

Macoun







Rückblick auf iOS 9 und 10

Joachim Kurz, XING SE
(@cocoafrog)






Enthalten / Nicht Enthalten

- ✓ Neue und geänderte iOS APIs
- ✗ macOS, watchOS, tvOS
- ✗ Xcode Neuerungen und anderes das unabhängig von iOS ist (z.B. CloudKit JS)
- ✗ Swift
- ✗ Metal
- ✗ Dinge, die man mit dem neuem System unterstützen **muss**
- Mehr iOS 10 als iOS 9

Neu und Unabhängig

- Universal Links 
- Safari Content Blocker 
- CallKit 
- SiriKit 
- iMessage Apps 
- Core Spotlight, Audio Unit Extension, Network Extension 

Neu, Nützlich, Benutzt es!

- `SFSafariViewController` 
- `CoreLocation`: Einmaliges Location Update 
- Objective-C Contacts Framework ersetzt C AddressBook Framework 
- `CoreImage`, `AVFoundation`, `Photos` unterstützen alle Live Photos 
- `UIRefreshControl` auf allen `UIScrollView`s 

Neuerungen in iOS





AutoLayout: API

- Viel lesbarer:

```
saveButton.trailingAnchor.constraintEqualToAnchor(space1.leadingAnchor)
```

- Type-Checking
 - Kann nur Layout-Anker des selben Typs kombinieren
 - NSLayoutConstraint, NSLayoutConstraint, NSLayoutConstraint

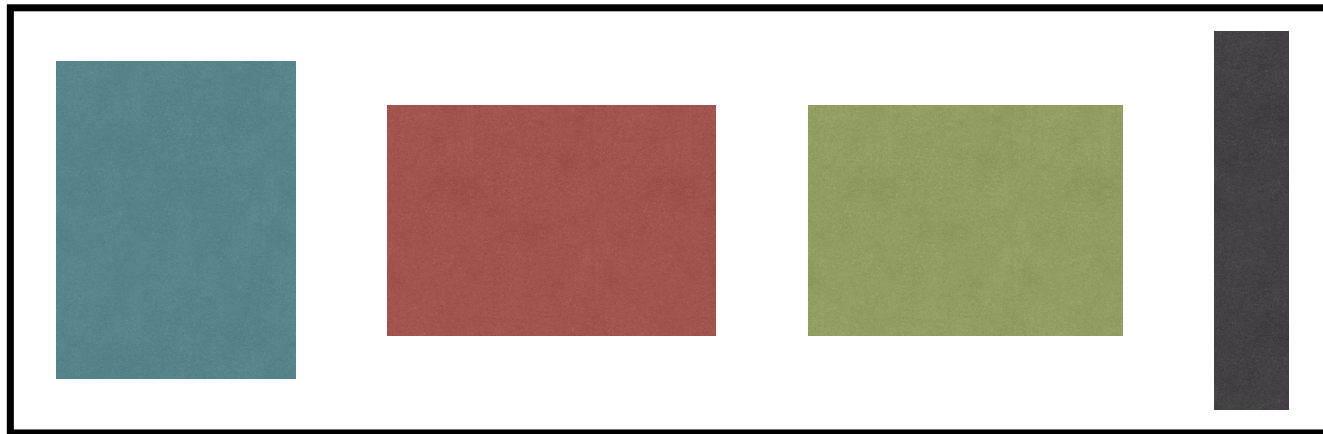


UILayoutGuide

- Nimmt an AutoLayout teil ohne ein UIView zu sein
 - Kein Mehraufwand für den View, z.B. Hit-Testing, Drawing, etc.
- Kann andere Views enthalten oder Constraints zu ihnen aufbauen.
 - Keine Platzhalter-Views mehr!

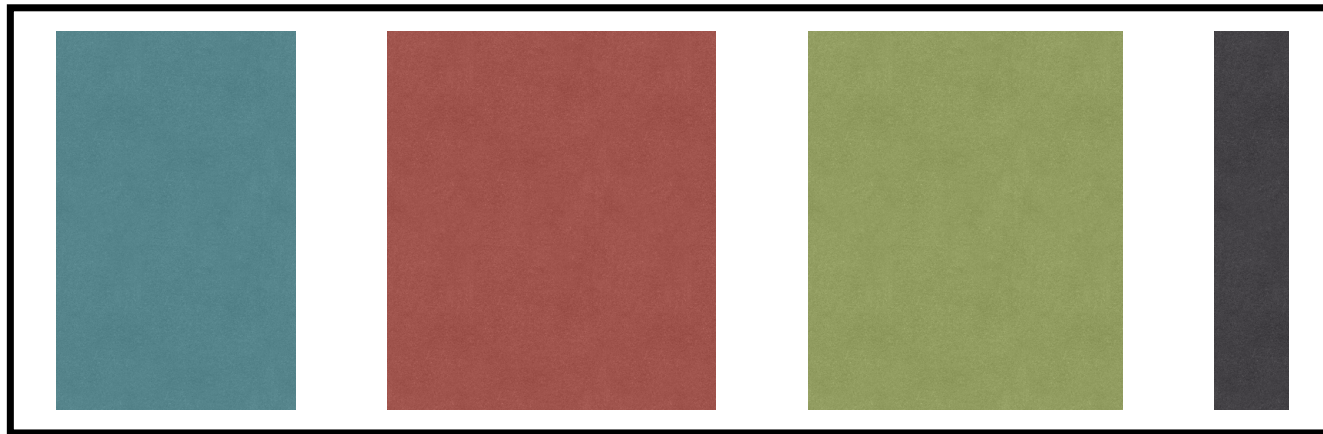


UIStackView



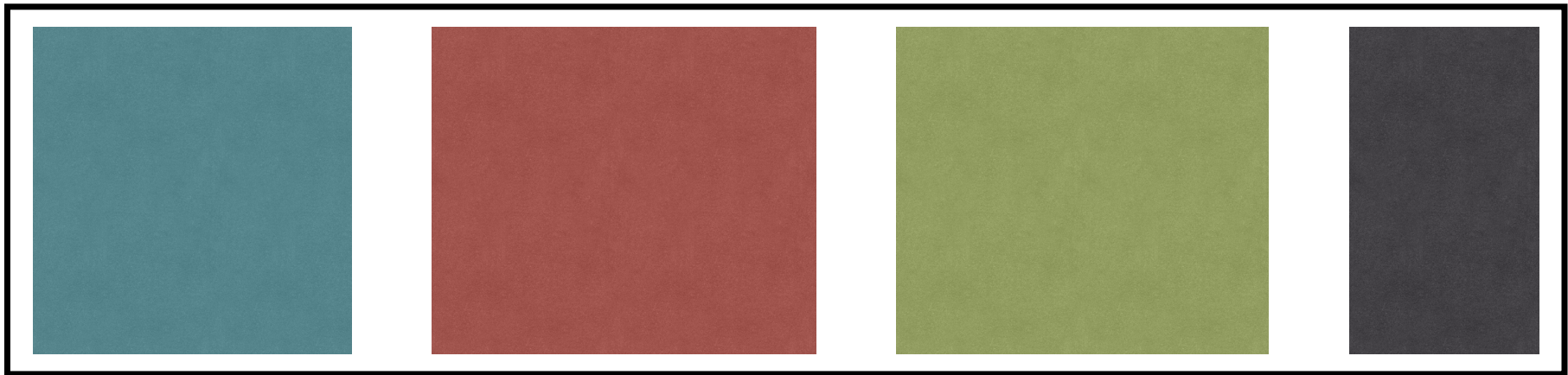


UIStackView



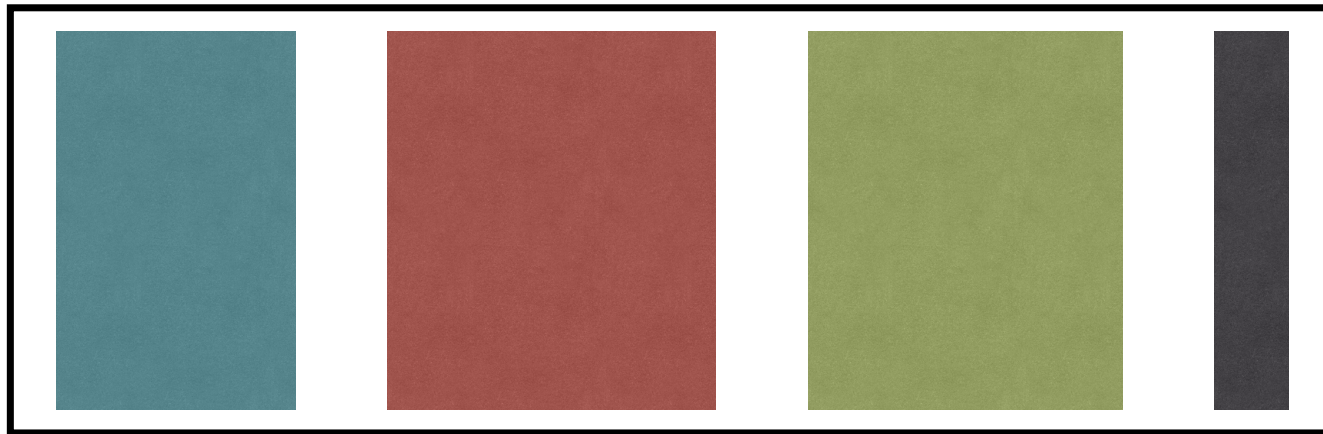


UIStackView



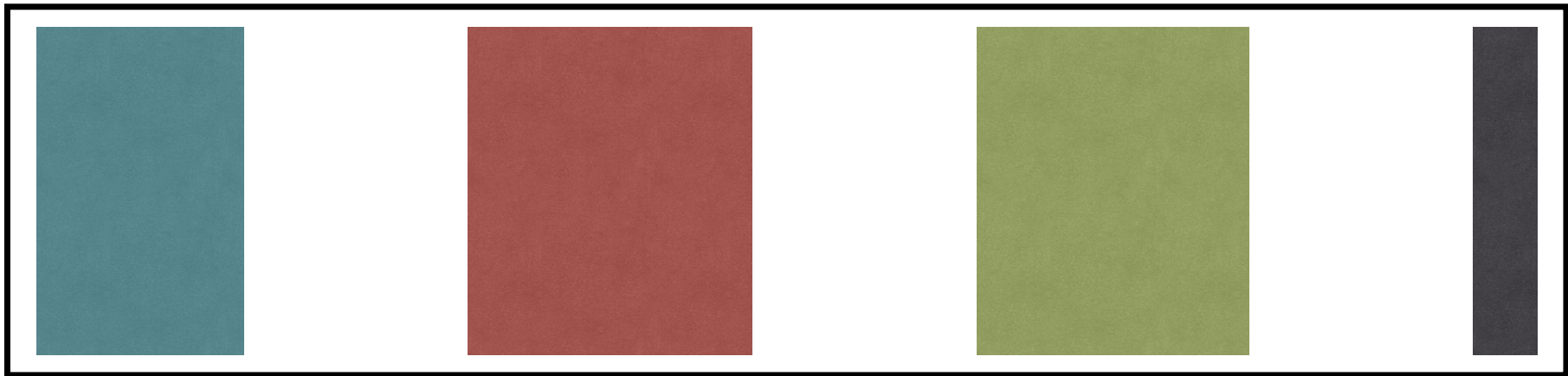


UIStackView



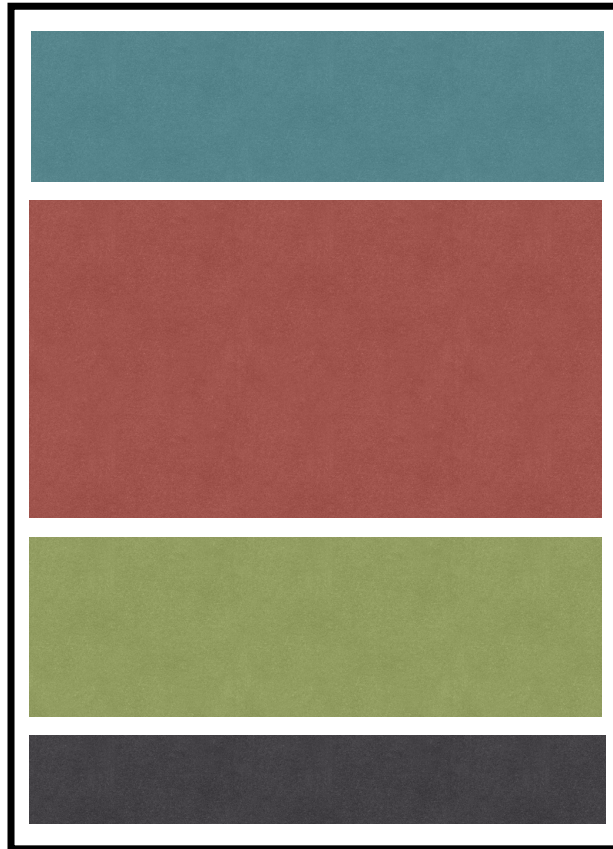


UIStackView





UIStackView





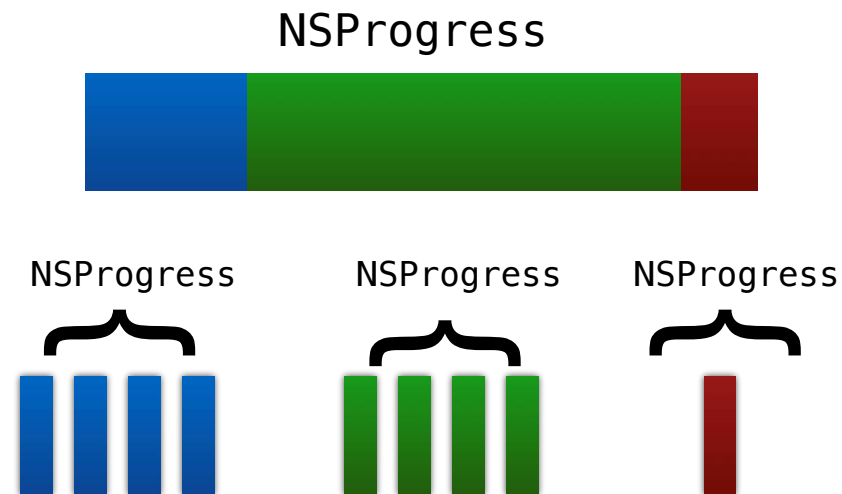
Internationalisierung

- Unterstützung von Sprachen die von rechts nach links laufen
- `NSNumberFormatter:OrdinalStyle` und weitere `CurrencyStyles`
- `NSPersonNameComponentsFormatter`
- `CNContactFormatter`
- `CNPostalAddressFormatter`
- `–[NSString variantFittingPresentationWidth]`



NSProgress

- `UIProgressView` kann jetzt ein `NSProgress` object haben





Außerdem noch...

- Neues UIControlEvent: `UIControlEventPrimaryActionTriggered`
- CoreData: `destroyPersistentStoreAtURL:`
- Neuer Kompressionsalgorithmus: `lzsf`
- `+[NSError setUserInfoValueProviderForDomain: provider:]`
- MapKit kann nun 3D flyover, Verkehr und ÖPNV Routendauerberechnung
- String Transformationen, z.B. kyrillische nach lateinische Schrift



NSUserActivity

„Lightweight way to capture app state so that it can be restored later“

— NSUserActivityDocumentation



NSUserActivity



- Activities können in Spotlight-Suchergebnissen auftauchen
- Metadaten um Aktivität vollständig zu beschreiben
- Kann in Siri-Vorschlägen auftauchen
- `expirationDate`

NSUserActivity

- Kann Orte/Adressen enthalten
 - `mapItem`
- Proaktive Vorschläge in Maps
- Bessere Autovervollständigung in Textfeldern
 - `UITextField/UITextView` können mit erwartetem Datentyp konfiguriert werden
 - `URL, addressCity, emailAddress, familyName, userName`



UIKit: 3DTouch

- Peek & Pop
- UIPreviewInteraction 
 - Erzeugt das selbe Gefühl wie Peek & Pop, aber mit eigener UI
- UIFeedbackGenerator, für haptisches Feedback 
 - Impact, Selection, Notification



Games

- Model I/O
- MetalKit
- ReplayKit
- Metal Performance Shaders
- GamePlayKit
- GameKit: kombiniert alle games-bezogenen Frameworks




Games

- Model I/O
- MetalKit
- ReplayKit
- Metal Performance Shaders
- GamePlayKit
- GameKit: kombiniert alle games-bezogenen Frameworks






ReplayKit

- Spiel aufnehmen (Video)
- Video beschneiden
- Begleitkommentare
- Eingebautes Sharing und Speichern
- Live-Streaming 



Metal Performance Shaders

- Häufig benutzte Image Processing Kernel für Effekte auf Metal Textures
 - Histogram, Lanczos Resampling, Thresholding, Gaussian Blur
- Convolutional Neural Networks 
- Farbkonvertierungen 
- Matrixmultiplikation 



GamePlayKit

- State Machines
- Agenten
- Wegfindung
- Zufallsquellen
- Regelsysteme
- Künstliche Intelligenz (für Gegner)




GamePlayKit

- State Machines
- Agenten
- Wegfindung
- Zufallsquellen
- Regelsysteme
- Künstliche Intelligenz (für Gegner)






GamePlayKit: Agenten

- Autonomes „Wesen“, das von Zielen gesteuert wird
 - Suchen, Abfangen, Ausweichen, Ausrichten, einem Pfad folgen
- Realistische Bewegungen
 - Trägheit, Beschleunigung, etc.
- Auch in 3D 


GamePlayKit: Künstliche Intelligenz



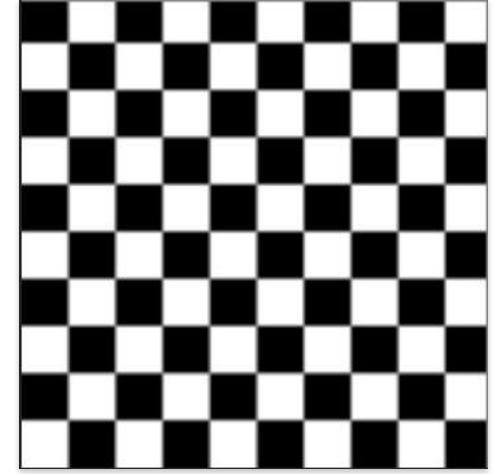
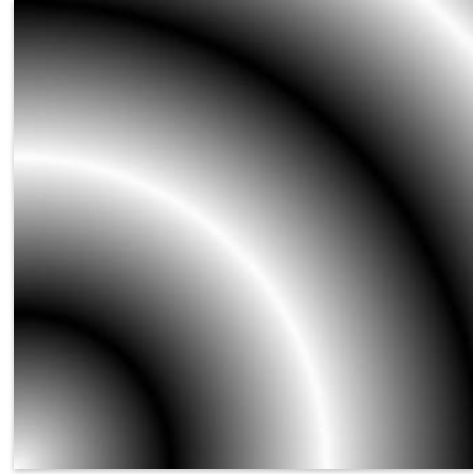
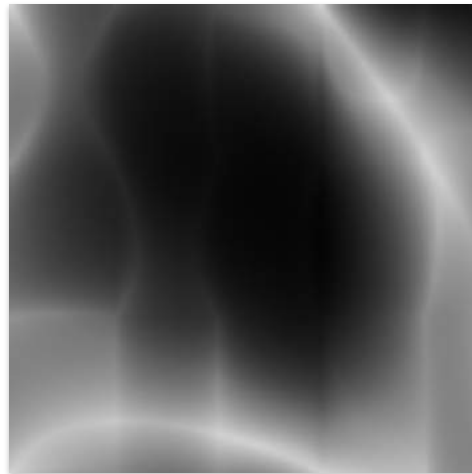
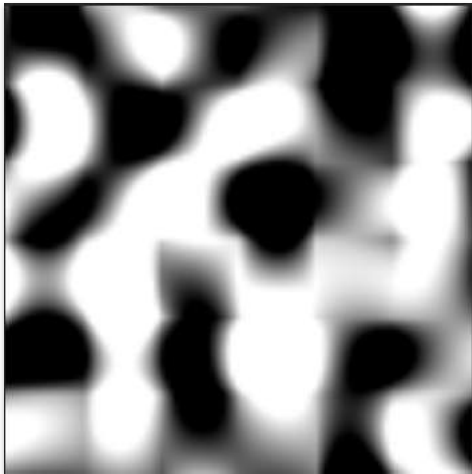
- Techniken um Intelligenz im Spiel zu simulieren
- Min-Max-Strategie
- Monte-Carlo-Strategie 
- Eigene Strategien implementieren 
- Entscheidungsbaum 
 - Manuell oder erlernt



GamePlayKit: Zufallsquellen

- Einfache Zufallsquellen
 - Deterministisch, Serialisierbar, Industriestandard
- Verteilungen
 - Echter Zufalls, Gauss, Anti-clustering
- Komfortfunktionen für Würfelzufallsfunktionen (W6, W8, W20, etc.)
- Rauschfunktionen 

GamePlayKit: Rauschfunktionen



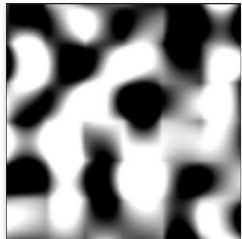
GamePlayKit: Rauschfunktionen

Kohärent

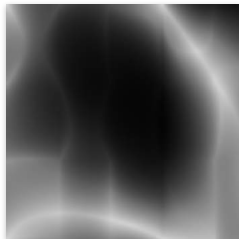
Geometrisch

Konstant

Perlin



Voronoi

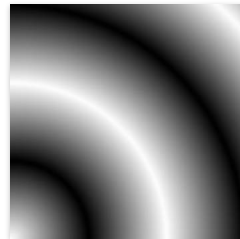


Ridges

Billows

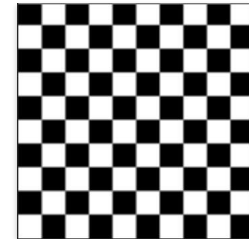
Spheres

Cylinders



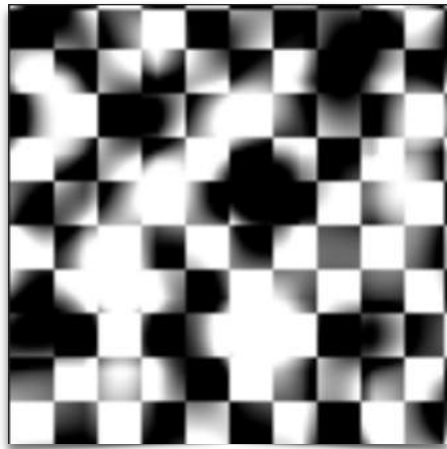
Constant

Checkerboard



GamePlayKit: Rauschfunktionen

- Kombinieren: Addition, Multiplikation, Min, Max, a^b , Verschieben
- Transformieren: Skalieren, Rotation, Verschiebung, Terrassierung, curve-map
- Modifikation: Betrag, eingrenzen, Turbulenz, Invertierung





GameCenter: GKGameSession

- Generalisierten Interface um Daten zu speichern und zu teilen
- Basiert auf CloudKit
- Mit jedem iCloud User spielen
- Turn-based: 1-100 (vorher: 2-16)
 - kein starres Konzept eines Zuges mehr
- Real-time: 1-16 (vorher: 2-4)



SpriteKit + SceneKit

- SpriteKit
 - TileMaps
 - Warp Transformation
- SceneKit
 - Physically-Based-Rendering

Neuerungen in iOS

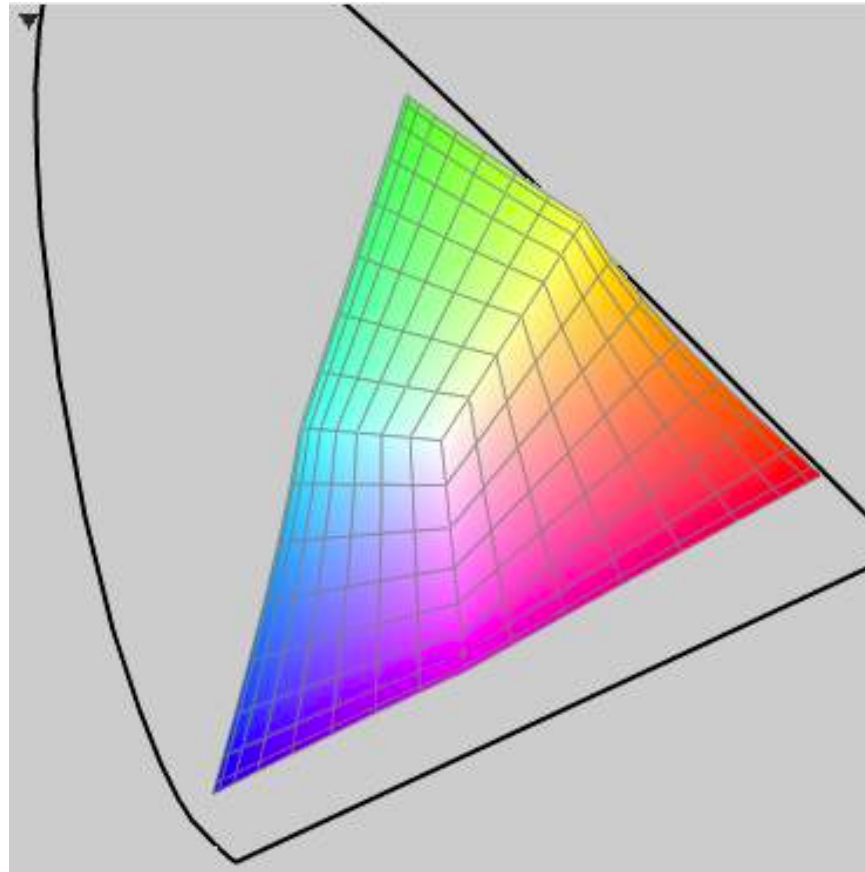


Color

Wide Color

Wide Color

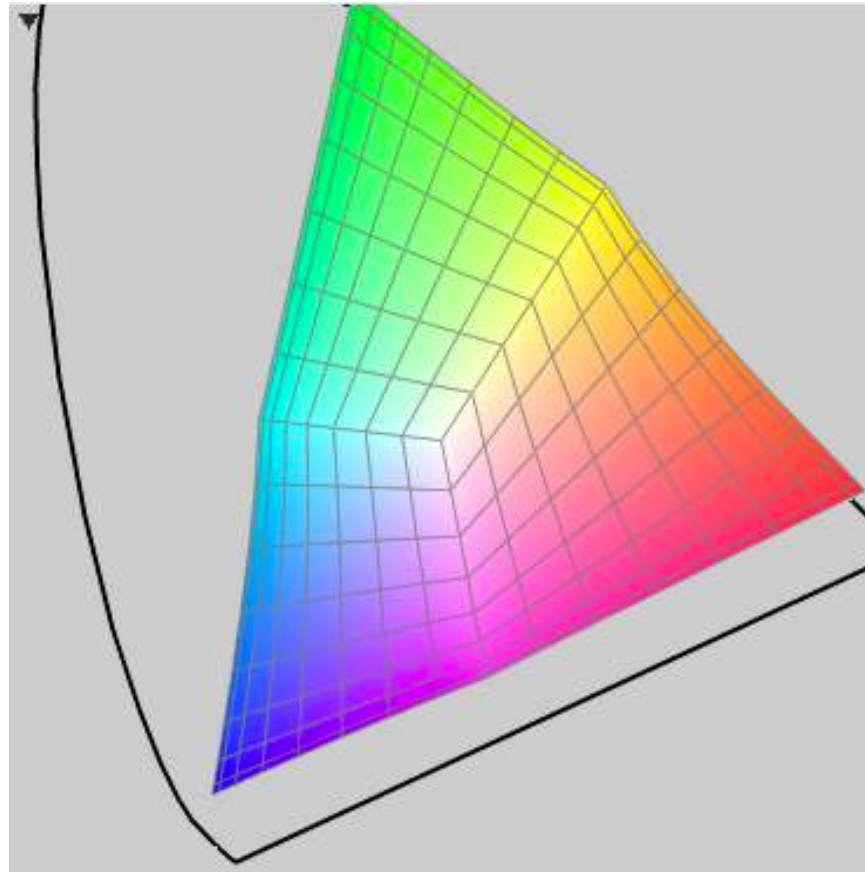
sRGB



nicht-Pro iPads
iPhone ≤ 7 (Plus)
iPod Touch

Wide Color

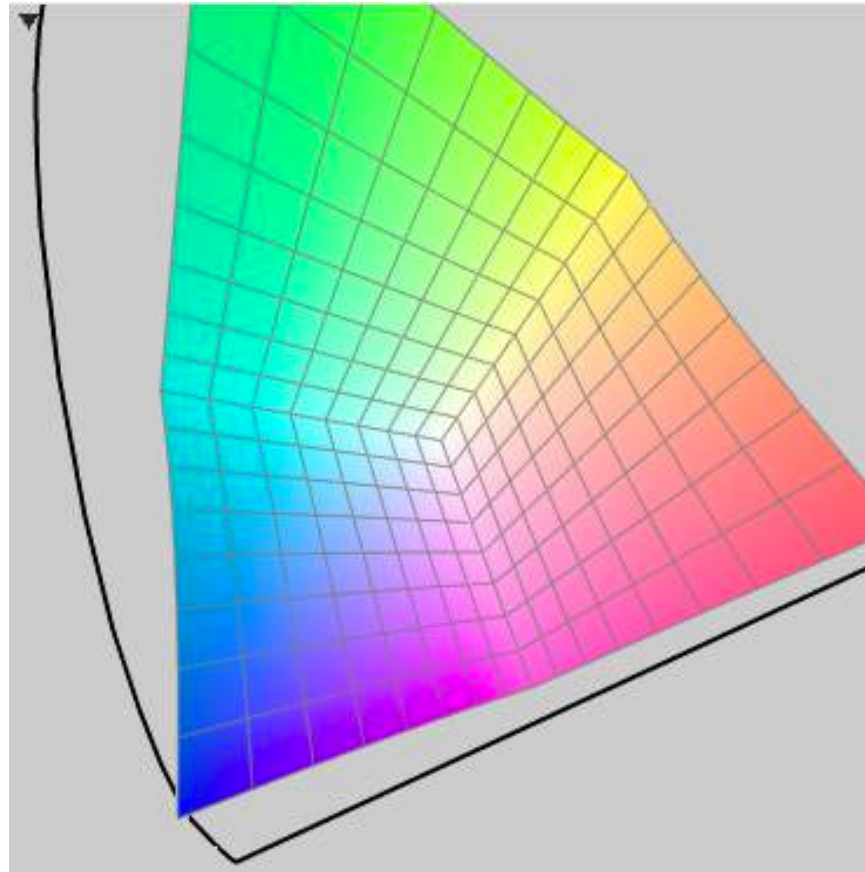
Display P3



iPad Pros
iPhone 8 (Plus)
iPhone X

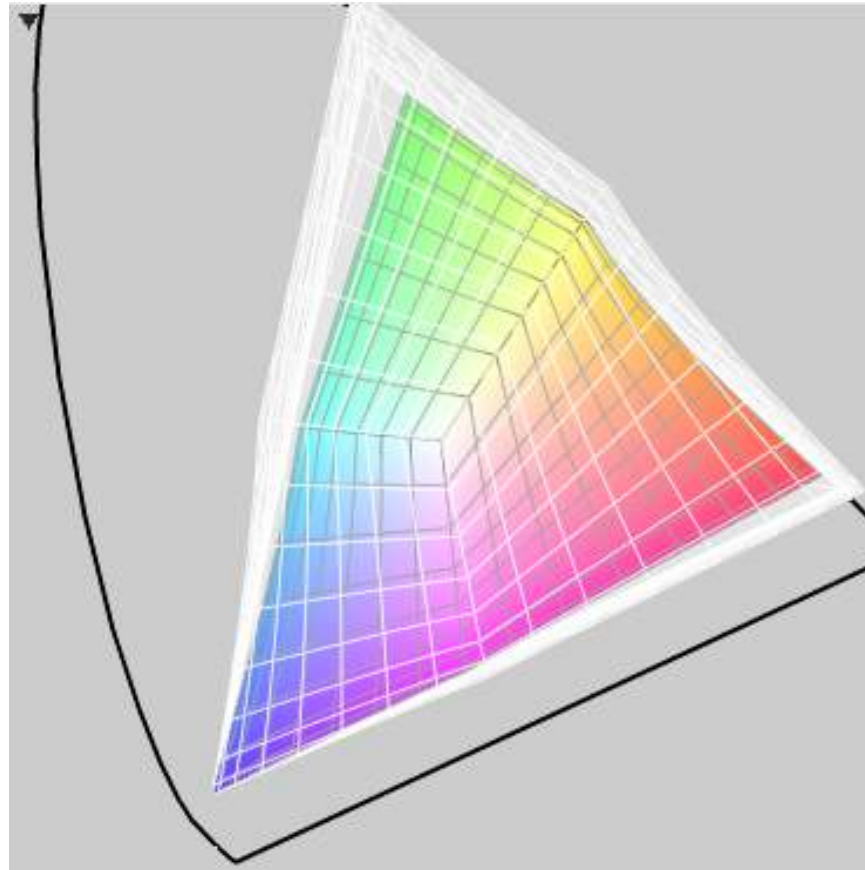
Wide Color

ACES CG Linear



Wide Color

sRGB
vs.
Display P3



Testet selbst unter: <https://webkit.org/blog-files/color-gamut/comparison.html>

Wide Color

sRGB



100%

≈

85%



Display P3



Wide Color



Wide Color

8 Bit (exemplarisch)

sRGB



8 Bit (exemplarisch)?

Display P3



Wide Color

8 Bit (exemplarisch)

sRGB



8 Bit (exemplarisch)?

Display P3



Wide & Deep Color

8 Bit (exemplarisch)

sRGB



16 Bit! (exemplarisch)

Display P3



Wide & Deep Color

- Werte 0-255/0x0-0xFF sind nicht mehr ausreichend
 - Nutzt stattdessen $[0...1]$, kompatibel mit 8bit und 16bit
- Farbwert ohne Angabe eines Color Space ist bedeutungslos
- Extended sRGB hat Farbwerte <0.0 und >1.0
- Passt eure Kommunikation mit den Designern daran an!



Wide & Deep Color API

- `UIColor(red:green:blue:alpha)` akzeptiert Werte <0.0 und >1.0
- `UIColor(displayP3Red:green:blue:alpha:)`
- Alle (?) APIs unterstützen Wide Color. Photos, UIImage, CoreImage, Metal, SpriteKit, SceneKit, CoreGraphics, etc.
- Neue block-basierte `UIGraphicsRenderer` Klasse für Bilder und PDFs statt `UIGraphicsBeginImageContext()`



UIKit News

- `UICollectionView` unterstützt automatisches schätzen der Zellgröße
- Prefetching, implizit und mit Delegate
- `TabBar`: Eigene Badge-Farbe, eigene Farbe für nicht gewählt Tabs
- Traits: `WideGamut` und `LayoutDirection`
 - Kann geladenen Bildern traits zuweisen
- Neue `UIDataDetectors`: Flugnummer, Pakettrackingnummer, Lookup Vorschlag

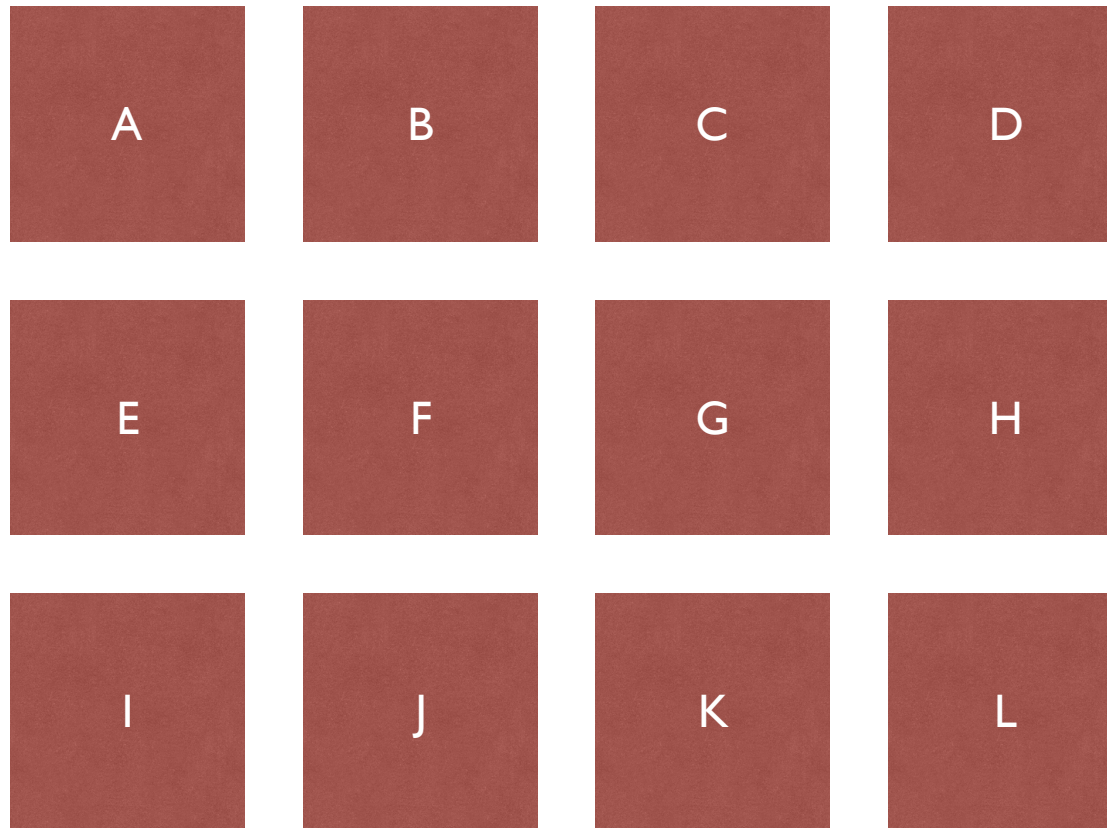


UIKit: Property Animator

- Unterbrechbar, scrubbable, umkehrbar
- Gut anpassbares Timing: `UITimingCurveProvider`
- Properties unterstützen KVO: `running`, `reversed`, `fractionComplete`
- Builder-Pattern für Animationen
- Animation kann während des Ablaufs verändert werden
- Hit-Test für sich bewegende Views
- Auch `UIViewControllerTransitions`

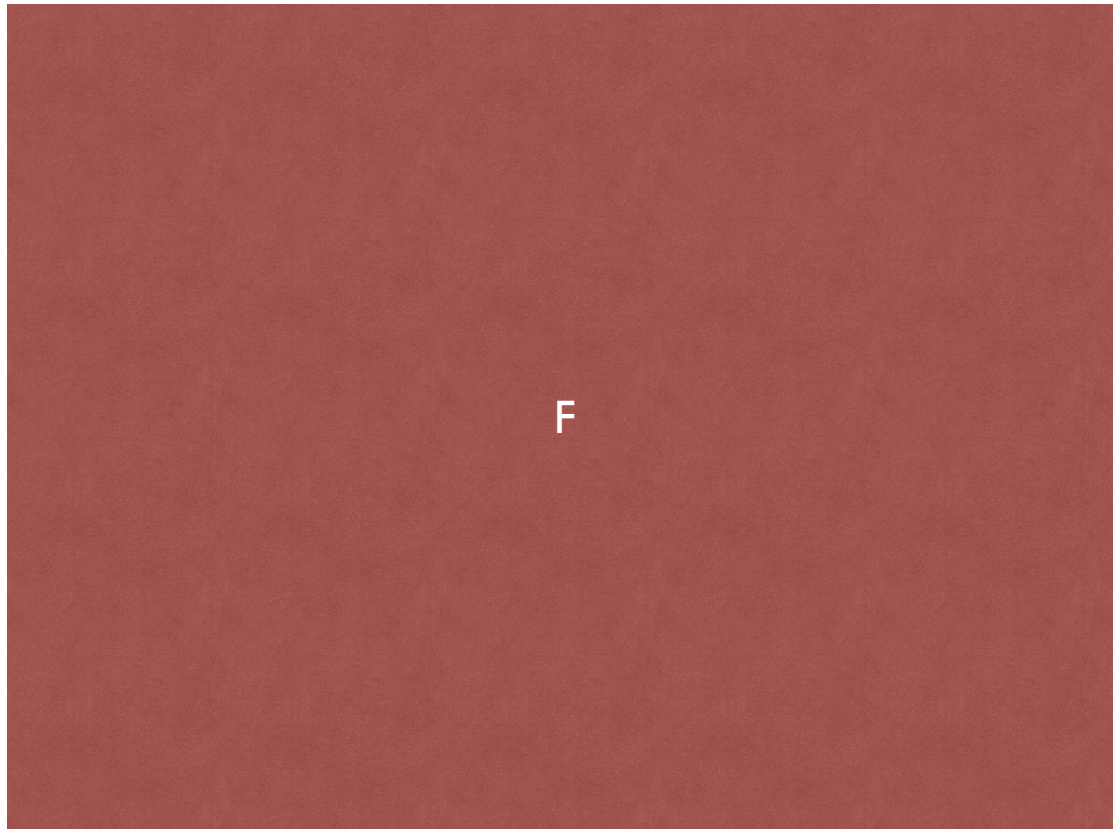


UIKit: Property Animator



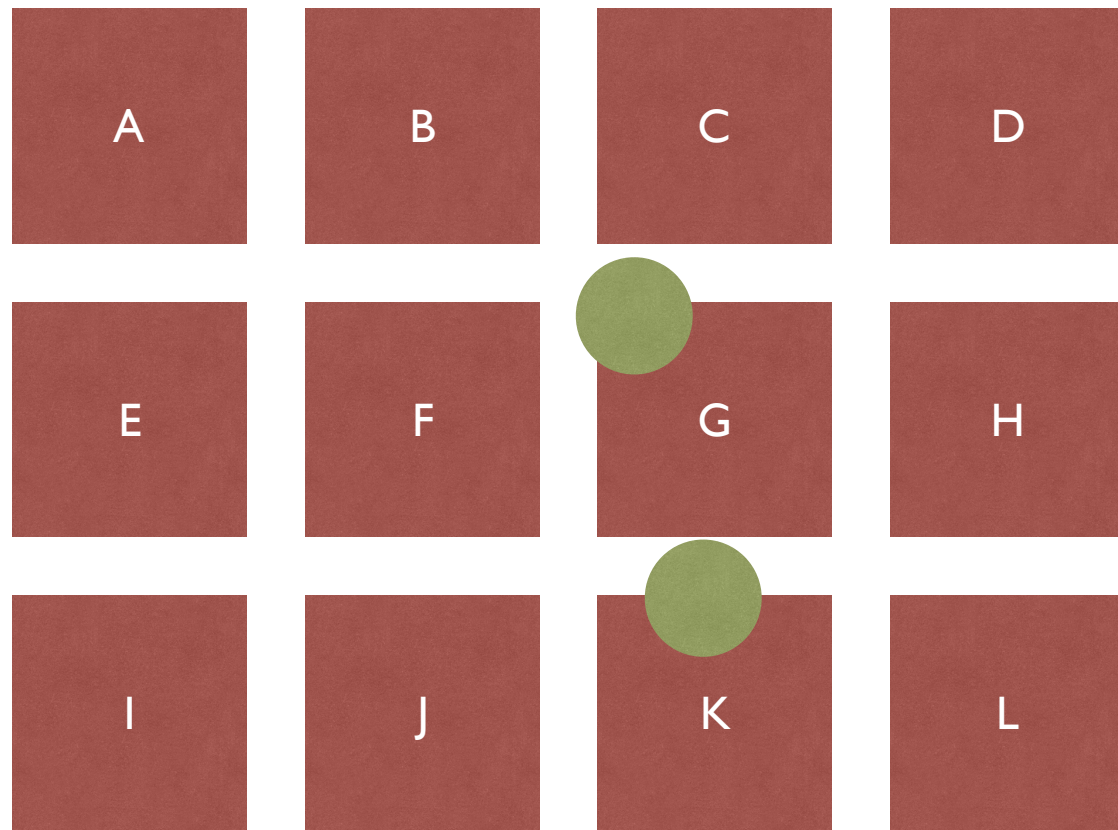


UIKit: Property Animator





UIKit: Property Animator



interaktiv

nicht-interaktiv



Foundation News

- `ISO8601DateFormatter`
- `NSDateInterval`
- `NSLocale` `localizedStringFor`
 - `CalendarIdentifier`, `CollationIdentifier`, `CurrencyCode`, etc.

NSURLSession

- NSURLConnection deprecated seit 2015!
- HTTP/2 Server Push
- Verbindungsstatistiken
 - Verbindungsaufbau, Gesamtdauer, Anzahl Redirects, etc.



Measurements

- value: Double
- Unit (z.B. Kilometer)
 - Symbol („m“, „km“, „ft“, „s“, „Ω“, „m/s²“)
 - Dimension: Gruppierung von zusammengehörenden Einheiten (z.B. Länge)
- Berechnungen, Konvertierungen, Vergleiche, Ranges

100in > 2m?

451°F > 200°C?

1Lichtjahr in km?

Measurements

- value: Double
- Unit (z.B. Kilometer)
 - Symbol („m“, „km“, „ft“, „s“, „Ω“, „m/s²“)
 - Dimension: Gruppierung von zusammengehörenden Einheiten (z.B. Länge)
- Berechnungen, Konvertierungen, Vergleiche, Ranges
- Eigene Einheiten
- MeasurementsFormatter ersetzt viele andere Formatter

true

true

9.461.000.000.000 km



Unified Logging

- Vereinheitlicht Logging API, Performance Verbesserung
- Logging Levels
 - Fault, Error
 - Default (=NSLog), Info, Debug
- Jedes Level kann an/aus sein und nur im Speicher oder on Disk
- Berücksichtigt Privatsphäre der Nutzer: Automatische Anonymisierung von nicht-statischer Information
- Activity API Verbesserungen



UserNotifications Framework

- Vereinheitlichte API für lokale und Remote Notifications
- Strukturierter Inhalt: Titel, Untertitel, **Body**, Media Attachments
 - Image, Video, Audio, GIF?
- Kann wartende oder schon zugestellte Notifications ändern/löschen
- Notifications trotz App im Vordergrund anzeigen
- Extension: Entschlüsselung E2E-verschlüsselter Nachrichten
- Extension: Eigene UI, mit begrenzter Interaktivität

User Notifications: Begrenzte Interaktivität



- Eigener Notification View ist nicht interaktiv
- Aber Input Accessory View ist interaktiv
- View kann aktualisiert werden nachdem Nutzer geantwortet hat



User Notifications: Begrenzte Interaktivität



- Eigener Notification View ist nicht interaktiv
- Aber Input Accessory View ist interaktiv
- View kann aktualisiert werden nachdem Nutzer geantwortet hat





CloudKit News

- Langlebige Operations
- `CKOperationTimeouts`
 - `timeoutIntervalForRequest`, `timeoutIntervalForResource`
- `CKDatabaseSubscription` statt Polling
- `CKFetchDatabaseChangesOperation` statt jede Record Zone einzeln abfragen
- `fetchAllChanges` property: Automatisches Paging



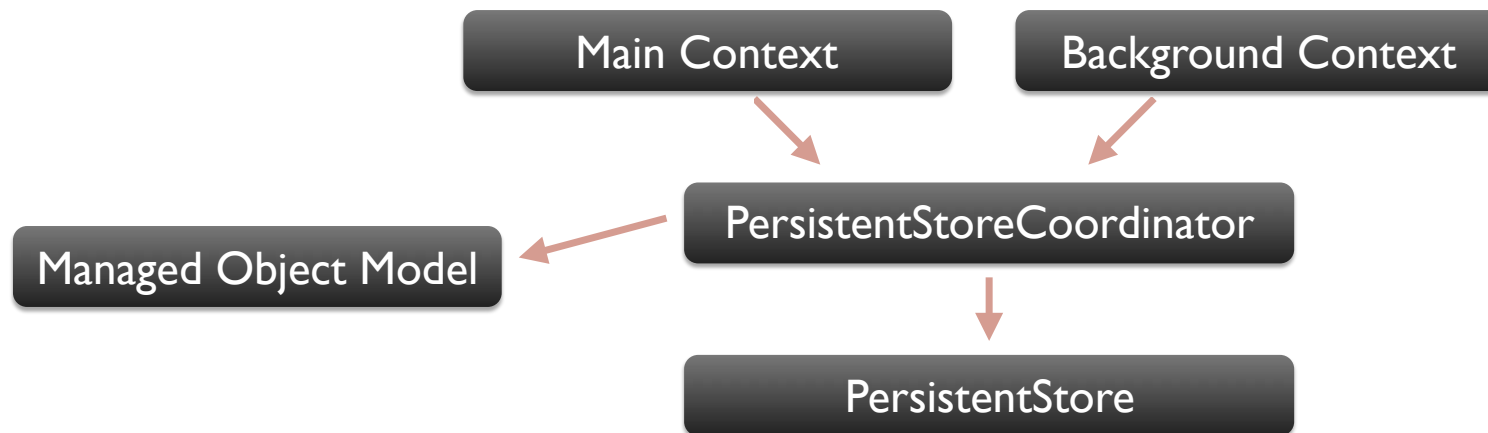
CloudKit Private Sharing

- Vorher: Komplette öffentlich oder komplett privat
- Jetzt: Kann Daten mit anderen ausgewählten Nutzern teilen
- Readwrite oder Readonly
- Daten gehen auf Kosten des erstellenden iCloud Accounts
- UICloudSharingController: Alles über das man einen Link versenden kann
 - iMessage, Twitter, Mail,

CoreData

- Query Generations: Sicherstellen, dass sich euer Context nur dann ändert wenn ihr es wollt
 - Kein „could not fulfil fault“ mehr
- API Verbesserungen
 - `Subclass.entity()` **statt** `NSEntityDescription.entity(forEntityName:"Entity" in:context)`
 - `Subclass.fetchRequest()` **statt** `NSFetchRequest(entityName: "Entity")`
 - `Subclass(context: managedObjectContext)` **statt** `NSEntityDescription(insertNewObjectforEntityForName("Entity", inManagedObjectContext:context)`

CoreData: NSPersistentContainer



CoreData: NSPersistentContainer





CoreData: NSPersistentContainer

- viewContext
- newBackgroundContext() Factory
- Viel schönere API
 - NSPersistentStoreDescription statt Dictionary
 - Asynchrones Hinzufügen von Stores
 - **Braucht nur einen Namen**
- performBackgroundTask()

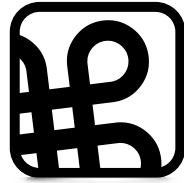
Außerdem noch...

- Spracherkennungs-API (Echtzeit, aber recht beschränkt)
- `DispatchQueue.AutoreleaseFrequency.autoreleaseWorkItem`
- Accelerate: API für Basic Neural Network Subroutines
- SecKey: Vereinheitlichte API, ersetzt CDSA und asymmetric SecTransforms
- CryptoTokenKit: Unterstützung für crypto-Geräte wie Smartcards und USB-Sticks
- `UIPasteboardOption:expirationDate, localOnly`

Zusammenfassung

Joachim Kurz
@cocoafrog

- AutoLayout API, Layout Guides, UIStackView
- „There is a Formatter for that“
- Unterstützt NSUserActivity und taggt eure Textfelder
- GameplayKit: State Machine, Agents, Zufallsquellen, Rauschfunktionen, AI, Pathfinding
- Daten teilen via CloudKit oder GKGameSession
- Setzt euch mit Wide & Deep Color auseinander
- Benutzt UIViewPropertyAnimator
- Benutzt endlich NSURLSession
- CoreData, UserNotifications, Contacts und vieles andere wurde aufgeräumt



Macoun