

# Mobile Applications

Betriebssysteme, Frameworks und Anwendungsprogrammierung



- Dies ist ein Übersichts-Seminar
  - Damit werden die Inhalte durch das Studium von Beispielen vertieft
  - Eigene Erweiterungen und Übungen sind bei Bedarf möglich und sinnvoll
  - Eine Musterlösung wird in elektronischer Form angeboten
    - Diese muss jedoch weder die eleganteste noch beste Lösung sein!
- Die Web Anwendungen können in jedem Browser getestet werden
  - Es empfiehlt sich jedoch, die Seiten in einem Mobilgerät zu betrachten
  - Dazu müssen die Seiten jedoch aus einer WIFI-Umgebung geladen werden können
    - Alternativ kann auch der Browser eines Simulators benutzt werden
      - Der Android-Simulator verbindet sich über die IP-Adresse 10.0.2.2 mit seinem Host

© Javacream

Javacream

Dr. Rainer Sawitzki

Alois-Gilg-Weg 6

81373 München

**Alle Rechte, einschließlich derjenigen des auszugsweisen Abdrucks,  
der fotomechanischen und elektronischen Wiedergabe vorbehalten.**

Einführung	6
Strategien zur Mobilen Anwendungsentwicklung	20
Mobile Plattformen	40
Vertiefung Android	67
Mobile Web Applications	107
Javascript für Mobile Web Applications	131
Hybrid Applications	163
iOS	174
Windows Phone	210

1

# EINFÜHRUNG

1.1

# ÜBERSICHT

- „Software that is developed for small low power handheld devices such as personal digital assistants, enterprise digital assistants or mobile phones. “
  - Bestenfalls der Versuch einer eindeutigen Definition
  - Smartphones und insbesondere Tablets sind nicht unbedingt „klein“
- Andere Versuche einer Annäherung
  - Mobile Anwendungen
    - Installation und die Benutzung an jedem beliebigen Ort
  - Mobile Daten
    - Zentrale Datenhaltung auf Servern in der Cloud
  - Mobilität des Geräts
    - Potenzielle Lücken in der Netzwerk-Abdeckung



- Naive Definition
  - Anwendungen, auf die die eben angegebenen Begriffe zutreffen
    - Damit hat praktisch jede neu konzipierte Anwendungen einen mobilen Anteil
- Fachliche Definition
  - Anwendungen, die Informationen wie Geolokation benötigen
- Technische Definition
  - Anwendungen, die für den heutigen Markt von Mobilten Endgeräten entwickelt werden
  - Aktuell drei unterschiedliche Ansätze
    - Native Applikationen
    - Mobile Web Applikationen
    - Hybrid-Lösungen

- Native App
  - Neue Konzeption und Entwicklung
  - reichhaltige Benutzer-Interaktionen
  - Verwendung und Definition permanent laufender Hintergrund-Dienste
  - Zugriff auf alle Ressourcen und Sensoren des Geräts
  
- Mobile Web Anwendungen
  - können häufig durch Anpassungen bereits vorhandener Internet-Auftritte realisiert werden
    - Konzeptions- und Dokumentationsaufwand wird dadurch verringert.
  - Mobile Geräte haben einen relativ gut standardisierten Browser
    - gilt leider nur für moderne Smartphones und Tablets

- Web Anwendungen
  - Chat
  - Datenorientierte Anwendungen mit häufig wechselnden Daten
  - Spiele (Multiplayer)
  - Kataloge, Listen
  - Kalender und Aufgaben synchronisiert mit mehreren Benutzern
- SDK Anwendungen
  - Address-Bücher, Kontakte
  - Animierte Grafiken
  - Datenorientierte Anwendungen mit kritischen Informationen
  - Komplexe Spiele
  - Location-aware Anwendungen
  - Photo-/Video-Anwendungen

1.2

## **DIE ZIELPLATTFORMEN – EIN EXTREM FRAGMENTIERTER BEREICH**

- Mobile Phones
  - Nichts anderes als eine Telefon ohne Apps
- Feature Phone
  - Zugriff auf das Internet durch integrierten Browser
    - Teilweise nur Textbasiert mit Tab-Navigation
- Social Phone
  - Vorinstallierter Satz weiterer Anwendungen: E-Mail, Twitter, Facebook, ...
- Smartphone
  - Touch Display, Apps, Sensoren...
  - App Stores ermöglichen die Installation weiterer Software

Android version	Codename	Released on	Optimized for
1.0 & 1.1		2008, 2009	Smartphones (deprecated)
1.5	Cupcake	1Q 2009	Smartphones (deprecated)
1.6	Donut	3Q 2009	Smartphones (deprecated)
2.0 & 2.1	Eclair	4Q 2009	Smartphones
2.2	Froyo	2Q 2010	Smartphones
2.3	Gingerbread	4Q 2010	Smartphones
3.0, 3.1 & 3.2	Honeycomb	2011	Tablets
4.0	Ice Cream Sandwich (ICS)	4Q 2011	Smartphones & tablets
4.1 & 4.2	Jelly Bean (JB)	2Q 2012	Smartphones & tablets
Not confirmed	Key Lime Pie (KLP)		Smartphones & tablets

Browser	Current platforms	Engine	Proxied	Navigation
Safari	iOS	WebKit	No	Multitouch
Android browser	Android up to 4.1	WebKit	No	Multitouch and focus
Chrome	Android > 4.0 (iOS as a pseudo-browser)	WebKit	No	Multitouch and focus
Nokia Browser	S40 6th Edition	WebKit	No	Cursor, touch, and focus
	S40 new platform (Nokia Xpress browser)	Gecko	Yes	Focus, touch, and multitouch
	Symbian	WebKit	No	Cursor, multitouch, and focus
	MeeGo	WebKit2	No	Multitouch
webOS browser	webOS, Open webOS	WebKit	No	Multitouch and focus
BlackBerry browser	BlackBerry OS 5.x	Custom	In some situations	Cursor and multitouch
	BlackBerry OS 6.0, 7.0	WebKit	No	Cursor and multitouch
	Tablet OS / BB10	WebKit	No	Multitouch and cursor
Internet Explorer	Windows Phone, Windows	Trident	No	Multitouch
Firefox	Android, MeeGo, Firefox OS	Gecko	No	Multitouch
NetFront	Low-end devices	Custom	No	Focus or touch <sup>a</sup>
Opera Mobile	Android, Symbian	Presto/WebKit	Yes/No <sup>b</sup>	Focus
Opera Mini	Android, iOS, Symbian, Java, BlackBerry	Presto/WebKit <sup>c</sup>	Yes	Cursor or touch <sup>d</sup>
Bada Browser	Bada	WebKit	No	Touch
UC Browser	Android, iOS, Java, BlackBerry	Custom	Yes	Multiple

# Und dazu kommen noch:

- Apps, die den Browser intern benutzen
  - Web Views
  - „Pseudo Browser“: Native Apps, die die Rendering Engine des Betriebssystems nutzen
- Cloud-basierte Browser
  - Der Client kommuniziert mit einem Proxy-Server, der die eigentliche Seite lädt und dabei diverse Optimierungen vornimmt
- Transcoder
  - Optimierungsversuche der Mobilfunk-Betreiber



# Wie soll das alles organisiert werden?



- Die Anwendung muss auf den verschiedenen Geräten getestet werden
  - Simulatoren
  - Device Labs
    - Kostenpflichtige Anmietung von Geräten
  - Remote Test Labs

# Beispiel: Samsung Remote Test Lab

**SAMSUNG DEVELOPERS** » DEVELOP DISTRIBUTE DEVICES FORUM EVENTS

Sign out | My Page English »

Search

**Mobile Device Specs**

**SmartTV Specs**

**Remote Test Lab**

[Device List](#)

[Test History](#)

[Issues / Suggestions](#)

[About Remote Test Lab](#)

## Device List

RTL(Remote Test Lab) requires a Modern browser with JavaScript and a Java runtime environment (JRE v1.6 +).

Configuration requirements:

Configuration check result : **Pass** [Show Details](#)


Once your configuration is OK, Continue to select a device.

RTL(Remote Test Lab) allows users to install and test applications over the web.

You now have 1.8 credit(s) to use. For more details, please go to [Test History](#)

Hot Android bada [MY RESERVATIONS](#)

**Galaxy S4**




– OS Version –

– Device List –

Reserve for 30 minutes

[Start](#)

**Galaxy Note II**




– OS Version –

– Device List –

Reserve for 30 minutes

[Start](#)

**Galaxy Note 10.1**




– OS Version –

– Device List –

Reserve for 30 minutes

[Start](#)

**Galaxy S III**




– OS Version –

– Device List –

Reserve for 30 minutes

[Start](#)

**Galaxy S II**




– OS Version –

– Device List –

Reserve for 30 minutes

[Start](#)

**Galaxy Tab 10.1**



– OS Version –

– Device List –

Reserve for 30 minutes

[Start](#)

2

# **STRATEGIEN ZUR MOBILEN ANWENDUNGSENTWICKLUNG**

2.1

## **NATIVE APPLIKATIONEN**

- Installation über App Store
- Realisierung durch Programmiersprachen, die auf dem Mobile Device zur Verfügung gestellt werden
  - Damit muss die Anwendung in verschiedenen Zweigen mit völlig unterschiedlichen Quellcodes realisiert werden
- Hardware-nahe Programmierung und damit Optimierungen möglich
- Zugriff auf Ressourcen des Mobilgeräts möglich
  - Dateisystem
  - Threading und Hintergrund-Services
  - Embedded Datenbank
  - Client-Server-Zugriff durch http-Sockets
- GUI-Programmierung durch Verwendung von Betriebssystem-Widgets
  - Und damit (fast) automatisch ein einheitliches Look&Feel

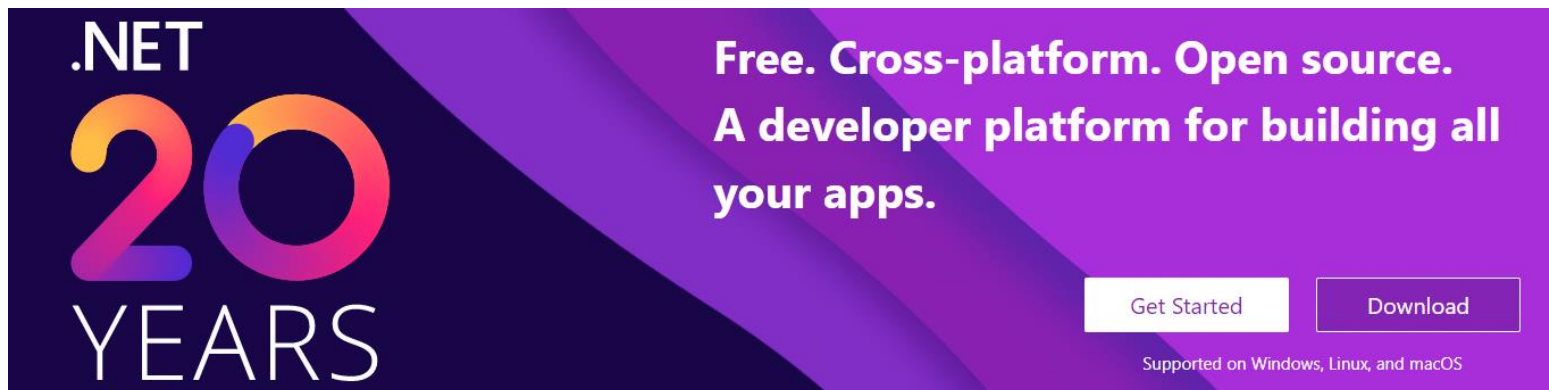


# Die drei Marktführer (2015)



android





**.NET**  
**20**  
**YEARS**

**Free. Cross-platform. Open source.**  
**A developer platform for building all your apps.**

[Get Started](#) [Download](#)

Supported on Windows, Linux, and macOS

- Jede native Anwendung ist eine komplett eigene Implementierung
  - Wiederverwendet werden kann „nur“ das Fachkonzept bzw. die Dokumentation
- Damit entsteht unvermeidbar pro Plattform beträchtlicher Aufwand
- Strategie: Konzentration auf den/die Marktführer
  - Andere Plattformen können bei Bedarf auch noch später unterstützt werden
  - Funktioniert innerhalb eines Unternehmens recht gut, da hier die Zielplattformen einigermaßen homogen vorgegeben werden können

2.2

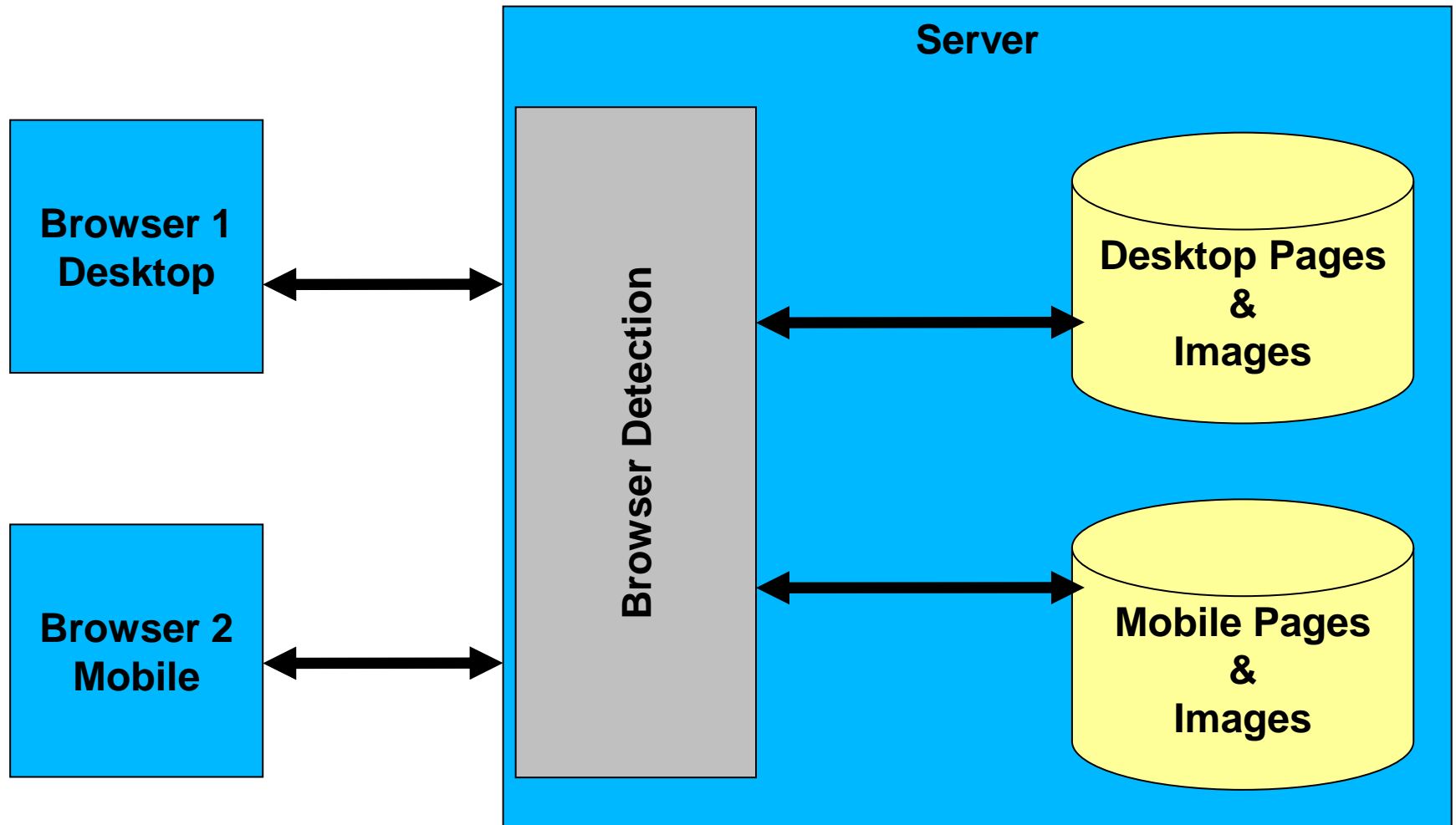
## **MOBILE WEB APPLICATIONS**

- Bereitstellung der Seiten durch den Server
- Das Mobilgerät muss nur einen Browser zur Verfügung stellen
  - Damit können auch noch Feature Phones erreicht werden
- GUI-Programmierung durch Verwendung der vom Browser zur Verfügung gestellten Widgets
- Bei der Programmierung können nur die Technologien des mobilen Browsers benutzt werden
  - Caniuse.com bietet eine Übersicht

- Low-end Geräte
  - Basic (X)HTML, Bildschirmbreite maximal 176 Pixel, Basic CSS (text color, background color, font size), kein JavaScript
- Mid-range Geräte
  - Basic (X)HTML, Durchschnittliche Bildschirmbreite 240 Pixel, Medium CSS support (box model, images), Basic JavaScript
- High-end Geräte
  - (X)HTML oder HTML 4, Durchschnittliche Bildschirmbreite 320 Pixel, Advanced CSS (Desktop-Browser), Ajax und DOM
- Smartphones
  - HTML5, Relativ große Bildschirme mit High Resolution, Touch support, CSS3 (Animations, Effects), Ajax, Local Storage, Geolocation
- Web App für Smartphones
  - Zusätzlich Offline support, Full-Screen und Icon Installation, Native Integration, Device APIs

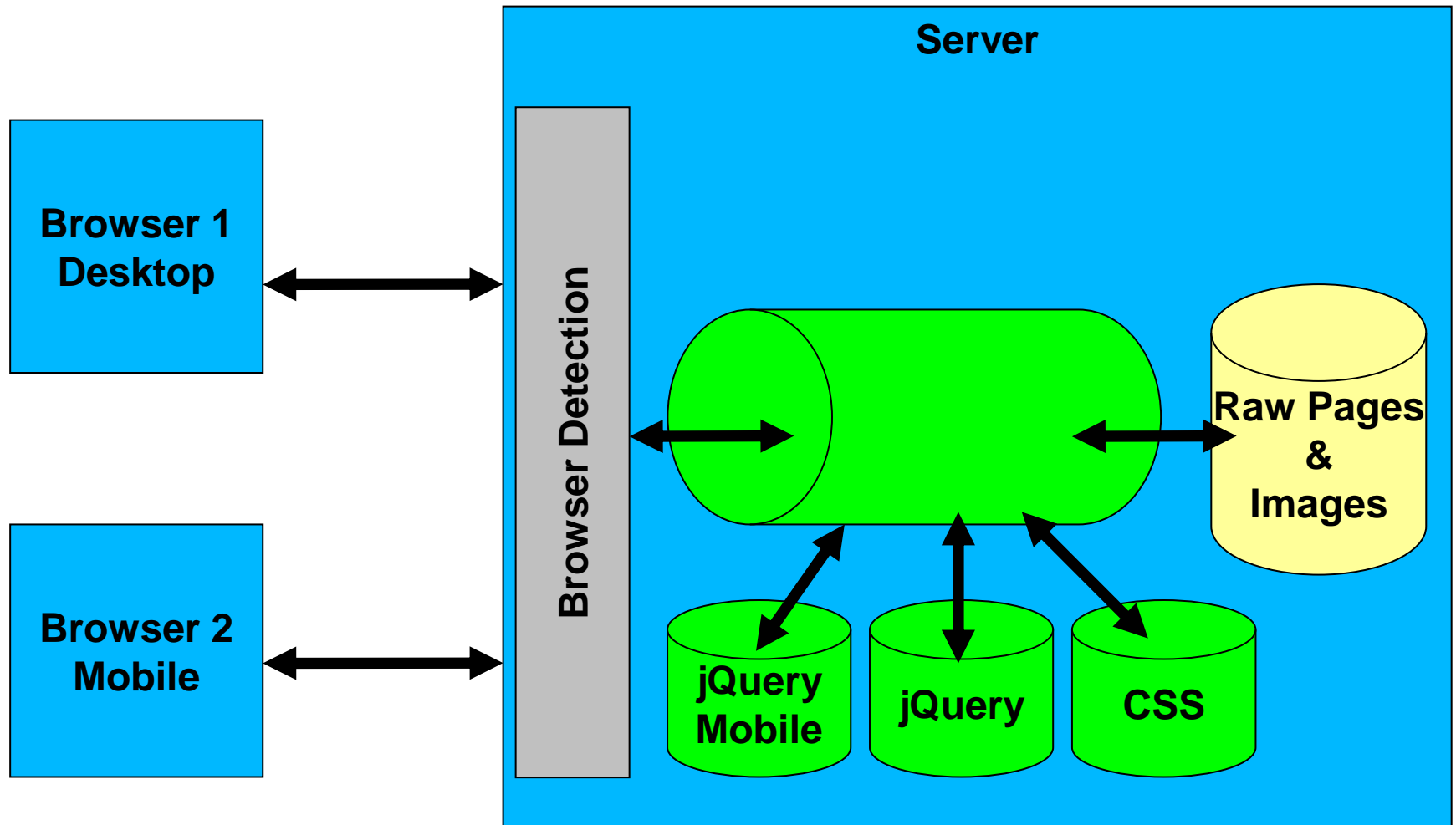
- Durch die ebenfalls vorhandene Fragmentierung des Browser-Markts müssen auch hier Strategien entwickelt werden, um
  - Wartbarkeit und
  - Wiederverwendbarkeit zu ermöglichen
- Zur Realisierung können sowohl Server- als auch Client-seitige Ansätze benutzt werden
  - Server
    - Server-side Adaption
    - Progressive Enhancement
  - Client
    - Graceful Degradation
    - Regressive Enhancement
    - Responsive Design

- Der Server stellt für jeden identifizierten Browser eine eigene Seite zur Verfügung
  - Desktop versus Mobile Version
- Die Unterscheidung übernimmt
  - der Server automatisch
    - beispielsweise mit einem Reverse Proxy
  - der Client durch Auswahl der Aufruf-URL
    - m.xyz.de





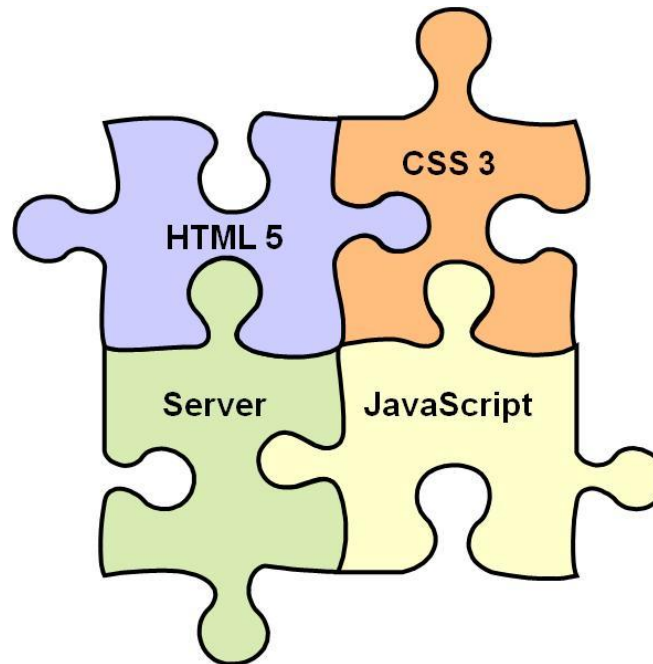
- Eine modulare Form der Server-seitigen Adaption
- Die Funktionalität der Anwendung wird durch verschiedene Schichten erreicht
  - Diese werden in dem Moment aktiv, in dem eine bestimmte Funktionalität vom Browser zur Verfügung gestellt wird
  - Der „raw“ Seite bleibt jedoch für alle Seiten identisch



- Der Browser entscheidet über das Design und Verhalten der Web-Seite
- Um dem Browser diese Entscheidung zu ermöglichen, müssen die Seiten jedoch bestimmten Vorgaben genügen
  - Einfaches Beispiel: Die HTML-Seite darf keine festen Größen- und Positionsangaben enthalten
- Die Browser muss Abfragelogik bereitstellen
  - Browsertyp
  - Bildschirm-Größe und Auflösung
  - Ändern der Darstellung bei einem Wechsel der Orientierung
- Gegebenenfalls muss die Seite in verschiedenen Umgebungen komplett anders dargestellt werden
  - Desktop: Feste Menüstruktur und „Tabbed Panes“
  - Kleingerät: Einblendbares Menü und Navigation zwischen Subseiten
- Beispielseiten
  - [mediaqueri.es](http://mediaqueri.es)

- Graceful Degradation
  - Umkehrung des Progressive Enhancements
  - Features, die von alten Browsern nicht unterstützt werden, werden von diesem „fehlerfrei ignoriert“
    - Allerdings ist damit die Funktionalität nicht garantiert!
- Regressive Enhancement
  - Hier werden alte Browser durch Ergänzungen/Workarounds/Hacks in die Lage versetzt, moderne Features zu unterstützen
    - Polyfills sind meistens JavaScript-basierte Lösungen
    - Web Shims implementieren ein API neu
    - HTML5 Shivs stellen HTML5-Features zur Verfügung

- Responsive Web Design verlagert die Verantwortung für die Anpassung der Seitendarstellung auf den Browser
  - Dies alleine kann nicht optimal funktionieren
    - Beispiel: Eine große Bilddatei müsste komplett Übertragen werden, um vom Browser dann skaliert zu werden
- Eine Kombination der Strategien ist notwendig



2.3

## **HYBRID APPLIKATIONEN**

- Eine Hybrid-Application benötigt ein Produkt, das auf allen unterstützten Plattformen installiert werden kann und intern eine homogene Programmier-Umgebung zur Verfügung stellt
- Die Programmierung der Seite erfolgt in einer Plattform-unabhängigen Sprache
  - Beispielsweise durch HTML5, CSS3, JavaScript
- Zugriff auf Ressourcen des Mobilgeräts möglich
  - Hierzu erweitert die Hybrid-Plattform beispielsweise das JavaScript-API um die entsprechenden Objekte
- Die Anwendung selbst ist native und wird über den Plattform-abhängigen Store verteilt
  - Damit ist pro zu unterstützendem Device ein eigener Build-Prozess notwendig

- Die Java-Laufzeitumgebung ist auf allen relevanten Betriebssystemen und Plattformen vorhanden
  - Insbesondere auf dem Server und dem Desktop
- Auch für Mobil-Geräte kann Java eingesetzt werden
  - Dafür existiert sogar eine eigene „Java Micro Edition“-Spezifikation (JME)
- Allerdings ist aktuell wenig Akzeptanz zu erkennen, Java-Standards von Oracle zu benutzen
  - Android
    - Eigene Java-Version ohne Bezug zu JME
  - iOS
    - Keinerlei Bereitschaft zur offiziellen Unterstützung von Java
  - Windows
    - Noch keine klare Linie zu erkennen
    - Android Apps für Windows möglich



3

## **MOBILE PLATTFORMEN**

3.1

## **INFRASTRUKTUR**

- Android
  - Linux-basiert
    - Prinzipiell Open Source
    - C-basierte Anwendungsprogrammierung möglich (Native Development Kit, NDK)
- iOS
  - Unix-basiert
  - Geschlossen mit streng reglementiertem Lizenzmodell
- Windows Phone
  - Windows
  - Geschlossen mit teilweiser Öffnung zu Providern

- Android
  - Programmiersprache Java
    - Ergänzt durch natives C
  - Entwicklungsumgebungen als Plugins für Standard-Java-Werkzeuge
    - Eclipse, Netbeans
  - Debugger, Profiler, Simulator
- iOS
  - Programmiersprache Objective C
    - Apple propagiert stark die neue Sprache „Swift“
  - XCode, frei erhältlich für MAC
- .NET
  - Programmiersprachen C#, VB.NET
  - Visual Studio

- Android
  - Android-interner Browser
  - Chrome
  - Beliebige Browser-Apps
    - Firefox, Opera
- iOS
  - Safari
- Windows Phone
  - Internet Explorer
  - Beliebige Browser-Apps
    - Firefox, Opera

- Android
  - Prinzipiell kann das Android-Betriebssystem selbst compiliert und installiert werden
  - Standard-Funktionen
    - https
    - VPN
    - Installation von Zertifikaten durch Benutzer möglich
- iOS und Windows Phone
  - Geschlossene Systeme
  - Enterprise Umgebungen für Benutzer-Verwaltung, Installation von Zertifikaten etc.
  - https, VPN

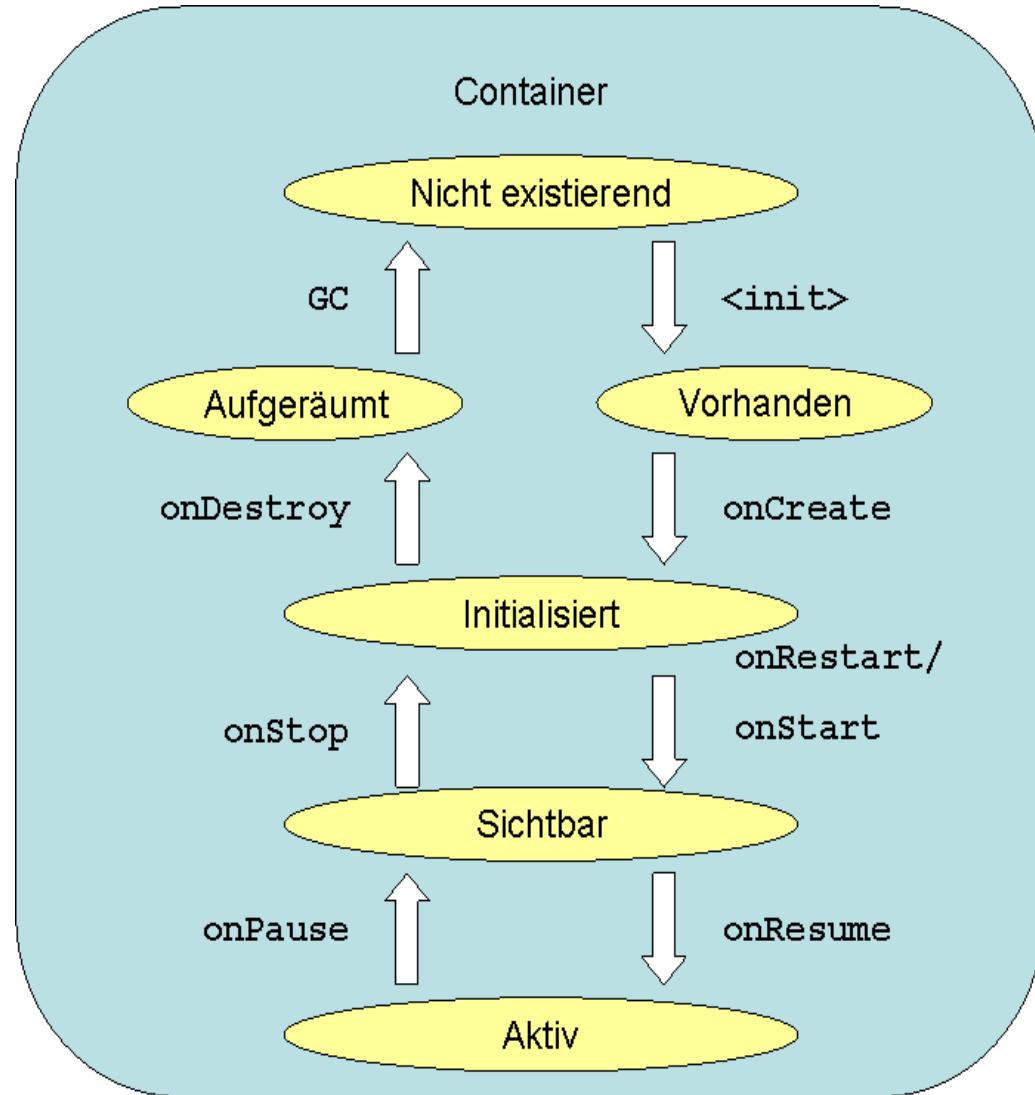
- Android
  - Wenig bis keine Einschränkungen und Prüfungen
  - Freie erhältliche Lizenz notwendig
- iOS
  - Strenge Vorgaben mit Prüfung der Einhaltung
  - Kostenpflichtige Lizenz
- .NET-Apps
  - Strenge Vorgaben
  - Freie Lizenz möglich

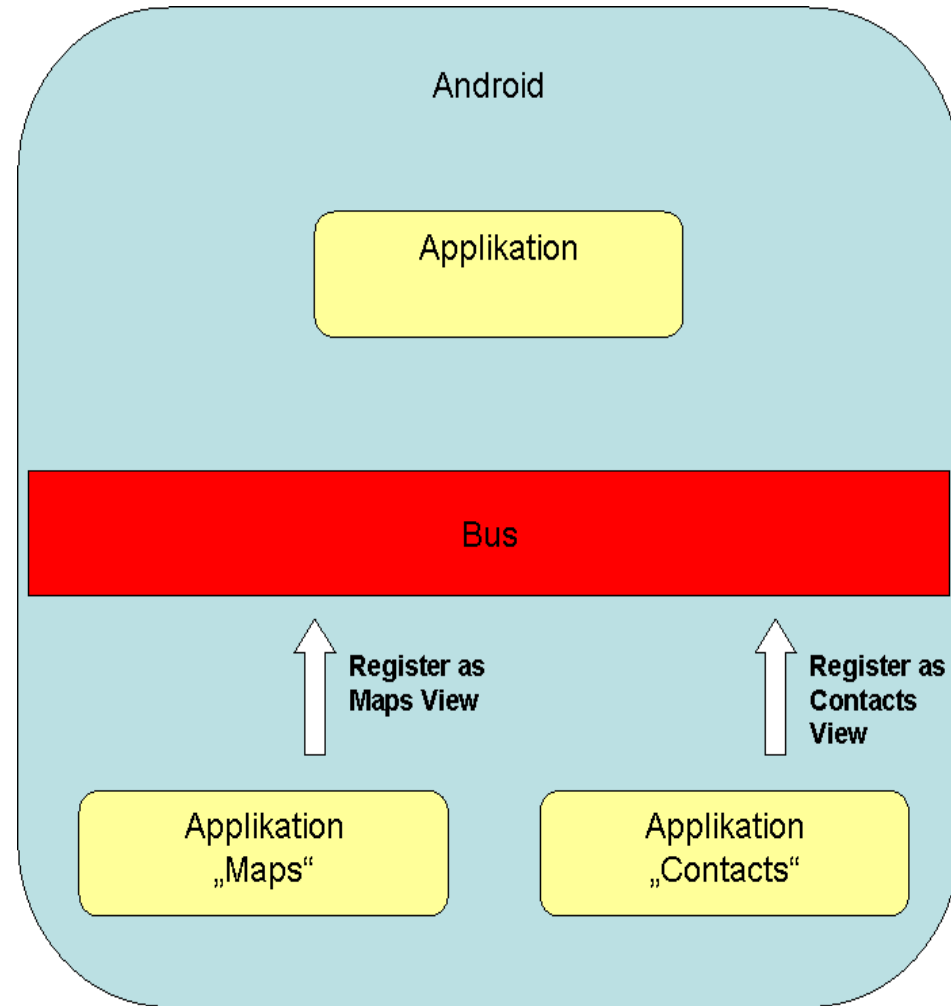
3.2

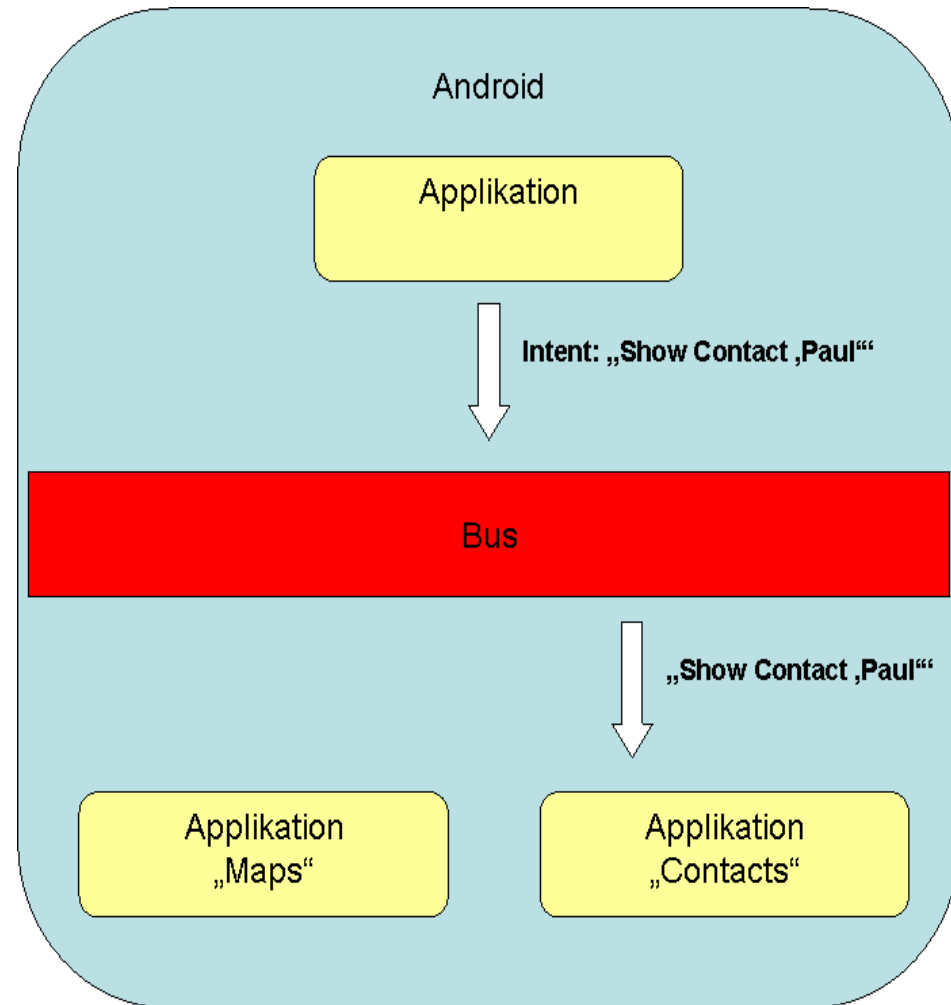
## **PROGRAMMIERUNG**

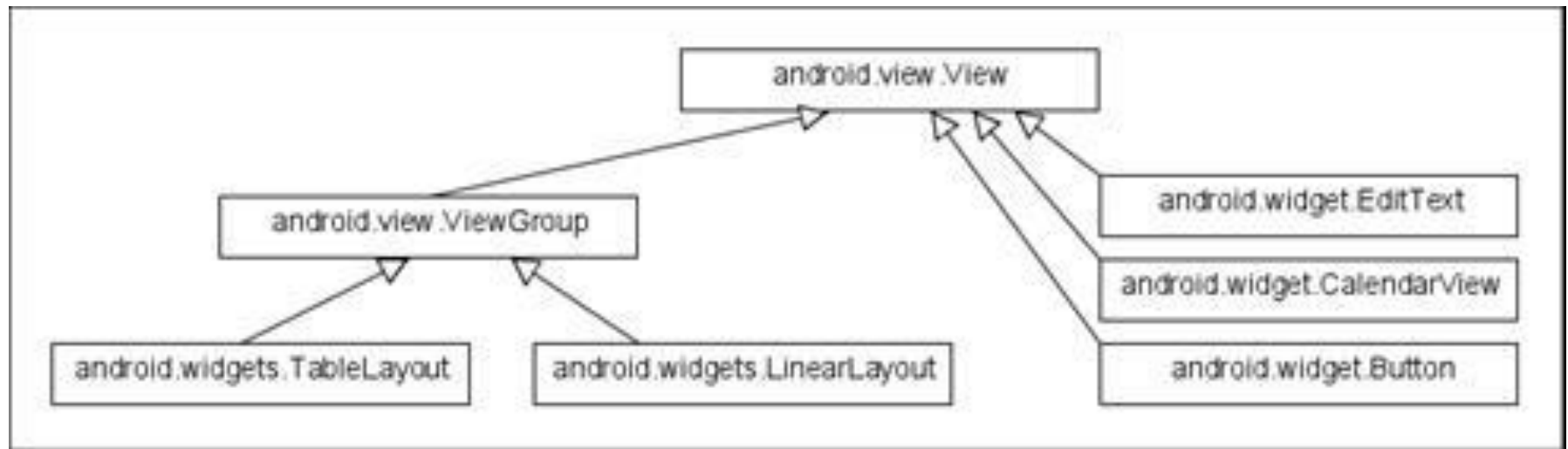


- Container und Lifecycle
- Benachrichtigungen und internes Bus-System
- Widget-Bibliotheken und Layouts
- Event-Modell
  - Event Queue und Dispatching
- Anwendungs-Design mit Entwurfsmuster Model-View-Controller (MVC)



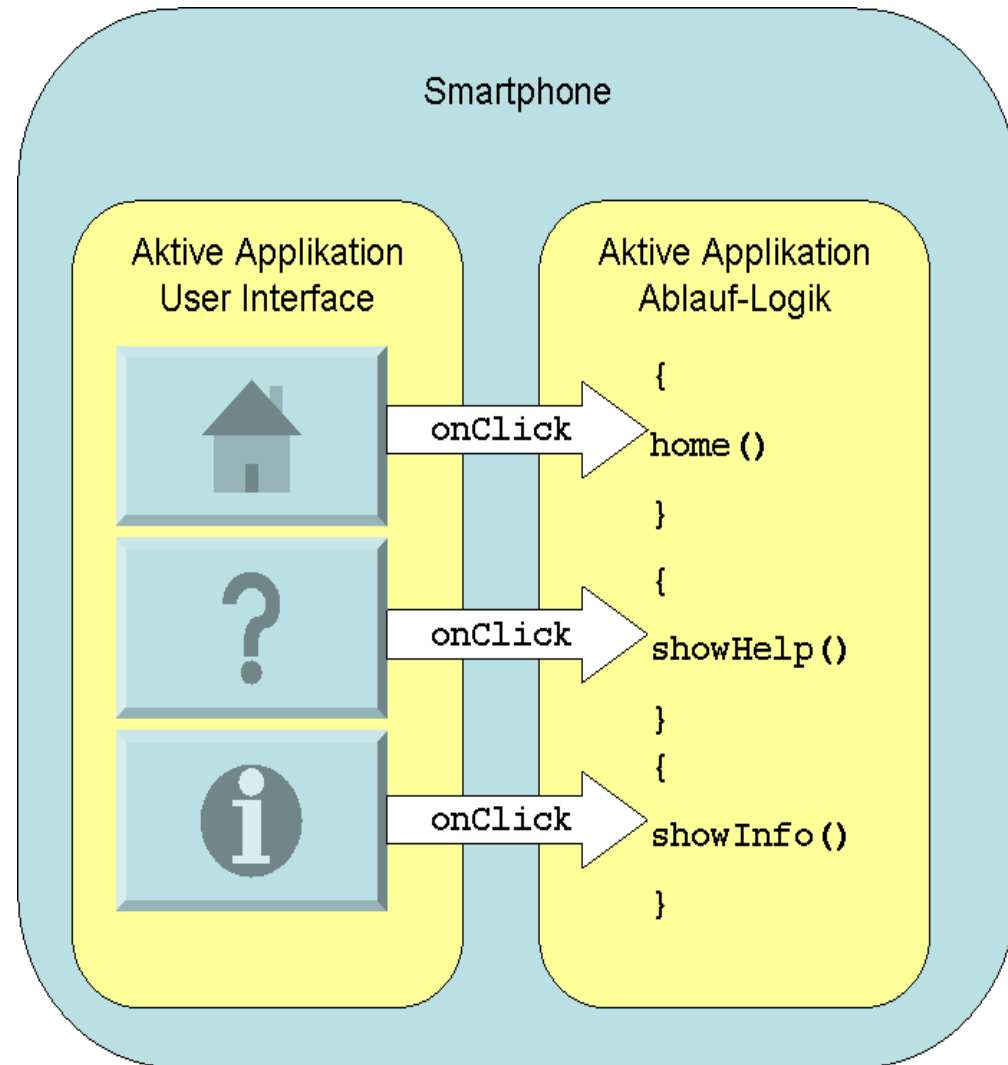


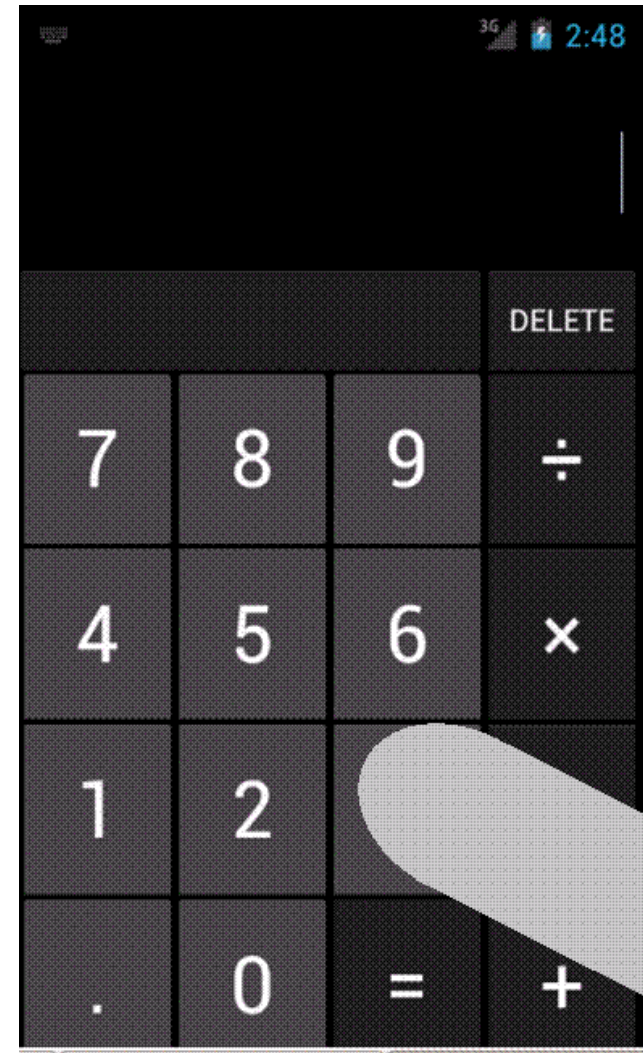




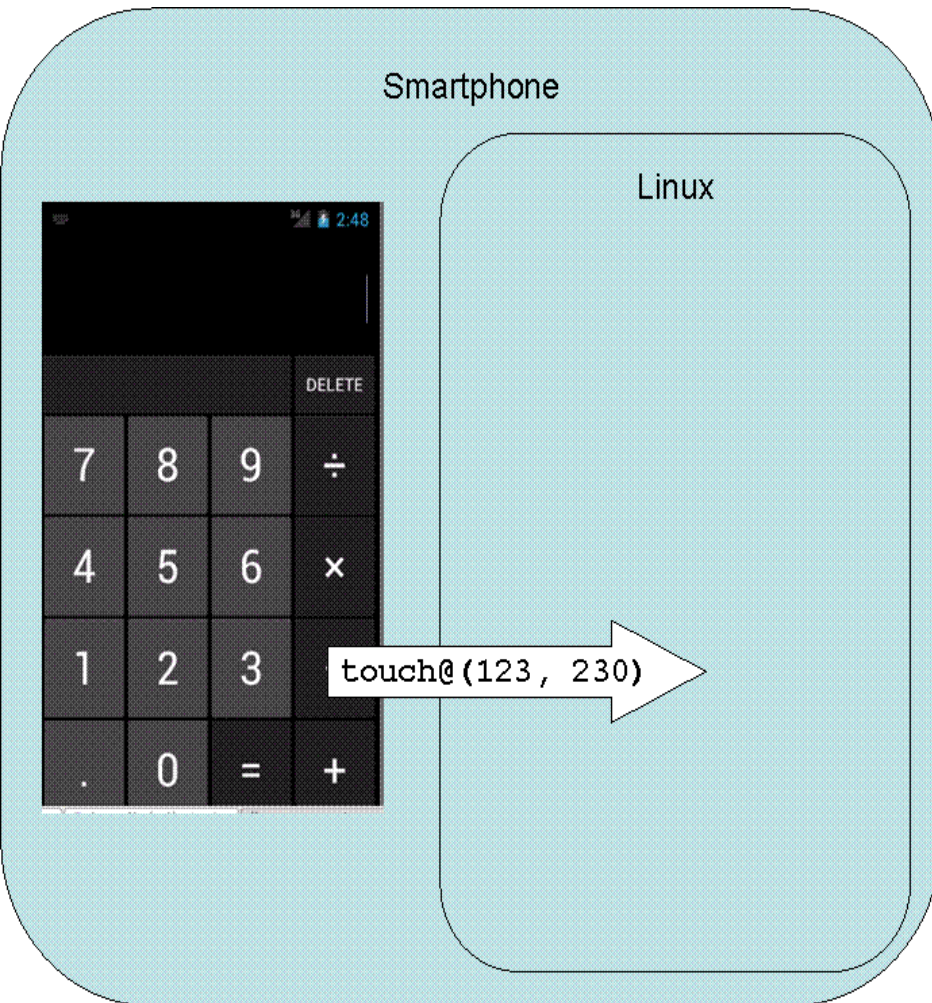
3.3

## **EVENT-MODELL**

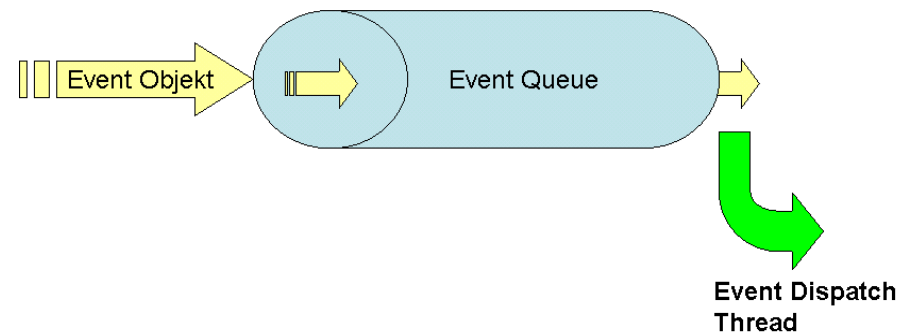
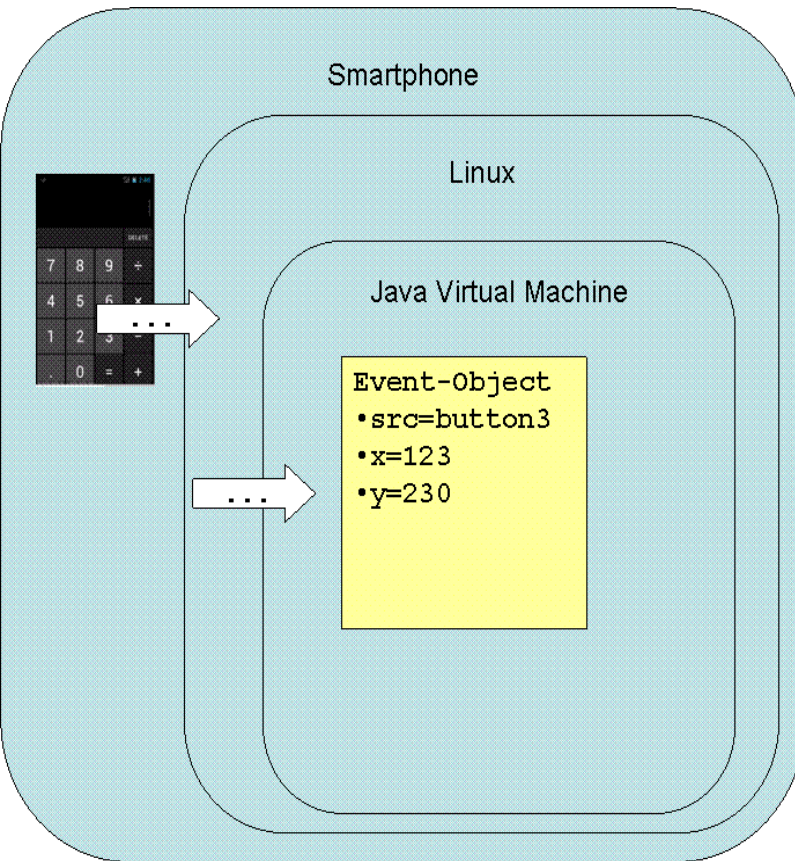


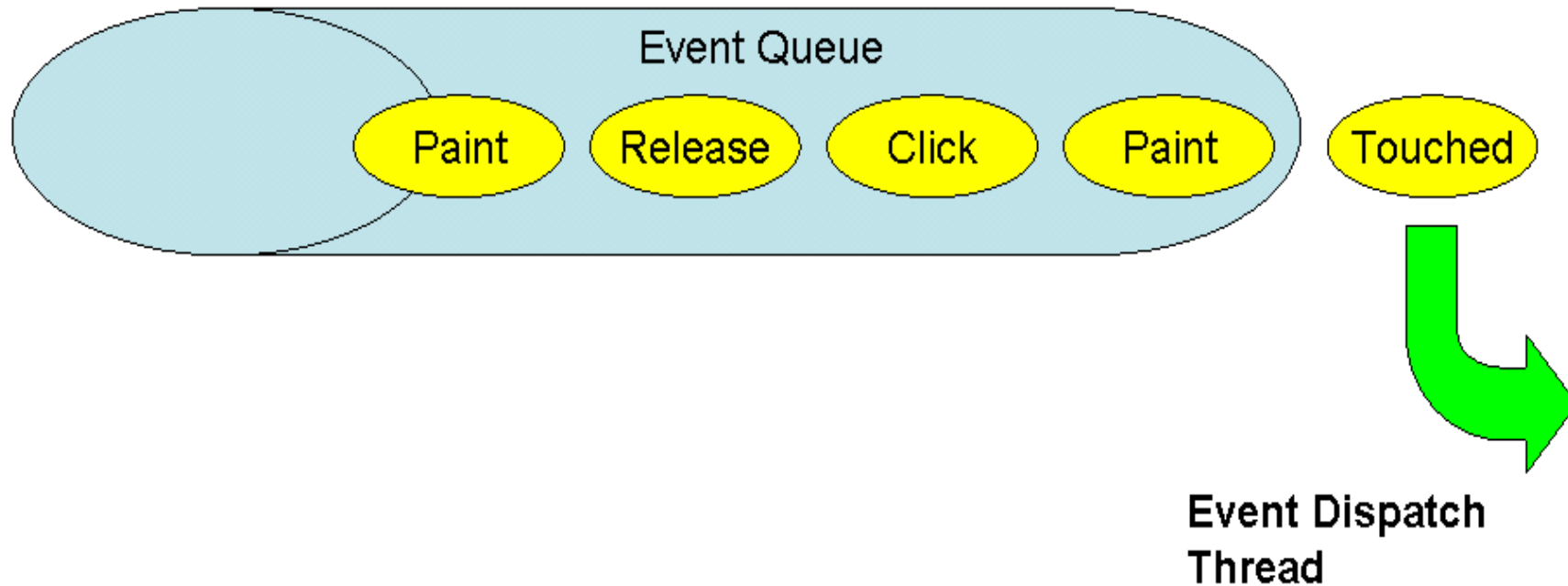






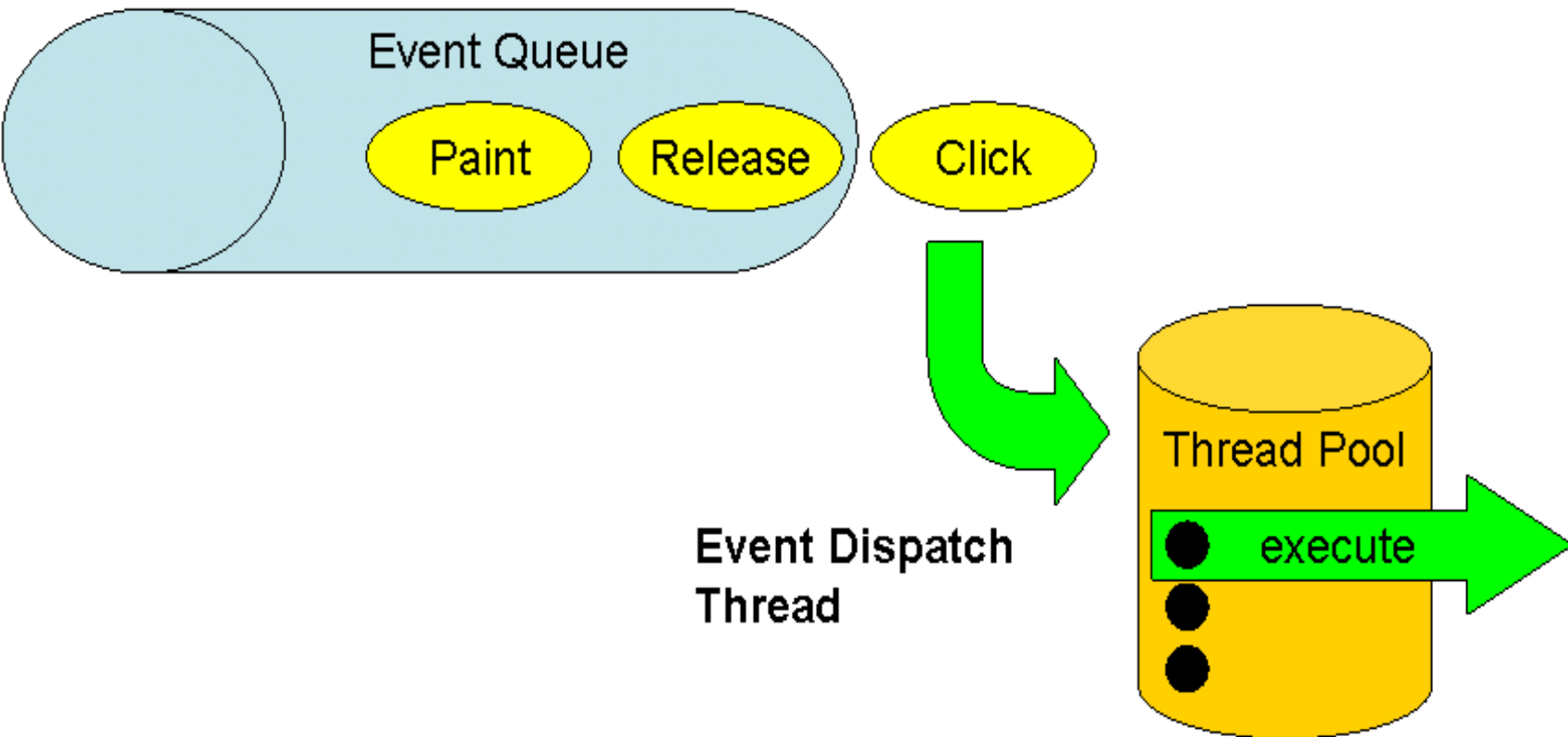
# Dispatching an die Virtuelle Maschine



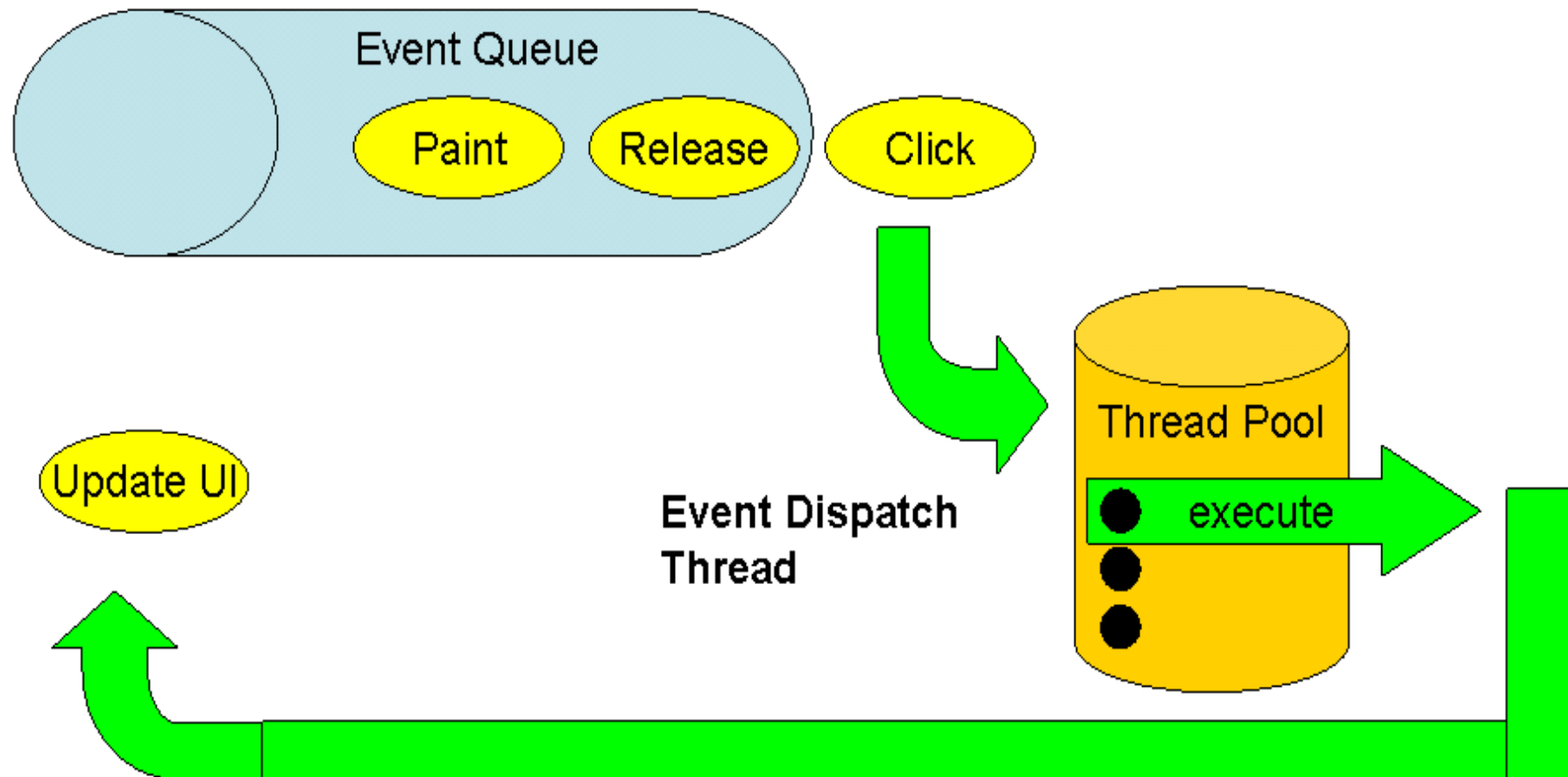


<b>View.OnClickListener</b>	<b>Klick auf die View</b>
<b>View.OnFocusChangeListener</b>	<b>Fokus-Wechsel</b>
<b>View.OnHoverListener</b>	<b>Bewegung über die View</b>
<b>View.OnKeyListener</b>	<b>Drücken einer Taste</b>
<b>View.OnLongClickListener</b>	<b>Drücken und Halten der View</b>
<b>View.OnTouchListener</b>	<b>Berühren der View</b>

- Listener-Methoden dürfen nicht lange dauern
  - Aufmerksame Power-User bemerken bereits Verzögerungen ab 50 (!) Millisekunden
  - Damit müssen länger-laufende Aktionen von einem Worker abgearbeitet werden
    - Die Aktualisierung der Oberfläche muss dann aber wieder über einen Event initiiert werden
    - Pro Aktion sind damit mindestens zwei Thread-Kontextwechsel notwendig



# Aktualisierung der Oberfläche

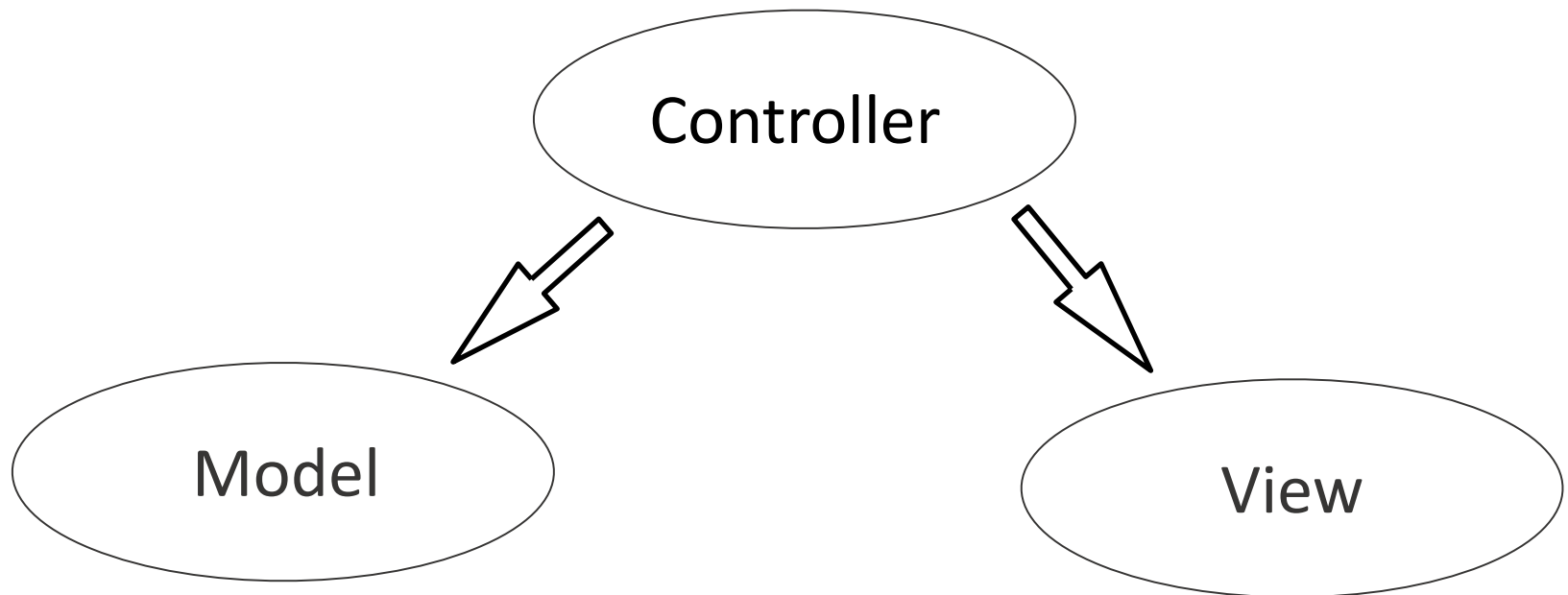


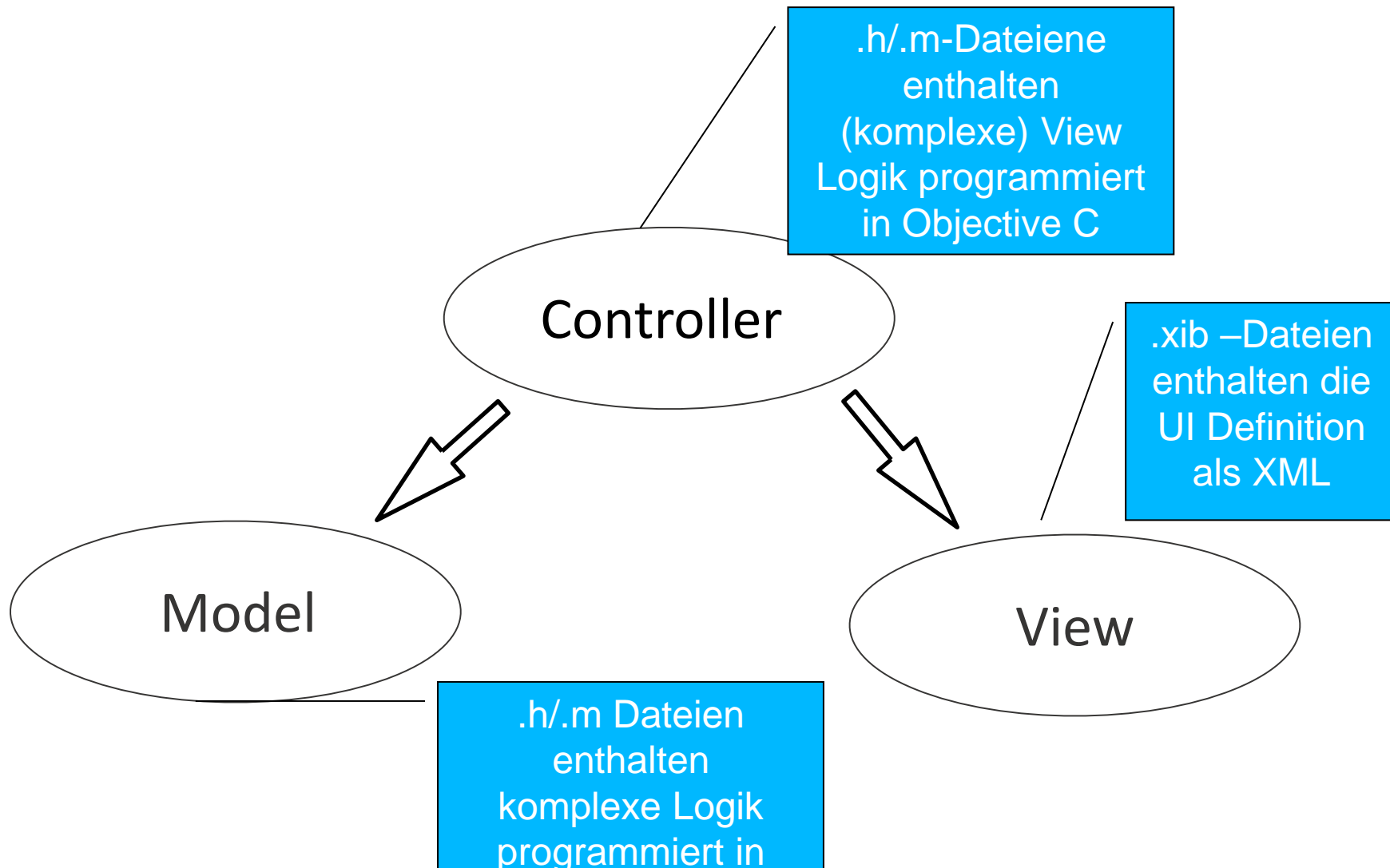
- `android.os.AsyncTask<Param, Process, Result>`
  - Durch ein mehr oder weniger simples Erben von dieser Klasse können Aktionen in den jeweils richtigen Threads ausgeführt werden:
    - `doInBackground` bekommt Parameter des angegebenen generischen Param-Typs und läuft in einem externen Thread
    - `onPostExecute` bekommt als Parameter den Result-Typ und läuft im Event Dispatch Thread
  - Die Typen `Param` und `Process` sind hier nicht notwendig



3.4

## **MODEL VIEW CONTROLLER**





4

## **VERTIEFUNG ANDROID**

4.1

## **DAS BETRIEBSSYSTEM**



android

- Die Hardware-Ausstattung von Smartphones ist im Gegensatz zu den mittlerweile doch sehr stark vereinheitlichten Desktop-Systemen noch äußerst heterogen
  - Dies betrifft nicht nur eher triviale Unterschiede in der Bildschirmauflösung und im vorhandenen Speicher sondern beispielsweise auch Dienste wie Bluetooth, GPS oder die Nahfeld-Kommunikation
- Android selbst verlangt vom Hersteller eines Smartphones oder Tablets relativ wenig standardisierte Hardware und läuft deshalb auch auf einer sehr breiten Produktpalette
  - Weiterhin werden bei Updates auch ältere Plattformen in erstaunlichem Umfang unterstützt



















- Diese potenziell heterogenen Hardware-Voraussetzungen werden am Besten durch ein Linux-basiertes Betriebssystem gekapselt
- Linux ist ein Open Source-Produkt und kann deshalb auf einen weiten Hardware-Bereich hin optimiert und kompiliert werden
- Google spezifiziert die notwendigen Features des Android-Kernels im Gegensatz zur darunter liegende Hardware-Plattform exakt


























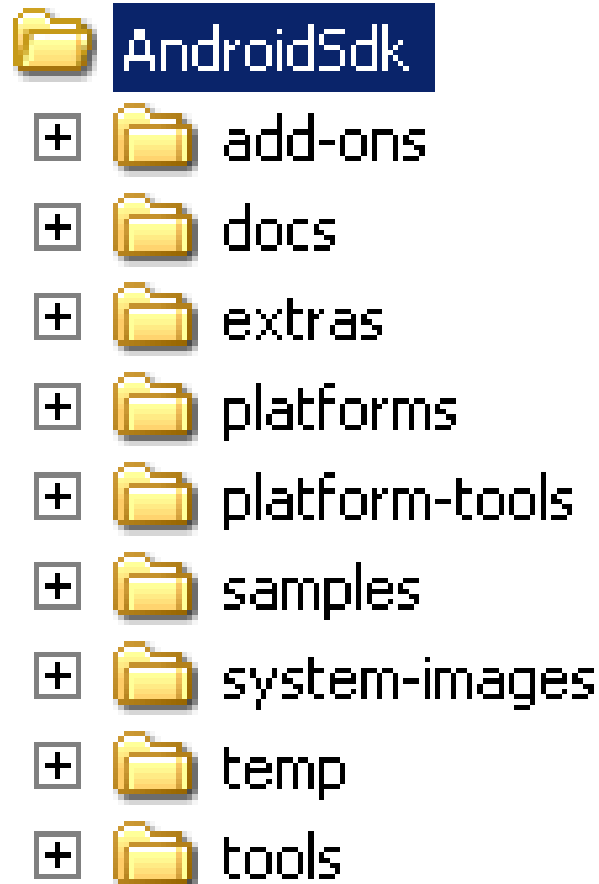
- Die JVM selbst ist eine native C-basierte Anwendung, die für alle gängigen Betriebssysteme implementiert wurde
- Die JVM ist ein Interpreter für compilierte Java-Programme, dem so genannten Bytecode
  - Bytecode ist Plattform-unabhängig konzipiert und deshalb laufen Java-Anwendungen auf allen Systemen, für die eine JVM existiert
- Die JVM enthält als zentrales Konzept einen Garbage Collector
- Sämtliche Ressourcen-Zugriffe werden über einen Security Manager kontrolliert
- Die JVM stellt einen Fehler-toleranten Exception-Mechanismus zur Verfügung
- Die allermeisten Android-Applikationen laufen in einer speziell auf den Android-Kernel hin optimierten Virtual Machine, der Dalvik-VM

- Die Entwicklung von Android-Applikationen ist pures Java:
  - Syntax,
  - Sprachspezifikation,
  - Bytecode-Format
  - etc. sind komplett übernommen.
- Ebenso können etablierte Verfahren wie grundlegende Design-Umsetzungen, Programmierrichtlinien und Best Practices direkt übernommen werden
- Weiterhin sind alle Java-Werkzeuge wie IDEs, Debugger und Profiler sowie vorhandene Open Source-Bibliotheken zumindest teilweise für Android geeignet

# Oracle JDK versus Android SDK

Archive Edit View Help			
<div>  Open ▾  Extract   </div>			
<div> Back    Location:  /</div>			
Name ▾	Size	Type	Date Modified
 com	20.8 MB	Folder	
 java	5.5 MB	Folder	
 javax	7.4 MB	Folder	
 META-INF	2.3 KB	Folder	
 org	1.0 MB	Folder	
 sun	12.2 MB	Folder	
 sunw	644 bytes	Folder	
7 objects (46.9 MB)			

Archive Edit View Help			
<div>  Open ▾  Extract   </div>			
<div> Back    Location:  /</div>			
Name ▾	Size	Type	Date Modified
 android	2.2 MB	Folder	
 assets	233.8 KB	Folder	
 com	245 bytes	Folder	
 dalvik	14.5 KB	Folder	
 java	1.2 MB	Folder	
 javax	213.3 KB	Folder	
 junit	17.0 KB	Folder	
 META-INF	71 bytes	Folder	
 org	504.6 KB	Folder	
 res	3.5 MB	Folder	
 AndroidManifest.xml	49.9 KB	XML docu...	03 February 2011, 14:49
 resources.arsc	3.7 MB	unknown	03 February 2011, 14:49
12 objects (11.7 MB)			



Packages Tools

SDK Path: /home/user/Documents/tools/android-sdk-linux/

Packages

Name	API	Rev.	Status
Android 3.1 (API 12)			
Android 3.0 (API 11)			
Android 2.3.3 (API 10)			
SDK Platform	10	2	Installed
Samples for SDK	10	1	Installed
DTS Add-On by KYOCERA Corporation	10	1	Installed
DTS Add-On by KYOCERA Corporation	10	1	Installed
EDK 1.1 by Sony Ericsson Mobile Comm	10	1	Installed
DTS Add-On by KYOCERA Corporation	10	1	Installed
Google APIs by Google Inc.	10	2	Installed
Extras			

Show: ☒ Updates/New ☒ Installed ☐ Obsolete Select [New](#) or [Updates](#)

Sort by: ☒ API level ☐ Repository [Deselect All](#)







Install packages...




Selects all items that are either new or updates.






Delete packages...

Done loading packages.

Archiv Bearbeiten Ansicht Hilfe

  Öffnen ▾ |  Entpacken |   | 

◀ Zurück ▶   Ort:  /

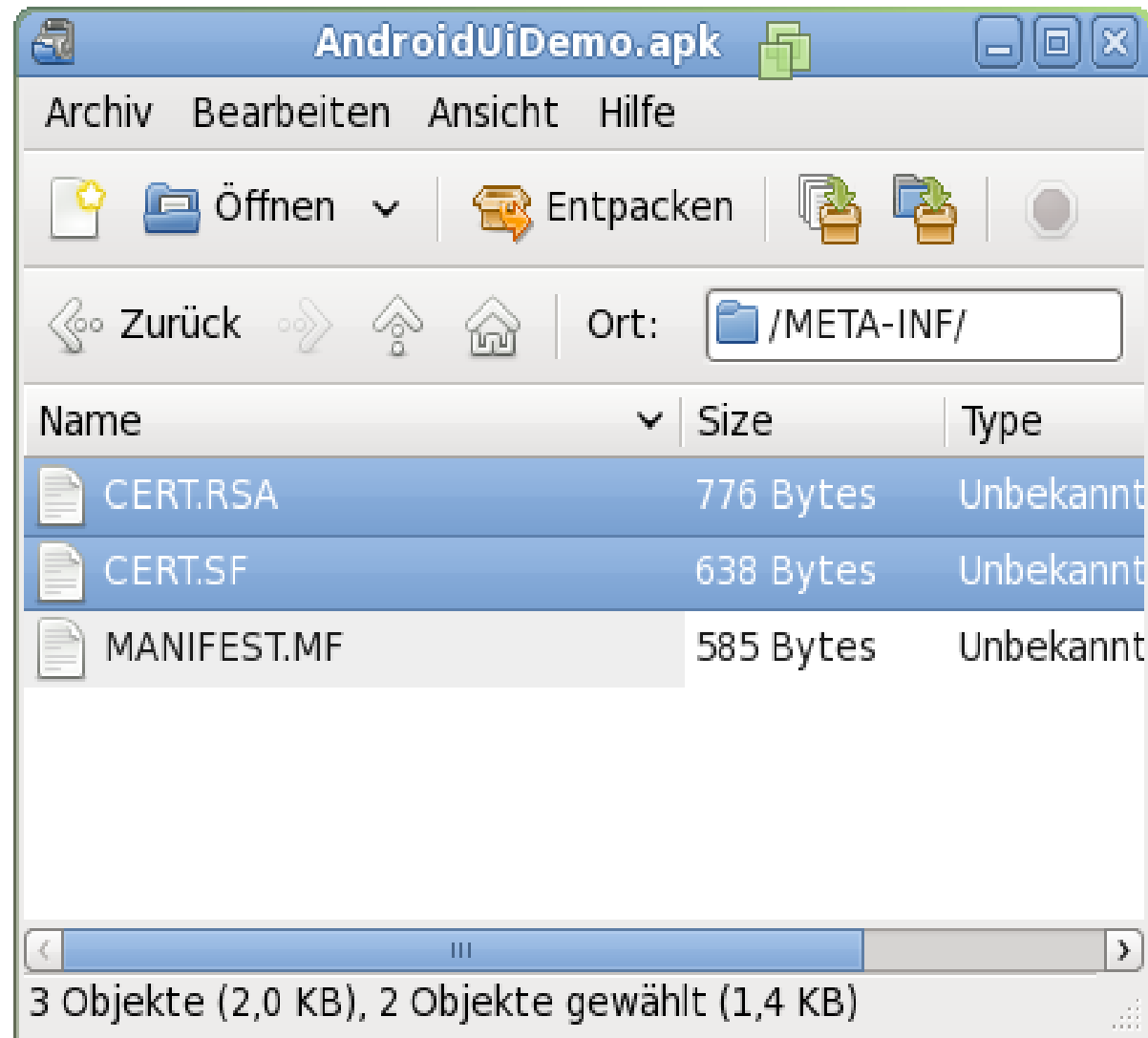
Name ▾	Size	Type
 META-INF	2,0 KB	Ordner
 res	8,2 KB	Ordner
 AndroidManifest.xml	1,5 KB	XML-Doku
 classes.dex	3,0 KB	Unbekannt
 resources.arsc	1,3 KB	Unbekannt

5 Objekte (15,9 KB)

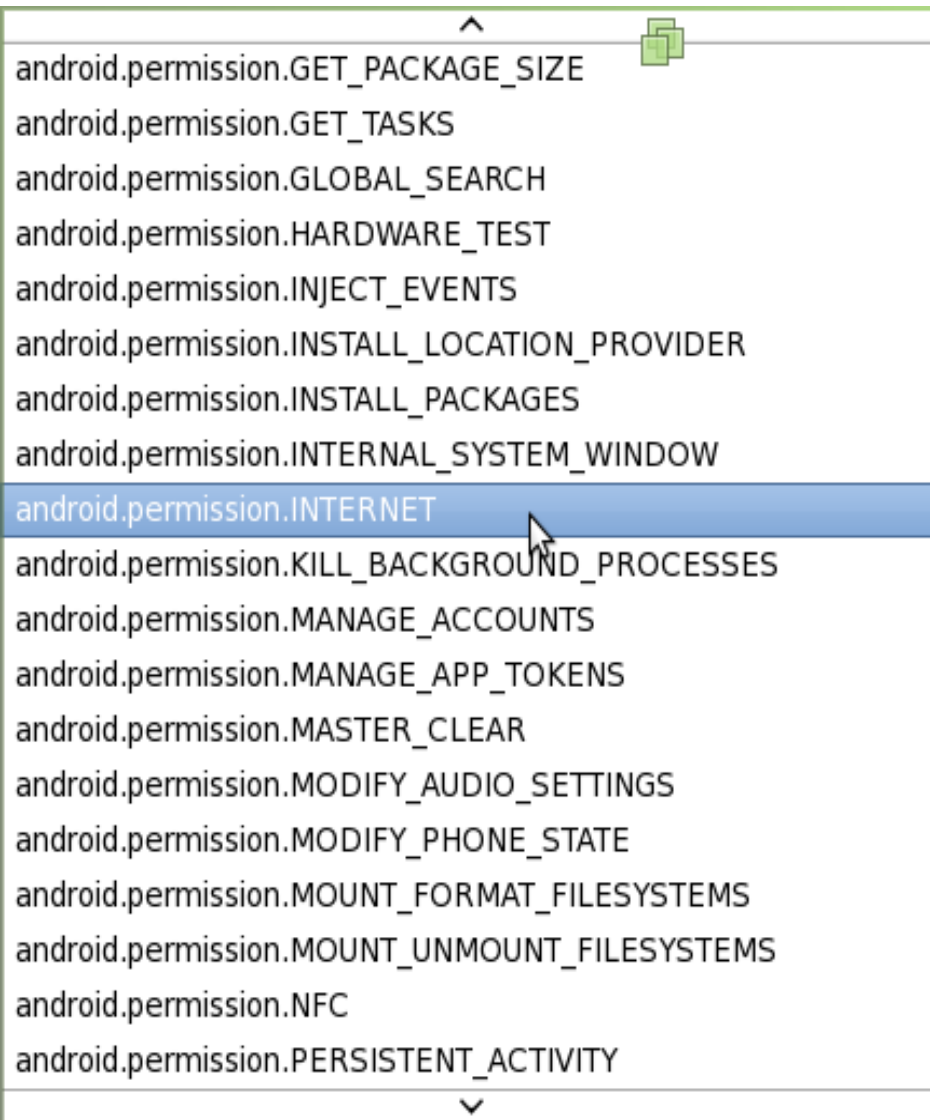
- Das Deployment der Anwendung kann auf mehrere Arten erfolgen:
  - Aus der Entwicklungsumgebung durch Starten eines Emulators
  - Auf ein über USB-Kabel verbundenes Device im Debug-Modus
  - Über den Google Play Store
  - Bei allen Verfahren wird effektiv das erzeugte Archiv transferiert. Allerdings verlangt der Google Market eine Registrierung sowie ein signiertes Archiv. Auf Security-Aspekte wird im Folgenden Abschnitt eingegangen.

- Signierte Anwendungen
  - Das Android-Archiv kann mit normalen Java-Werkzeugen signiert werden. Dies ist insbesondere deshalb wichtig, weil ein Android-Gerät ausschließlich signierte Anwendungen ausführen lässt.
  - Die Android Tools erzeugen und benutzen dafür ein Debug-Zertifikat.
  - Dieses ist jedoch für ein Verteilen der Anwendung über den Google Play-Markt nicht gültig und muss durch ein Developer-Zertifikat ersetzt werden, in dem der Hersteller eindeutig zugewiesen werden kann.
- Die Signatur verhindert nachträgliche Änderungen der Dateien





# Permissions

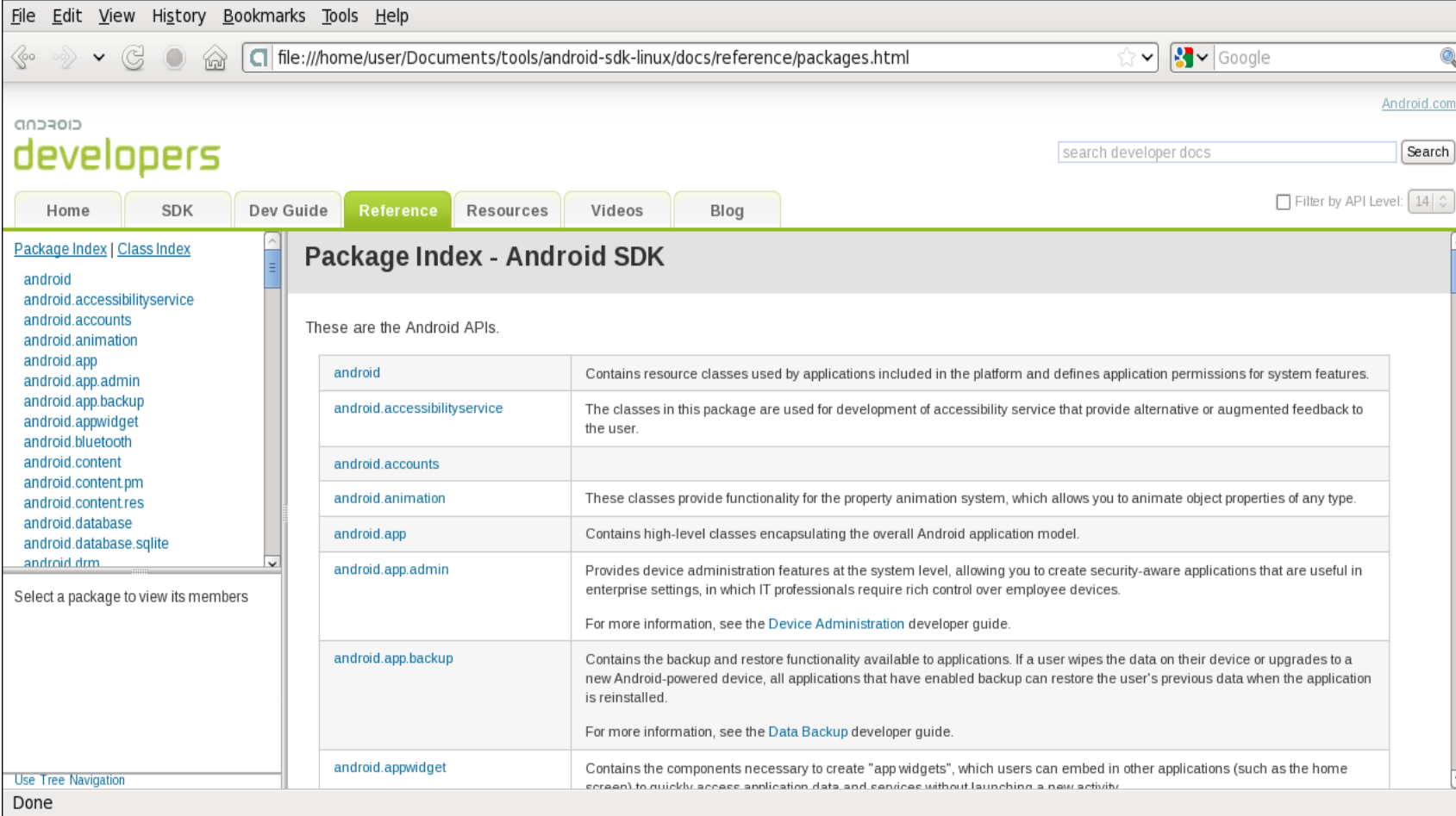


- Als Programme mit Benutzer-Oberfläche sind Android-Anwendungen nicht trivial zu testen
- Android integriert in seiner Klassenbibliothek das etablierte JUnit-Framework und stellt selbst Erweiterungen zur Verfügung
- JUnit-Tests
  - Mit Hilfe der Basis-Klasse `junit.framework.TestCase` können sämtliche Klassen der Anwendung, die keinen UI-Bezug besitzen sowie keine Dienste des Devices benötigen getestet werden
  - Näheres zu diesen JUnit-Tests ist im Anhang zu finden.

4.2

## **UI-PROGRAMMIERUNG**

- Android enthält in seiner Java-Bibliothek einen kompletten Satz von Oberflächen-Komponenten, den sogenannten Widgets, die zur Definition eines User Interfaces benutzt werden können
- Neben den Widget-Komponenten, die eine direkte visuelle Repräsentation besitzen, existieren die Layouts
  - Diese sind Container für Widgets
- Layouts und Widgets befinden sich im Paket `android.widget`. die gemeinsame Basisklasse ist `android.view.View`
- Die Widget-Klassen selbst stehen in einer flachen Vererbungshierarchie zueinander und enthalten Methoden und Eigenschaften, die eine Anpassung der Oberfläche an die geforderten Vorgaben ermöglichen.

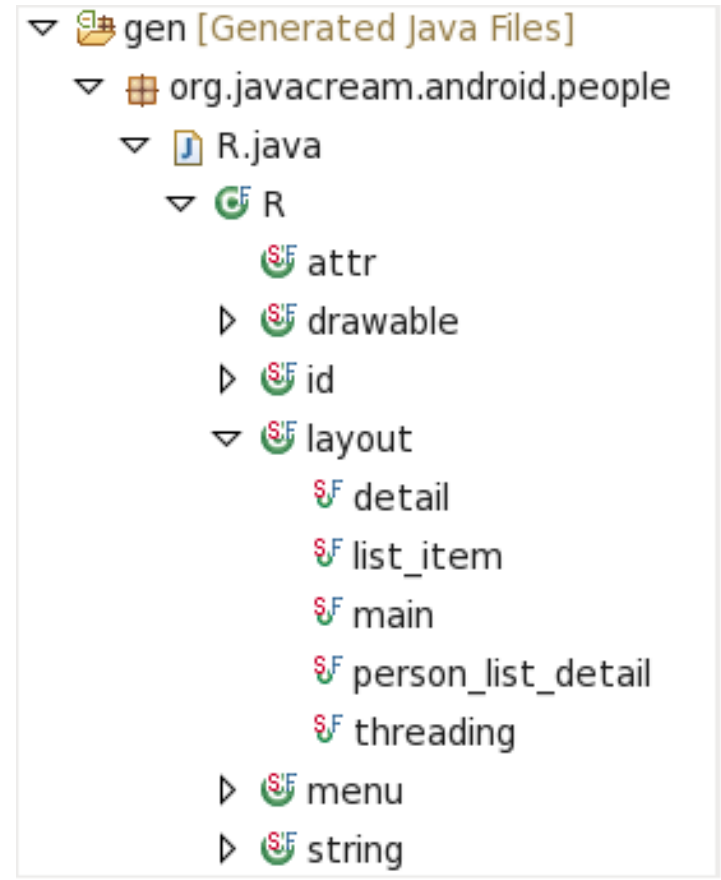
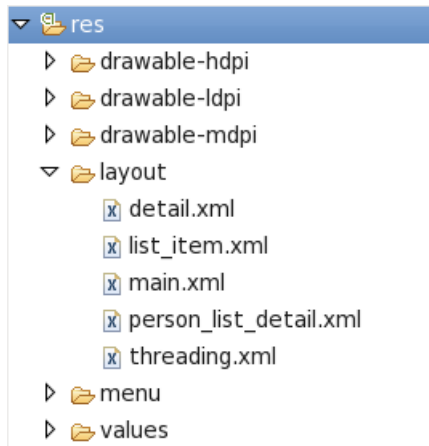


The screenshot shows a web browser window displaying the Android SDK Reference documentation. The address bar shows the file path: `file:///home/user/Documents/tools/android-sdk-linux/docs/reference/packages.html`. The page title is "Package Index - Android SDK". The left sidebar contains a list of packages under the heading "Package Index | Class Index". The main content area lists several packages with their descriptions.

**Package Index - Android SDK**

These are the Android APIs.

<a href="#">android</a>	Contains resource classes used by applications included in the platform and defines application permissions for system features.
<a href="#">android.accessibilityservice</a>	The classes in this package are used for development of accessibility service that provide alternative or augmented feedback to the user.
<a href="#">android.accounts</a>	
<a href="#">android.animation</a>	These classes provide functionality for the property animation system, which allows you to animate object properties of any type.
<a href="#">android.app</a>	Contains high-level classes encapsulating the overall Android application model.
<a href="#">android.app.admin</a>	Provides device administration features at the system level, allowing you to create security-aware applications that are useful in enterprise settings, in which IT professionals require rich control over employee devices.  For more information, see the <a href="#">Device Administration</a> developer guide.
<a href="#">android.app.backup</a>	Contains the backup and restore functionality available to applications. If a user wipes the data on their device or upgrades to a new Android-powered device, all applications that have enabled backup can restore the user's previous data when the application is reinstalled.  For more information, see the <a href="#">Data Backup</a> developer guide.
<a href="#">android.appwidget</a>	Contains the components necessary to create "app widgets", which users can embed in other applications (such as the home screen) to quickly access application data and services without launching a new activity.



- Dafür stellt die Klasse Activity die Methode findViewById zur Verfügung
- Das Prefix @+id bewirkt, dass die generierte Klasse eine Enumeration Namens inputLastname bekommt, die zum Zugriff auf das Eingabeelement benutzt werden kann
  - Values
  - Auch Konstanten der Anwendung werden in XML-Dateien externalisiert und durch Enumerationen angesprochen
  - Diese Werte können sowohl innerhalb der Java-Anwendung als auch innerhalb einer Layout-Definition benutzt werden
  - In letzterem Falle wird wiederum ein spezielles Prefix, nämlich @string, benutzt



- Komplexere Widgets wie Listen oder Auswahlfelder benötigen für ihre Darstellung komplexere Datenstrukturen
  - So könnte beispielsweise die Aufgabe darin bestehen, eine Personen-Informationen in einer Liste darzustellen.
  - Die Klasse Person selbst ist eine einfache JavaBean ohne Bezug zum Android-API
- Layout
  - Eine Liste benötigt eine eigene Layout-Definition
  - Diese wird wie üblich im Layout-Ordner gespeichert
- Adapter
  - Layout und Daten werden von einem Adapter gekoppelt
  - Dieser erbt vom Android-BaseAdapter und implementiert dessen abstrakte Methoden
- Zur Darstellung der Liste genügt nun eine einfache Instanziierung der obigen Implementierung und Verwendung innerhalb einer ListView

- Toast
  - Mit Hilfe der Toast-Klasse kann ein simpler Dialog eingeblendet werden, der auch ohne Benutzer-Interaktion nach Ablauf einer gewissen Zeit verschwindet
  - Die Länge der Darstellung wird dabei durch eine der statischen Konstanten LENGTH\_SHORT und LENGTH\_LONG bestimmt
- Statuszeile
  - Auch in die Statuszeile des Mobilgeräts können Meldungen integriert werden

4.3

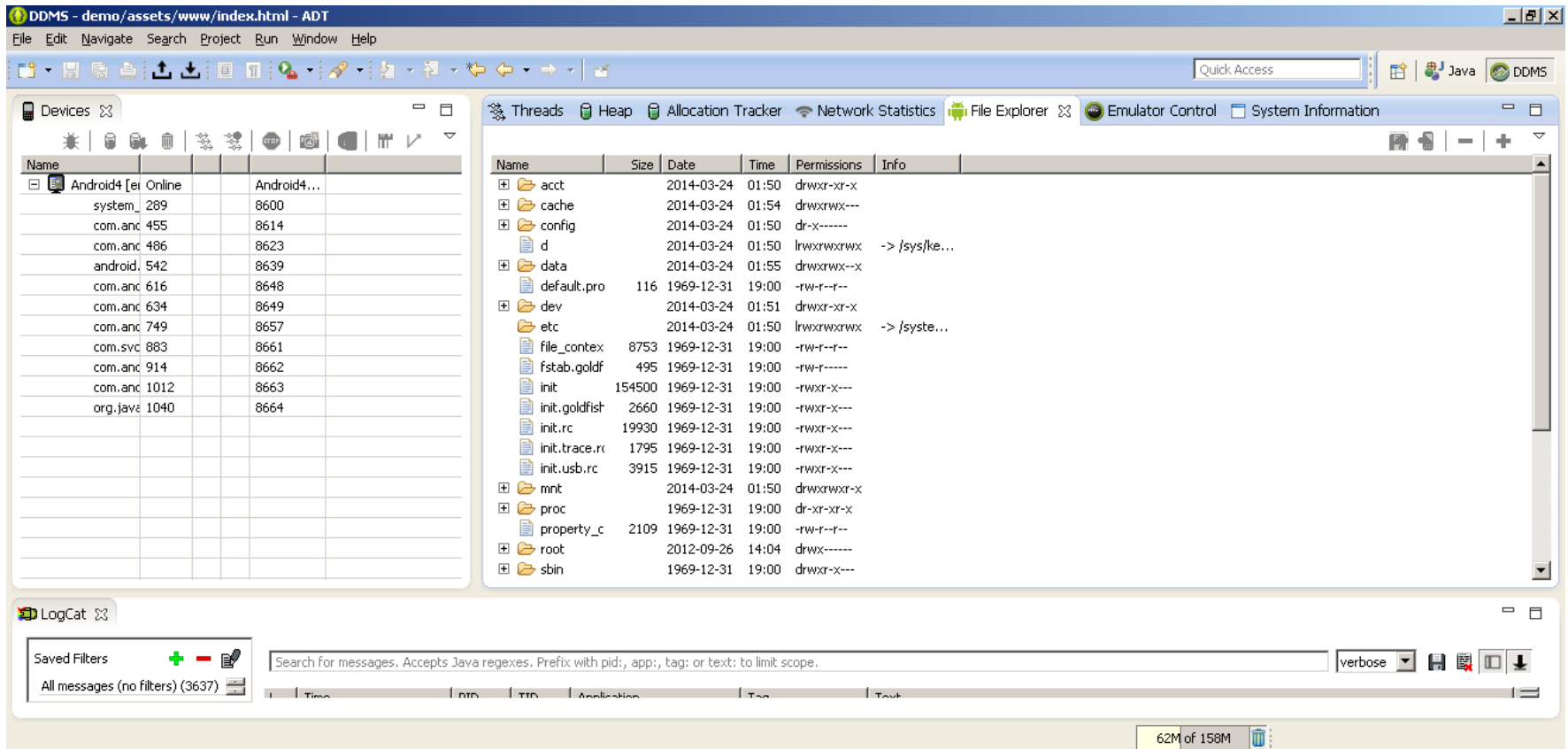
## **RESSOURCEN-ZUGRIF**

- Eine Ressource
  - Wird vom Betriebssystem und der Hardware verwaltet.
  - Der Applikation über einen „Handle“ zur Verfügung gestellt. Dieser wird von der Java Virtual Machine als Objekt zur Verfügung gestellt.
  - Ressourcen sind relativ teuer und beschränkt und müssen deshalb im Rahmen der Anwendung gesondert angefordert und, noch viel wichtiger: unbedingt auch wieder geschlossen werden.
- Ressourcen werden im Folgenden zum Senden und Empfangen von Daten benutzt
  - Shared Preferences. Dies sind flache typisierte Properties, die intern in Dateien abgelegt werden. Android übernimmt hierbei die gesamte Ressourcen-Verwaltung der File-Handles.
  - Ein internes Dateisystem. Dies ist Bestandteil des Mobilgeräts und kann zum Lesen und Schreiben beliebiger Dateien benutzt werden.
  - Falls installiert: Ein externes Dateisystem, beispielsweise eine Flash-Karte.
  - Eine komplette SQLite Datenbank.
  - http-basierte Client-Server-Programmierung.

- Beim Erstellen von Dateien können verschiedene Modi angegeben werden, die grob die Lese- und Schreibberechtigung kontrollieren können:
  - `MODE_PRIVATE`
    - der default, beschränkt den Zugriff auf die Applikation, die die Datei erzeugt hat
  - `MODE_WORLD_READABLE`
    - erlaubt den lesenden Zugriff für alle Applikationen
  - `MODE_WORLD_WRITABLE`
    - analog für den schreibenden Zugriff

# Der Dalvik Debug Monitor Service (DDMS)

- Die Android-Implementierung einer Java Virtual Machine stellt einen hervorragenden Debug- und Monitor Service zur Verfügung. Dieser wird über die DDMS-Perspektive in Eclipse visualisiert



The screenshot displays the DDMS interface within the Eclipse ADT environment. The top toolbar includes icons for File, Edit, Navigate, Search, Project, Run, Window, and Help. The main interface is divided into several panes:

- Devices:** A table listing various Android devices and their properties.
- Threads, Heap, Allocation Tracker, Network Statistics, File Explorer, Emulator Control, System Information:** A row of tabs for different monitoring tools.
- File Explorer:** A table showing the file system structure of the selected device.
- LogCat:** A pane at the bottom for viewing log messages.

**Devices Table:**

Name	System	Android
Android4 [Online]	Android4...	
system_289	8600	
com.anc_455	8614	
com.anc_486	8623	
android_542	8639	
com.anc_616	8648	
com.anc_634	8649	
com.anc_749	8657	
com.svc_883	8661	
com.anc_914	8662	
com.anc_1012	8663	
org.java_1040	8664	

**File Explorer Table:**

Name	Size	Date	Time	Permissions	Info
acct		2014-03-24	01:50	drwxr-xr-x	
cache		2014-03-24	01:54	drwxrwx--	
config		2014-03-24	01:50	dr-x-----	
d		2014-03-24	01:50	lrwxrwxrwx	-> /sys/ke...
data		2014-03-24	01:55	drwxrwx--x	
default.pro	116	1969-12-31	19:00	-rw-r--r--	
dev		2014-03-24	01:51	drwxr-xr-x	
etc		2014-03-24	01:50	lrwxrwxrwx	-> /system...
file_context	8753	1969-12-31	19:00	-rw-r--r--	
fstab.goldf	495	1969-12-31	19:00	-rw-r-----	
init	154500	1969-12-31	19:00	-rwxr-x---	
init.goldfish	2660	1969-12-31	19:00	-rwxr-x---	
init.rc	19930	1969-12-31	19:00	-rwxr-x---	
init.trace.rc	1795	1969-12-31	19:00	-rwxr-x---	
init.usb.rc	3915	1969-12-31	19:00	-rwxr-x---	
mnt		2014-03-24	01:50	drwxrwxr-x	
proc		1969-12-31	19:00	dr-xr-xr-x	
property_c	2109	1969-12-31	19:00	-rw-r--r--	
root		2012-09-26	14:04	drwx-----	
sbin		1969-12-31	19:00	drwxr-x---	

**LogCat:** Saved Filters: All messages (no filters) (3637). Search for messages. Accepts Java regexes. Prefix with pid, app, tag, or text: to limit scope. verbose

- Das Mobilgerät stellt ein Dateisystem zur Verfügung
  - Mit welcher Technologie die „Festplatte“ realisiert wird ist für die Entwicklung von Android-Applikationen fast gleichgültig
  - Das Betriebssystem kapselt den Zugriff und stellt Java-Streams zur Verfügung
- Um eine Datei zu lesen oder zu schreiben werden Methoden der Activity-Klasse benutzt, um die Streams zu erhalten
- Das Dateisystem des Mobilgerätes wird als Bestandteil der DDMS-Perspektive dargestellt

- Android beinhaltet eine SQLite-Datenbank
- Der Zugriff hierauf erfolgt durch die Methode `openOrCreateDatabase` der Activity
- Dabei werden wieder die Access-Modi berücksichtigt
- Mit Hilfe der Datenbank können SQL-Statements abgesetzt werden
- Dies erfolgt entweder
  - gekapselt über `execSQL`
  - oder mit einem Raw-Statement unter Verwendung eines Cursors

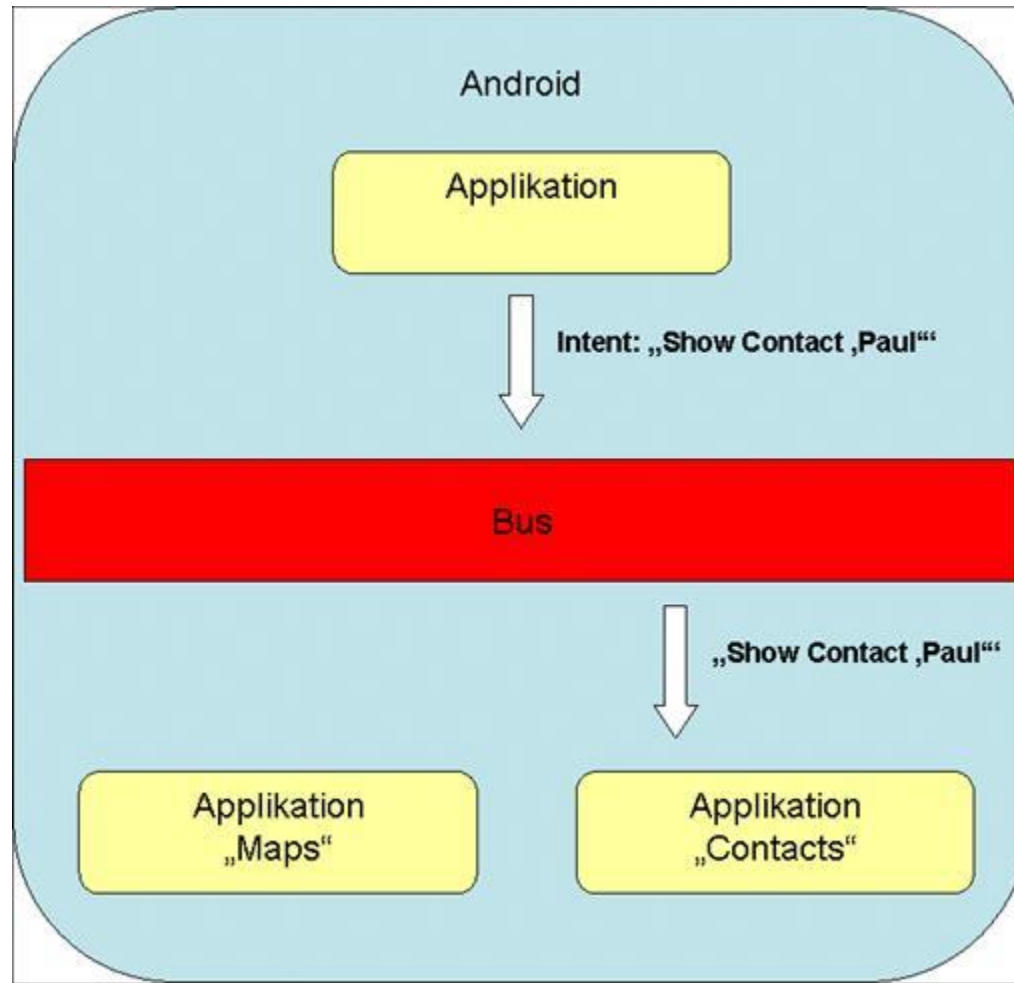


- Zum Absenden eines http-Requests benutzt Android eine Open Source-Bibliothek der Apache-Group
  - HttpComponents, vormals http Client
  - Die Erzeugung der Verbindung, das Setzen der Header und die Fehler-Behandlung werden dabei sehr gekapselt
- Damit können auch einfach RESTful Web Services aufgerufen werden
  - Datenaustausch über JSON
  - org.json-Bibliothek mit den Klassen JSONObject und JSONArray als Bestandteil der Android-Distribution

4.4

## **INTENTS**

- Mit Hilfe von Intents wird eine Interaktion innerhalb einer Applikation aber auch zwischen Applikationen erreicht
- Jede Anwendung kann aktiv einen Intent signalisieren bzw. sich als Listener für Intents registrieren



- Ein Intent wird ist nichts anderes als eine Instanz vom Typ `android.content.Intent`
- Ein Intent benötigt in jedem Falle eine Ziel-Aktivität.
- Jede Aktivität kann einen so genannten Intent Filter hinterlegen
  - Dieser beruht
    - Auf einer beliebigen Zeichenkette bzw. einer statischen Konstante
    - Auf dem MIME-Type der durch das Intent transportierten Daten
- Das Intent kann Daten transportieren
  - Diese werden bei der Konstruktion oder durch den Aufruf der Methode `setData(android.net.Uri data)` gesetzt.

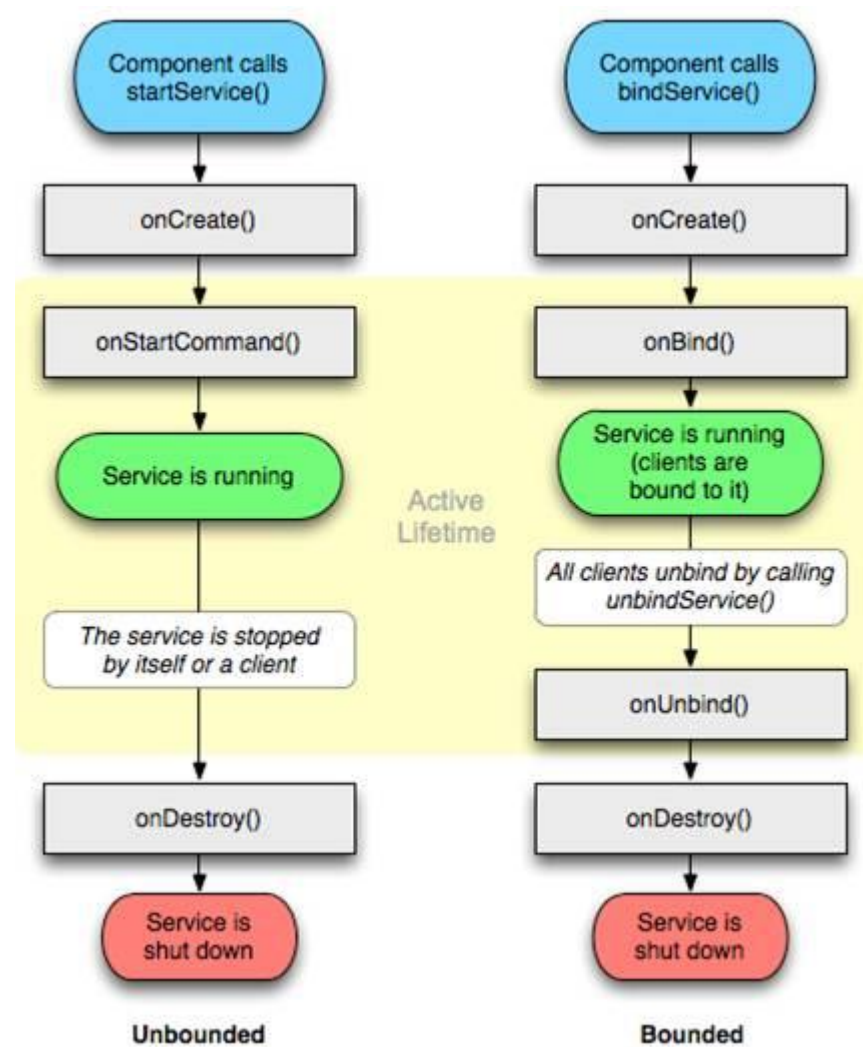
- Der eigentliche Aufruf eines Intents erfolgt über Methoden der Activity-Klasse.
  - `startActivity`
    - Eine andere Aktivität wird aufgerufen. Es wird kein Rückgabewert gemeldet.
  - `startActivityForResult`
    - Eine andere Aktivität wird aufgerufen. Neben dem Intent wird eine Request-Code mit übergeben
    - Beim Beenden der Ziel-Aktivität wird die Callback-Methode `protected void onActivityResult(int requestCode, int resultCode, Intent data)` mit dem Request-Code übergeben

4.5

## **SERVICES**

- Services sind, knapp gesprochen, Aktivitäten ohne visuelle Oberflächen. Sie werden durch einen Eintrag im Android-Manifest definiert und konfiguriert.
- Im Rahmen einer Anwendung übernehmen Services häufig die Aufgabe, lang-laufende Hintergrund-Prozesse abzubilden.
- Der Service wird, beispielsweise auf Grund einer Benutzer-Aktion in einer Aktivität, gestartet.
- Hat der Service seine Aufgabe erledigt wird dieser automatisch beendet.
- Eine andere Art von Services sind die gebundenen („bound“) Services.
  - Diese werden in der Service-Registry angemeldet
  - Auf diese Art und Weise sind auch System-Services realisiert





4.6

## **LITERATUR**



5

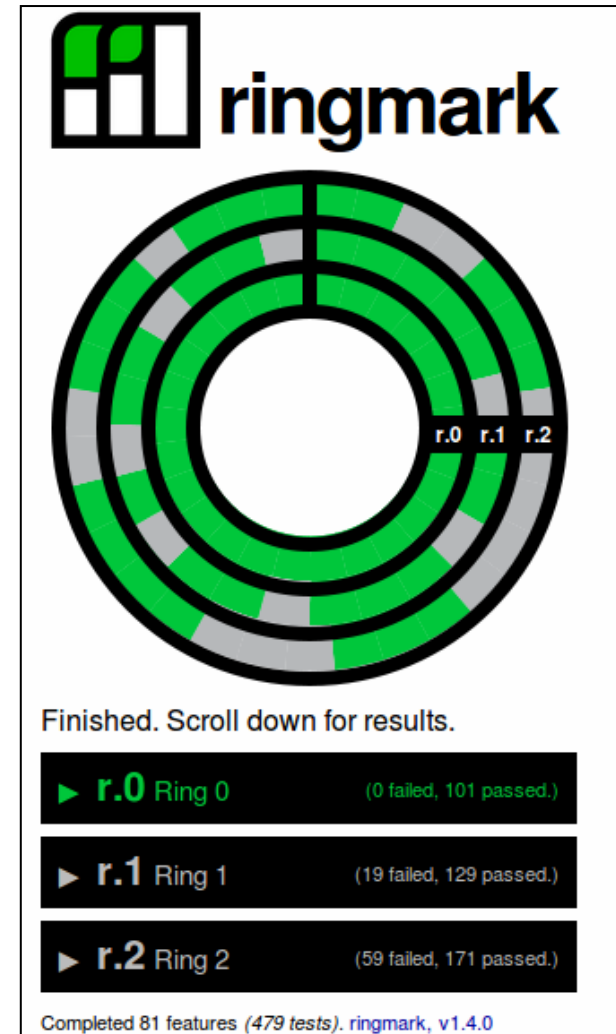
## **MOBILE WEB APPLICATIONS**

5.1

## **HTML 5: EINFÜHRUNG**

- 2003 Erste Basics
- 2004 Übernahme durch WHATWG
- 2008 Übernahme und erster öffentlicher Arbeitsentwurf
- 2012 (Dezember): Funktionsumfang festgeschrieben
- 2014 (geplant): Offizielle Verabschiedung von HTML5

- HTML 5 ist noch nicht „fertig“
  - Allerdings existiert eine sehr große Einigkeit über die Minimalanforderungen
- HTML 5 steht mehr oder weniger stark in der Kritik
  - Und zwar von allen Seiten:
    - „Zu wenig formal“
    - „Zu wenig erweiterbar“
    - „Keine klare Strategie“
  - Aktuell entwickelt sich HTML schneller als die Spezifikation nachkommt...





5.2

## **MOBILE HTML5**

- Auch Mobile Browser führen auf den unterschiedlichen Plattformen neue proprietäre HTML-Features ein
  - Erhöhter Aufwand durch leider notwendige Browser-Weichen
- Wirklich notwendige Erweiterungen werden erfahrungsgemäß auf anderen Plattformen in neuen Releases ebenfalls implementiert
  - Allerdings leider nicht immer 100% kompatibel

- Home Screen Icons
  - Eingeführt von Apple
  - Die Web Anwendung wird wie eine native Applikation behandelt
- Viewport
  - Skalierung der Darstellung
  - Prinzipielle Arbeitsweise:
    - Der Viewport definiert eine prinzipiell beliebige Screen-Größe
    - Der Browser rendered die Seite genau so, als stände diese Größe real zur Verfügung
    - Nach diesem Vorgang wird diese Seite auf die real existierende Bildschirmgröße des Mobilgeräts transformiert, ohne dass Elemente neu positioniert werden
  - Nützlicher Artikel: <http://www.quirksmode.org/mobile/viewports2.html>
  - Viewport-Definitionen im head werden in Zukunft durch eine spezielle Media Query abgelöst

Attribute	Possible values	Description
width	Integer value (in CSS pixels) of the constant device-width	Defines the viewport width
height	Integer value (in CSS pixels) of the constant device-height	Defines the viewport height
initial-scale	Floating value (0.1 to $n$ ); 1.0 is no zoom	Defines the initial zoom scale of the viewport
user-scalable	no or yes	Defines whether we will allow the user to zoom in and out in the viewport
minimum-scale	Floating value (0.1 to $n$ ); 1.0 is no scale	Defines the minimum zoom scale of the viewport
maximum-scale	Floating value (0.1 to $n$ ); 1.0 is no scale	Defines the maximum zoom scale of the viewport
target-densitydpi	Integer value (70 to 400) in DPI, or one of the constants device-dpi, high-dpi, medium-dpi, or low-dpi	Defines how to manage different resolutions regarding the viewport's width

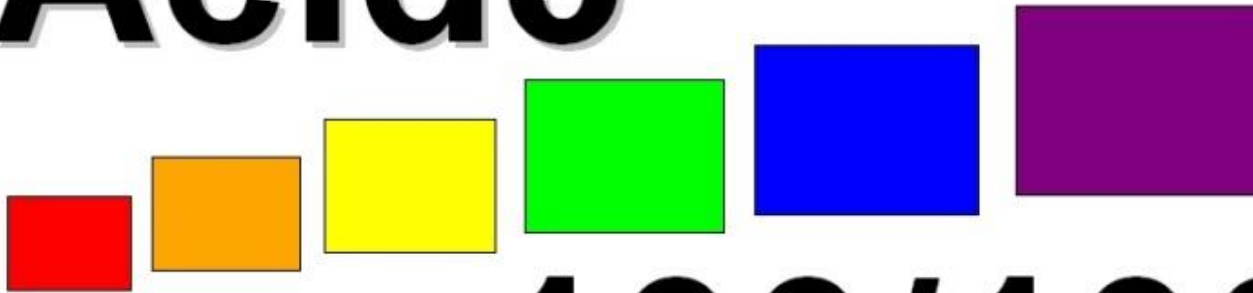
- Links aus Web Seiten heraus können von Diensten des Mobilgeräts empfangen werden
- `<b>Phone:</b> <a href="tel:555-555-5555">555-555-5555</a><br/>`
- `<b>Email:</b> <a href="mailto:a@b.com">a@b.com</a>`
- `<b>SMS:</b> <a href=" sms://5551112222"> sms://5551112222 </a>`

5.3

## **CSS 3**

- An der Version CSS3 wird bereits seit 2000 entwickelt
  - Sie ist also nach wie vor in der Entwurfsphase und genau wie HTML5 noch kein vom W3C verabschiedeter Standard
- Die Entwicklung von CSS3 soll nun modular vorangetrieben werden
  - Dadurch bekommen einzelne Module/Teiltechniken einen eigenen Release-Rhythmus
- Die kommende CSS3-Spezifikation führt zahlreiche neue und interessante Selektoren ein
  - Eine Übersicht über CSS ist im Internet zu finden

# Acid3



# 100/100

To pass the test, a browser must use its default settings, the animation has to be smooth, the score has to end on 100/100, and the final page has to look exactly, pixel for pixel, like [this reference rendering](#).



5.4

## **MEDIA QUERIES**

- Allgemeine Form
  - Embedded im CSS
    - @media [not|only] type [and] (expr) {
    - rules
    - }
  - oder External in Links
    - <link rel="stylesheet" href="print.css" media="print" />
- Bestandteile
  - Type
    - Geräte- oder Target-Plattform-Typ
  - Expressions
    - Logische Abfragen gegen Geräte-Features
  - Schlüsselwörter
    - AND, OR, ONLY
  - Rules
    - CSS-Styles

- Mit „Media Queries“ besteht die Möglichkeit, Formatierungen für bestimmte Medien durch einen logischen Ausdruck einzuschränken
- Ein Media Query besteht aus einem Medium-Typ und mindestens einem weiteren Ausdruck, der den Geltungsbereich des Medium-Typs einschränkt
- Zum Beispiel:
  - `@media screen and (max-width: 300px) {`
  - `/* eine oder mehrere CSS Regeln */`
  - `}`
- Diese Stylesheetangabe ist nur aktiv, wenn die Seite auf einem Bildschirm dargestellt wird und die Breite des Browserfenster kleiner als 300px ist

TYPE	TARGET DEVICES
all	All devices (default)
braille	Braille tactile feedback devices
embossed	Paged braille printers
handheld	Handheld devices (typically small screen and possibly monochrome)
print	Printing or print preview
projection	Projected presentations
screen	Color computer screen
speech	Speech synthesizers
tty	Media using a fixed-pitch character grid (terminals or teletypes)
tv	Television devices

FEATURE	DEFINITION	VALUE	MIN/MAX
width	Describes the width of the display area of the device	<length> (e.g., 320)	yes
height	Describes the height of the display area of the device	<length> (e.g., 600)	yes
device-width	Describes the width of the rendering surface of the device	<length> (e.g., 320)	yes
device-height	Describes the height of the rendering surface of the device	<length> (e.g., 600)	yes
orientation	Indicates if the device is in portrait (height greater than width), or landscape (width greater than height)	portrait landscape	no
aspect-ratio	Ratio of the value of the width feature to the value of the height feature	<ratio> (e.g., 16/9)	yes
device-aspect-ratio	Ratio of the value of the device-width feature to the value of the device-height feature	<ratio> (e.g., 16/9)	yes
color	Number of bits per color component of the device (returns zero if the device is not a color device)	<integer> (e.g., 1)	yes
color-index	Number of entries in the color look-up table for the device	<integer> (e.g., 256)	yes
monochrome	Number of bits per pixel on a monochrome device (returns zero if the device is not monochrome)	<integer> (e.g., 8)	yes
resolution	Resolution (pixel-density) of the device (resolution may be expressed in dots per inch [dpi] or dots per centimeter [dpcm])	<resolution> (e.g., 118dpcm)	yes
scan	Scanning process of "tv" devices	progressive   interlace	no
grid	Returns whether the device is a grid device (1) or a bitmap device (0)	<integer> (e.g., 1)	no

- Bei dem weiteren Beispiel könnten unterschiedliche Styles zur Anwendung kommen, wenn ein Device gedreht wird:
  - `body {background:#fff;}`
  - `@media all and (orientation:portrait) {`
    - `body {background: #ff0000;}`
    - `}`
  - `@media all and (orientation:landscape) {`
    - `body {background: #ff00ff;}`
    - `}`

```
■ <style type="text/css">
■ @media (min-width: 950px) {
■     /* breite Browserfenster */
■     body {background-color:green;}
■ }
■ @media (min-width: 450px) and
■     (max-width: 950px) {
■     /* Darstellung auf Netbooks */
■     body {background-color:red;}
■ }
■ @media (max-width: 450px) {
■     /* mobile Geräte */
■     body {background-color:blue;}
■ }
■ </style>
```

5.5

## **FEATURE DETECTION**





**Can I use...** [Suggestions](#) [Feed](#) [Twitter](#) 635 [Filter](#)

Compatibility tables for support of HTML5, CSS3, SVG and more in desktop and mobile browsers.

Latest update: [Five new features added: Templates, JPEG-XR, Channel messaging, Opus & Pointer events](#) (May 31, 2013)

Search:

[Index](#) [Tables](#) [Import stats](#) [FAQ](#) [Resources](#) [Embed](#)

### CSS

- @font-face Web fonts
- calc() as CSS unit value
- 2.1 selectors
- Counters
- Feature Queries
- Filter Effects
- Generated content
- Gradients
- Grid Layout
- Hyphenation
- inline-block
- Masks
- minmax-width/height
- outline
- position:fixed
- Regions
- Repeating Gradients
- resize property
- Table display
- 3D Transforms
- Animation
- Background-image options
- Border images
- Border-radius (rounded corners)
- Box-shadow
- Box-sizing
- Colors
- Media Queries
- Multiple backgrounds
- Multiple column layout
- object-fit/object-position
- Opacity
- Overflow-wrap
- selectors
- Text-overflow
- Text-shadow
- Transforms
- Transitions
- word-break

### HTML5

- Audio element
- Canvas (basic support)
- Color input type
- contenteditable attribute (basic support)
- Datalist element
- dataset & data-\* attributes
- Date/time input types
- Details & Summary elements
- Download attribute
- Drag and Drop
- Form validation
- HTML5 form features
- input placeholder attribute
- New semantic elements
- Number input type
- Offline web applications
- Progress & Meter
- Range input type
- Ruby annotation
- sandbox attribute for iframes
- Session history management
- Text API for Canvas
- Toolbar/context menu
- Video element
- WebGL - 3D Canvas graphics
- **All HTML5 features**

### SVG

- Inline SVG in HTML5
- SVG (basic support)
- SVG effects for HTML
- SVG filters
- SVG fonts
- SVG fragment identifiers
- SVG in CSS backgrounds
- SVG in HTML img element
- SVG SMIL animation
- **All SVG features**

### Other

- async attribute for external scripts
- classList (DOMTokenList)
- Content Security Policy
- Data URIs
- defer attribute for external scripts
- ECMAScript 5 Strict Mode
- [getComputedStyle](#)
- [getElementsByName](#)
- HTML templates
- JPEG XR image format
- MathML
- MPEG-4/H.264 video format
- Mutation Observer
- Ogg/Theora video format
- Opus
- PNG alpha transparency
- Pointer events
- [querySelector/querySelectorAll](#)
- Shadow DOM
- Strict Transport Security
- Touch events
- WAI-ARIA Accessibility features
- WebM/VP8 video format
- WebP image format
- WOFF - Web Open Font Format

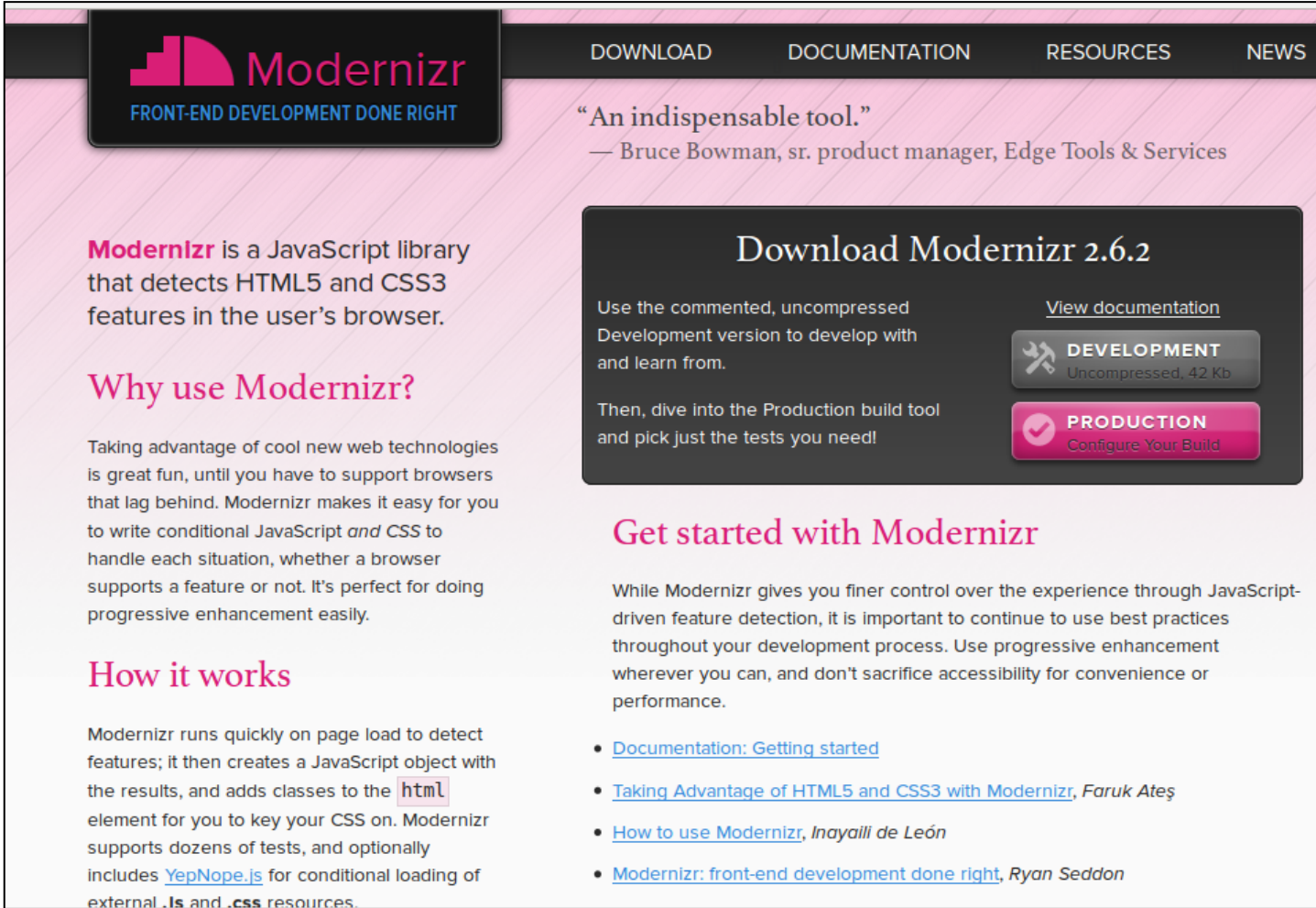
### JS API

- Blob URLs
- Channel messaging
- Cross-document messaging
- Cross-Origin Resource Sharing
- DeviceOrientation events
- File API
- FileReader API
- Filesystem & FileWriter API
- Full Screen API

# html5mobile.org: Kompakt für mobile Browser

Feature	Safari on iOS	Android Browser	Google Chrome	Amazon Silk	BlackBerry Browser		Nokia Browser		Internet Explorer		Opera Mobile	Opera mini	Firefox	webOS Browser		
Platform	iPhone, iPad	Phones & Tablet	Android 4.0+	Kindle Fire	Phones	Tablet	MeeGo - N9	Symbian	Windows Phone 7.5	Windows 8	Android & Symbian	Java/JOS Android	Android, MeeGo	HP Phones	HP TouchPad	
Versions tested	3.2 to 6.1	1.5 to 4.2	18 to 25b	1.0 to 2.0	5.0 to 7.1	BB10	1.0 to 2.1	1.2	^3 to Belle FP2	9	10 (metro)	11 to 12.1	5 to 7	6 to 18	1.4 to 2.0	3.0
<b>Application Cache</b> <u>W3C API</u> Offline package installation.	✓	✓ 2.1+	✓	✓	✓ 6.0+	✓	✓	✓	✓ Belle FP2+		✓	✓		✓	✓	✓
<b>Web storage</b> <u>W3C API</u> Persistent and session storage.	✓	✓ 2.0+	✓	✓	✓ 6.0+	✓	✓	✓	✓ Belle FP2+	✓	✓	✓		✓	✓	✓
<b>Web SQL storage</b> <u>W3C API (no active)</u> Persistent SQLite storage.	✓	✓ 2.0+	✓	✓	✓ 6.0+	✓	✓	✓	✓ Belle FP2+			✓			✓	✓
<b>Geolocation</b> <u>W3C API</u> Geolocation & tracking using GPS, cells or Wi-Fi.	✓	✓ 2.0+	✓	✓ 2.0+	✓ 6.0+	✓	✓	✓	✓ Belle+	✓	✓	✓		✓	✓	✓
<b>Multimedia</b> <u>W3C API</u> Video & Audio Players	✓	✓ 2.3+	✓	✓	✓ 7.0+	✓	✓	✓	✓ Belle+	✓	✓	✓		✓	✓	✓
<b>Server-Sent Events</b> <u>W3C API</u> EventSource pattern to maintain the connection to the server open	✓ 4.1+		✓	✓ 2.0+		✓ 2.0+	✓	✓				✓		✓		
<b>Web Sockets</b> <u>W3C API</u> Newbidirectional protocol over HTTP	✓ 4.2+		✓	✓ 2.0+	✓ 6.1+	✓	✓		✓ Belle FP2+		✓	✓ 12.1+		✓ 7+		✓ 3.0.5+

- modernizr.com



The screenshot shows the Modernizr website with a pink and black color scheme. The header includes a logo and navigation links: DOWNLOAD, DOCUMENTATION, RESOURCES, and NEWS. The main content area is divided into several sections: a description of Modernizr as a JavaScript library, a 'Why use Modernizr?' section, a 'How it works' section, a 'Download Modernizr 2.6.2' section with buttons for 'DEVELOPMENT' and 'PRODUCTION', and a 'Get started with Modernizr' section with a list of links.

**Modernizr**  
FRONT-END DEVELOPMENT DONE RIGHT

DOWNLOAD DOCUMENTATION RESOURCES NEWS

“An indispensable tool.”  
— Bruce Bowman, sr. product manager, Edge Tools & Services

**Modernizr** is a JavaScript library that detects HTML5 and CSS3 features in the user’s browser.

### Why use Modernizr?

Taking advantage of cool new web technologies is great fun, until you have to support browsers that lag behind. Modernizr makes it easy for you to write conditional JavaScript *and* CSS to handle each situation, whether a browser supports a feature or not. It’s perfect for doing progressive enhancement easily.

### How it works

Modernizr runs quickly on page load to detect features; it then creates a JavaScript object with the results, and adds classes to the `html` element for you to key your CSS on. Modernizr supports dozens of tests, and optionally includes [YepNope.js](#) for conditional loading of external `.js` and `.css` resources.

## Download Modernizr 2.6.2

Use the commented, uncompressed Development version to develop with and learn from.

[View documentation](#)

**DEVELOPMENT**  
Uncompressed, 42 Kb

Then, dive into the Production build tool and pick just the tests you need!

**PRODUCTION**  
Configure Your Build

## Get started with Modernizr

While Modernizr gives you finer control over the experience through JavaScript-driven feature detection, it is important to continue to use best practices throughout your development process. Use progressive enhancement wherever you can, and don’t sacrifice accessibility for convenience or performance.

- [Documentation: Getting started](#)
- [Taking Advantage of HTML5 and CSS3 with Modernizr](#), Faruk Ates
- [How to use Modernizr](#), Inayalli de León
- [Modernizr: front-end development done right](#), Ryan Seddon

6

# JAVASCRIPT FÜR MOBILE WEB APPLICATIONS

6.1

## **JQUERY CORE: EINE EINFÜHRUNG**

- Die jQuery-Bibliothek wurde primär eingeführt, um die Manipulationen des DOM-Baums zu vereinfachen
- Dazu wurde ein vereinheitlichtes API geschaffen, das den Zugriff auf Elemente des DOM mit etablierten Abfrage-Ausdrücken wie XPATH und den CSS-Selektoren ermöglicht
- Browser-Inkompatibilitäten werden intern abhandelt
- Einheitliches Event-Modell
- Utilities
  - AJAX
  - Effekte

- Die zentrale Funktion der jQuery-Bibliothek ist `jQuery()`
- Der Rückgabewert ist abhängig vom Übergabe-Parameter:
  - CSS-Selektor
    - Eine Selektion vorhandener HTML-Elemente, auf die Kommandos angewendet werden können
  - HTML-Fragment
    - Eine Selektion von HTML-Elementen, die in die Seite integriert werden können
  - Funktionsobjekt
    - Das Funktionsobjekt wird zur `ready`-Funktion, die automatisch nach dem vollständigen Laden des HTML-Codes ausgeführt wird

- Werden auf einer Selektion ausgeführt
- Eine Iteration über die Selektion ist nicht notwendig
  - alle Kommandos werden auf allen Elementen ausgeführt
- Gruppen von Kommandos
  - Zuweisen von HTML-Eigenschaften, insbesondere Styles
  - Registrieren von Event-Handlern
  - Manipulation des DOM



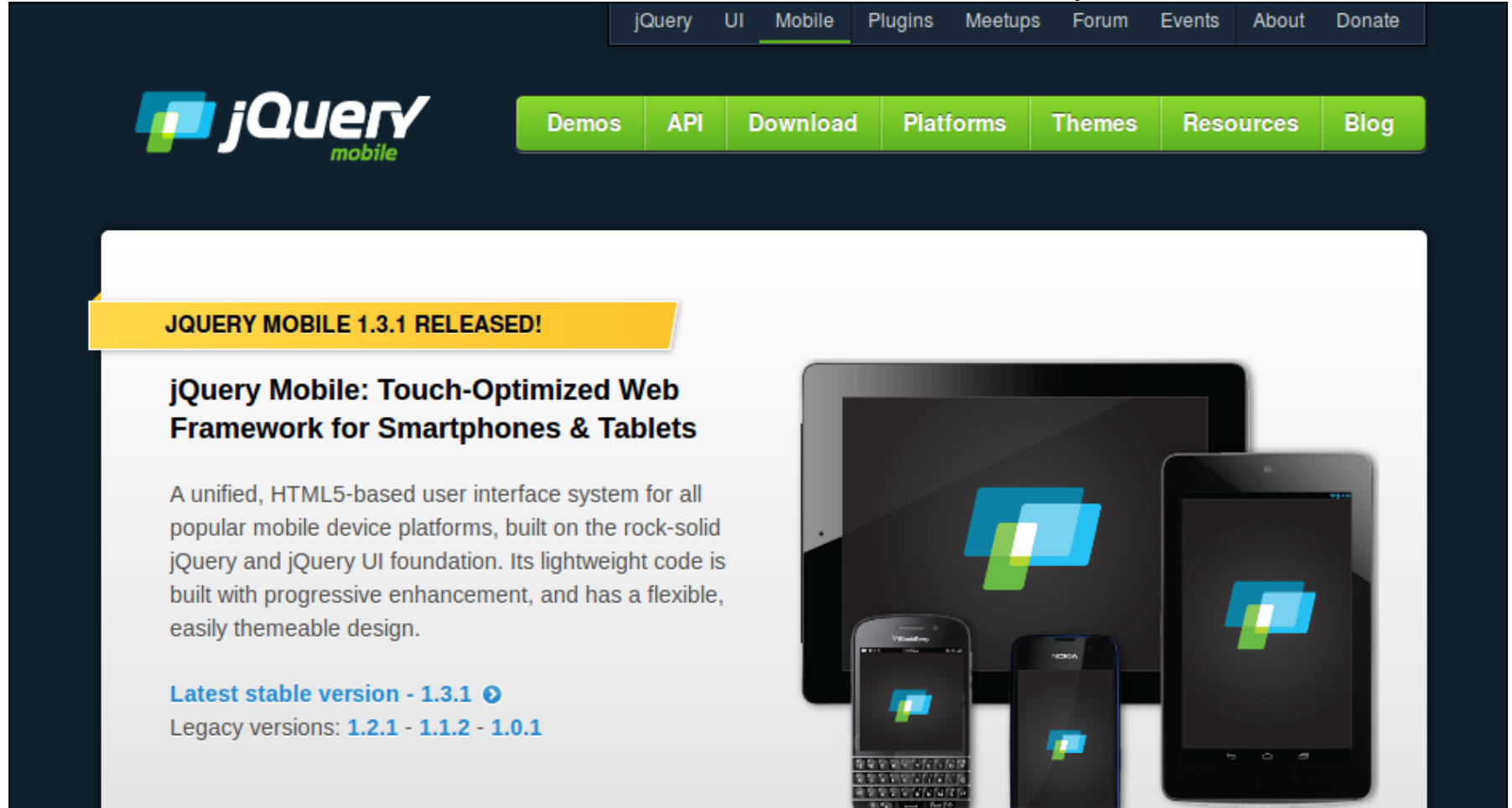
- Eine hervorragende und aktuelle Online-Dokumentation steht unter
  - [http://docs.jquery.com/Main\\_Page](http://docs.jquery.com/Main_Page)
- zur Verfügung

6.2

## **JQUERY MOBILE**

- Open Source
- Setzt auf der bekannten jQuery-Bibliothek auf
- Unterstützung der neuen Event-Typen für Touchscreens
- Eine Web Anwendung wird „Mobil“ gemacht, in dem HTML5-Attribute gesetzt werden
  - Diese werden von jQuery Mobile definiert und ausgelesen
  - Damit erfolgt beispielsweise eine Zerlegung einer Seite in mehrere Unterseiten inklusive Seitennavigation
- Konsistentes Oberflächen-Design mit Themes und Swatches
  - Eigene Themes werden mit dem Theme Builder erstellt
- Unterstützung verschiedener Plattformen
  - jQuery Mobile implementiert transparent die nötigen Browser-Weichen


- Aktuelle und ausführliche Dokumentation und Beispiele online



The screenshot shows the jQuery Mobile website homepage. At the top, there is a navigation bar with links: jQuery, UI, Mobile (highlighted), Plugins, Meetups, Forum, Events, About, and Donate. Below this is the jQuery mobile logo and a row of green buttons: Demos, API, Download, Platforms, Themes, Resources, and Blog. The main content area features a yellow banner announcing 'JQUERY MOBILE 1.3.1 RELEASED!'. Below the banner, the text reads 'jQuery Mobile: Touch-Optimized Web Framework for Smartphones & Tablets'. A paragraph describes it as a unified, HTML5-based user interface system for all popular mobile device platforms, built on the rock-solid jQuery and jQuery UI foundation. It mentions its lightweight code, progressive enhancement, and flexible, easily themeable design. Below this, it states 'Latest stable version - 1.3.1' with a download icon, and 'Legacy versions: 1.2.1 - 1.1.2 - 1.0.1'. To the right of the text, there is an image showing a tablet, a smartphone, and a feature phone, all displaying the jQuery Mobile logo.

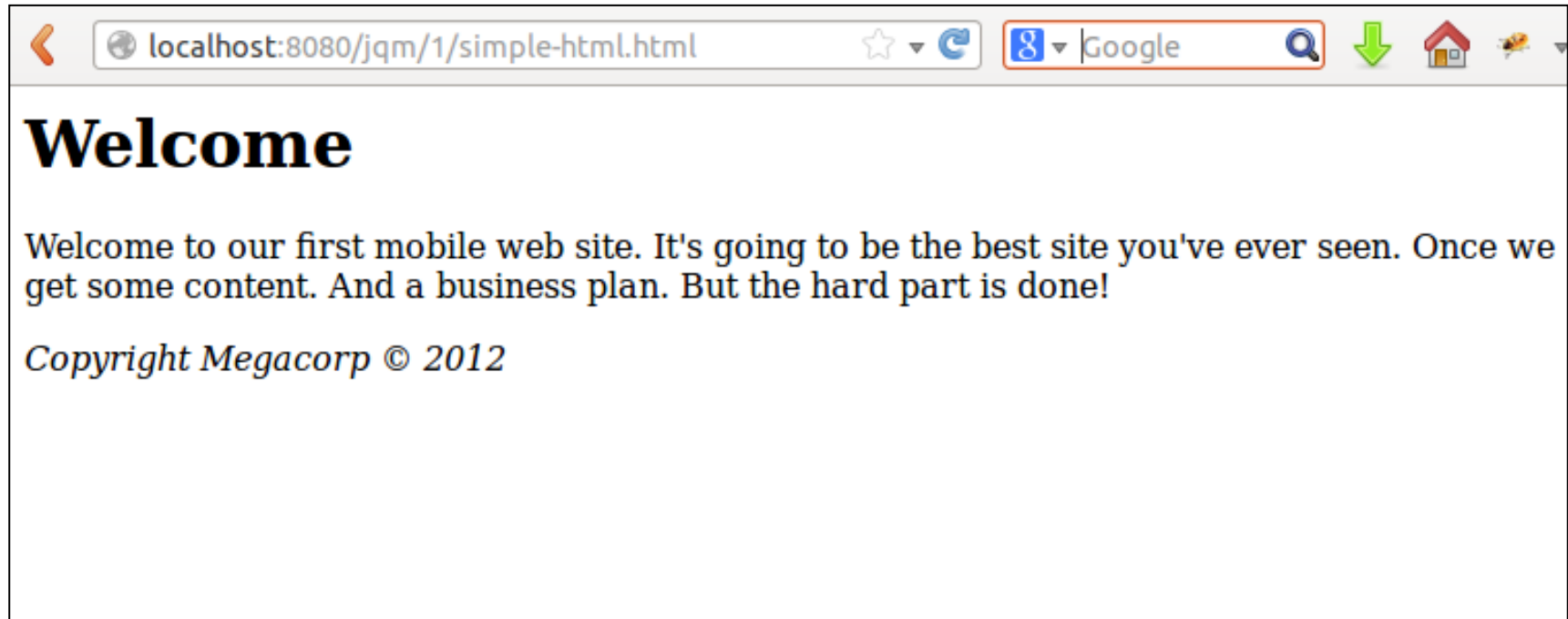
jQuery Mobile: Touch-Optimized Web Framework for Smartphones & Tablets

A unified, HTML5-based user interface system for all popular mobile device platforms, built on the rock-solid jQuery and jQuery UI foundation. Its lightweight code is built with progressive enhancement, and has a flexible, easily themeable design.

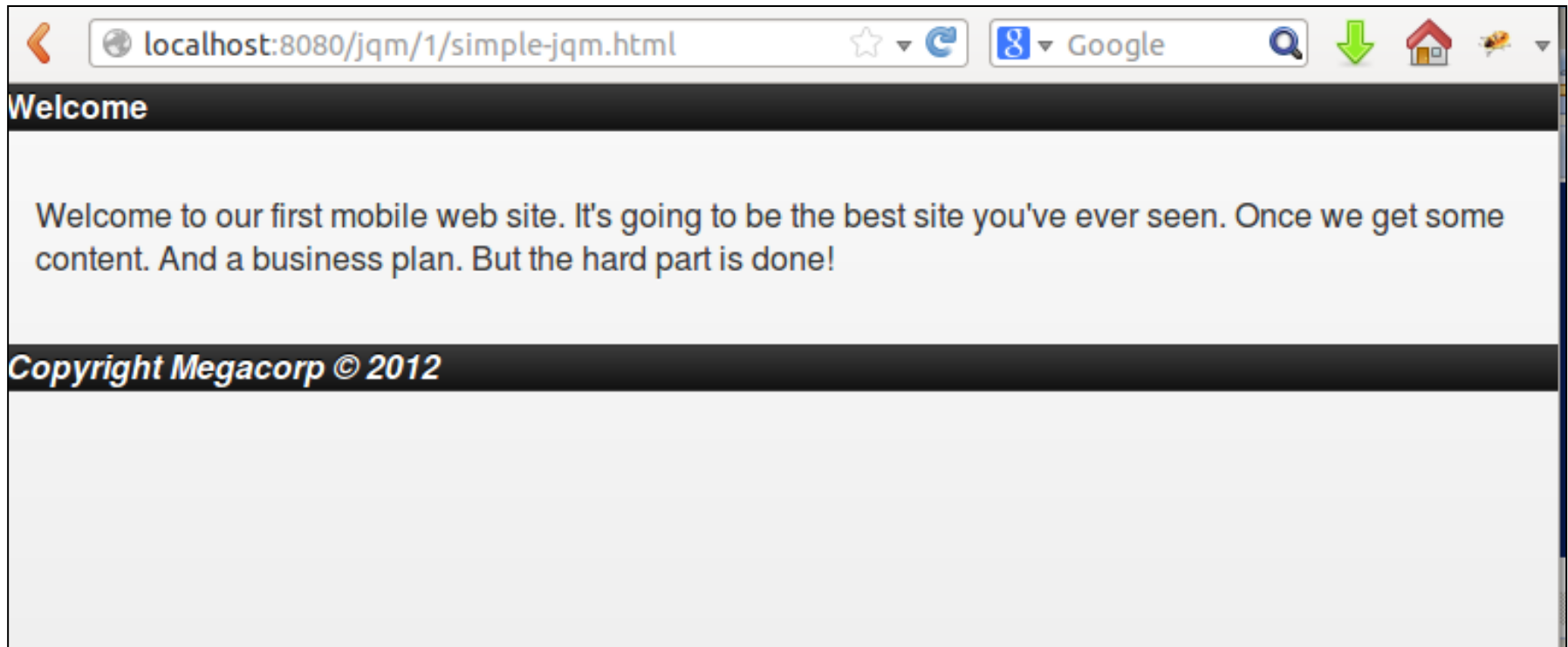
Latest stable version - 1.3.1 

Legacy versions: 1.2.1 - 1.1.2 - 1.0.1

- `<html>`
- `<head>`
- `<title>First Mobile Example</title>`
- `</head>`
- `<body>`
- `<h1>Welcome</h1>`
- `<p>...</p>`
- `</body>`
- `</html>`



- `<!DOCTYPE html>`
- `<html><head><title>First Mobile Example</title>`
- `<meta name="viewport" content="width=device-width, initial-scale=1">`
- `<link rel="stylesheet" href="/jQuery/jquery.mobile-1.3.1.min.css" />`
- `<script src="/jQuery/jquery-1.9.1.min.js"></script>`
- `<script src="/jQuery/jquery.mobile-1.3.1.min.js"></script>`
- `</head><body>`
- `<div data-role="page">`
- `<div data-role="header">Welcome</div>`
- `<div data-role="content"><p>...</p></div>`
- `<div data-role="footer"><i>Copyright Megacorp &copy;`  
`2012</i></div>`
- `</div>`
- `</body></html>`



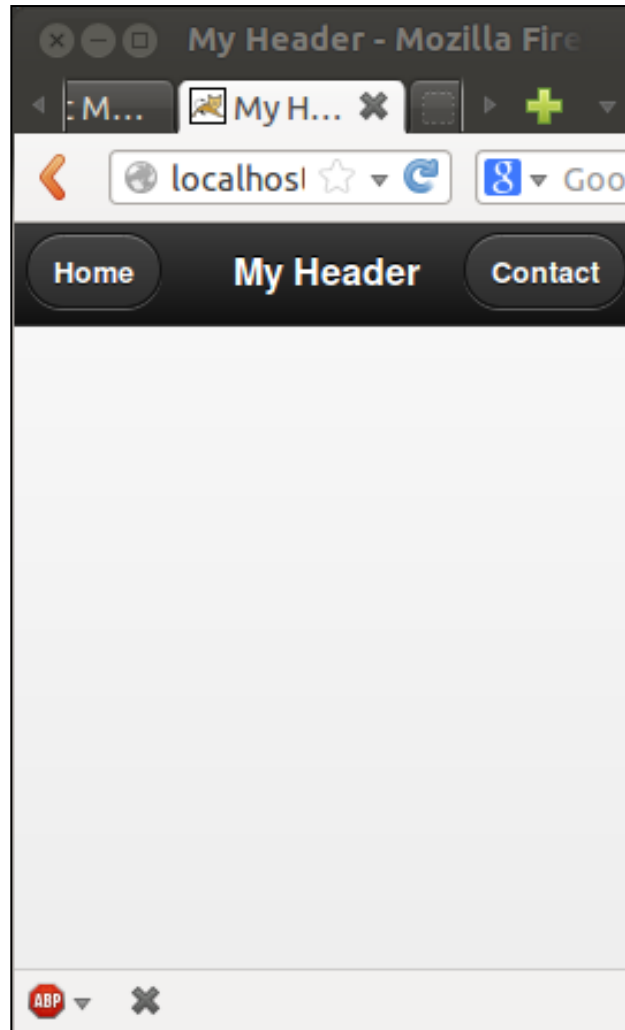


- jQuery Mobile
  - Stellt auf einer Seite jeweils nur einen page-Bereich dar
  - Intercepted sämtliche Links auf der Seite
    - Hashmarks führen zum Einblenden der angesprungenen Seite
    - Seiten der selben Server-Domain werden über einen AJAX-Request geladen und dem aktuellen DOM hinzugefügt
      - Damit werden die page-Bereiche der neuen Seite komplett in die vorhandene Seite integriert
    - Links auf externe Seiten werden nicht verändert
      - Die neue Seite wird normal geladen
  - Links können durch data-role="button" als Schaltfläche gerendered werden
    - Für kleine Bildschirme meistens besser

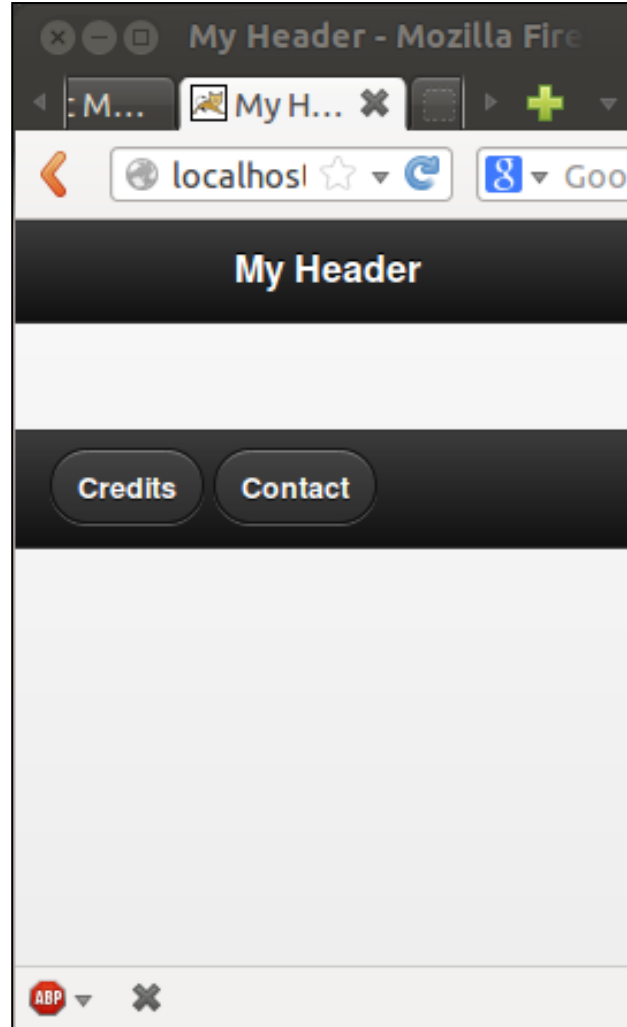
6.3

## **JQUERY MOBILE: UI-LAYOUT**

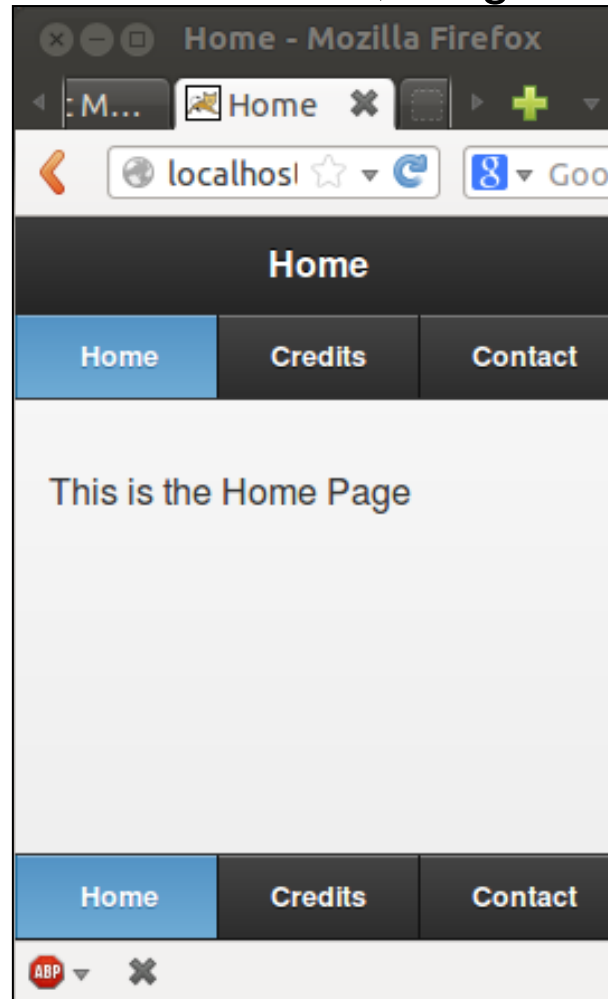
- Überschrift und maximal zwei Schaltflächen



- Text und maximal 5 Schaltflächen




- Bereich mit maximal 5 Schaltflächen, dargestellt ohne Abstand



6.4

## **WIDGETS**



Search...

## Widget reference

This section is a reference for all widget options.

Search widgets...

**Widget reference**

Accordion
AJAX Navigation
Autocomplete
Buttons
Checkboxes
Collapsibles
Controlgroup
Dialogs
Fixed toolbars

6.5

## **EVENTS**



- jQuery Mobile benutzt das Event-Modell von jQuery
  - Die Funktion bind registriert an einem Event-Typen eine Listener-Funktion
    - `jQuerySelection.bind(eventType, eventData, eventFunctionObject)`
  - Diese Funktion beim Auftreten der Benutzer-Interaktion als Parameter das jQuery-Event-Objekt
  - Bei den page Events wird als zweiter Parameter der pageContext mit übertragen
    - dieser enthält unter anderem die Ziel-Page

- tap
- taphold
- swipe
- swipeleft
- swiperight
- scrollstart
- scrollstop
- orientationchange

- Laden einer Page
  - pagebeforeload
  - pageload
  - pageloadfailed
- Darstellung einer anderen Seite
  - pagebeforechange
  - pagechange
  - pagechangefailed
- Transition zwischen Pages
  - pagebeforeshow
  - pageshow
  - pagebeforehide
  - pagehide

- Initialisierung
  - pagebeforecreate
  - pagecreate
  - pageinit
- Entfernen einer anderen Seite
  - pageremove
- Layout
  - updatelayout

6.6

## **MOBILE JAVASCRIPT**

- Überblick
  - Geolocation
  - Touch und Gesture-Events
  - Zugriff auf Sensoren
  - Ressourcen-Zugriff: Files, Network
  - Einbindung in das Notification-System des Geräts
  - Kamera
- Eine Plattform-übergreifende Realisierung ist aktuell nicht vorhanden
  - Teilweise muss die Web Anwendung als native App verpackt werden
    - Apache Cordova (vormals PhoneGap)

- Die Geolocation API ist eine API zur Unterstützung von "location aware webservices"
  - In Javascript ist das Objekt `navigator.geolocation` verfügbar
    - Unterstützt der Browser die API nicht oder hat der Benutzer sie deaktiviert, so ist `navigator.geolocation` nicht verfügbar
- Ist das Objekt verfügbar, so kann mit `getCurrentPosition()` eine Abfrage der Koordinaten abgesetzt werden
  - Als Parameter erwartet diese Methode eine Callback-Methode für die Verarbeitung des Ergebnisses und optional eine zweite Methode zur Fehlerbehandlung

```
var coords;

function getGeolocation() {
    if (navigator.geolocation) {
        navigator.geolocation.getCurrentPosition(saveCoords);
    } else {
        alert("Die Geolocation-Funktion wird
            von Ihrem Browser leider nicht unterstützt.");
    }
}

function saveCoords(position) {
    coords = position.coords.latitude + ',' +
        position.coords.longitude;
    alert(coords);
}

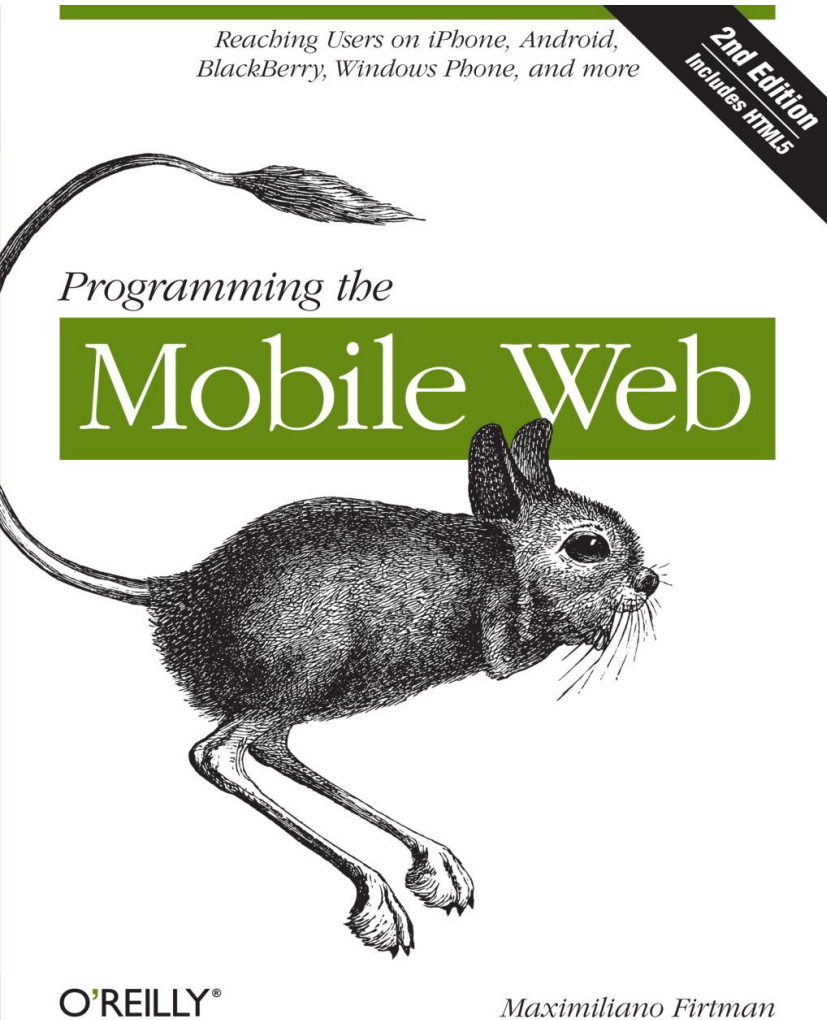
getGeolocation();
```



<code>position.timestamp</code>	Wann die Position ermittelt wurde
<code>position.coords.latitude</code>	Breitengrad
<code>position.coords.longitude</code>	Längengrad
<code>position.coords.accuracy</code>	Genauigkeit in Metern
<code>position.coords.altitude</code>	Höhenangabe falls verfügbar
<code>position.coords.altitudeAccuracy</code>	Genauigkeit der Höhenangabe in Metern
<code>position.coords.speed</code>	Geschwindigkeit der Bewegung in Metern
<code>position.coords.heading</code>	Bewegungsrichtung (Grad von Norden)

6.7

## LITERATUR



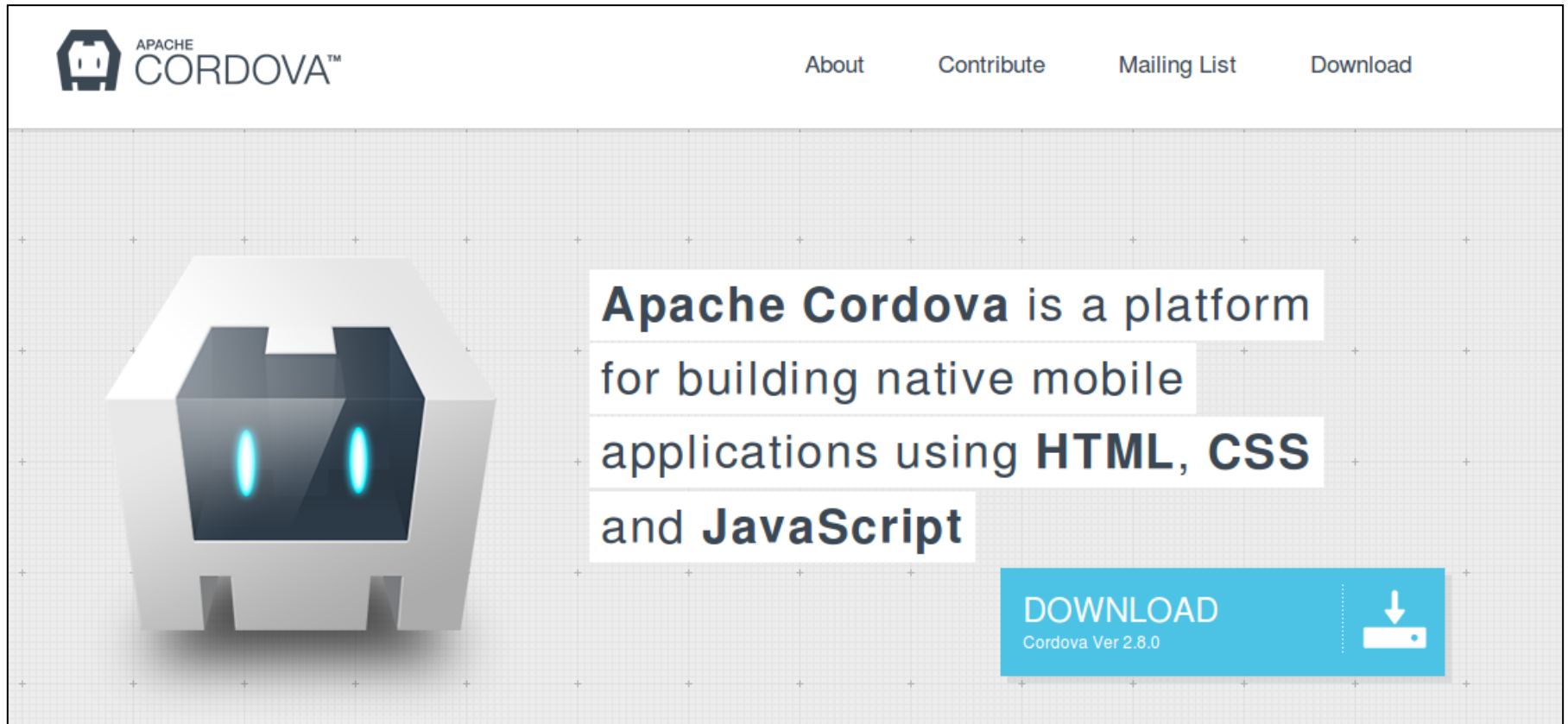
7

## **HYBRID APPLICATIONS**

7.1

## **GRUNDKONZEPT**

- Der mobile Browser hat nicht notwendigerweise den vollen Zugriff auf die Dienste bzw. Services des Mobilgeräts
- Deshalb müssen im konkreten Anwendungsfall doch native Anwendungen erzeugt werden
- Verschiedene Ansätze möglich:
  - Komplette Alternativentwicklung
  - Benutzung einer „Web View“-Komponente
    - Diese benutzt die native Render-Engine des Geräts und stellt damit innerhalb einer nativen App eine Webseite dar
  - Benutzung einer Wrapper-Applikation
    - Diese stellt den Zugriff auf die nativen Dienste und Services über JavaScript-Objekte zur Verfügung



The screenshot shows the Apache Cordova website. At the top left is the Apache Cordova logo, which consists of a stylized robot head icon and the text "APACHE CORDOVA™". To the right of the logo are four navigation links: "About", "Contribute", "Mailing List", and "Download". The main content area features a large, 3D-rendered robot head icon on the left. To its right, a text block states: "Apache Cordova is a platform for building native mobile applications using **HTML**, **CSS** and **JavaScript**". Below this text is a blue button with the word "DOWNLOAD" in white, and "Cordova Ver 2.8.0" in smaller white text below it. To the right of the button is a white download icon (a downward arrow pointing to a document) on a blue background.

APACHE CORDOVA™

About Contribute Mailing List Download

**Apache Cordova** is a platform  
for building native mobile  
applications using **HTML**, **CSS**  
and **JavaScript**

DOWNLOAD  
Cordova Ver 2.8.0

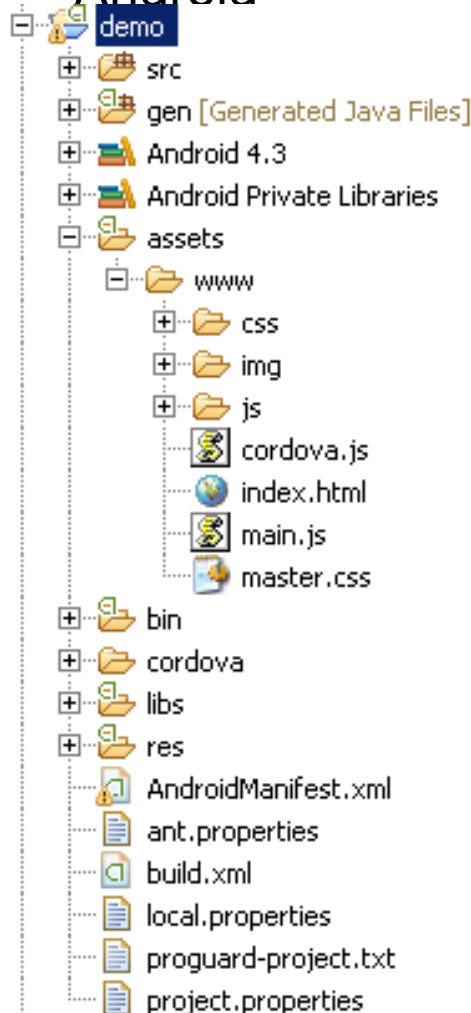
7.2

## **APACHE CORDOVA/PHONEGAP**

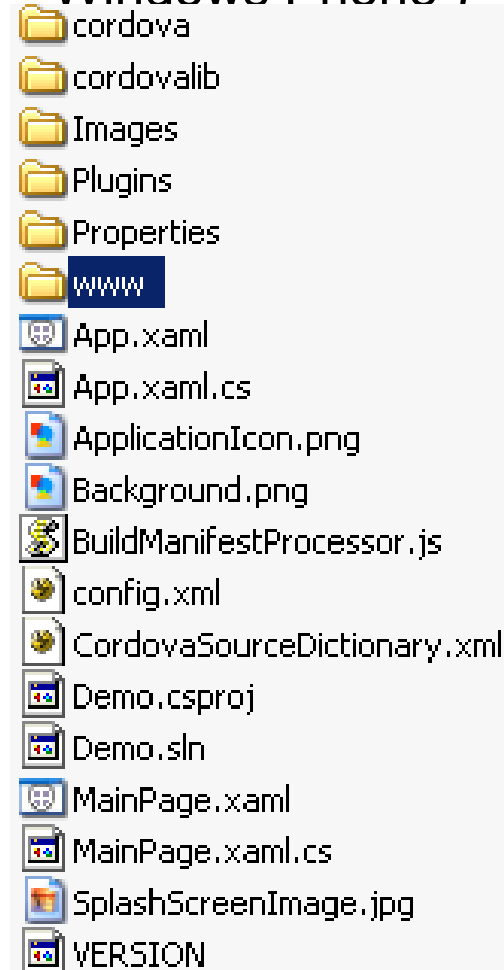


- Für die verschiedenen Plattformen wird durch ein Template das Grundgerüst einer nativen Applikation erzeugt
- Dieses enthält in der Plattform-abhängigen Projektstruktur jeweils einen Ordner www
  - Dieser enthält die Anwendung als HTML, CSS, JavaScript
  - Dieses wird beispielsweise über ein Versionsverwaltungssystem Projekt-übergreifend gehalten
- Die native Applikation wird über den Plattform-eigenen Build-Prozess erzeugt
  - Dafür kann auch der Adobe PhoneGap Build Service benutzt werden

## ■ Android



## Windows Phone 7



- Normales HTML, CSS und JavaScript
  - Cordova benutzt das Browser-API der Zielplattform

## API Reference

### Accelerometer

Tap into the device's motion sensor.

### Compass

Obtain the direction that the device is pointing.

### Device

Gather device specific information.

### Geolocation

Make your application location aware.

### Media

Record and play back audio files.

### Storage

### Camera

**Capture** a photo using the device's camera.

### Connection

Quickly check the network state, and cellular network information.

### Events

Hook into native events through JavaScript.

### Globalization

Enable representation of objects specific to a locale.

### Notification

Visual, audible, and tactile device notifications.

### Capture

**Capture** media files using device's media capture applications.

### Contacts

Work with the devices contact database.

### File

Hook into native file system through JavaScript.

### InAppBrowser

Launch URLs in another in-app browser instance.

### Splashscreen

Show and hide the applications splash screen.

## Camera

---



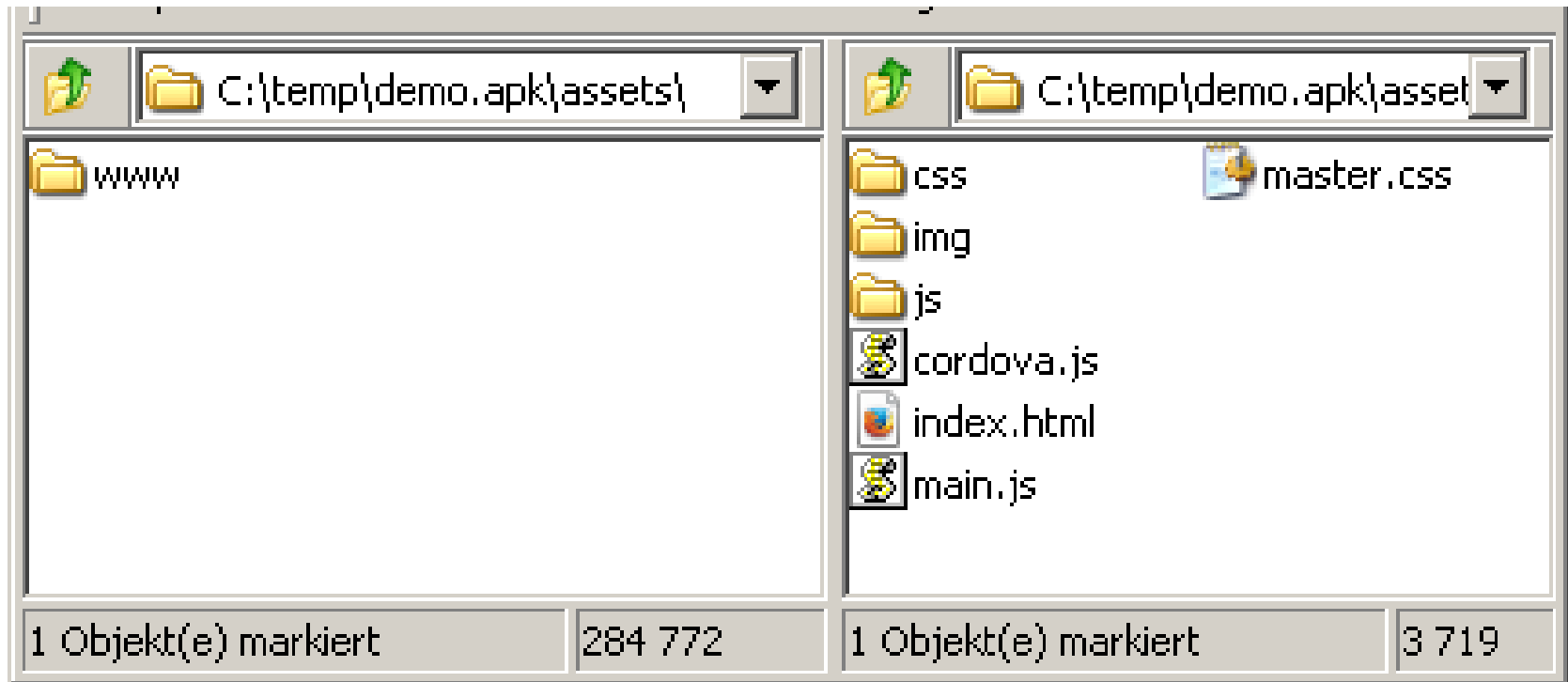
The `camera` object provides access to the device's default camera application.

**Important privacy note:** Collection and use of images from a device's camera raises important privacy issues. Your app's [privacy policy](#) should discuss how the app uses the camera and whether the images recorded are shared with any other parties. In addition, if the app's use of the camera is not apparent in the user interface, you should provide a just-in-time notice prior to your app accessing the camera (if the device operating system doesn't do so already). That notice should provide the same information noted above, as well as obtaining the user's permission (e.g., by presenting choices for "OK" and "No Thanks"). For more information, please see the [Privacy Guide](#).

### Methods :

- [camera.getPicture](#)
- [camera.cleanup](#)

# Packaging, Beispiel Android APK



8  
**IOS**



8.1

## **IPHONE/IPAD PROGRAMMIERUNG**



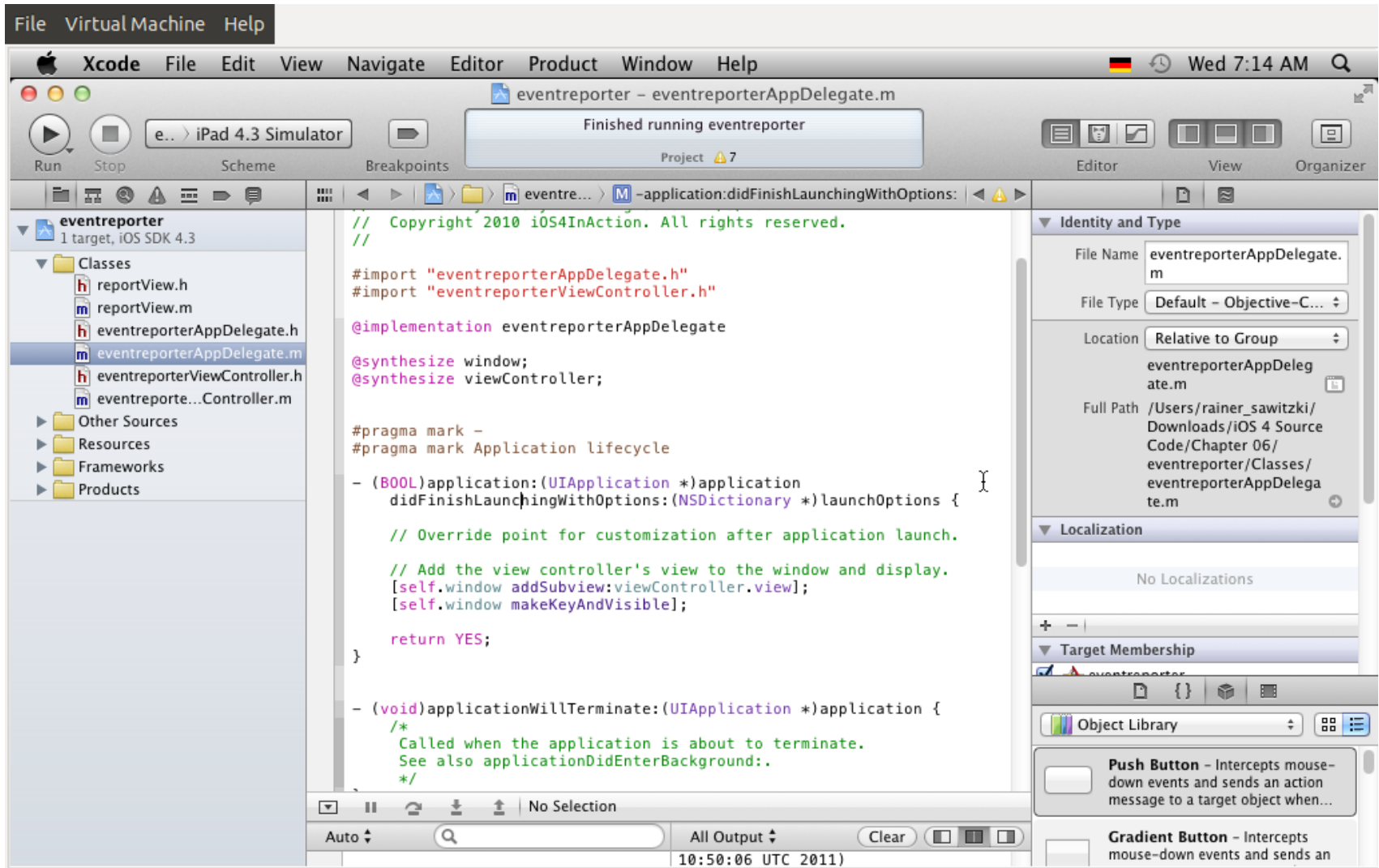
- Core Spezifikation
  - CPU, Hauptspeicher
- IO
  - Touch Screen
  - Networking
  - Dateisystem und Datenbank
  - Unterstützung verschiedener Media-Types und Grafik-Formaten
- Zusätzliche Dienste
  - Bluetooth
  - GPS
  - Bewegungssensoren
  - ...

- DAS Werkzeug zur Entwicklung von iOS/OS-Anwendungen
- „All in one“
  - Wizards
  - Editoren
  - Code-Unterstützung
  - Compile
  - Build
  - Versioning
  - iOS Simulator
  - Debugging
- Nur für Mac-Computer verfü



- App Store
- Developer Download Site
- Vorsicht:
  - Ein wirklich großer Download!
  - Voraussetzungen für OS Version und SDK müssen exakt erfüllt werden
  - Aktualisierungen sind problematisch, möglichst auf einem blanken System aufsetzen!
  - Häufige neue Releases

# Editoren, Dialoge, Wizards...



# Fortgeschrittene Werkzeuge: Code Analyzer





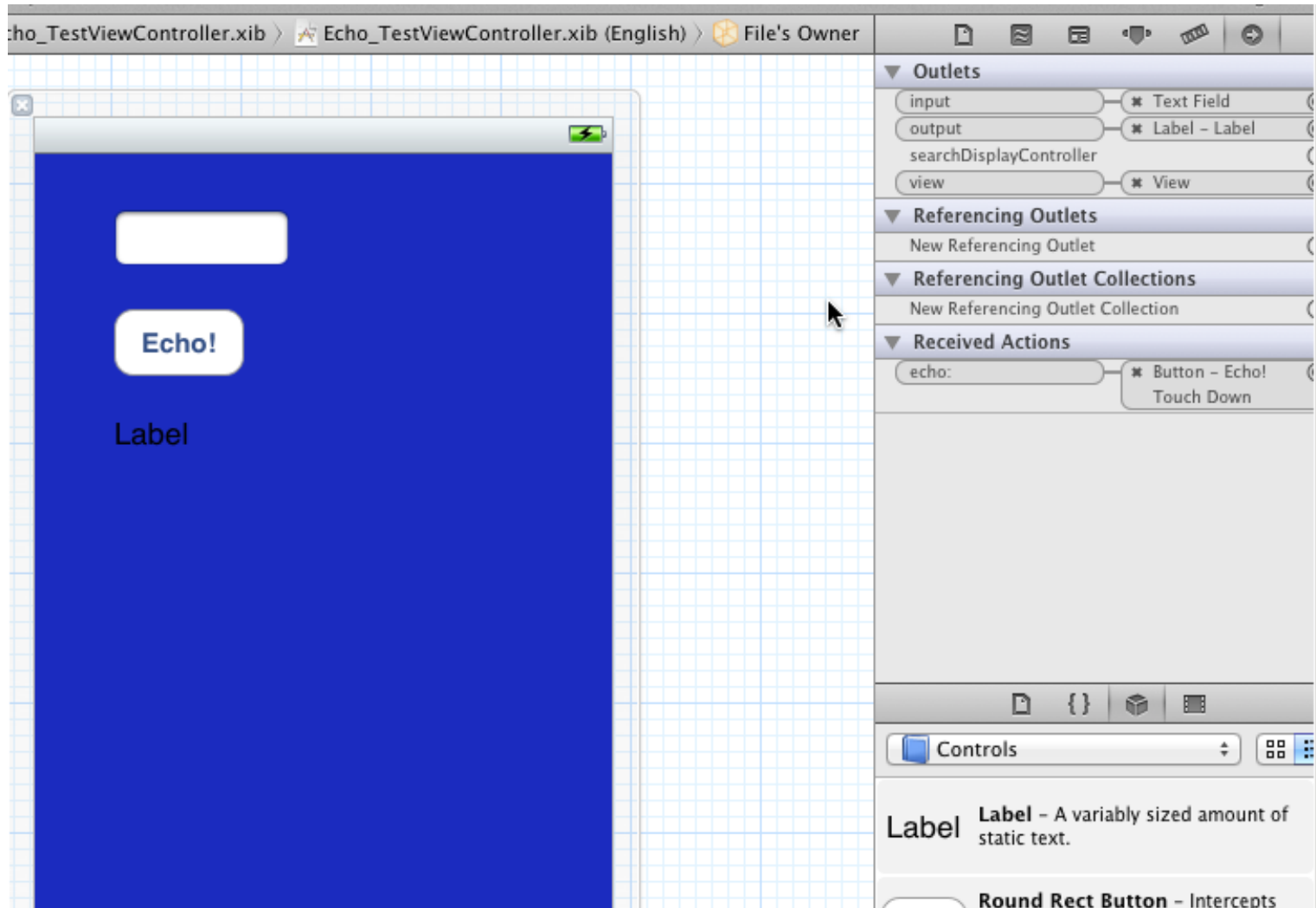
- Editor für
  - HTML
  - CSS
  - JavaScript
- Bestandteil der XCode-Umgebung
- Enthält eine iOS-spezifische Widget-Bibliothek



- Editor für die XML-basierte UI-Definition
- Bestandteil der XCode-Umgebung
- Drag&Drop Visual UI Builder



# Oberfläche des Interface Builders



- iOS Simulator
  - Test und Debugging einer iPhone Anwendung auf dem Entwickler-Rechner
    - Kein vollwertiger Ersatz für den Test auf dem Mobilgerät!
  - Unterstützung für iPhone/iPad
- Quartz Composer
  - Quartz ist Apples 2D-Grafik-Bibliothek

8.2

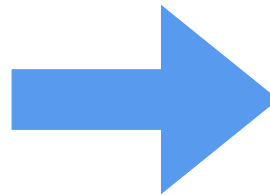
## **PROVISIONING**

- Ein „Bundle“
  - besteht aus der kompletten Anwendung inklusive Logik, Hilfsdateien und Deskriptoren
    - \*-Info.plist
    - Ein Zertifikat ist verpflichtend
      - und muss von Apple gekauft werden...
  - Packaging als gezippte Verzeichnisstruktur

# Ein Key für iOS

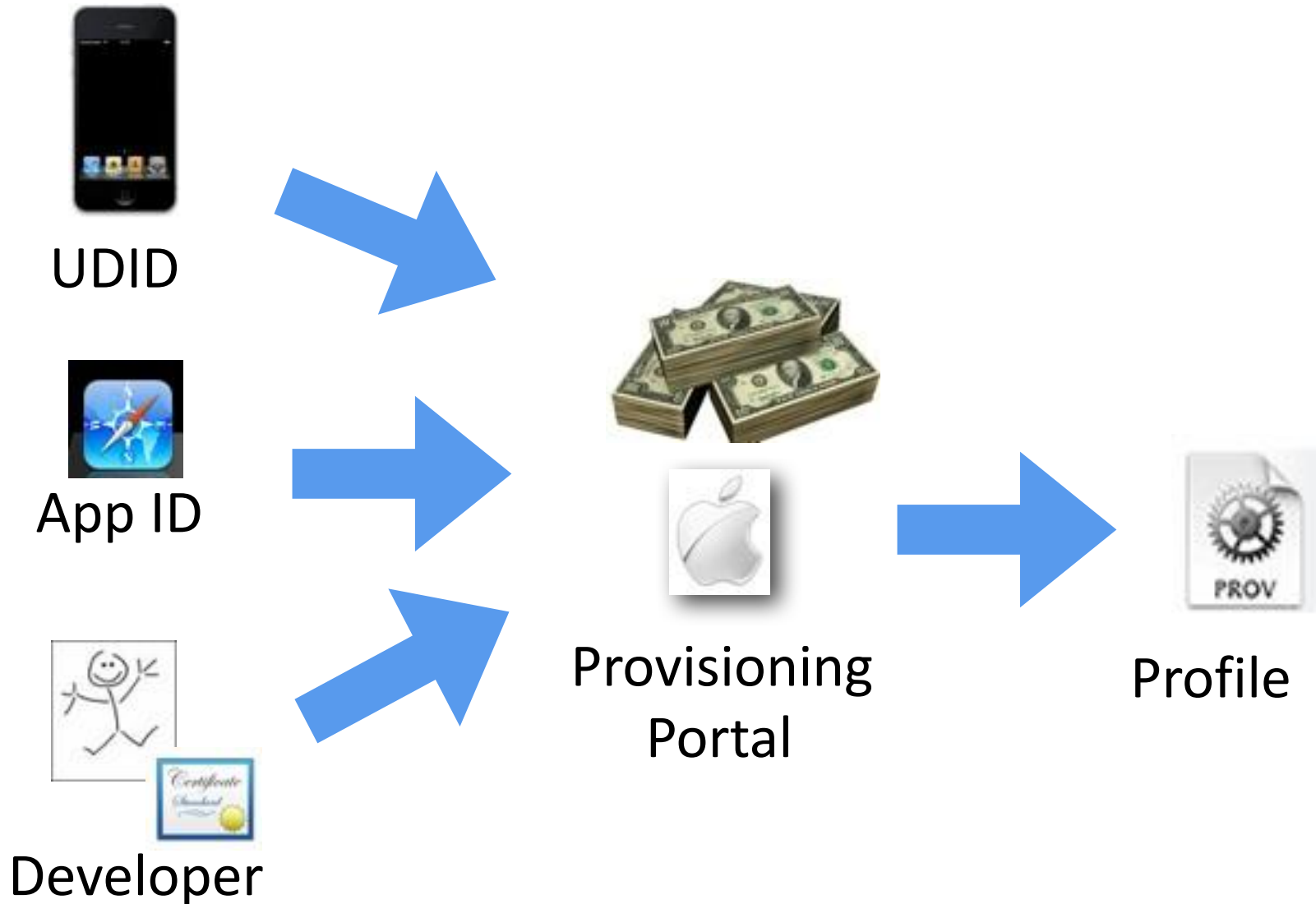
Key	Type	Value
Localization native development region	String	en
Bundle display name	String	\$(PRODUCT_NAME)
Executable file	String	\$(EXECUTABLE_NAME)
► Icon files	Array	(0 items)
Bundle identifier	String	de.integrata.training.\$(PRODUCT_NAME:rfc1034identifier)
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	APPL
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
Bundle version	String	1.0
Application requires iPhone environment	Boolean	YES
► Required device capabilities	Array	(1 item)
► Supported interface orientations	Array	(3 items)
► Supported interface orientations (iPad)	Array	(4 items)

# Provisioning: Ganz einfach!



- Kurze Zusammenfassung der notwendigen Schritte:
  - Registrierung bei Apple und Erzeugung eines Developer-Accounts
  - Kaufen des Entwickler-Zertifikats
  - Installation auf dem Keystore des Entwickler-Rechners
    - Vor einem Upgrade des Systems oder eines Wechsels auf einen anderen Rechner muss das Zertifikat exportiert werden, sonst ist es weg
  - Erzeugung eines Provisioning Profiles
    - Enthält das Entwickler-Zertifikat, iPhone Device IDs und App IDs
  - Ein Entwickler lädt dieses Profil und installiert es in XCode

Erst wenn alles passt ist es einfach...

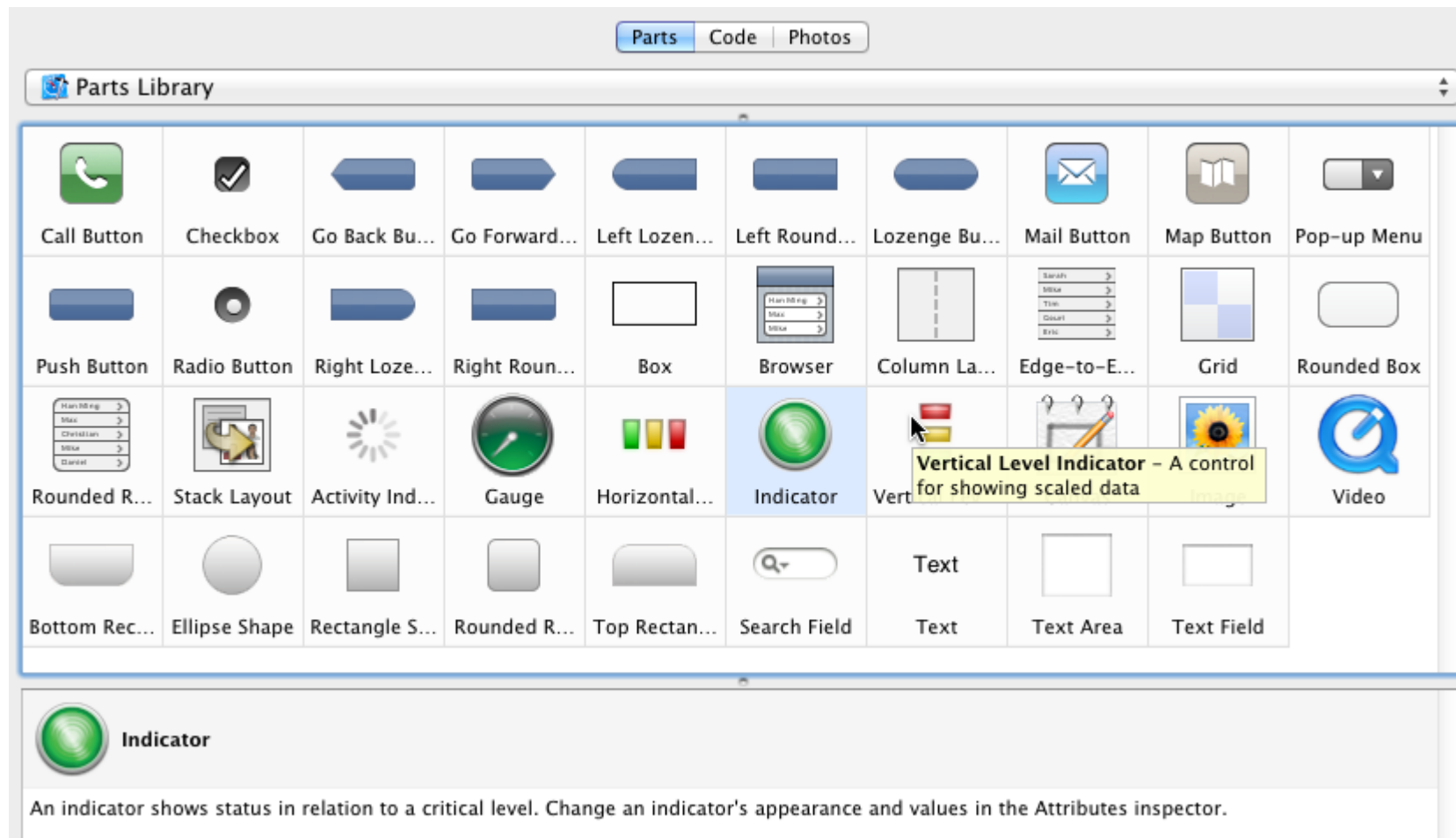




8.3

## **DASHCODE**

- Komplette IDE für die Entwicklung von iPhone Web Anwendungen
  - Visueller WYSIWYG Editor
  - Editoren für JavaScript, CSS
  - Integrierter Web Server
  - Anbindung an iOS Simulator
  - Debugger
- Bibliothek von Widgets



8.4

## **IOS KURZEINFÜHRUNG**

- *“The Swift Programming Language greatly benefited from the experiences hard-won by many other languages in the field, drawing ideas from Objective-C, Rust, Haskell, Ruby, Python, C#, CLU, and far too many others to list.”*
  - Chris Lattner

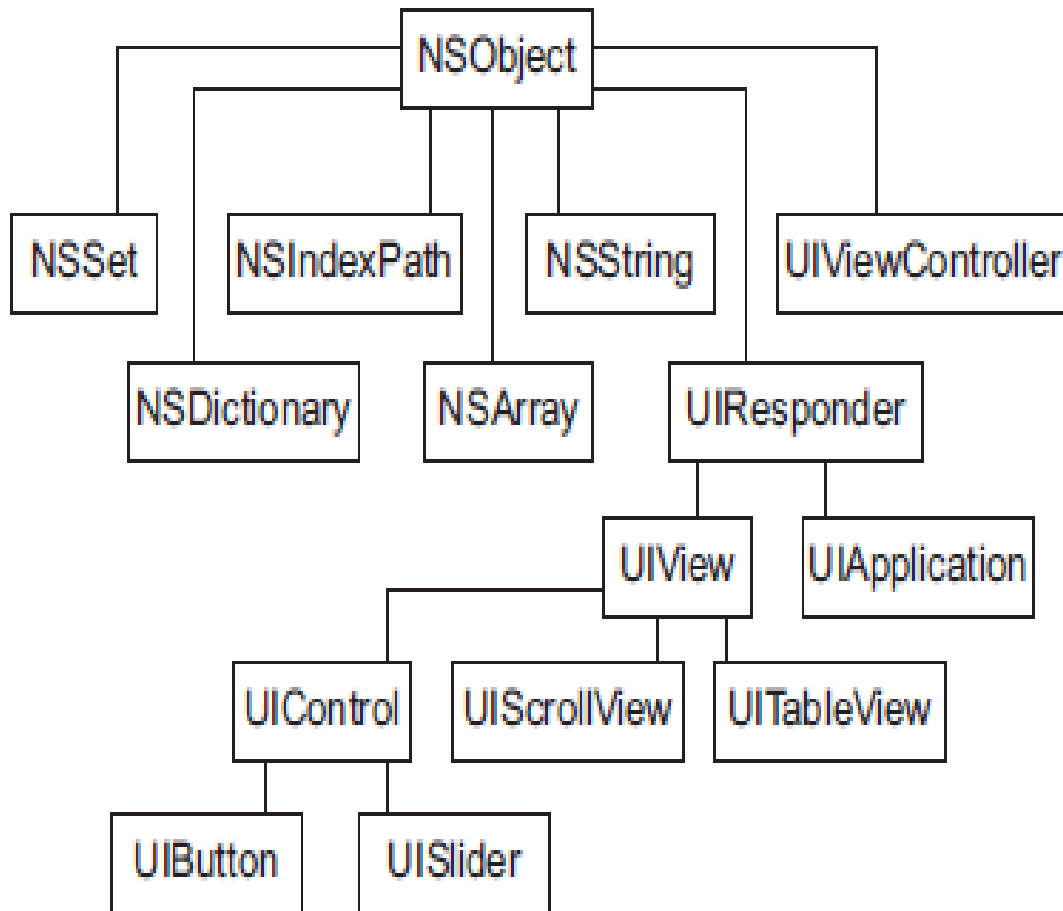
- Grundlegenden Konzeption
  - OO-Sprache, die etablierte Elemente aus anderen Sprachen übernimmt
  - Moderne Sprache, die Funktionale Programmierung unterstützt
    - Funktionsobjekte
    - Closures
  - Integration vorhandener Objective-C-Frameworks soll möglich sein
    - Wiederverwendung des Cocoa-Frameworks
  - Plattformübergreifende Versionen für
    - iOS
    - Linux
  - Open Source

Cocoa Touch

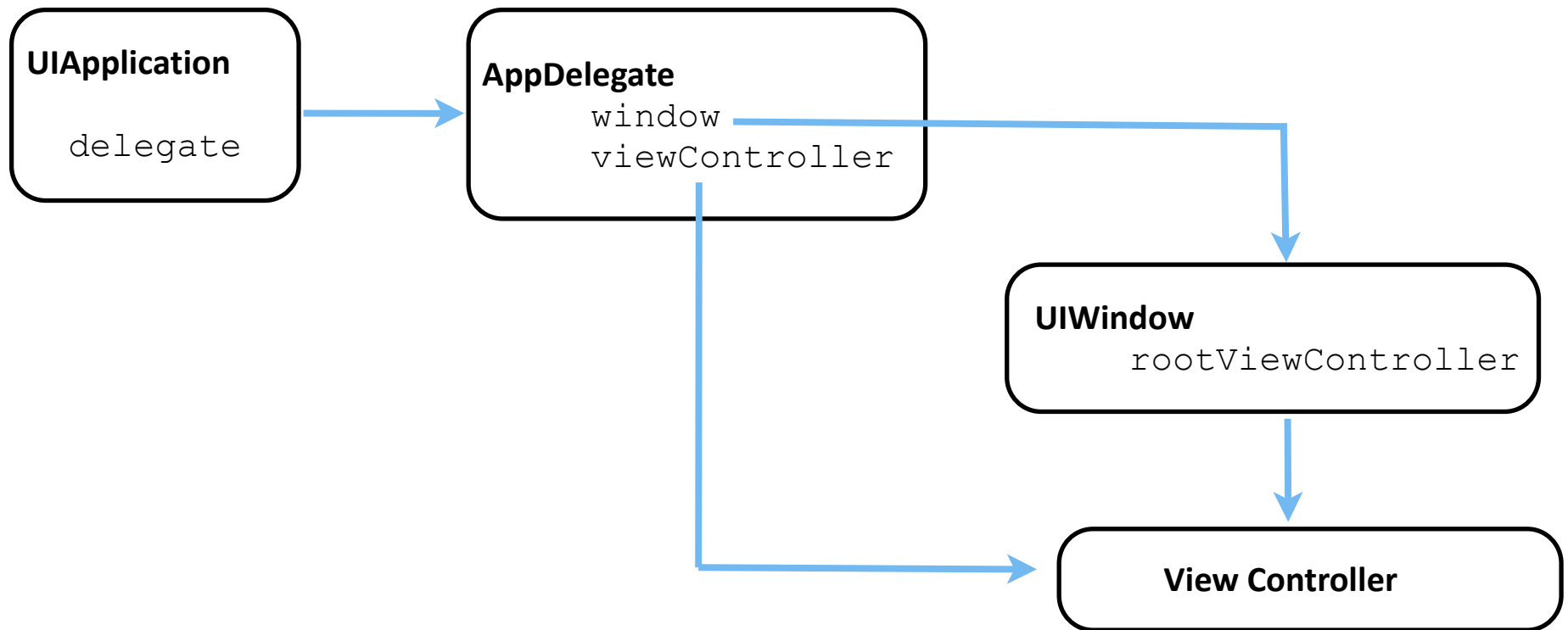
Media

Core Services

Core OS





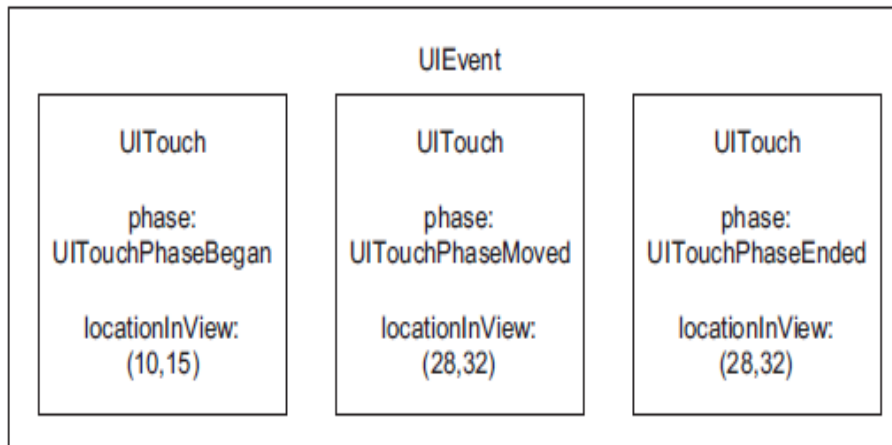


- Ein delegate wird über ein protocol definiert
- Damit werden beispielsweise für UI-Klassen Lifecycle-Callbacks definiert
- AppDelegate
  - application started
  - application activated
  - application sent to background
  - application stopped
  - ...
- Delegates werden auch für den Zugriff auf iPhone Services benutzt
  - Vereinfacht: Eine Implementierung, die beispielsweise einen LocalizationManager benötigt implementiert das Protokoll, und registriert sich beim Localization Service



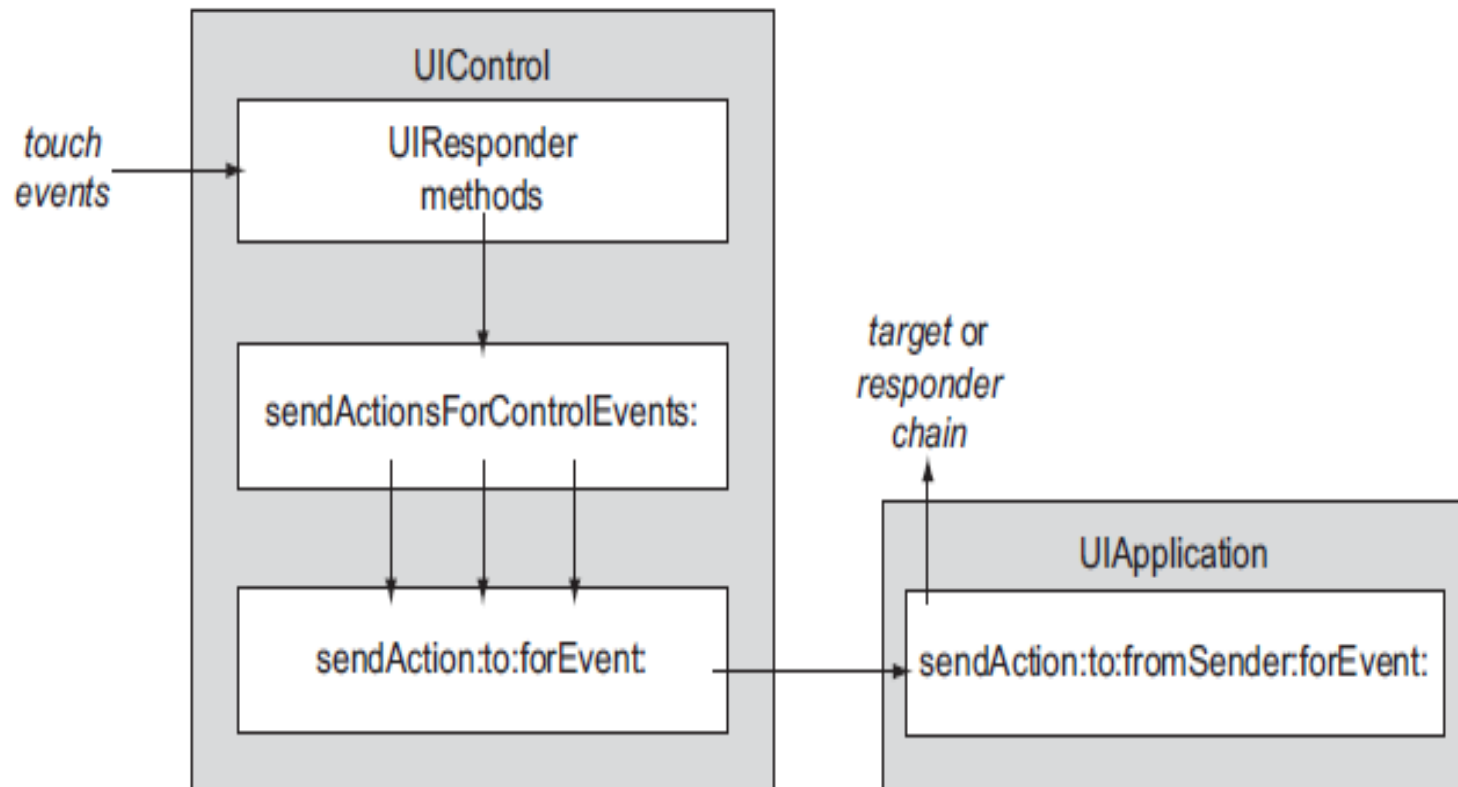
- Stellen das fundamentale View-Management für iPhone Anwendungen zur Verfügung
  - Laden eines .xib-file
- Darstellung der zugehörigen View
  - Parsen und Interpretieren der .xib
- Modale Views und Dialoge
- Rotieren der Darstellung
  
- Subklassen:
  - UINavigationController
  - UITabBarController
  
- ViewController (.h) und View (.xib) sind verbunden
  - IBOutlet, IBAction

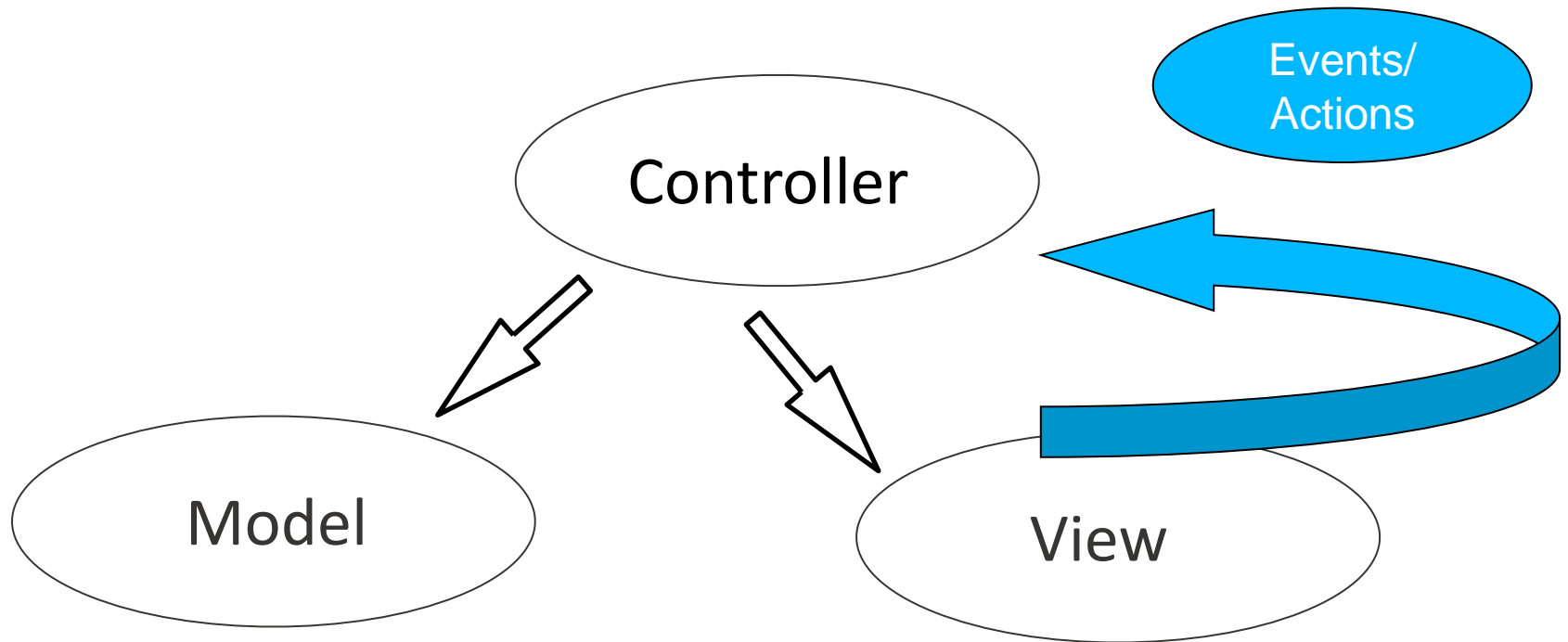
- **UITouch**
  - Basis-Klasse für Touch-Events
  - Properties beschreiben das Event als Objekt
    - view
    - tapCount
    - ...
- **UIEvent**



- Der UIResponder fängt Events
  - touchesBegan
  - touchesMoved
  - touchesEnded
  - touchesCancelled

- Events werden selten direkt programmiert
- Eine UIControl transformiert Events in Actions

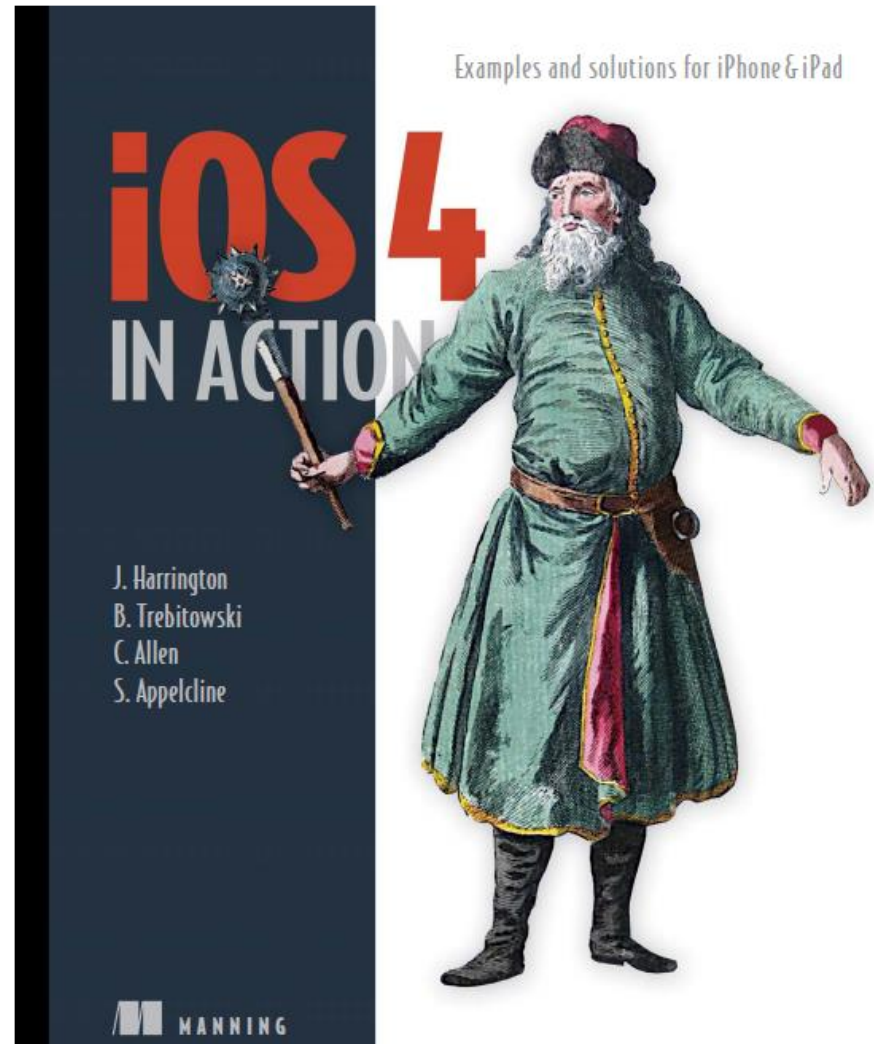
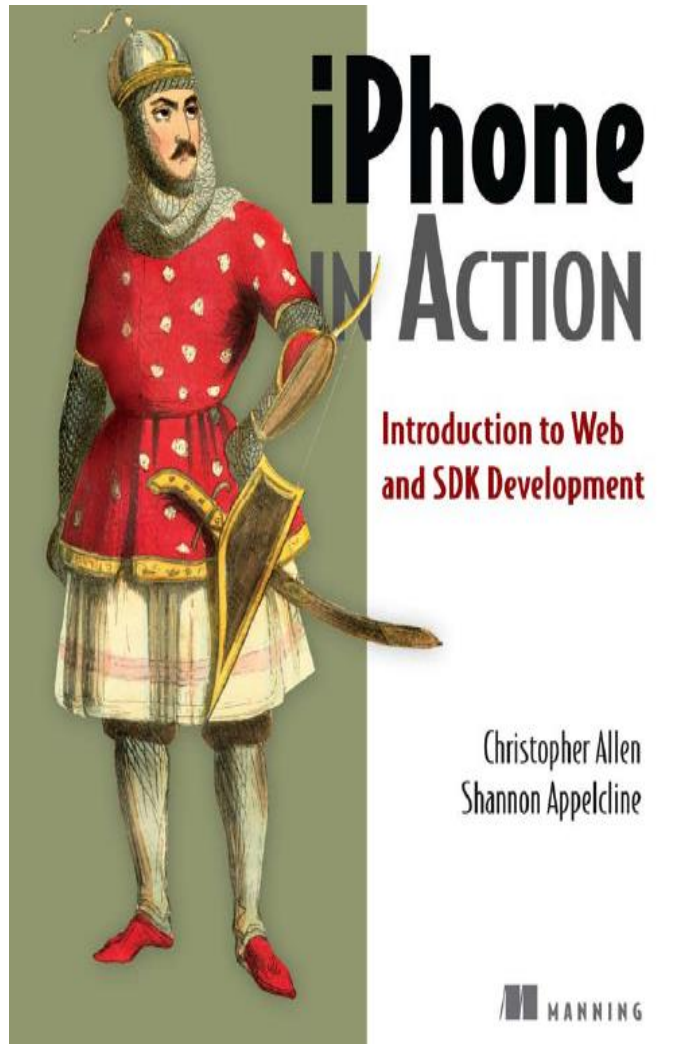






8.5

## LITERATUR



9

## WINDOWS PHONE



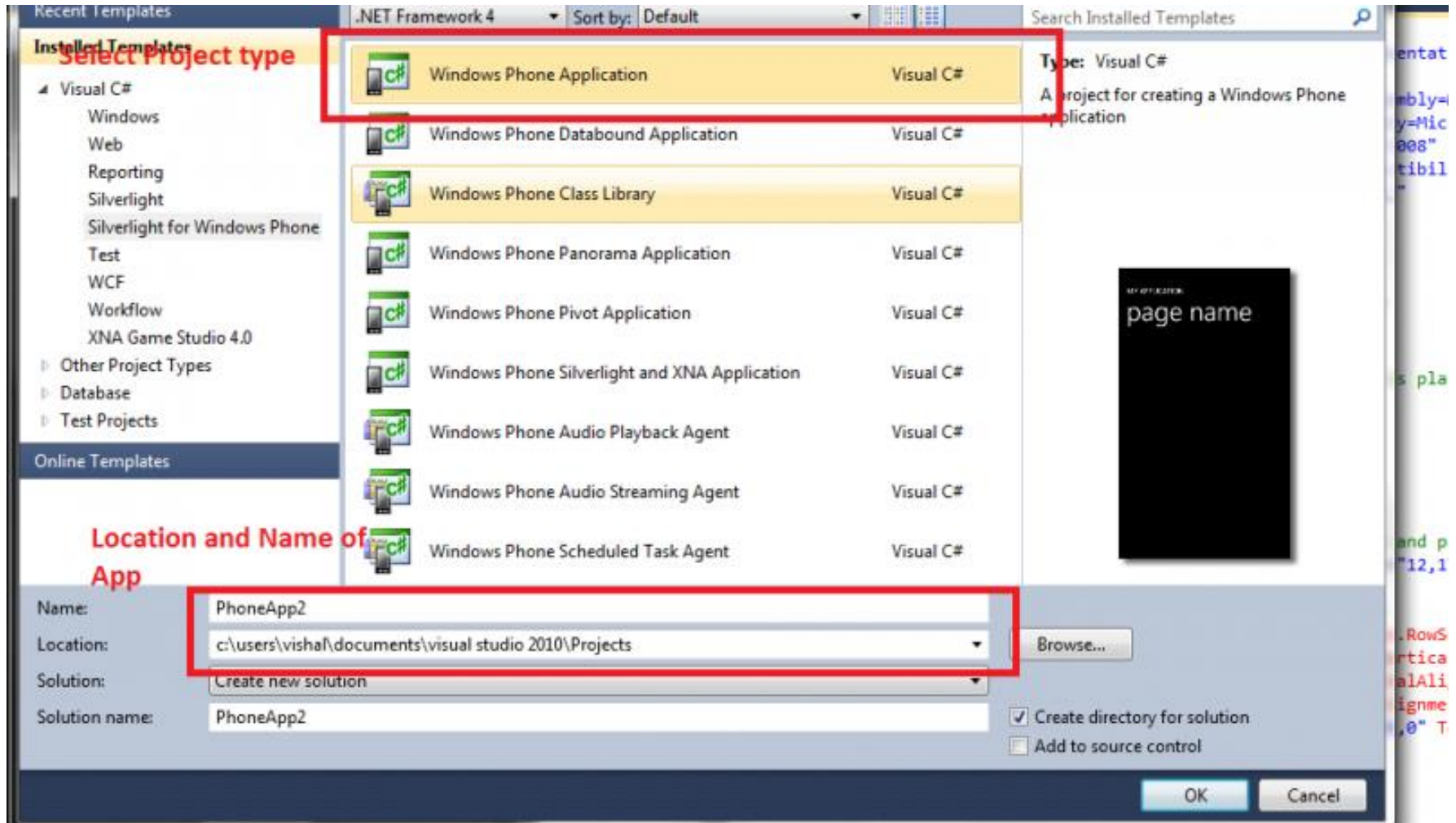
Windows<sup>®</sup>  
phone

9.1

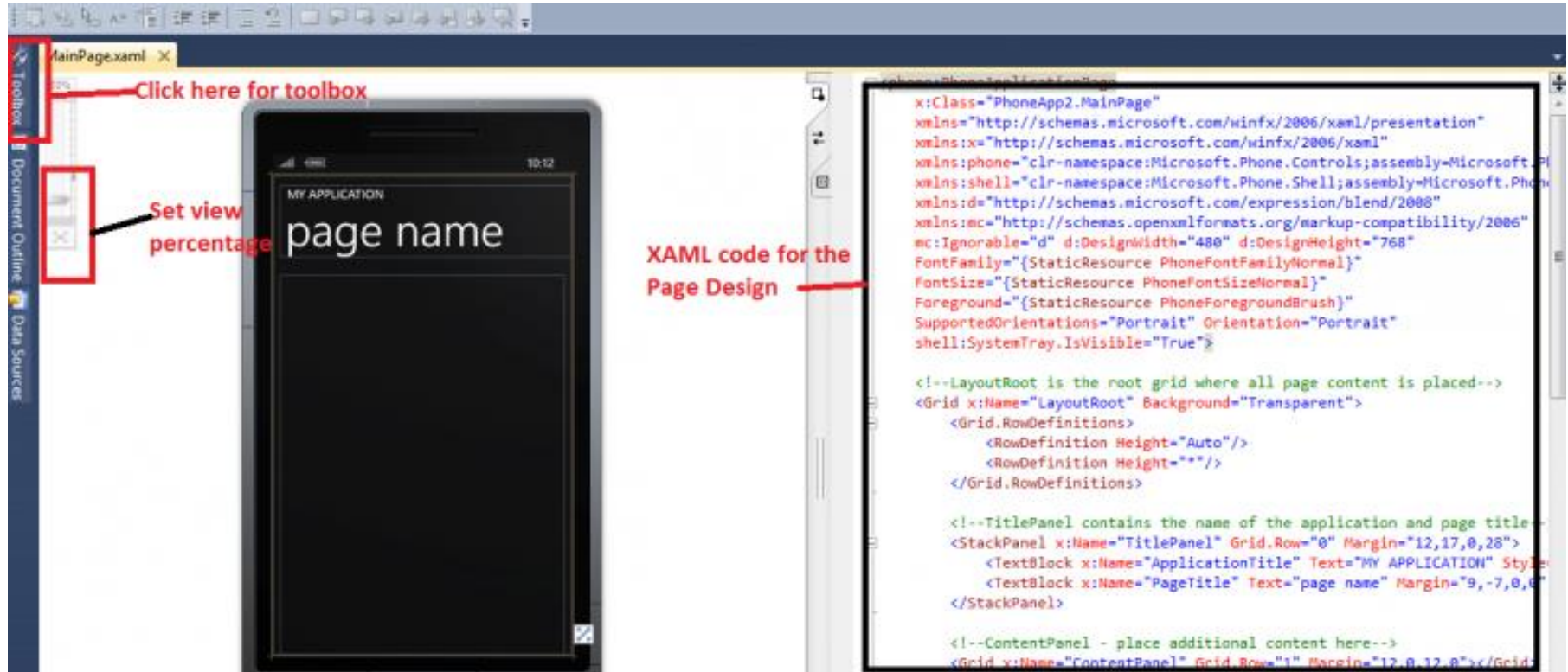
## **ANWENDUNGS-ENTWICKLUNG**



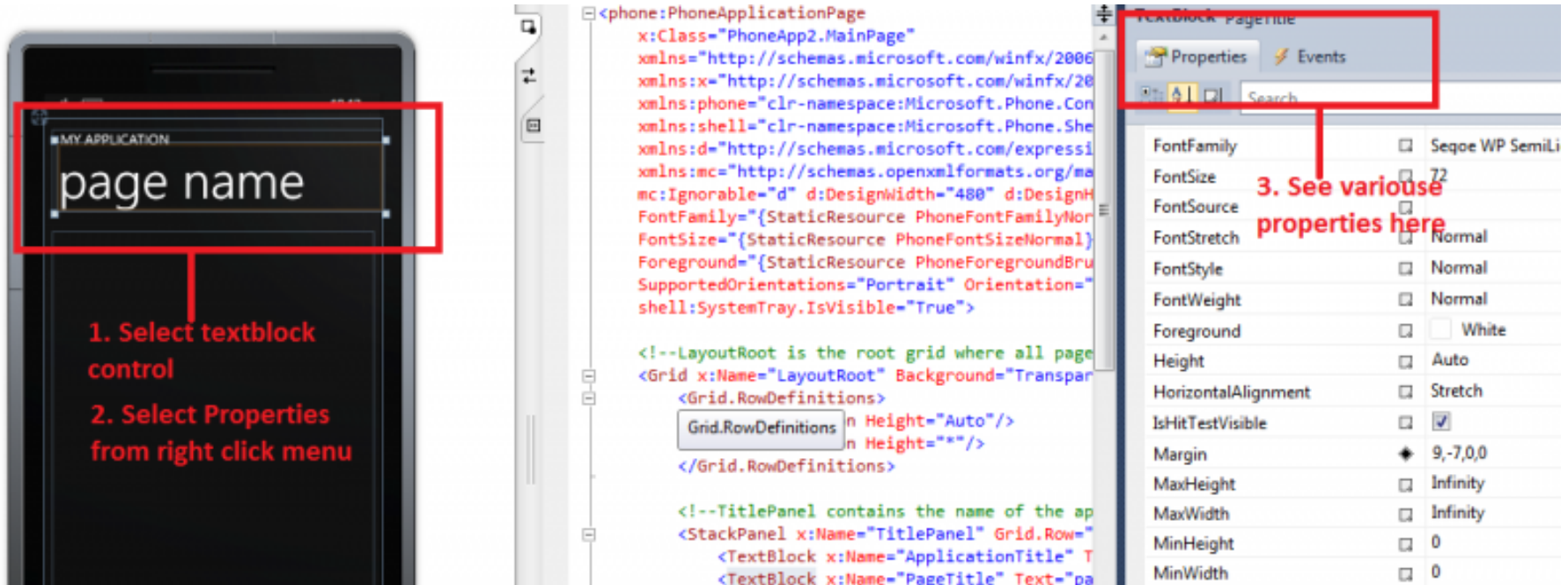
# Projekt anlegen



# Das erzeugte Projekt







1. Select textblock control

2. Select Properties from right click menu

3. See various properties here

```
<phone:PhoneApplicationPage
  x:Class="PhoneApp2.MainPage"
  xmlns="http://schemas.microsoft.com/winfx/2006/08/xaml"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/08/xaml"
  xmlns:phone="clr-namespace:Microsoft.Phone.Controls;assembly=Microsoft.Phone.Controls"
  xmlns:shell="clr-namespace:Microsoft.Phone.Shell;assembly=Microsoft.Phone.Shell"
  xmlns:d="http://schemas.microsoft.com/expression/2010/deployment/xaml"
  xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
  mc:Ignorable="d" d:DesignWidth="480" d:DesignHeight="768"
  FontFamily="{StaticResource PhoneFontFamilyNormal}"
  FontSize="{StaticResource PhoneFontSizeNormal}"
  Foreground="{StaticResource PhoneForegroundBrush}"
  SupportedOrientations="Portrait" Orientation="Portrait"
  shell:SystemTray.IsVisible="True">

  <!--LayoutRoot is the root grid where all page content is placed-->
  <Grid x:Name="LayoutRoot" Background="Transparent">
    <Grid.RowDefinitions>
      <Grid.RowDefinition Height="Auto"/>
      <Grid.RowDefinition Height="*"/>
    </Grid.RowDefinitions>

    <!--TitlePanel contains the name of the application-->
    <StackPanel x:Name="TitlePanel" Grid.Row="0" Grid.Column="0">
      <TextBlock x:Name="ApplicationTitle" Text="Application Title" />
      <TextBlock x:Name="PageTitle" Text="Page Title" />
    </StackPanel>
  </Grid>
</phone:PhoneApplicationPage>
```



```
namespace PhoneApp2
{
    public partial class MainPage : PhoneApplicationPage
    {
        // Constructor
        public MainPage()
        {
            InitializeComponent();
        }

        private void btnLogin_Click(object sender, RoutedEventArgs e)
        {
            if (txtPIN.Text == "1234")
            {
                NavigationService.Navigate(new Uri("/Page1.xaml", UriKind.Relative));
            }
            else
            {
                MessageBox.Show("You have enetered wong PIN !");
            }
        }
    }
}
```

Type this code

9.2

## LITERATUR

