

MAX SEELEMANN

HALLO TEXTKIT!

GESCHICHTLICHER ABRISS

TEXT AUF IOS

# HISTORIE

## IOS 2

- Einfache Text-Anzeige: `UILabel`
- Einfache Text-Eingabe: `UITextField`
- Eingabe für längere Texte: `UITextView`
- Keine Formatierungen, Text-Auswahl, Farben

# HISTORIE

## IOS 3

- Text-Selektion
- Kopieren, Ausschneiden und Einsetzen
- Data Detectors: Telefonnummern, Links

# HISTORIE

## IOS 3.2

- CoreText — low-level Textsatz-Framework
  - portiert von Mac OS X 10.5
- Abfangen der Tastatureingabe: **UITextInput**-Protokoll

# HISTORIE

## IOS 4

- Nichts Neues
- WWDC-Anekdote: "yeah... File a radar."

# HISTORIE

## IOS 5

- Wieder nichts Neues
- WWDC-Anekdote: "we don't see a lot of requests..."

# HISTORIE

## IOS 6

- Formatierter Text in `UITextView`
  - Fett, Kursiv, Unterstrichen, Groß, Klein
  - Kaum anpassbar
- System-eigene Text-Selektion für Subklassen von `UITextView`



HISTORIE

IOS 7

HALLO TEXTKIT!

FEATURES, FEATURES, FEATURES

FEATURES

# FEATURES

## KERNING

- Text-Unterschneidung: bessere Lesbarkeit

Taten

mit Kerning

Taten

ohne Kerning

# FEATURES

## LIGATUREN

- Spezielle Zeichen für kombinierte Buchstaben

finden

mit Ligaturen

finden

ohne Ligaturen

# FEATURES

## BILDER

- Im Text eingebettete Bilder
- Können vom Nutzer selektiert und bearbeitet werden

# FEATURES

## SILBENTRENNUNG

- Automatischer Umbruch: besseres Schriftbild

Denn er war ein  
phlegmatischer Junge und  
hatte keinerlei Abenteuer  
und kühne  
Unternehmungen. Während  
Tom nun seine Suppe aß  
und nach Möglichkeit  
Zuckerstückchen stahl,

Denn er war ein phlegmatis-  
cher Junge und hatte keiner-  
lei Abenteuer und kühne Un-  
ternehmungen. Während  
Tom nun seine Suppe aß  
und nach Möglichkeit Zuck-  
erstückchen stahl, stellte  
Tante Polly allerhand Fragen

# FEATURES

## ANPASSBARKEIT

- Komponenten-basiertes System
- Delegates und Subklassen erlauben weite Anpassung

# FEATURES

## NEUE TEXTATTRIBUTE

- Unterstreichungs- und Durchstreichungsstile:
  - Doppelt, Dick, Gestrichelt, Gepunktet
- Text-Grundlinie verschieben
- Hintergrundfarben



# FEATURES

## UND SONST NOCH...

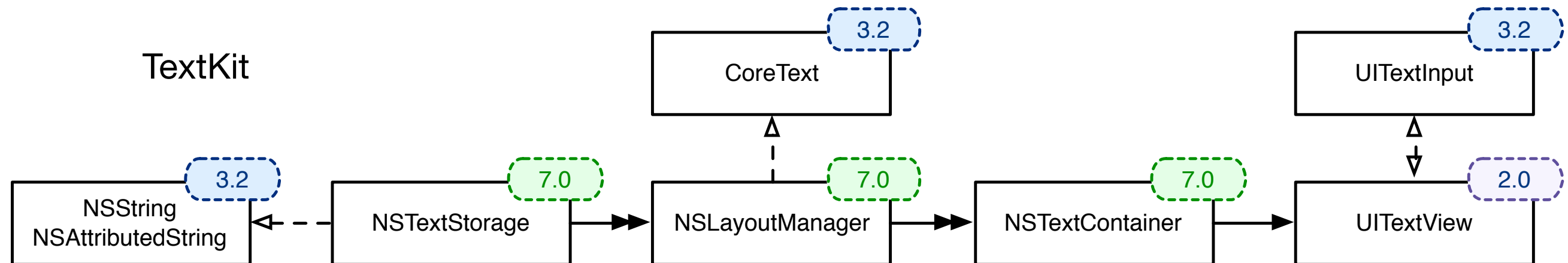
- Serialisierung von Text mit Attributen
- Textstile: systemweit einheitliche Schriften
  - Einstellbar vom Nutzer (Größe, Fett)
- Texteffekte. Ohne Worte.

STRUKTUR, ZUSAMMENSETZUNG, SCHEMA

AUFBAU

# TEXTKIT

## KLASSENÜBERSICHT



# AUFBAU

## LAYOUT-ABLAUF

- 1 `NSLayoutManager` überwacht `NSTextStorage`
- 2 Zeichen in Glyphen übersetzen
- 3 `NSTextContainer` bestimmt verfügbaren Bereich
- 4 Bereiche mit Zeilen füllen
- 5 Zeilen mit Zeichen füllen: Umbruch, Trennung
- 6 `UITextView` invalidieren, `NSLayoutManager` zeichnet

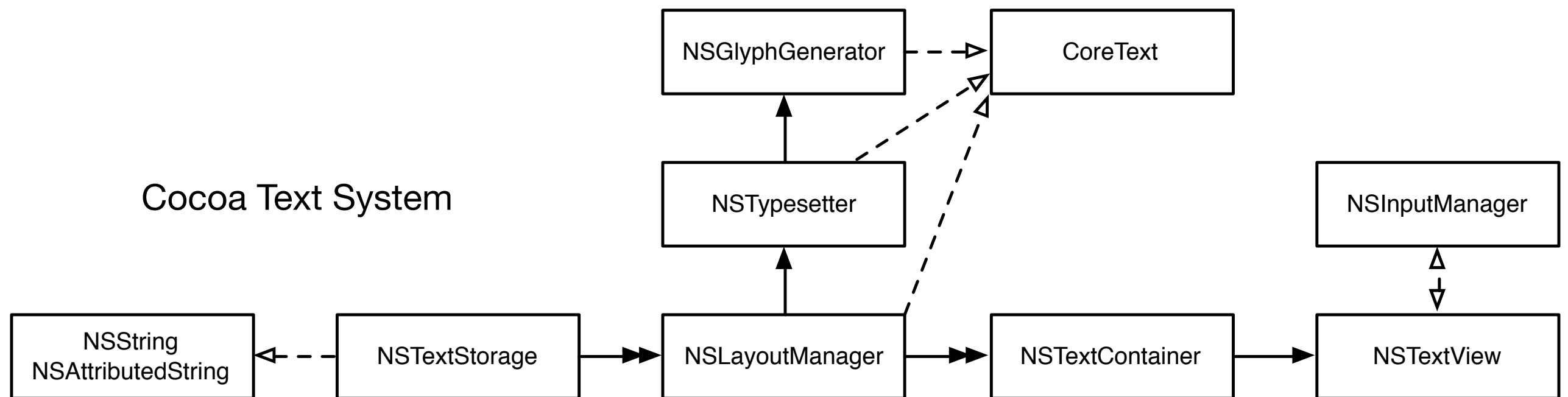
ALLES GANZ NEU?

VERGLEICH ZU OS X

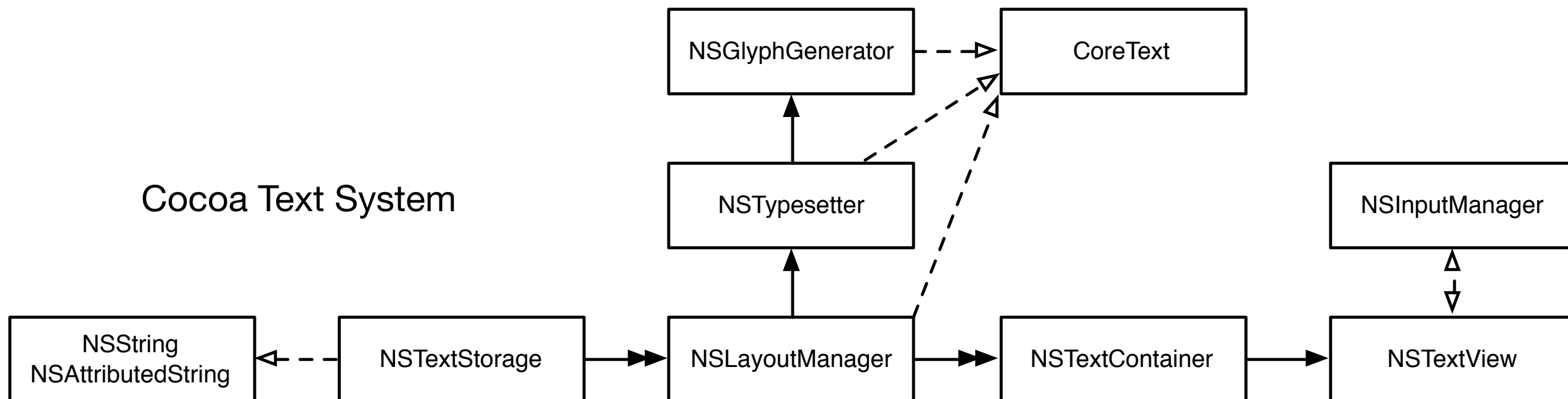
# COCOA TEXT SYSTEM

## KLASSENÜBERSICHT

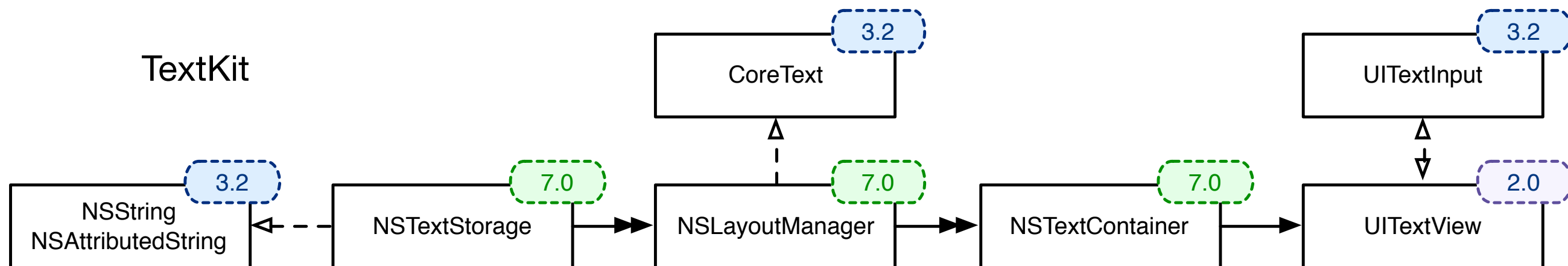
Cocoa Text System



## Cocoa Text System



## TextKit



SYNTAX HIGHLIGHTING

BEISPIEL



# BEISPIEL

## HERANGEHEN

- Farben bestimmt NSTextStorage > ableiten

```
- (NSString *)string;  
- (NSDictionary *)attributesAtIndex:(NSUInteger)location effectiveRange:(NSRange *)range;  
  
- (void)replaceCharactersInRange:(NSRange)range withString:(NSString *)str;  
- (void)setAttributes:(NSDictionary *)attrs range:(NSRange)range;
```

# BEISPIEL

## NSTEXTSTORAGE ABLEITEN

```
@implementation TKDHighlightingTextStorage
{
    NSMutableAttributedString *_imp;
}

- (id)init
{
    self = [super init];

    if (self) {
        _imp = [NSMutableAttributedString new];
    }

    return self;
}
```

# BEISPIEL

## GETTER-METHODEN

```
- (NSString *)string
{
    return _imp.string;
}

- (NSDictionary *)attributesAtIndex:(NSUInteger)location effectiveRange:(NSRange *)range
{
    return [_imp attributesAtIndex:location effectiveRange:range];
}
```

# BEISPIEL

## SETTER-METHODEN

```
- (void)replaceCharactersInRange:(NSRange)range withString:(NSString *)str
{
    [_imp replaceCharactersInRange:range withString:str];
    [self edited:NSTextStorageEditedCharacters range:range
              changeInLength:(NSInteger)str.length - (NSInteger)range.length];
}

- (void)setAttributes:(NSDictionary *)attrs range:(NSRange)range
{
    [_imp setAttributes:attrs range:range];
    [self edited:NSTextStorageEditedAttributes range:range changeInLength:0];
}
```

# BEISPIEL

## TEXTSTORAGE INSTALLIEREN

```
_textStorage = [TKDHighlightingTextStorage new];  
[_textStorage addLayoutManager: self.textView.layoutManager];
```

# BEISPIEL

## "SYNTAX" FESTLEGEN

```
- (void)processEditing
{
    static NSRegularExpression *iExpression;
    iExpression = iExpression ? [NSRegularExpression regularExpressionWithPattern:
        @"i[\\p{Alphabetic}&\\p{Uppercase}][\\p{Alphabetic}]+" options:0 error:NULL];
}
```

# BEISPIEL

## "SYNTAX" ERKENNEN

```
NSRange paragraphRange = [self.string paragraphRangeForRange: self.editedRange];  
[self removeAttribute:NSForegroundColorAttributeName range:paragraphRange];  
  
[iExpression enumerateMatchesInString:self.string options:0 range:paragraphRange  
    usingBlock:^(NSTextCheckingResult *result, NSMatchingFlags flags, BOOL *stop)  
{  
    [self addAttribute:NSForegroundColorAttributeName  
        value:[UIColor redColor]  
        range:result.range];  
}];
```

DEMO! DEMO! DEMO!

DEMO? DEMO? DEMO?



[HTTPS://GITHUB.COM/MACGURU/TEXTKITDEMO](https://github.com/macguru/textkitdemo)

DANKE!