UNIKASSEL VERSITÄT

Fachgebiet für Kommunikationstechnik

Prof. Dr.-Ing. Klaus David

ComTec

Communication Technologies 2

Einführung in das iOS SDK

Dipl.-Inf. Daniel Wilhelm
Kassel, 17.02.2014

Übersicht

ComTec

- Voraussetzungen
- Mitgelieferte Werkzeuge
- Einschränkungen
- iPhone OS (iOS)
- Programmiermodelle

- Quellen

ComTec

Einführung in das iOS SDK

VORAUSSETZUNGEN

Voraussetzungen (1)

ComTec

- Mitgliedschaft im Entwicklerprogramm von Apple
  - Online, University (kostenlos)
  - Standard, Enterprise (kostenpflichtig)

Developer Program	Download des SDKs	Test im Simulator	Test auf Endgerät	Vertrieb: Ad Hoc	Vertrieb: App Store	Vertrieb: In-House
Online	✓	✓	×	×	×	×
Standard	✓	✓	✓	100	✓	×
Enterprise	✓	✓	✓	✓	×	✓
University	✓	✓	✓	(200)	×	×

Quelle [1]

- iOS SDK

Voraussetzungen (2)

ComTec

- Mac auf Intel-Basis (aktuell: Mac OS X v10.9.1)
- Testgeräte unterschiedlicher Generationen sind sinnvoll:
  - iPod touch G1 / G2 / G3 / G4 / G5
  - iPhone / 3G / 3GS / 4 / 4S / 5 / 5C, 5S
  - iPad G1 / G2, mini G1 / G3 / G4 / Air, mini G2
- Zugang zu iTunes Connect (Verkauf von Apps)
- Kenntnisse von Objective-C

ComTec

Einführung in das iOS SDK

MITGELIEFERTE WERKZEUGE

Werkzeuge (1)

ComTec

- **Xcode** ist eine auf Open-Source-Programmen aufbauende Entwicklungsumgebung mit:
  - Projektmanagement
  - Quellcodebearbeitung
  - Dokumentation
  - Grafischem Debugger
  - ...

Werkzeuge (2)

ComTec

- **Interface Builder** (integriert) erlaubt:
  - Die grafische Entwicklung von App-Oberflächen
  - Das Verlinken der Oberflächen mit Code-Objekten
- **Simulator** nutzt Intel-x86-kompilierten Code zum Testen von Apps auf dem Mac
- **Instruments** dient zur Auslastungs-Analyse:
  - Prüft Speicherauslastung und Leistung
  - Macht Speicherlecks ausfindig

ComTec

Einführung in das iOS SDK

EINSCHRÄNKUNGEN

Simulator vs. Endgerät (1)

ComTec

- Simulator:
  - \+ Ermöglicht schnelle Tests & Verwendung von Tastatur
  - \+ Kann Speicherwarnungen simulieren
  - \+ Kann alle möglichen Auflösungen simulieren
  - – Verwendet Mac-Frameworks & -Hardware
  - – Unterstützt nicht alle Funktionen (z.B. Kamera, GPS, Beschleunigung, Vibration, Multitouch, Push-Dienste, Schlüsselbund), einige können aber simuliert werden

Simulator vs. Endgerät (2)

ComTec

- iPod touch / iPhone / iPad:
  - \+ Test unter „realen“ Bedingungen (Sensoren, Speicher, Prozessor, Frameworks etc.)
  - – Unterschiedliche Hardwareausstattung (z.B. Kamera, Aufösung)
  - – Kabel muss (zum Debuggen) angesteckt bleiben

Einschränkungen (1)

ComTec

- Ausführung der App in einer Sandbox, d.h. kein Zugriff auf Daten anderer Applikationen
- Kein virtueller Speicher vorhanden → sparsamer Umgang mit Ressourcen ist wichtig
- Keine Garbage Collection (mittlerweile „ARC“)
- Benutzerverhalten → App wird häufig nur kurz benutzt (regelmäßig speichern etc.)

Einschränkungen (2)

ComTec

- Kein echtes Multitasking, d.h. die App
  - Kann jederzeit beendet werden
  - Sollte dabei den Speicher „aufräumen“
  - Kann auch wieder aktiviert werden
- Akkulaufzeit → auf Rechenleistung achten
- Kleiner Bildschirm, d.h.
  - Einfach anzutippende Schaltflächen und alternative Eingabemöglichkeiten sind wichtig
  - Texteingabe ist nicht unkritisch

ComTec

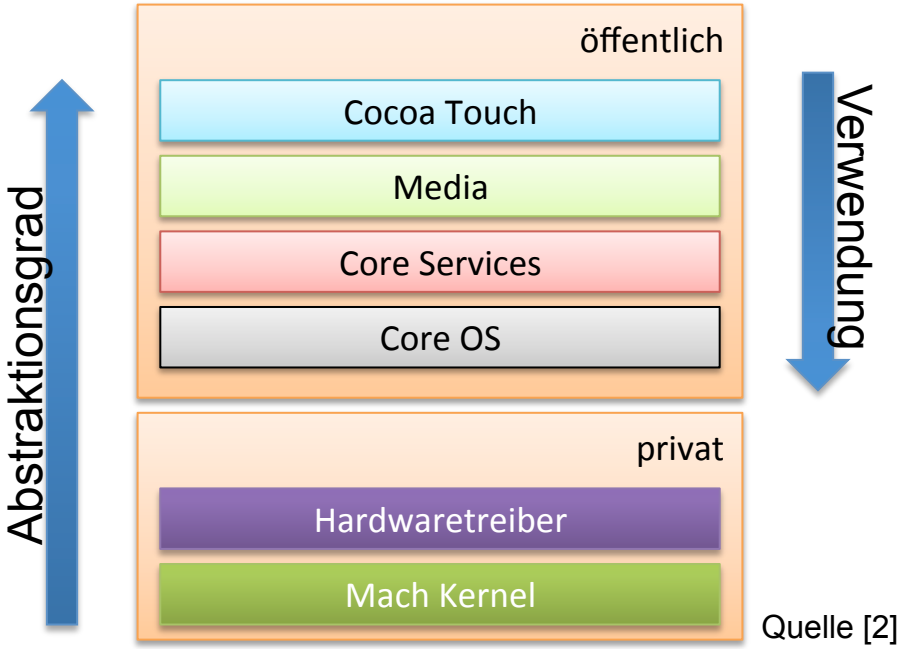
Einführung in das iOS SDK

IPHONE OS (IOS)

iOS (1)

ComTec

- BS für iPod touch, iPhone, iPad und AppleTV
- Applikationen haben keinen direkten Hardware-zugriff; erfolgt über Abstraktionsschichten



Quelle [2]

iOS (2)

ComTec

- Core OS Layer (Nutzung mittels C)
  - POSIX-Threads
  - BSD-Sockets
  - Dateisystemzugriff
  - Standard-I/O
  - Bonjour, DNS
  - Zugriff auf Informationen zur Sprache
  - Allokierung von Speicher
  - Mathematische Berechnungen

iOS (3)

ComTec

- Core Services Layer
  - Adressbuch-Zugriff
  - Core Foundation (Basisdienste zur Programmierung; z.B.: Listen, Zeichenketten, Datum & Zeit)
  - CFNetwork (Netzwerkkommunikations-Basisdienste; z.B.: SSL/TLS, DNS, HTTP & HTTPS, FTP)
  - Core Location (Basisdienste für Aufenthaltsorte)
  - Security (Anwendungssicherheit)
  - SQLite
  - XML

iOS (4)

ComTec

- Media Layer
  - Grafik (Quartz, Core Animation, OpenGL ES)
  - Audio (Core Audio, OpenAL)
  - Video
- Cocoa Touch Layer
  - Frameworks, die als Basis für grafische Oberflächen eingesetzt werden
  - z.B.: Push Notifications, AddressBook UI, Map Kit

ComTec

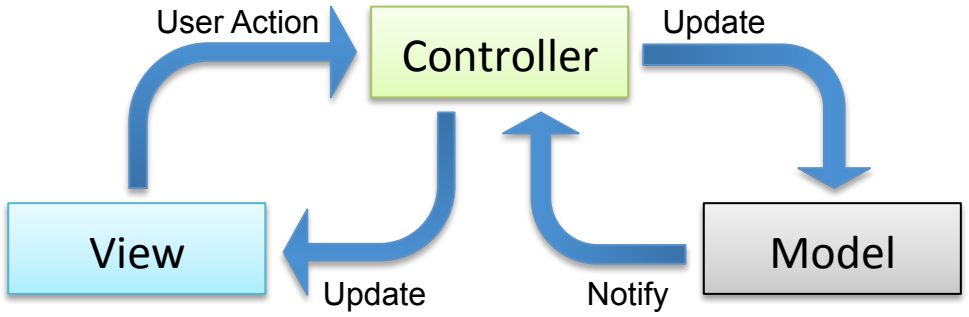
Einführung in das iOS SDK

PROGRAMMIERMODELLE

Programmiermodelle

ComTec

- Objektorientierte Programmierung (Objective-C):
  - Klassen
  - (Mehrfach-)Vererbung
- MVC-Pattern:
  - Trennung von Model (Datenmodell), View (Ansicht) und Controller (Steuerung)



Quelle [3]

MVC-Pattern (1)

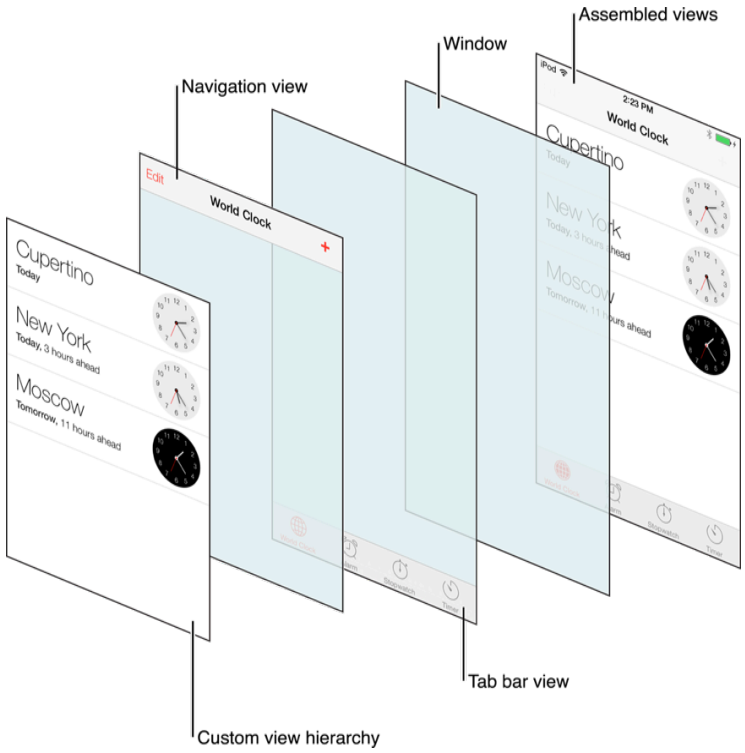
ComTec

- Datenmodell:
  - Modellmethoden liefern Daten (per Callback-Methoden)
  - Meistens im `ViewController` implementiert (bei einfachen Quellen)

MVC-Pattern (2)

ComTec

- Ansicht:
  - Komponenten sind Kinder der Klassen `UIView` + `UIResponder` und `UIViewController`
  - Views sind hierarchisch in Bäumen angeordnet; eine `UIWindow`-Instanz bildet die Wurzel



Quelle [4]

MVC-Pattern (3)

ComTec

- Steuerung:
  - Komponenten allgemein gehalten (Verwendung kann nicht erahnt werden)
  - **Delegation**: (vorher definierte) Callback-Methoden werden auf einer anderen Klasse aufgerufen → reduziert Ableitungen
  - **Target-Actions**: Umleitung einer bestimmten Aktion (ähnlich Delegation aber einfacher)
  - **Benachrichtigungen**: Kommunikation mittels Abo von per Broadcast gesendeten Nachrichten

Einführung in Objective-C

QUELLEN

Quellen (1)

ComTec

- Stäuble, Markus: „Programmieren fürs iPhone“ (dpunkt.verlag GmbH, 1. Auflage 2009)
- Sadun, Erica: „Das große iPhone Entwicklerbuch“ (Addison-Wesley Verlag, 2010)
- Dr. Koller, Dirk: „iPhone-Apps entwickeln“ (Franzis Verlag GmbH, 2010)
- Apple Inc.: „iOS Developer Library“ (https://developer.apple.com/library/ios/navigation/)

Quellen (2)

ComTec

- [1] Apple Inc.: „Apple Developer Programs – Apple Developer“ (http://developer.apple.com/programs/, 2013)
- [2] Apple Inc. „iOS Technology Overview: About the iOS Technologies“ (https://developer.apple.com/library/ios/documentation/miscellaneous/conceptual/iphoneostechoverview/Introduction/Introduction.html, 2013)
- [3] Apple Inc.: „Cocoa Core Competencies: Model-View-Controller“ (https://developer.apple.com/library/ios/documentation/General/Conceptual/DevPedia-CocoaCore/MVC.html, 2013)
- [4] Apple Inc.: „UIKit User Interface Catalog: About Views“ (https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/UIKitUICatalog/, 16.12.2013)

Stand: 11. Februar 2014