your situation */
        );
        //on iOS 4, we were getting heights of 14.9 where the line height was 15, so it was getting cut off.
        after we stop supporting iOS 4, we could safely kill these lines.
        suggestedSize.width = round(suggestedSize.width);
        suggestedSize.height = round(suggestedSize.height);
    }else{
        suggestedSize = size;
    }
    return suggestedSize;
}
```


Pro Tip: Estimating Text Size

```
- (CGSize)sizeThatFits:(CGSize)size{
    if(!self.text || self.text.length == 0){
        return CGSizeZero;
    }
    CGSize suggestedSize = CGSizeZero;
    if(!attrString){
        [self createAttributedString];
    }
    if(framesetter){
        if(size.width < 1){ size.width = 10000; }
        if(size.height < 1){ size.height = 10000; }
        suggestedSize = CTFramesetterSuggestFrameSizeWithConstraints(
            framesetter, /* Framesetter */
            CRangeMake(0, CFAttributedStringGetLength(attrString)), /* String range (entire
string) */
            NULL, /* Frame attributes */
            size, /* Constraints (CGFLOAT_MAX indicates unconstrained) */
            NULL /* Gives the range of string that fits into the constraints, doesn't matter
in your situation */
        );
        //on iOS 4, we were getting heights of 14.9 where the line height was 15, so it was getting cut off.
        after we stop supporting iOS 4, we could safely kill these lines.
        suggestedSize.width = round(suggestedSize.width);
        suggestedSize.height = round(suggestedSize.height);
    }else{
        suggestedSize = size;
    }
    return suggestedSize;
}
```

Available on GitHub

❖ <http://github.com/newsme/>

Demo — Core Text Sample App

A few quick tips

Quick Tip: Entity Extraction

- ❖ NSLinguisticTagger
- ❖ Extract entities from any text
- ❖ People, Places, Organizations, Verbs, etc

Quick Tip: Entity Extraction

“Rangel and His Campaign to Pay \$23,000 Fine Over Misuse of Rent Law”

Organizations

- Congress
- Congressional
- Federal Election
- Commission
- Federal Election
- Commission Records
- House
- House Ethics Committee
- National Legal
- The New York Times
- The Times
- Times
- Ways and Means
- Committee

People

- Hannah Kim
- Harlem
- Lenox Terrace
- Mr Rangel
- Representative Charles B
- Rangel

Places

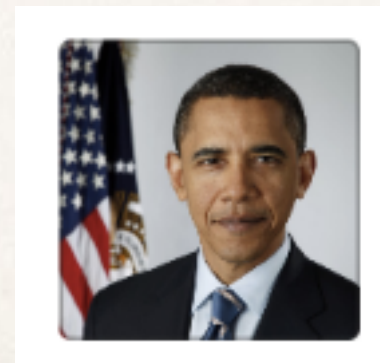
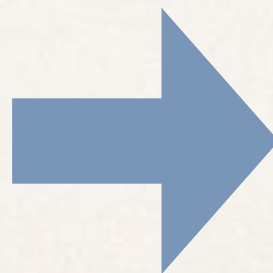
- Center
- City College
- Dominican Republic
- New York
- New York City
- Upper Manhattan
- Washington

Quick Tip: *Accessibility – VoiceOver*

- ❖ Super easy to implement
- ❖ Important to blind & visually impaired
- ❖ UIAccessibility protocol
 - ❖ accessibilityLabel
 - ❖ accessibilityHint
- ❖ MattGemmell.com – great article

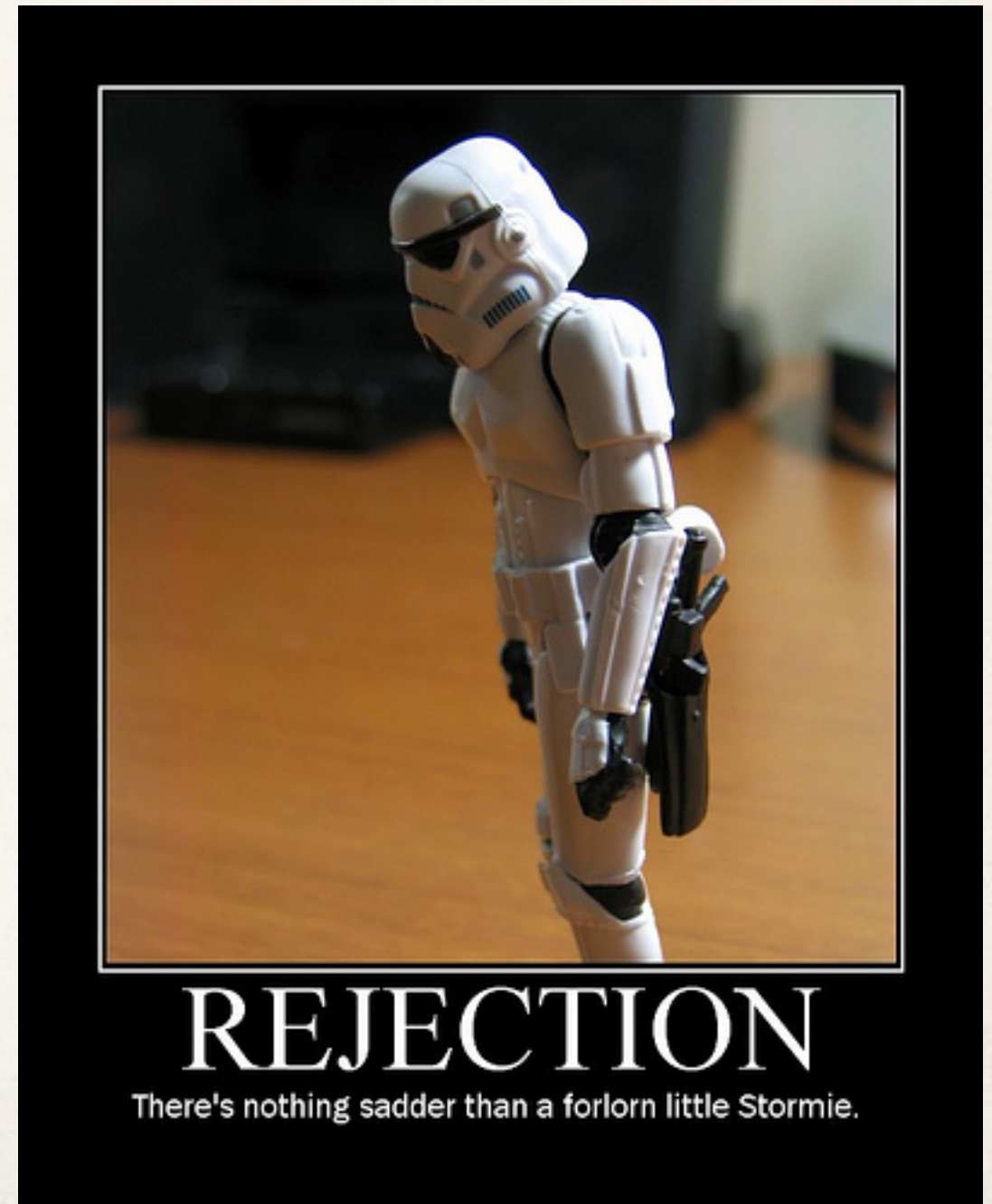
Quick Tip: CoreImage – Face Detection

- ❖ CImage – core image object
- ❖ CIDetector – detects faces
- ❖ CFaceFeature – defines a face in an image



Rejection

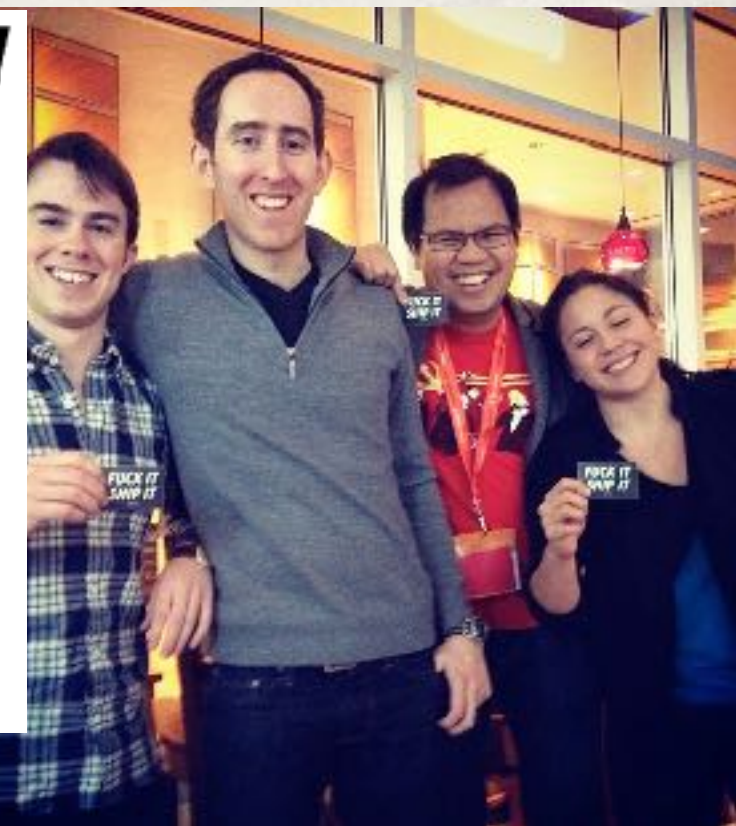
Don't take it personally







FUCK IT SHIP IT

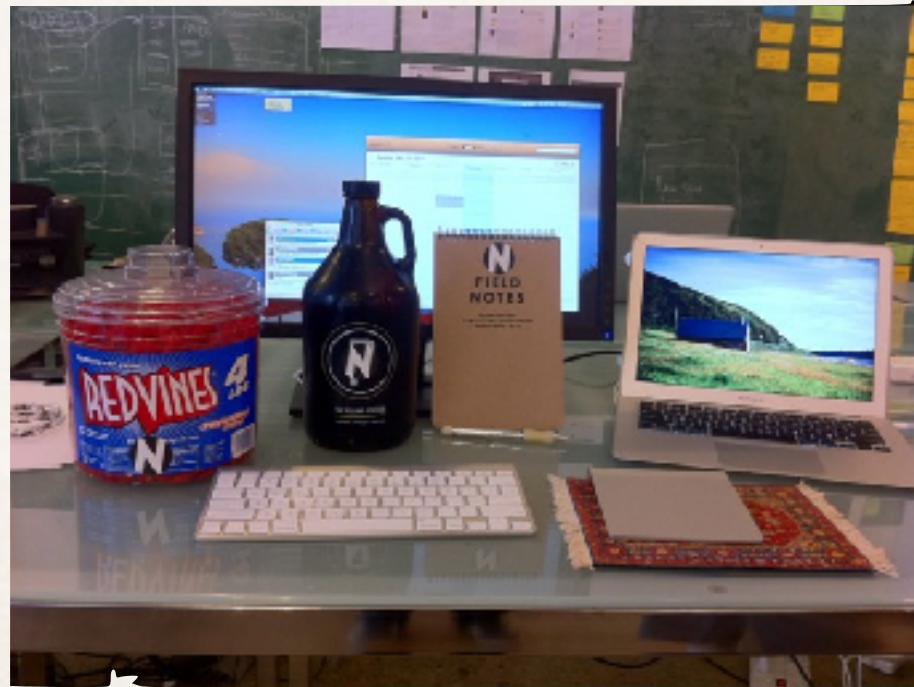


second is
exporter of clothing

FUCK IT
SHIP IT

We're Hiring!

- ❖ iOS + Python + Frontend



Contact – bit.ly/bkiosmeetup

- ✧ Me (Rob Haining)
 - ✧ Twitter – @tolar
 - ✧ Email – rob@news.me
- ✧ News.me
 - ✧ Twitter – @newsdotme
 - ✧ Email – feedback@news.me
 - ✧ Code – github.com/newsme