

# Magic Words

## Harnessing the Power of Text Kit

Gene Whitaker

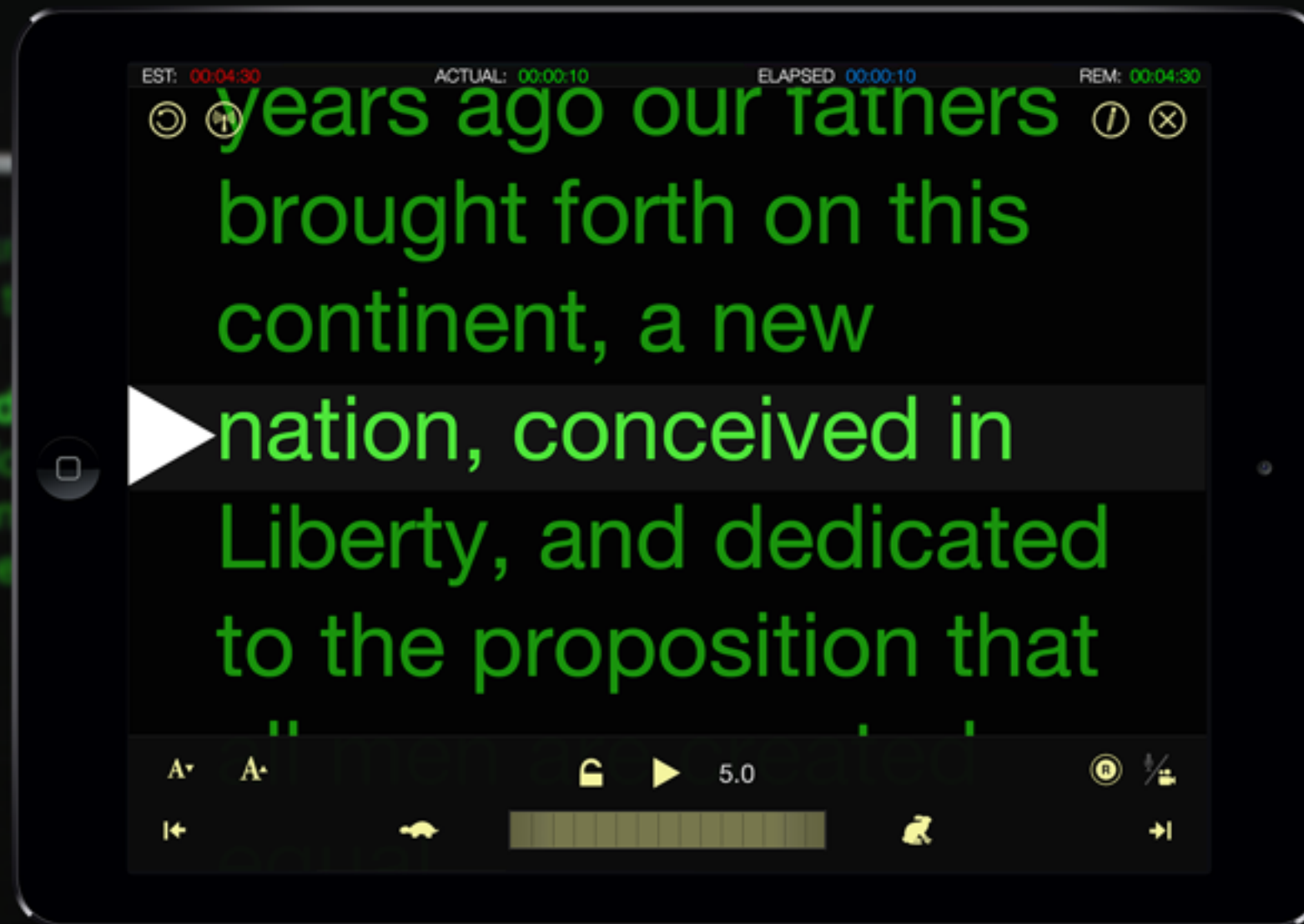


@gene\_whitaker

@bombingbrain

[gene@bombingbrain.com](mailto:gene@bombingbrain.com)

# Teleprompter



# Setlists

iPad 9:41 AM 100%

Little Things 7's Fears

Am7 Em Dm

I've got a rusty feeling deep down

Am7 Em Dm

So much I thought I'd lost what I'd found





Am7 Em Dm

My Soul's been out of touch with me

Am7 Em Dm

And I was too blind to see...

.....

Dan's Bar Set 1

Within And Without

Before

Beautiful

Introspection

Little Things

My Favorite Song

Just A Little Bit Of L...

A Moment's Notice

Notes

Mid-tempo. Guitar chords up front 2x before vocals come in.

# Super-Brief History of Text on iOS

*Pre-iOS 6*

UILabel

UITextField

UITextView

```
label.font = [UIFont fontWithName...
```

# Super-Brief History of Text on iOS

*iOS 6*

UILabel

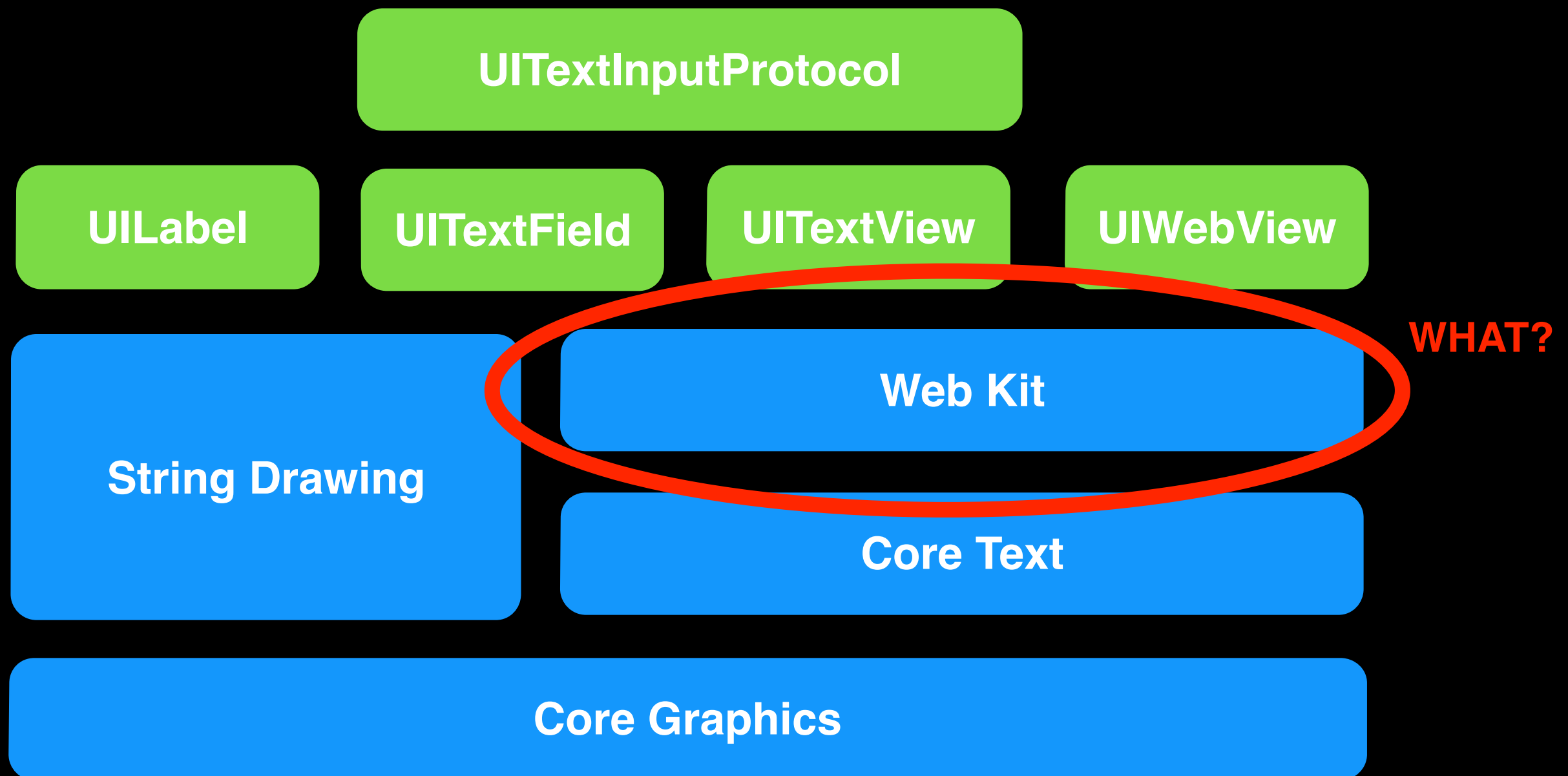
UITextField

UITextView

NSAttributedString

# Super-Brief History of Text on iOS

*iOS 6*



# Super-Brief History of Text on iOS

## *iOS 6*

- Rich text formatting or layout unsupported
- “Roll your own” using Core Text



# Super-Brief History of Text on iOS

*iOS 7*

# Text Kit



Rich Text Editing

Formatting/Syntax Highlighting

Custom Layout

Annotation

User Interaction

# **Text Kit Capabilities**

Text Effects

Text Highlighting

Custom Text Input Behavior

Interaction with UI Elements

# What is Text Kit?

3 interdependent classes:



- Manage the storage, geometry and layout of *glyphs*

# What is a *Glyph*?

Character

Glyph

“A”



*A*

“ff”

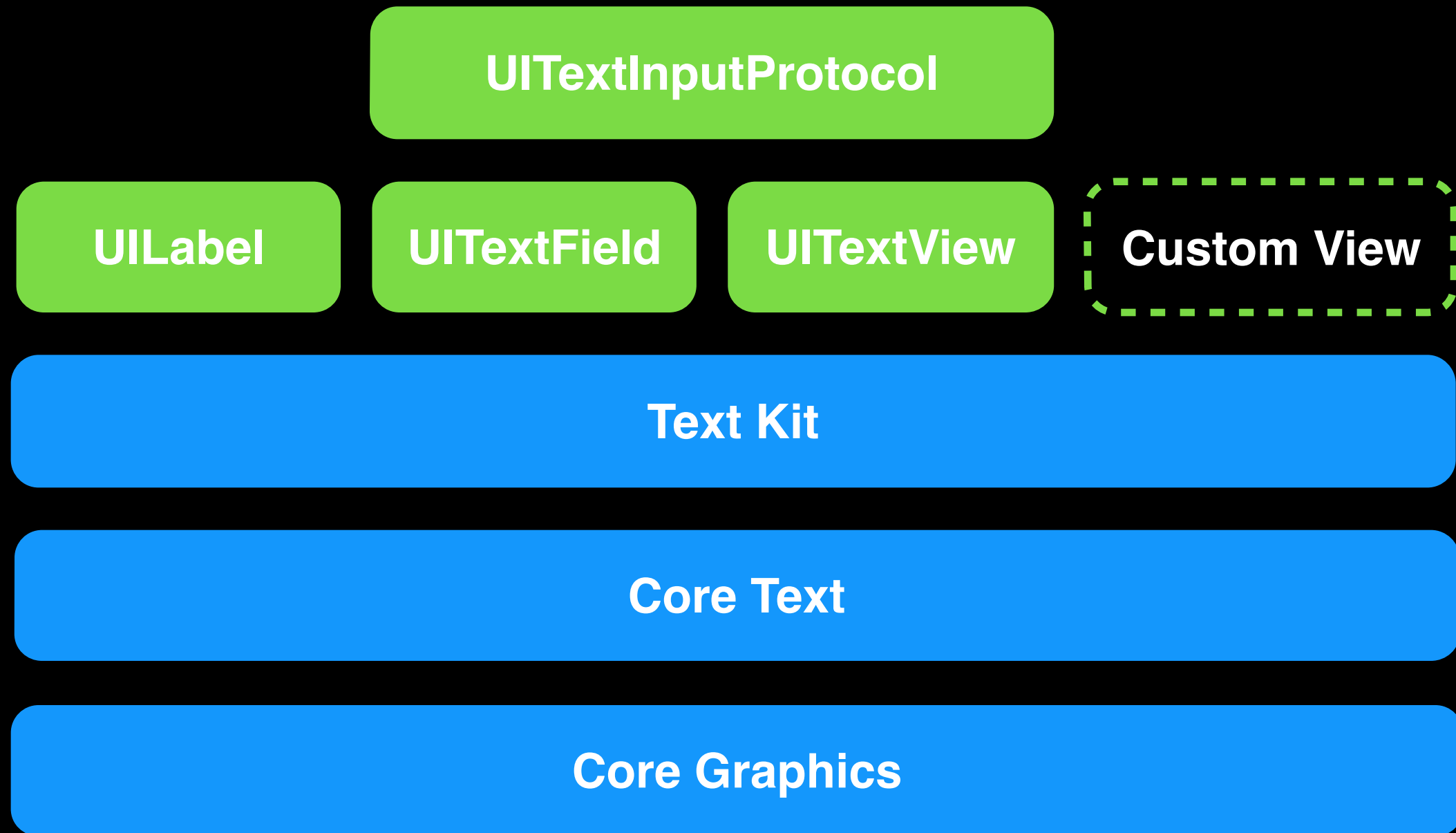


*ff*

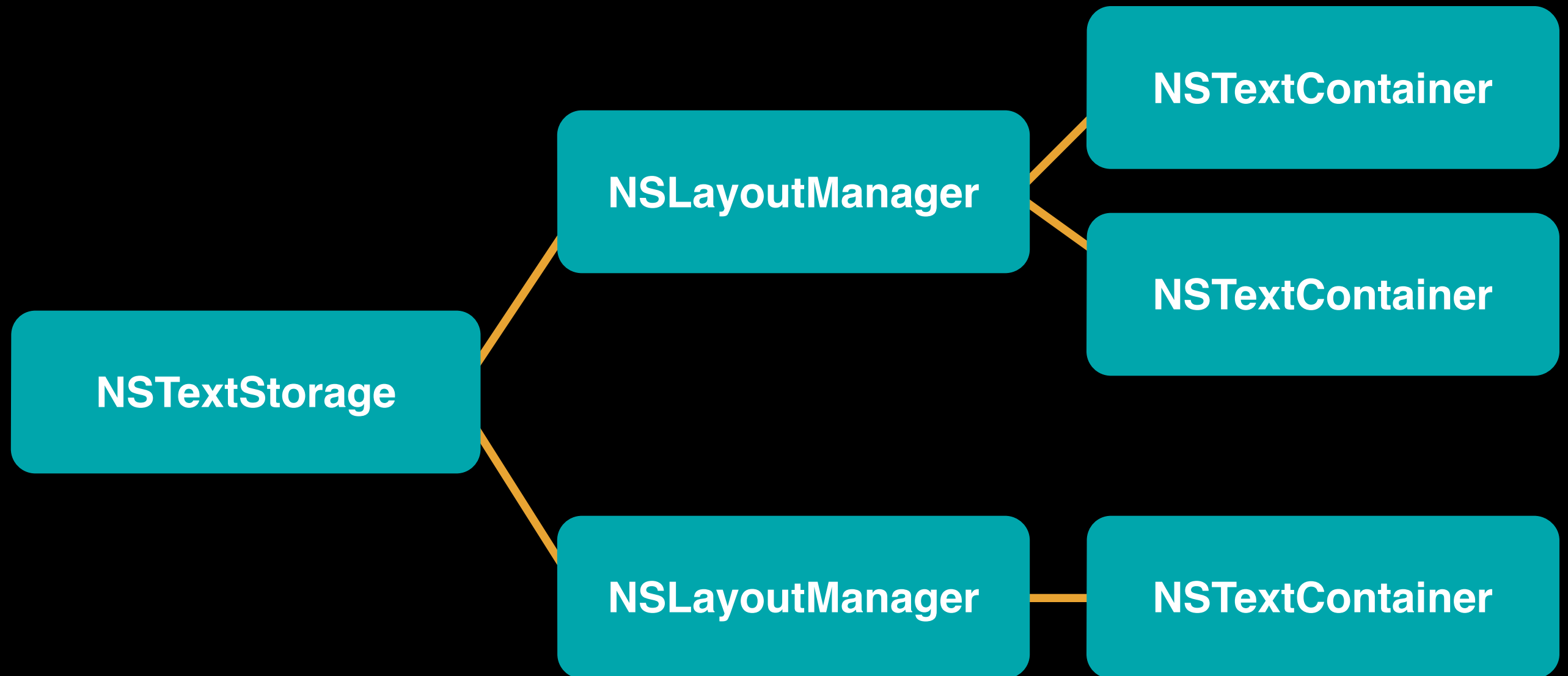
# Glyph Mapping

- (NSRange) **glyphRangeForCharacterRange:**  
(NSRange) charRange actualCharacterRange:  
(NSRangePointer) actualCharRange;
- (NSRange) **characterRangeForGlyphRange:**  
(NSRange) glyphRange actualGlyphRange:  
(NSRangePointer) actualGlyphRange;

# Architecture



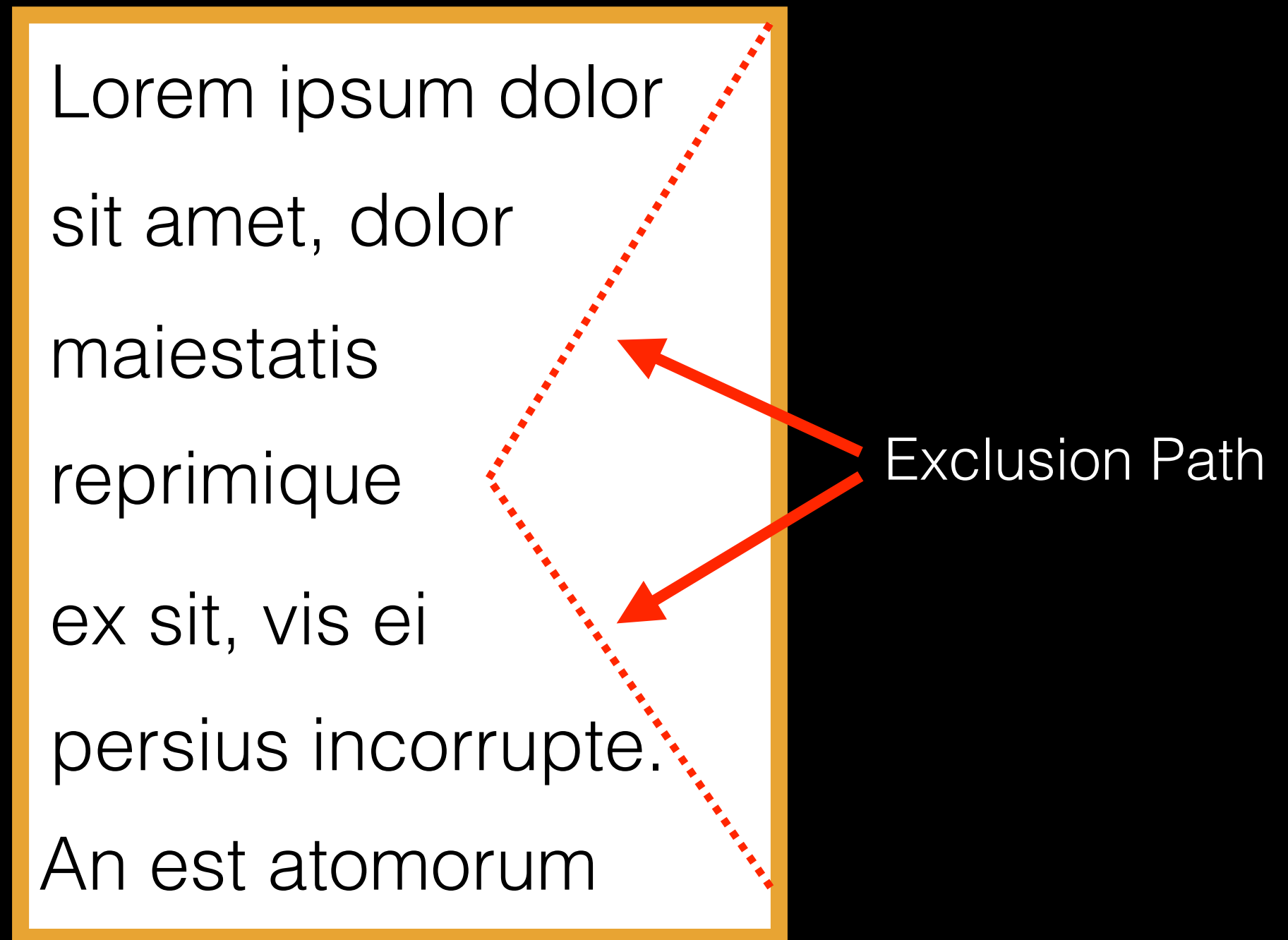
# Relationships



# NSTextContainer

- Identifies geometry in which glyphs will be laid out
- Not a view - identifies area in views where text can be laid out
- Allows *exclusion paths*
- Subclassing allows even more customization

# NSTextContainer





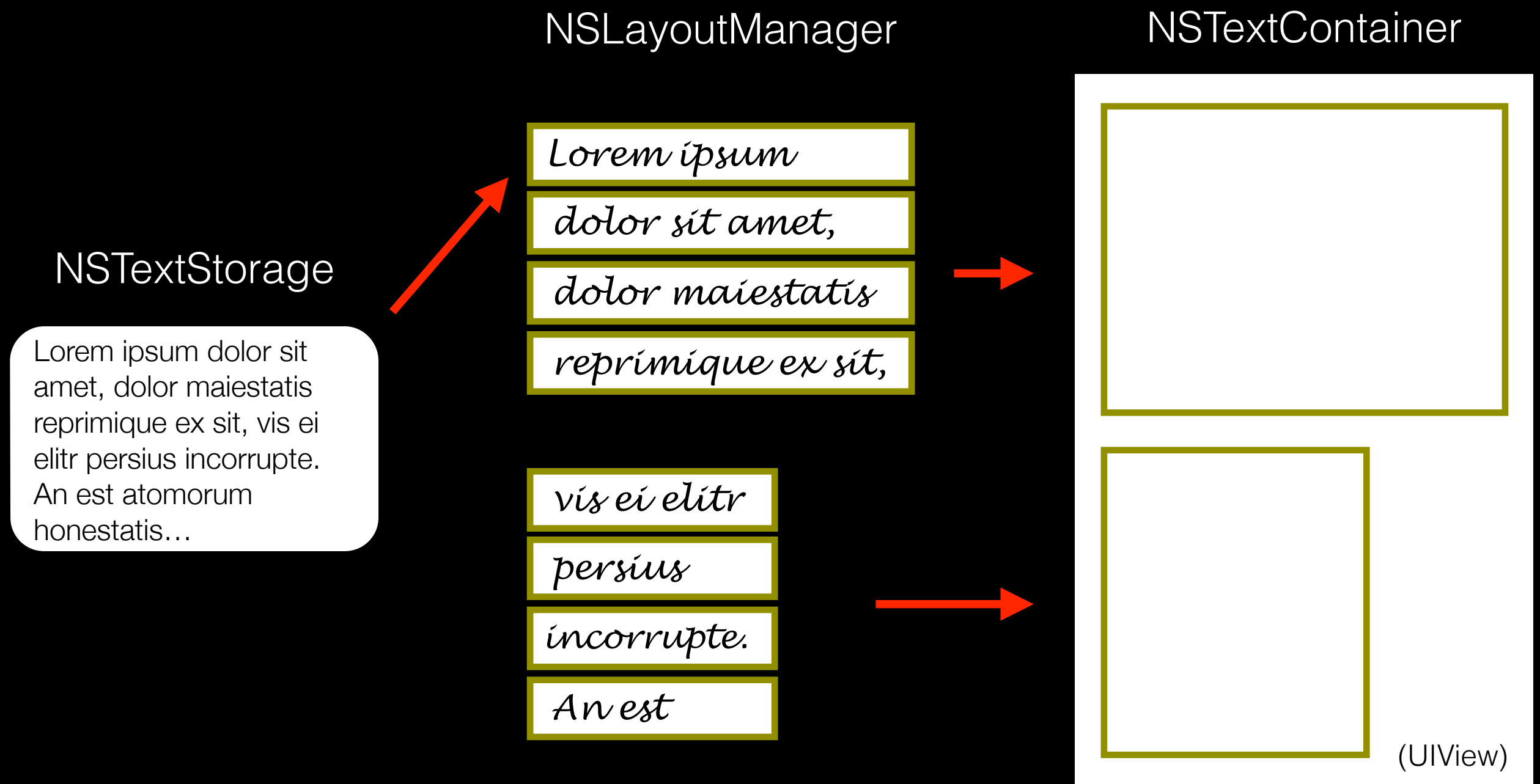
# NSStringStorage

- Stores raw strings, including formatting
- Subclass of `NSAttributedString`
- Notifies text layout system of content changes
- Subclassing allows customization of text storage and formatting

# NSLayoutManager

- Coordinates glyph layout for given text storage and containers
- Maps characters to glyphs
- Sets glyphs in a series of NSTextContainers
- Subclassing allows customization of glyphs

# Layout Process



# Customizing Layout and Behavior

- Direct manipulation
- Subclassing
- Delegates

# Subclassing NSTextStorage

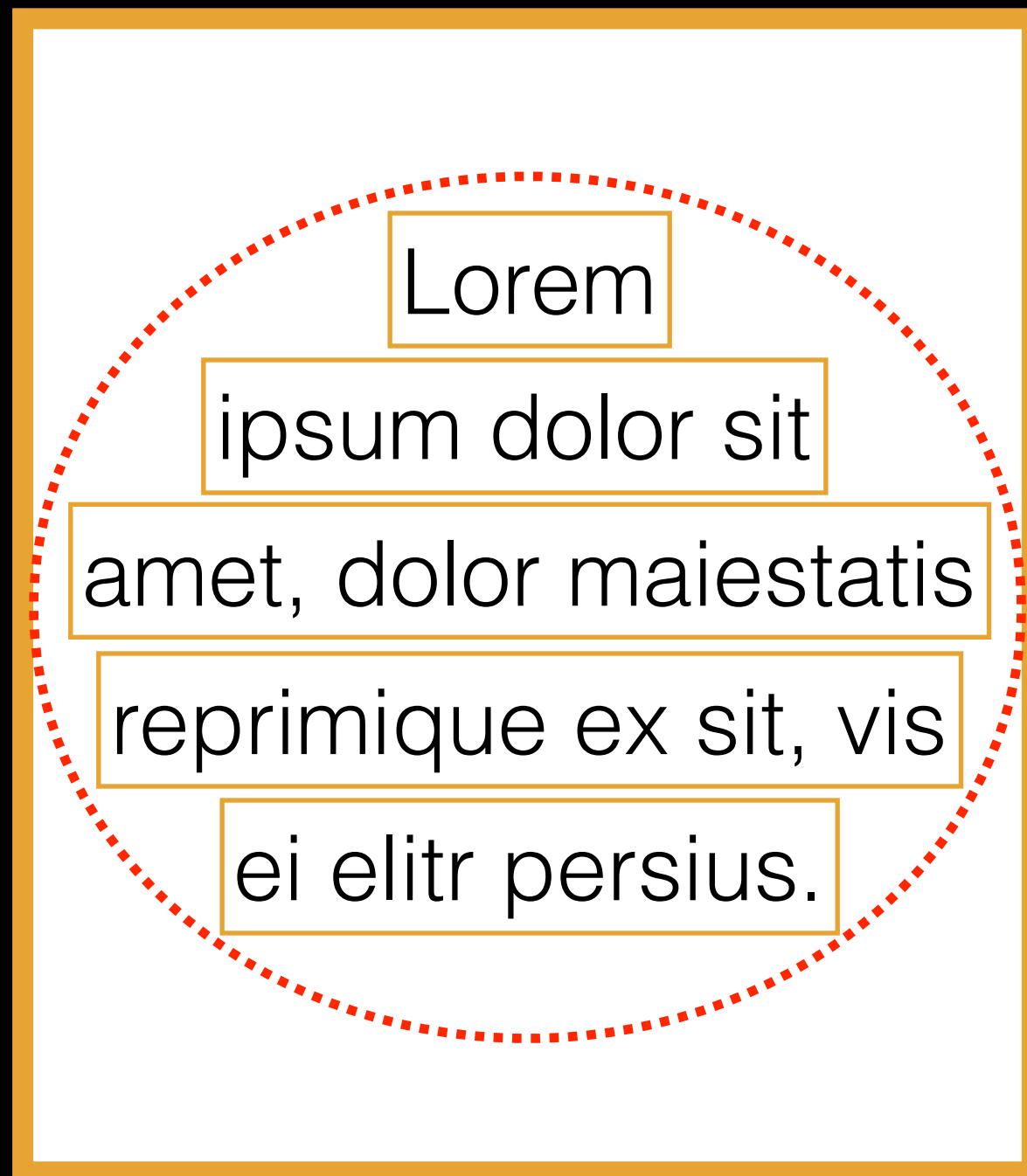
- Overriding how text or attributes are:
  - *stored (custom editing)*
  - *displayed (syntax highlighting)*

```
-(void)viewDidLoad {  
    [super viewDidLoad];  
    [[NSNotificationCenter defaultCenter] addObserver:self selector:@selector  
        (statusRequest:) name:BBI_NOTIF_STATUS_REQUEST object:nil];  
}
```

# Subclassing NSTextContainer

- Store implementation-specific information
  - Indexes, tags, coordinates
- Implementing complex drawing paths
  - Eg. “inclusion paths”

# Subclassing NSTextContainer



# Subclassing NSLayoutManager

- Glyph Substitution
- Custom line truncation



# NSLayoutManagerDelegate

- Allows customized glyph mapping

Password: ●●●●●●●●b

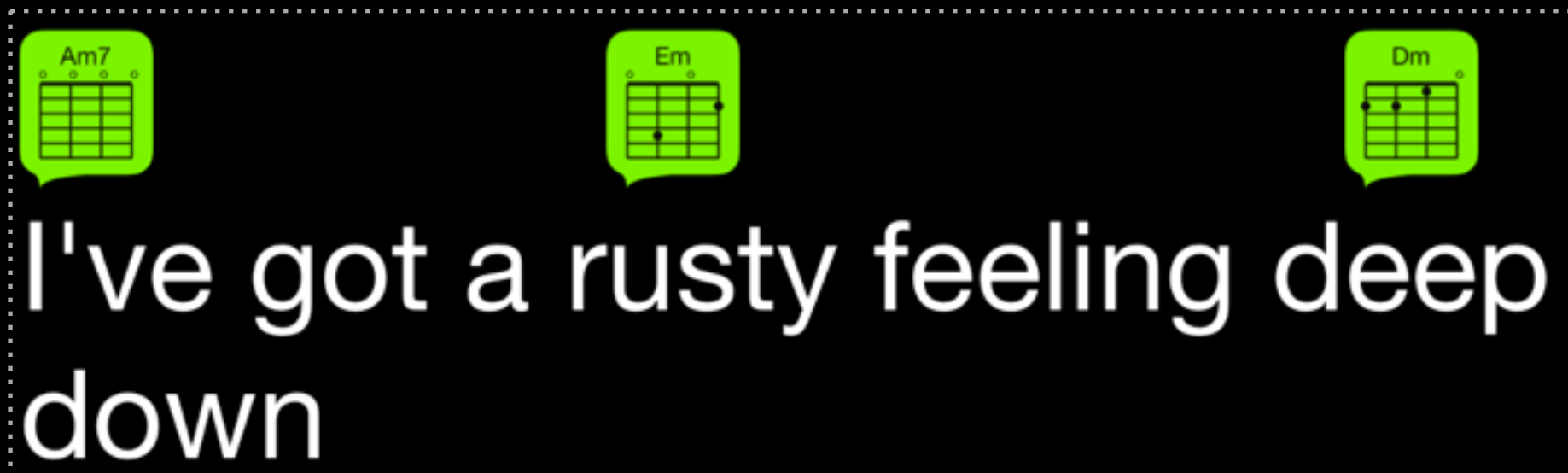
- Allows customization of layout behavior before layout
- Line/paragraph spacing, breaking, truncation

```
NSLog(NSStringFromRange(range));
```

⚠ Format string is not a string literal (potentially insecure)

# NSLayoutManagerDelegate

- Notifies delegate objects when layout operations are complete
- Allows delegate to position other UI elements based on text geometry



# Working With Text Kit



Demo

# Business Use Case

- Newstand Music Magazine App
  - Multi-column flexible text layout
  - Embedded images and audio
  - User interaction
- Editor App
  - Single column
  - Allows writer to embed audio content

# Flexible Layout

# Multiple Text Containers

--	--	--

# Exclusion Path(s)





# Rich Text Editing

# NSAttributedString Class

**String**

**+**

**NSDictionary of Attributes**

“360 iDev”

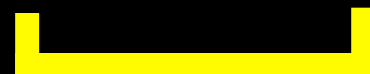
NSForegroundColorAttributeName  
NSParagraphStyleAttributeName  
NSUnderlineStyleAttributeName...

# NSAttributedString Class

```
NSMutableDictionary *attributes =  
@{NSForegroundColorAttributeName: [UIColor  
greenColor]}
```

```
[self.textStorage setAttributes:attributes  
range:NSMakeRange(5, 4)];
```

360 iDev



# Custom Attributes!

```
attributes["@AudioFile"] = @"song.mp4";
```

Interactivity

# Hit Testing

```
– (NSUInteger)characterIndexForPoint:  
  (CGPoint)point inTextContainer:(NSTextContainer  
  *)container  
fractionOfDistanceBetweenInsertionPoints:  
  (CGFloat *)partialFraction;
```



3323, "B"

# Dynamic Layout

# NSLayoutManagerDelegate

```
– (CGFloat)layoutManager:(NSLayoutManager  
*)layoutManager lineSpacingAfterGlyphAtIndex:  
(NSUInteger)glyphIndex  
withProposedLineFragmentRect:(CGRect)rect
```

- Called **before** layout is performed
- Watch out for circular calls



# NSLayoutManagerDelegate

```
-(void)layoutManager:(NSLayoutManager  
*)layoutManager  
didCompleteLayoutForTextContainer:  
(ColumnTextContainer *)textContainer atEnd:  
(BOOL)layoutFinishedFlag
```

- Called **after** layout is performed\*

\*Theoretically

# Beware of UITextView!

- NonContiguousLayout
  - High-Performance for most applications
  - Not reliable when UI elements placements depends on text layout
- NSLayoutManagerDelegate methods called multiple times

# Custom Input Behavior

# Subclassing NSTextStorage

- Class Cluster
- Subclass must implement:

`string`

`attributesAtIndex:effectiveRange:`

`replaceCharactersInRange:withString:`

`setAttributes:range:`

# Subclassing NSTextStorage

- `(void)processEditing;`
- Override to process text changes post-input
- Must call super!

# Thanks!

@gene\_whitaker  
@bombingbrain

[gene@bombingbrain.com](mailto:gene@bombingbrain.com)

# Resources

## Videos

### UIKonf 2014

Max Seelemann: TextKit for the rest of us

### WWDC 2013

210 Introducing Text Kit

220 Advanced Text Layouts and Effects with Text Kit

223 Using Fonts with Text Kit

## Awesome Articles

Andrew Hulsizer (Swift Yeti): TextEffects with NSLayoutManager

Colin Eberhardt, Gabriel Hauber ([raywenderlich.com](http://raywenderlich.com)): Text Kit Tutorial