



iOS Workshop

FHNW – Frühjahr 2017

Überblick

1. Termin: 27. Februar

- Überblick iOS
- Überblick Swift
- Einstieg Swift-Entwicklung

2. Termin: 6. März

- Swift-Entwicklung
- Einstieg iOS-Projekt
 - UI mit Storyboards

3. Termin: 13. März

- iOS-Projekt
 - UI mit Storyboards
 - Networking

4. Termin: 20. März

- Finalisierung iOS-Projekt
- Tipps & Tricks
- Empfehlungen Frameworks & Tools
- Fragen und Antworten

iOS

- Mobiles Betriebssystem von Apple
 - iPhone, iPad, iPod touch
- Darauf basierend: tvOS & watchOS
- iOS basiert auf macOS-Kern → Darwin → Unix
- Seit März 2008 SDK für iOS verfügbar
- Jährliche Aktualisierungen zur WWDC
- Voraussetzung für Entwicklung Apple-Hardware
- Entwicklung in Objective-C oder Swift
- Details: <https://en.wikipedia.org/wiki/IOS>



Walled Garden

- Jede App wird durch Apple auf Konformität zu den App-Store-Guidelines geprüft
 - Nicht jede Idee ist dadurch umsetzbar
 - Es wird gegenüber anderen Plattformen ein höheres Mass an Qualität und Sicherheit erreicht

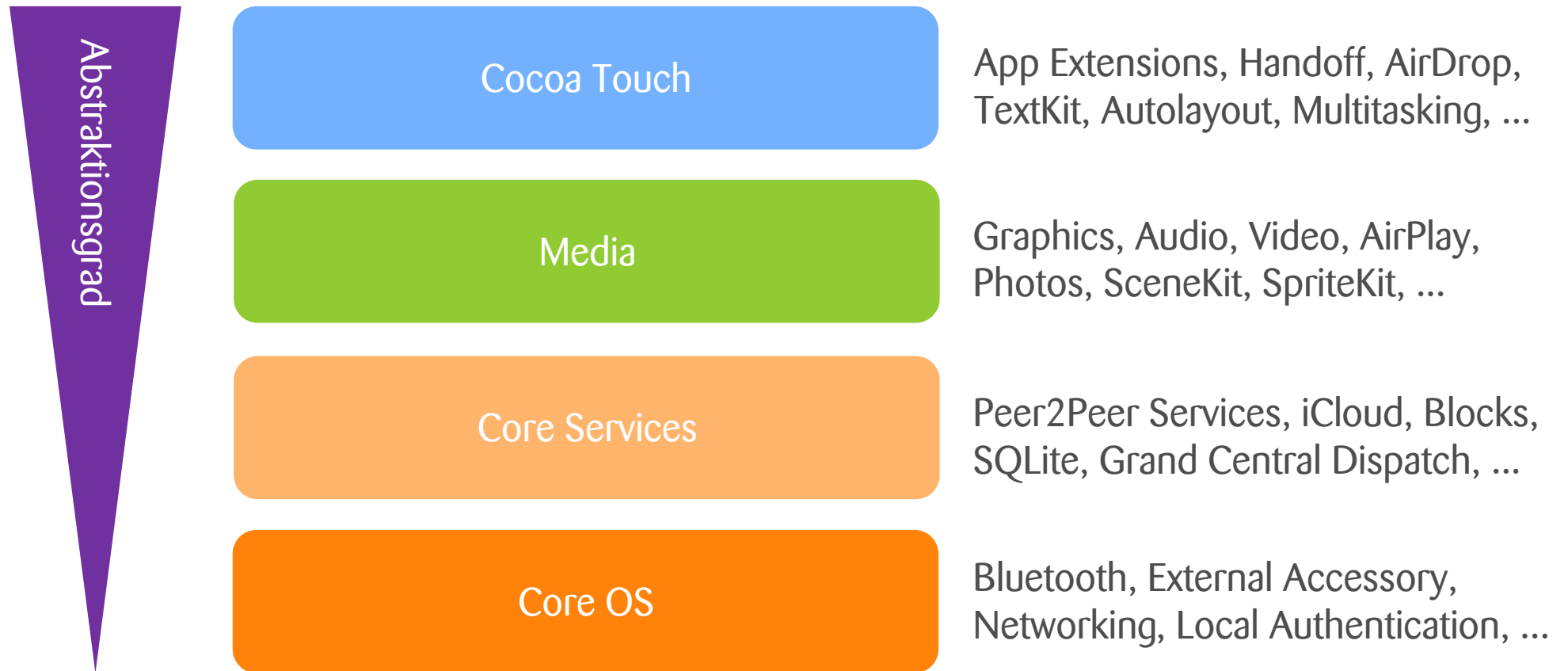
1.1.1 Defamatory or mean-spirited content, including references or commentary about religion, race, sexual orientation, gender, or other targeted groups, particularly if the app is likely to place a targeted individual or group in harm's way. Professional political satirists and humorists are generally exempt from this requirement.

1.1.2 Realistic portrayals of people or animals being killed, maimed, tortured, or abused, or content that encourages violence. "Enemies" within the context of a game cannot solely target a specific race, culture, real government, corporation, or any other real entity.

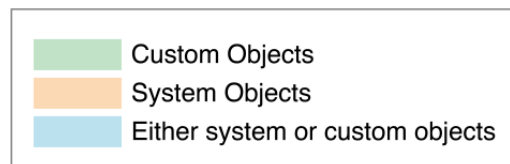
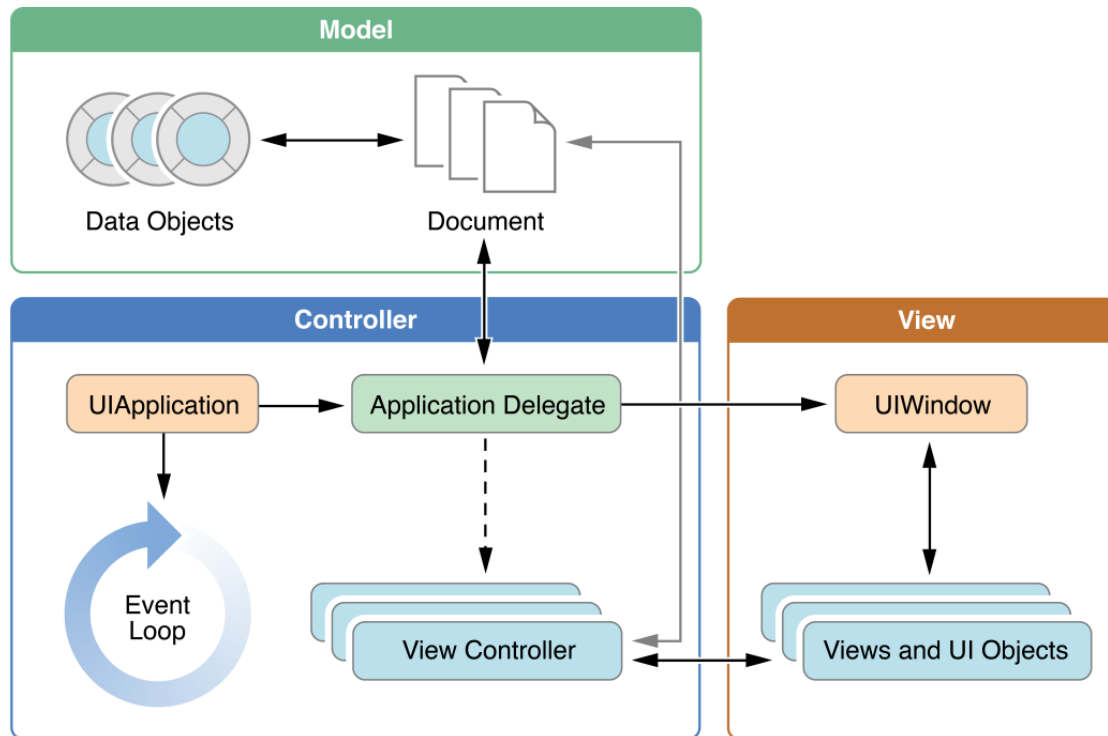
1.1.3 Depictions that encourage illegal or reckless use of weapons and dangerous objects, or facilitate the purchase of firearms.

1.1.4 Overtly sexual or pornographic material, defined by Webster's Dictionary as "explicit descriptions or displays of sexual organs or activities intended to stimulate erotic rather than aesthetic or emotional feelings."

iOS Architektur



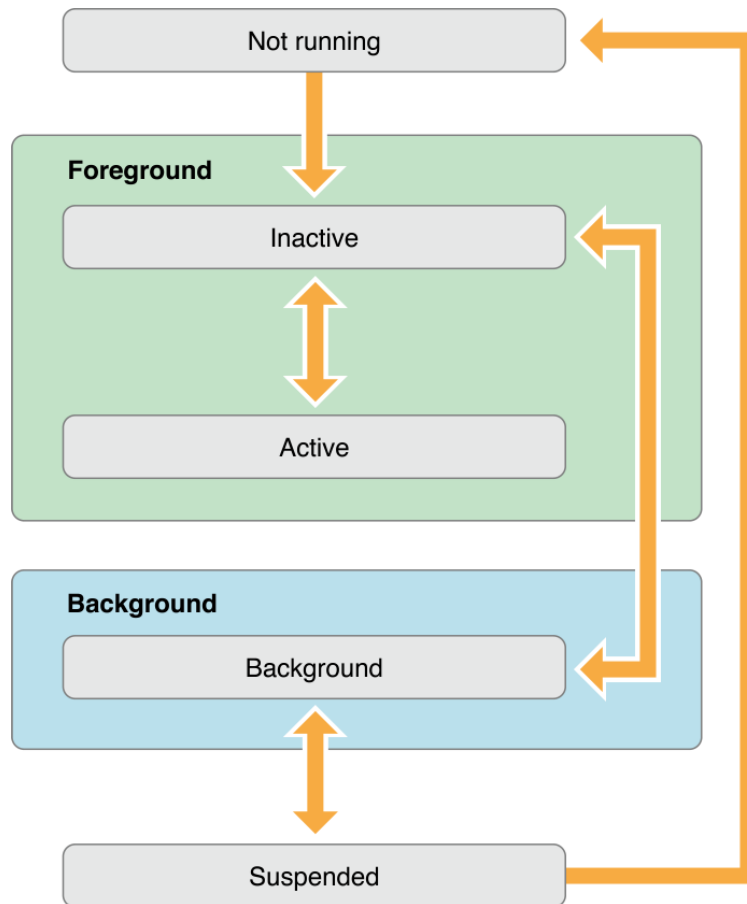
Struktur einer iOS-App



MVC-Pattern

Vergleich diverser iOS-Entwurfsmuster: [Link](#)

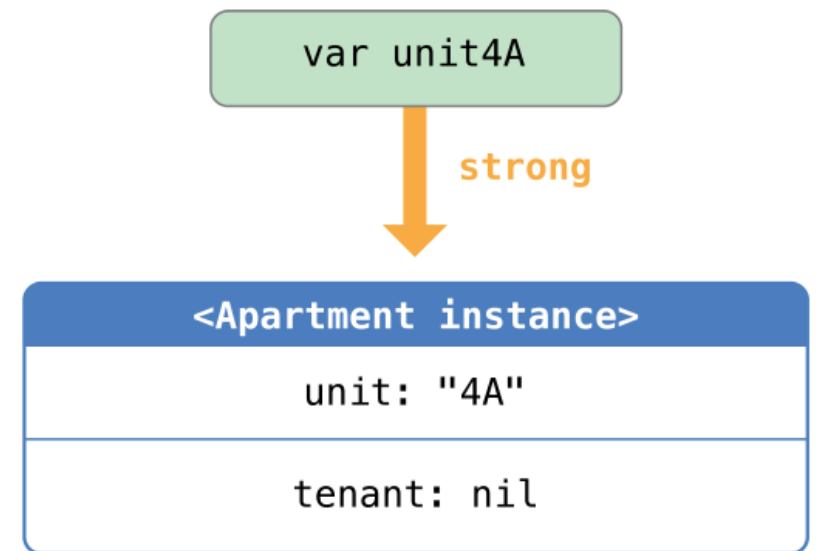
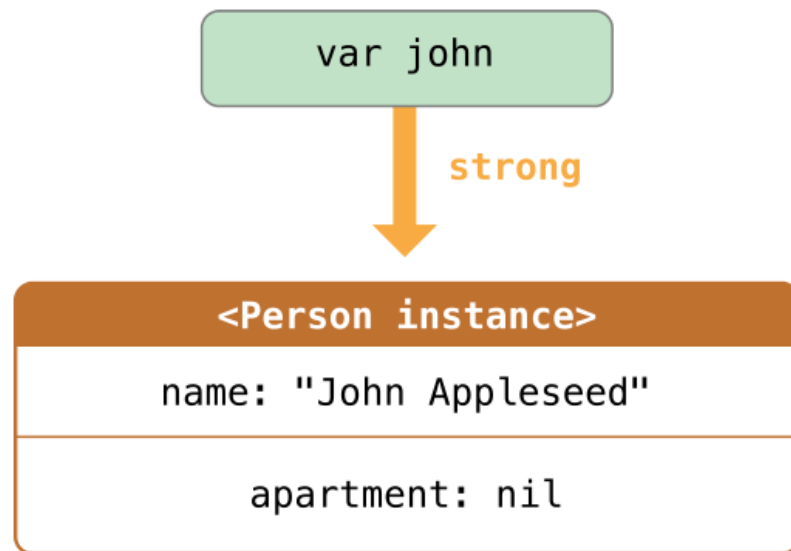
App Lifecycle



Diverse Delegate-Aufrufe im App-Delegate:

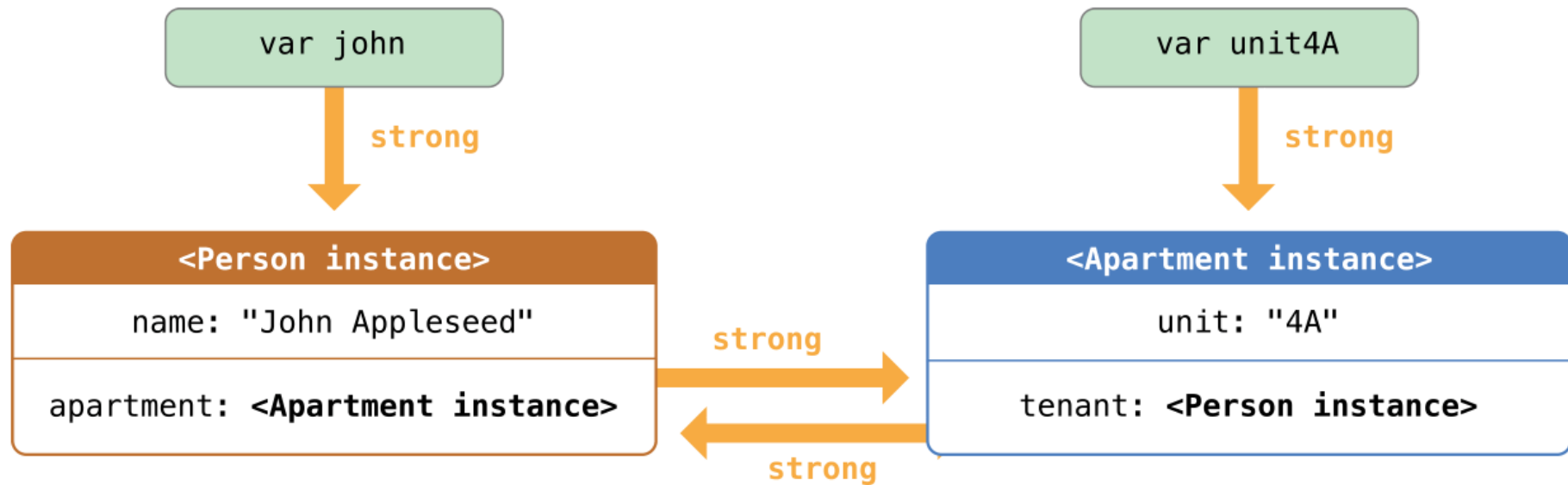
- `applicationWillFinishLaunchingWithOptions`
- `applicationDidFinishLaunchingWithOptions`
- `applicationDidBecomeActive`
- `applicationWillResignActive`
- `applicationDidEnterBackground`
- `applicationWillEnterForeground`
- `applicationWillTerminate`

Automatic Reference Counting



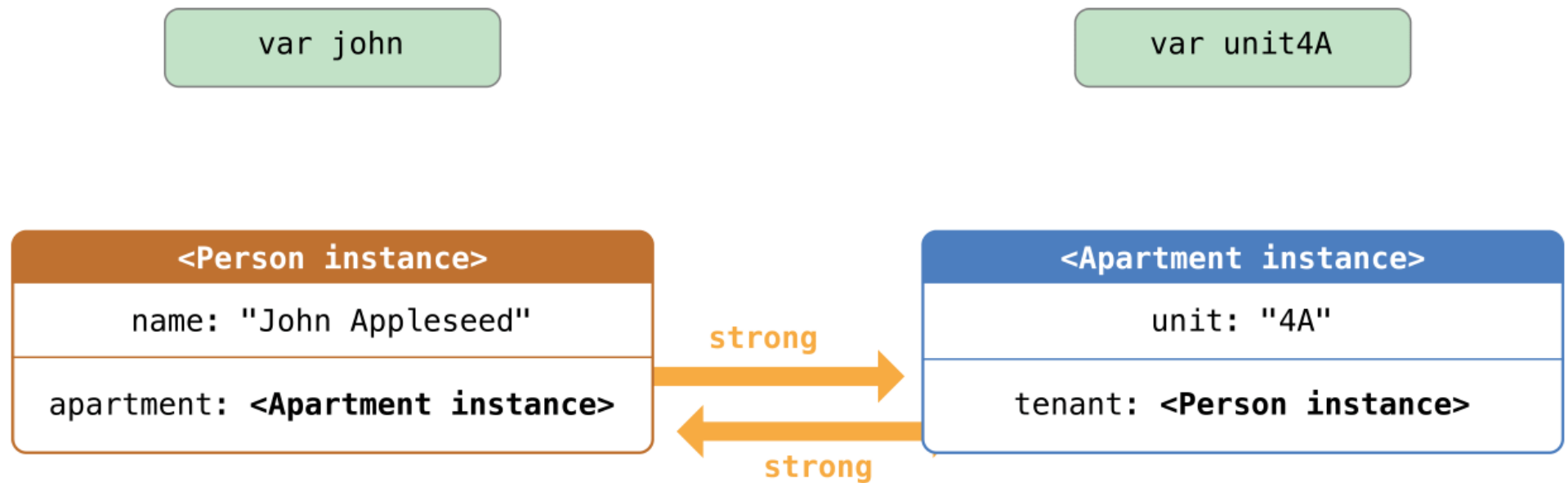
➡ [Apple-Doku](#)

Automatic Reference Counting



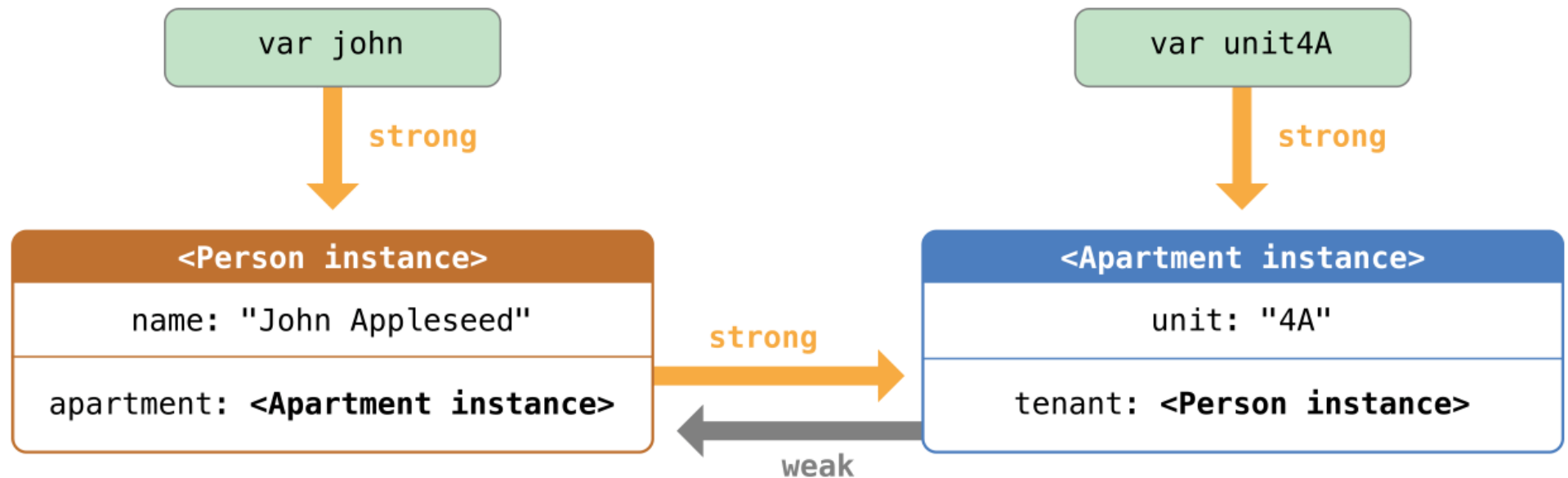
➡ [Apple-Doku](#)

Automatic Reference Counting



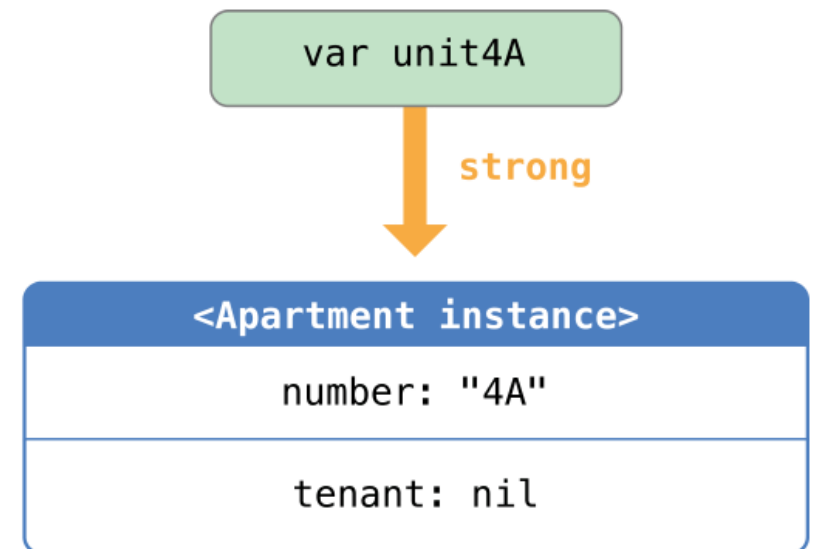
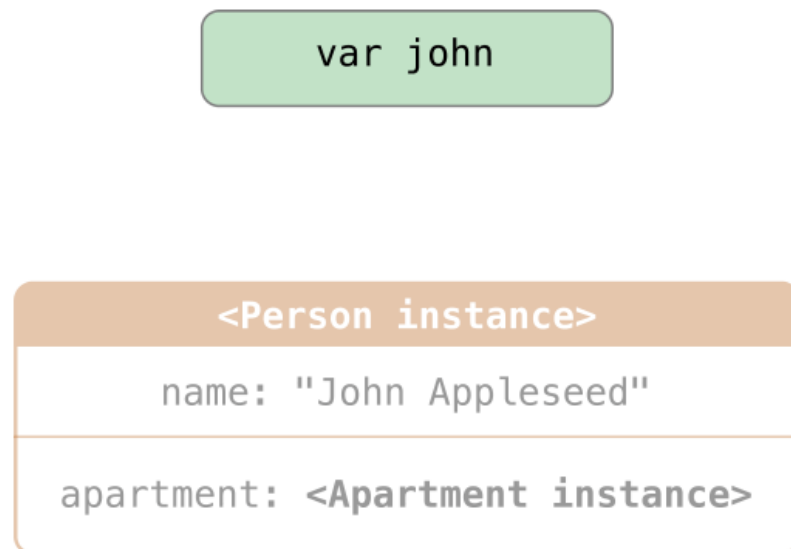
➡ [Apple-Doku](#)

Automatic Reference Counting



➡ [Apple-Doku](#)

Automatic Reference Counting



➡ [Apple-Doku](#)

Automatic Reference Counting

```
var john
```

<Person instance>	
name:	"John Appleseed"
apartment:	<Apartment instance>

```
var unit4A
```

<Apartment instance>	
unit:	"4A"
tenant:	nil

➡ [Apple-Doku](#)

iOS Entwicklungsressourcen

Referenz	Url	Thema
iOS API Reference	https://developer.apple.com/reference/	Offizielle Referenz von Apple zu allen Frameworks
Swift Programming Language	https://developer.apple.com/library/content/documentation/Swift/Conceptual/Swift_Programming_Language/index.html	Offizielle Doku zu Swift
Stanford iOS Kurs	https://itunes.apple.com/ch/course/developing-ios-10-apps-with-swift/id1198467120	Jährlicher iOS-Kurs der Stanford-University in iTunesU
WWDC Videos	https://developer.apple.com/videos/	Aufzeichnungen der WWDC Sessions
Swift Playgrounds	http://www.apple.com/swift/playgrounds/	iPad App zum Lernen von Swift
StackOverflow	https://stackoverflow.com/questions/tagged/swift+ios	Sehr grosse Community im iOS-Bereich

Swift

- General-Purpose, Multi-Paradigm, Compiled Programming Language
- Entwickelt durch [Apple](#)
- Plattformen: iOS, macOS, watchOS, tvOS & Linux
- Kompatibel zu Objective-C,
 - „Objective-C without the C“
- Wird durch LLVM Compiler verarbeitet
- [OpenSource](#) (Apache License)
- Diverse vielversprechende Portierungen, (z.B. durch [IBM](#))



Swift-Features

- Multiparadigm: functional, imperative, generic, OOP
- Memory safe by default
- Closures unified with function pointers
- Tuples and multiple return values
- Generics
- Fast and concise iteration over a range or collection
- Structs that support methods, extensions, and protocols
- Functional programming patterns, e.g., map and filter
- Powerful error handling built-in
- Advanced control flow with do, guard, defer, and repeat keywords
- ...



Einstieg Swift-Entwicklung

- XCode Playgrounds
 - Entwicklung ohne Projekt und Abhängigkeiten
 - Kombination aus Markdown und ausführbaren Code
 - Ideal für lebendige Tutorials, Dokumentation, Lernen, Experimente



Weiter in Playgrounds

