Cross-Platform Apps mit HTML5/JS/CSS/PhoneGap

Proseminar *Objektorientiertes Programmieren mit .NET und C#*

Florian Schulz

Institut für Informatik
Software & Systems Engineering

Einführung

- Was hat Cross-Plattform mit C# und .Net zu tun?
  - Die native Programmierung von Windows Phone erfolgt mit Hilfe von C# und .Net

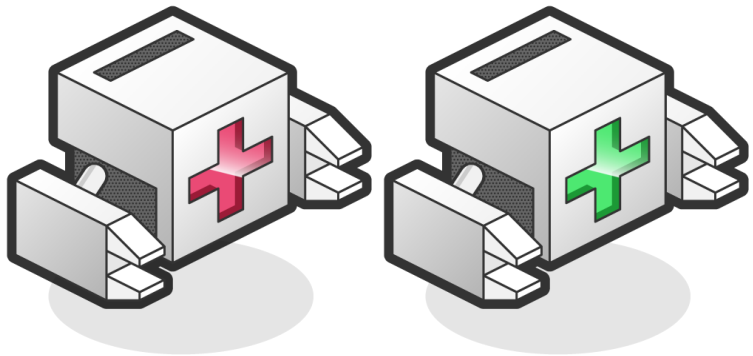
- Cross-Plattform, warum?
  - Zeitersparnis
  - Kostenersparnis
  - Größerer Markt

Agenda

- Einführung
- Ziele & Motivation
- Arten der Implementierung für mobile Anwendungen
- Aufbau einer PhoneGap Applikation
- Implementierung einer PhoneGap Applikation
- Demo
- jQueryMobile
- Demo
- Grenzen von PhoneGap
- Fazit

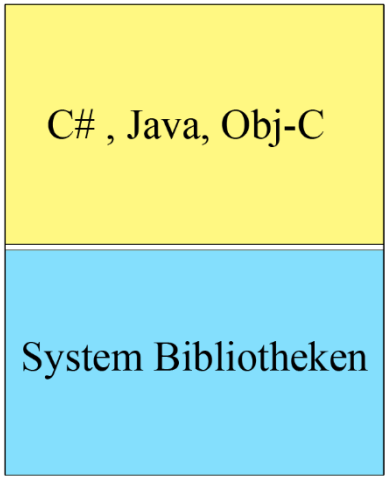
Ziele und Motivation

- Ziele
  - Einfachere & schnellere Entwicklung
  - Kostenersparnis
  - Plattform Unabhänigkeit
- Motivation
  - „Develop once, run everywhere“



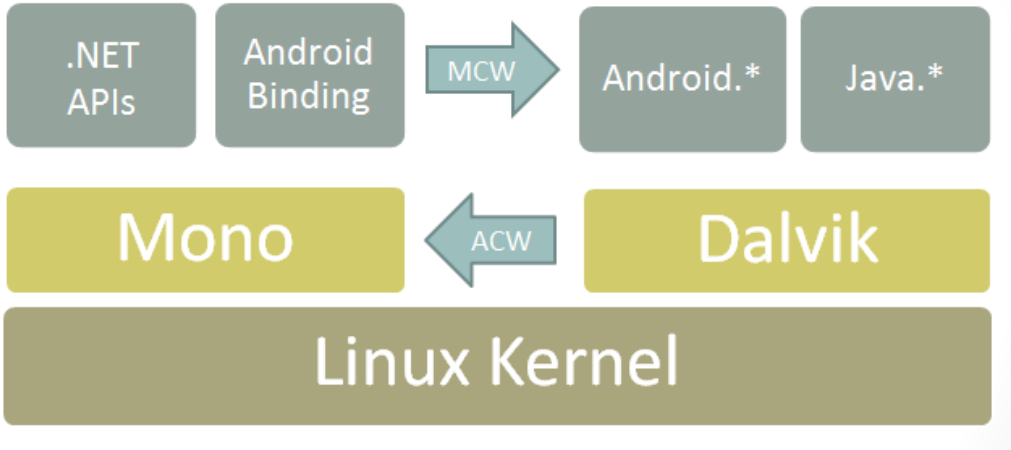
Arten der Implementierung (1) - Nativ

- Sprachen
  - Windows Phone 7/8 und Windows 8 in C#
  - Android in Java
  - iOS in Objective C
- Merkmale
  - Bedienelemente des Gerätes sind verfügbar
  - Native Benutzeroberfläche



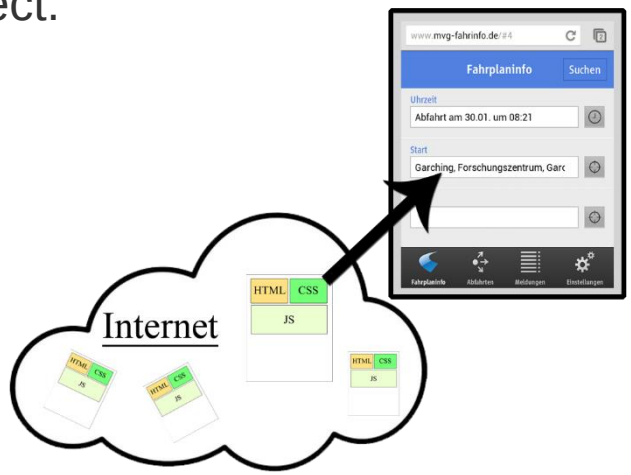
Arten der Implementierung (2) – Cross-Compiler

- Sprachen
  - Mono in C# + .Net
- Merkmale
  - iPhone: keine Mono-Runtime, vor der Laufzeit wird jeglicher C# Code in Maschinen-Code übersetzt
  - Android: eine Mono-Runtime + Projektionen auf die Android Bibliotheken

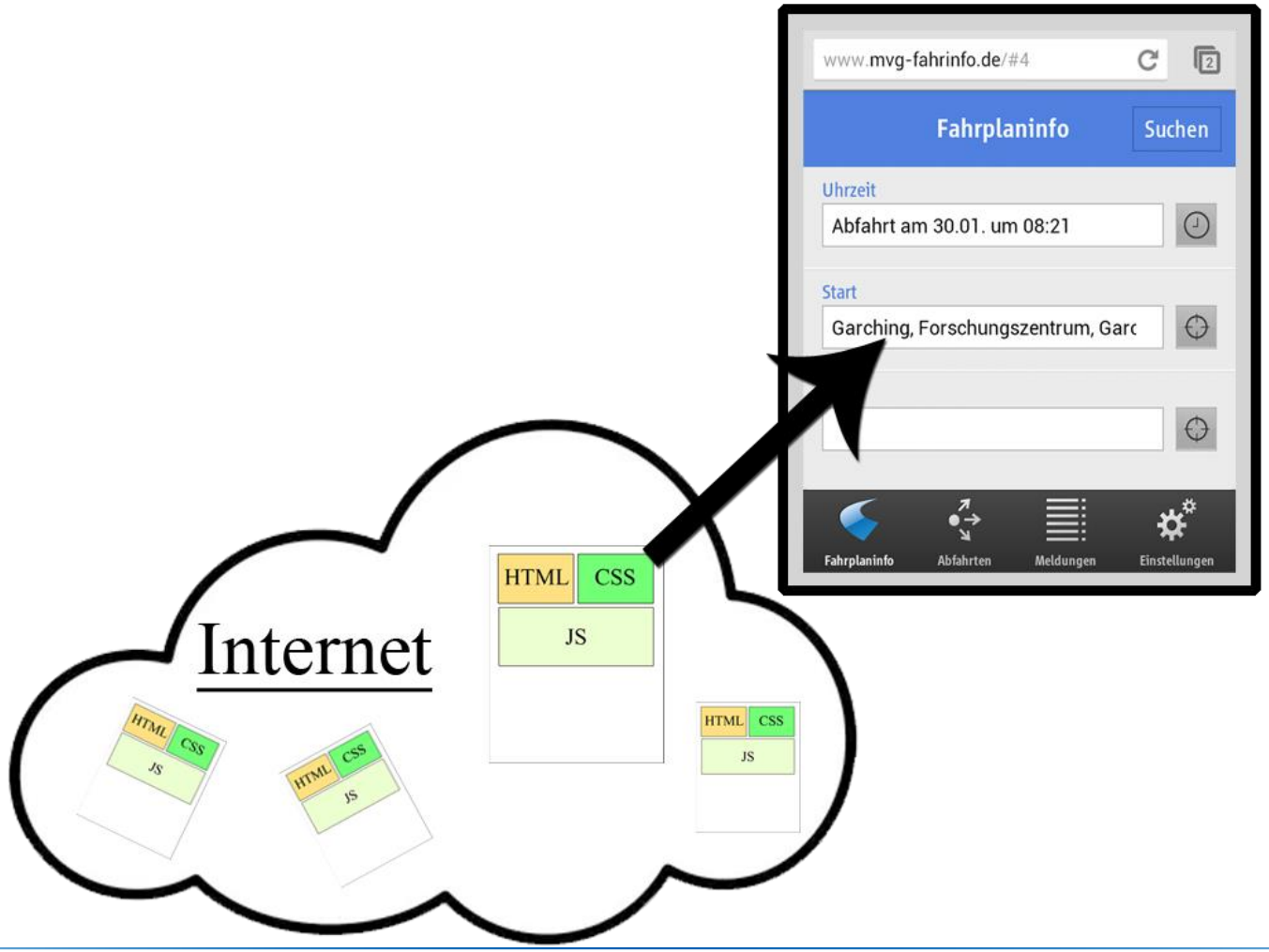


Arten der Implementierung (3) – Mobile Webseite

- Sprachen
  - HTML5, CSS, JavaScript
- Merkmale
  - Aufruf im Browser per URL
  - Nur Online verfügbar (durch Caching Offline)
  - Einiges ist durch HTML5 schon möglich, wie Ortsfunktionen, lokales Caching, ect.

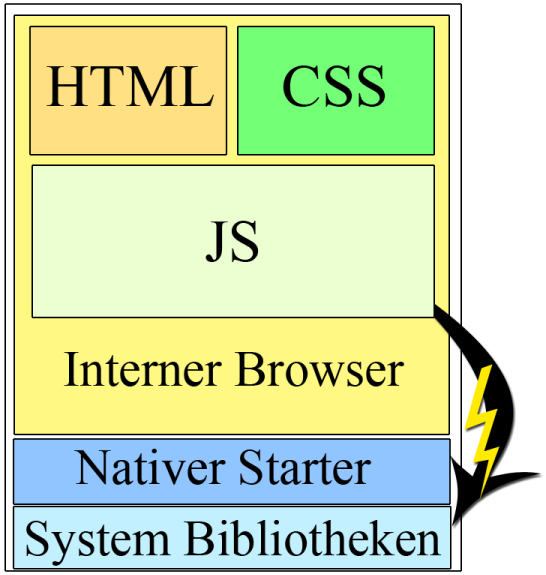


Arten der Implementierung (3) – Mobile Webseite



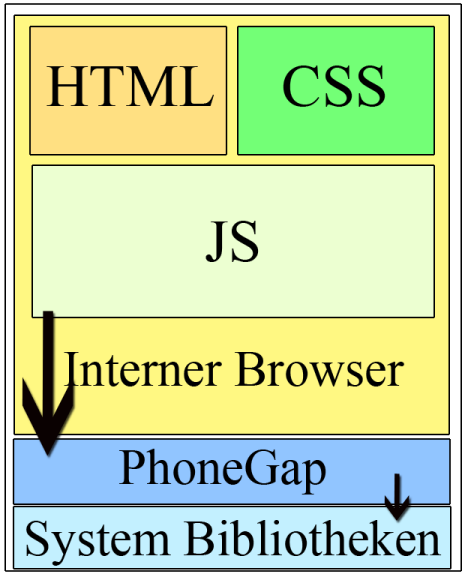
Arten der Implementierung (4) – Offline Webseite

- Sprachen
  - HTML5, CSS, JavaScript
- Merkmale
  - Die Webseite wird in eine Web Sicht eingebunden => Fullscreen
  - Anwendungspaket besteht im Wesentlichen aus HTML5 Seiten

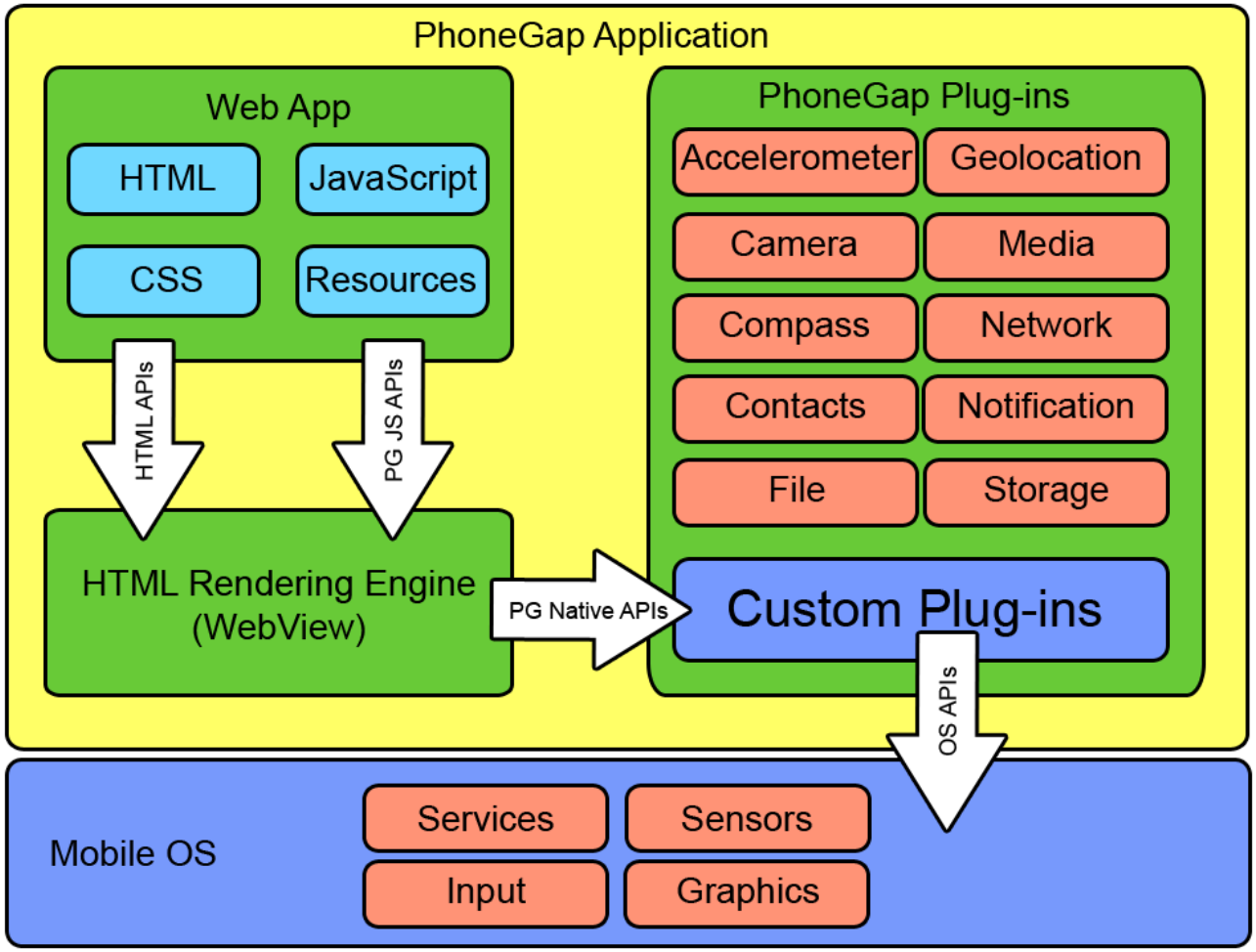


Arten der Implementierung (5) - PhoneGap

- Sprachen
  - HTML5, CSS, JS
- Merkmale
  - Verwendung der Systembibliotheken möglich, damit können z.B. Bewegungssensor und Kamera angesprochen werden



PhoneGap - Architektur



PhoneGap – Unterstüzte Funktionen

Unterstützte Funktionen	iPhone 3GS +	Android	BlackBerry OS 6.x	Windows Phone 7	Windows Phone 8	Windows 8
Beschleunigungssensor	✓	✓	✓	✓	✓	✓
Kamera	✓	✓	✓	✓	✓	✓
Kompass	✓	✓	x	✓	✓	✓
Kontakte	✓	✓	✓	✓	✓	✓
Dateien	✓	✓	✓	✓	✓	✓
Geo-Position	✓	✓	✓	✓	✓	✓
Medien	✓	✓	x	✓	✓	✓
Netzwerk	✓	✓	✓	✓	✓	✓
Benachrichtigung (Alarm)	✓	✓	✓	✓	✓	✓
Benachrichtigung (Sound)	✓	✓	✓	✓	✓	✓
Benachrichtigung (Vibration)	✓	✓	✓	✓	✓	✓
Speicher	✓	✓	✓	✓	✓	✓

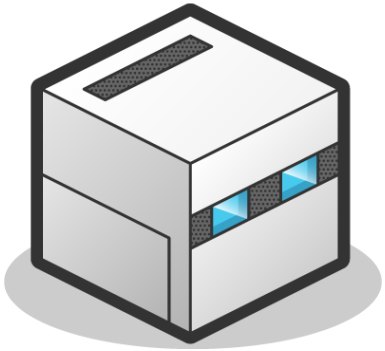
PhoneGap – Implementierung (1)

- Struktur der Implementierung
  - Jedes OS hat seine eigene, spezifische Entwicklungsumgebung und seine eigenen System Bibliotheken, deswegen werden überall verschiedene:

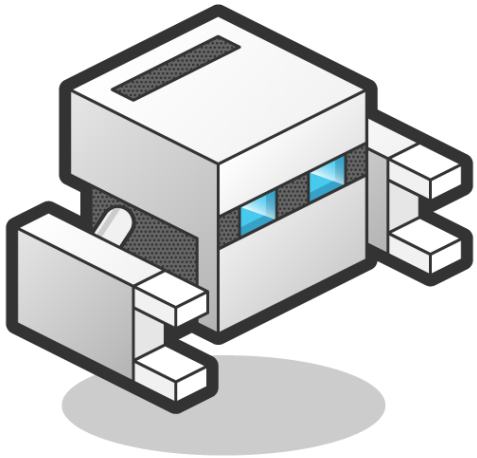
    Berechtigungen &

    Plugins eingebunden

- Demo am Beispiel von Visual Studio

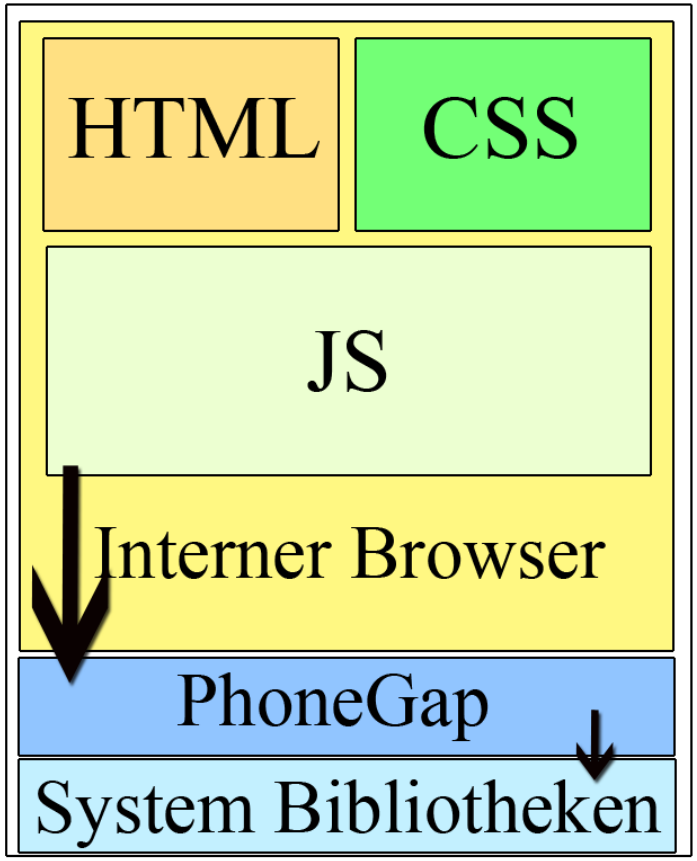


Demo



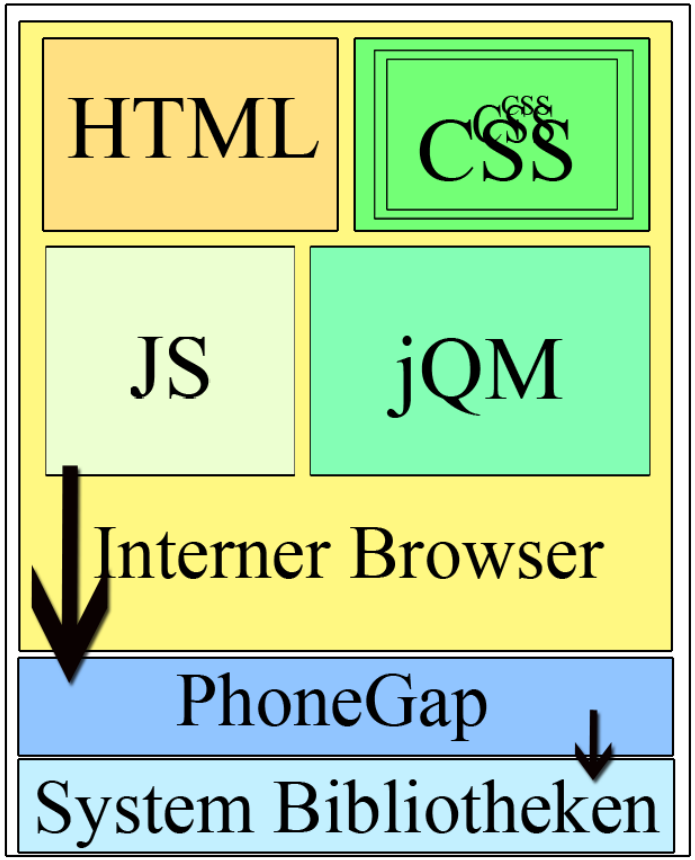
jQuery Mobile (1)

- Zur Erinnerung:



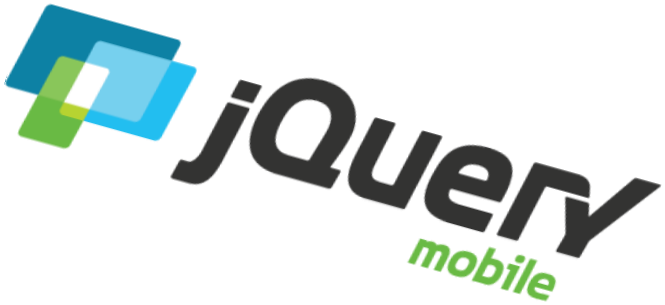
jQuery Mobile (2)

- jQueryMobile - Framework für die Benutzeroberfläche

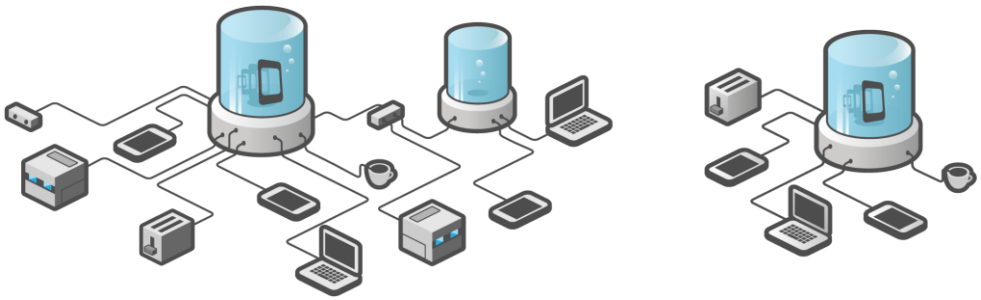


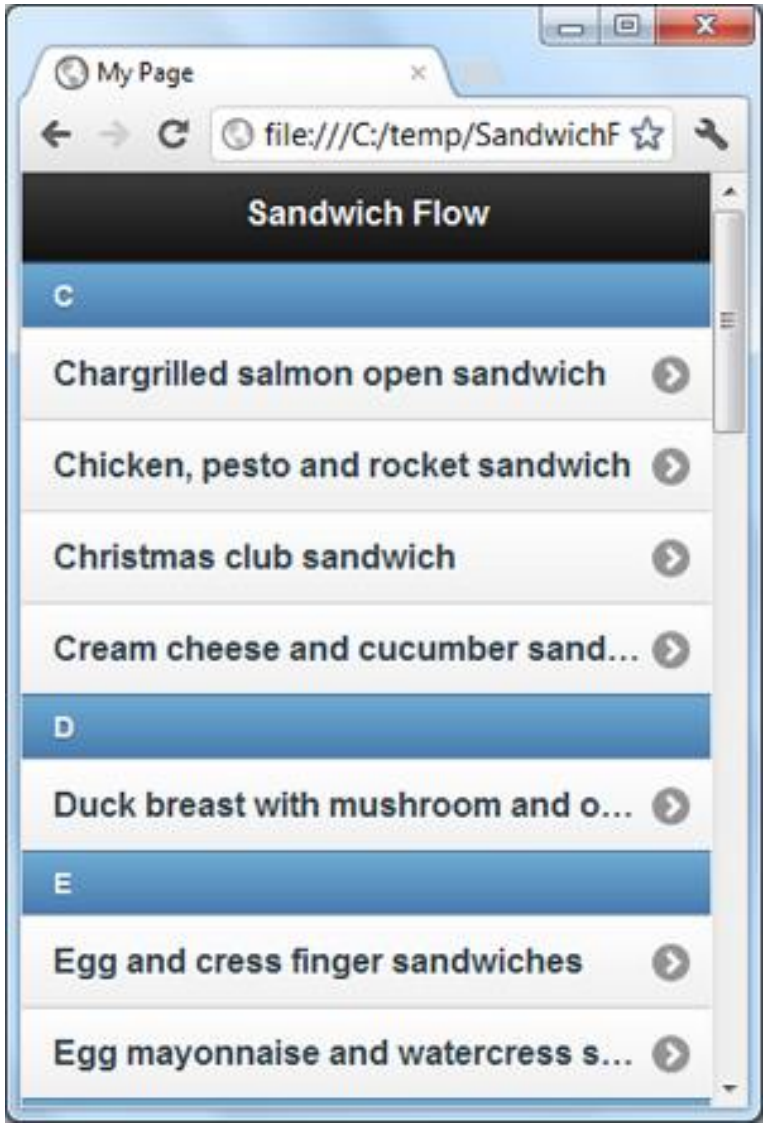
jQuery Mobile (3)

- Was macht jQuery Mobile genau?
  - Verwaltet die Benutzeroberfläche
  - Vereinfacht dynamische Ausgaben
  - Gibt die Möglichkeit Themes für jedes OS zu definieren.
- Wie macht es jQueryMobile
  - Manipuliert die DOM-Struktur
  - Liefert eine CSS mit

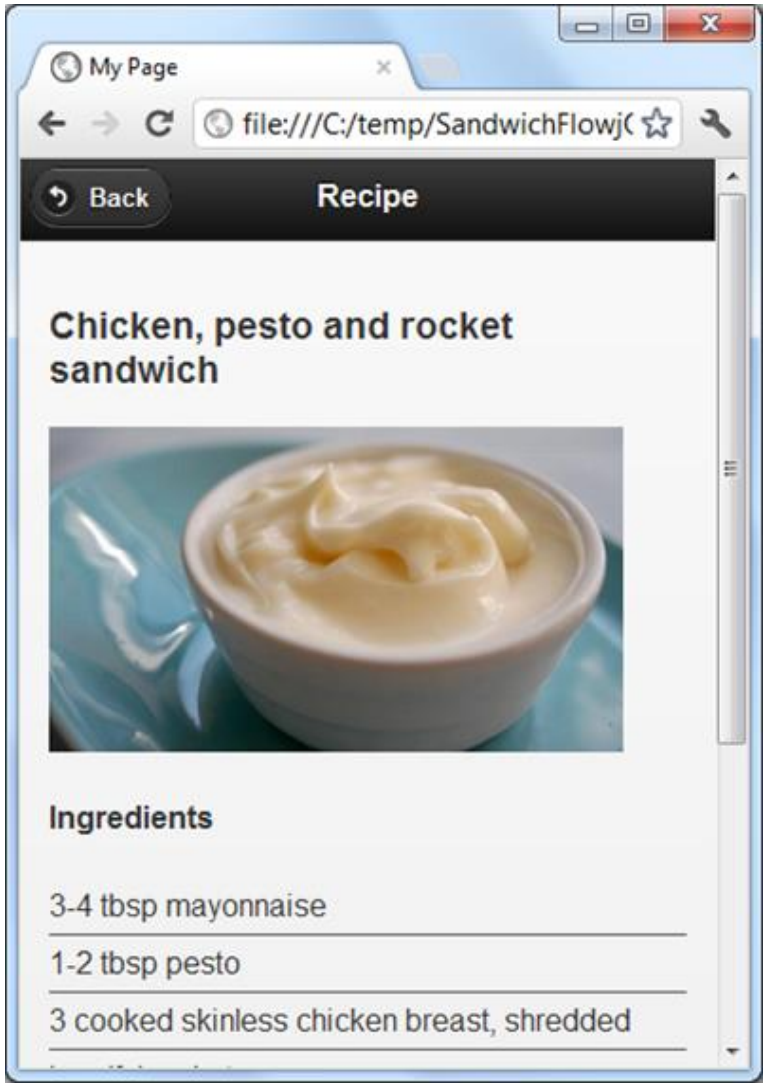


Demo




My Page
file:///C:/temp/SandwichF
Sandwich Flow
C
Chargrilled salmon open sandwich
Chicken, pesto and rocket sandwich
Christmas club sandwich
Cream cheese and cucumber sand...
D
Duck breast with mushroom and o...
E
Egg and cress finger sandwiches
Egg mayonnaise and watercress s...


My Page
file:///C:/temp/SandwichF
sandwich flow
C
Chargrilled salmon open sa...
Chicken, pesto and rocket s...
Christmas club sandwich
Cream cheese and cucumb...
D
Duck breast with mushroo...
E
Egg and cress finger sandw...
Egg mayonnaise and water...


My Page
file:///C:/temp/SandwichFlowj(
Back
Recipe
Chicken, pesto and rocket sandwich
Ingredients
3-4 tbsp mayonnaise
1-2 tbsp pesto
3 cooked skinless chicken breast, shredded


My Page
file:///C:/temp/SandwichFlowj(
recipe
Chicken, pesto and rocket sandwich
Ingredients
3-4 tbsp mayonnaise
1-2 tbsp pesto
3 cooked skinless chicken breast, shredded
handful rocket

PhoneGap - Grenzen

- Zugriff auf Systembibliotheken durch Projektion
- jQueryMobile behebt UI Schwächen
- Native Nutzererfahrung wird nie erreicht
- Durch Projektion geht Performance verloren
- Debugging absolut unzureichend
- Schwache Dokumentation
- Unterstützung von Features von OS zu OS unterschiedlich
- Größere Projekte ohne nativen Code nicht umsetzbar

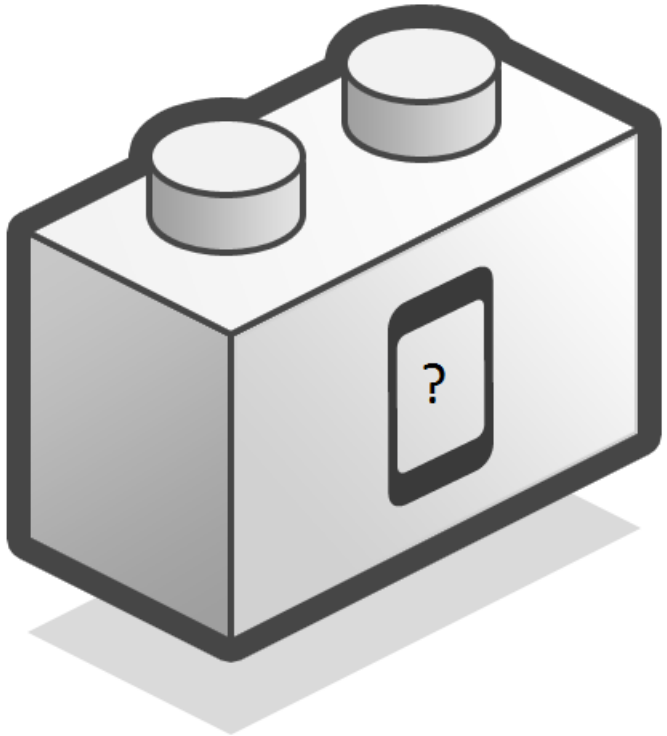
Fazit und Ausblick

- Wahl der Entwicklungsstrategie abhängig von Struktur der App:
  - Braucht die App Zugriff auf gerätespezifische Funktionen wie Kamera?
  - Wie hoch ist das verfügbare Budget?
  - Welche Endgeräte müssen unterstützt werden?
  - Braucht die Applikation Internetzugriff?
  - Welche Programmiersprachen beherrsche ich?
  - Wie wichtig ist mir die Geschwindigkeit und Perfomance der App?
  - Wie soll die Anwendung finanziert werdem?
  - Welchen Zweck hat die Anwendung?
  - Welches User-Experience ist erwünscht?

Fazit - Wann nehme ich was?

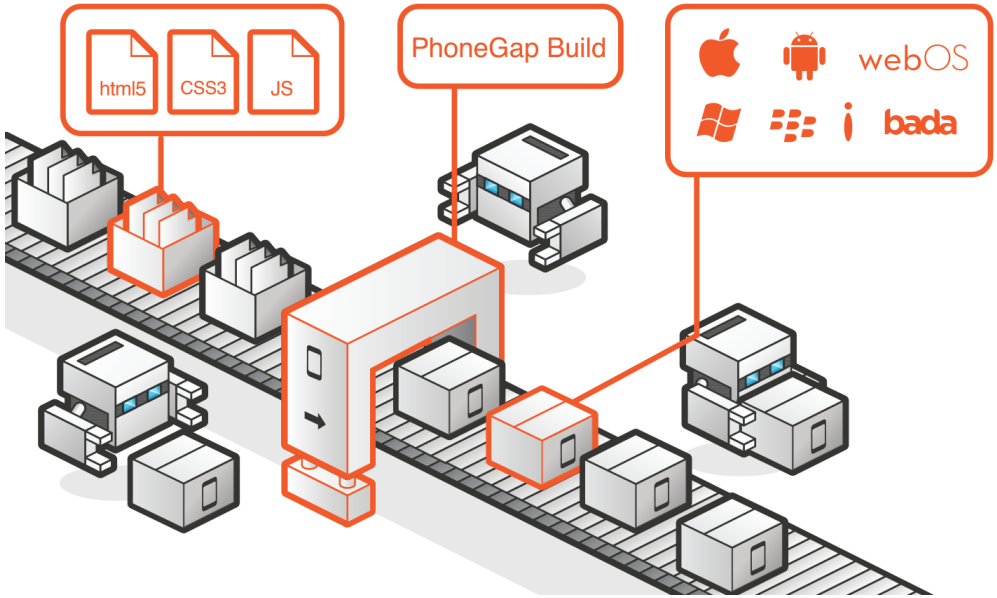
Inhalt	Nativ	WebApp	Offline	Cross-Compiler	PhoneGap
Informationsdarstellung					
Info + API Funktionen					
Grafische Anwendungen					
Performance Anwendungen					
Komplett Offline					
Einfach + API					
Komplex + API					
Marktplatz					
geringes Budget					

Noch Fragen?



PhoneGap Build

- Was ist PhoneGap Build?
  - Service zum Kompilieren der Anwendung für verschiedene mobile Betriebssysteme
- Was habe ich davon für einen Vorteil?
  - Kompilierung für alle Plattformen in der Cloud
- Was kostet dieser Dienst?
  - OpenSource: kostenfrei
  - Private Apps:
    - kostenfrei = 1
    - 9,99$/mo = 25



Vollständige Matrix für PhoneGap

Unterstützte Funktionen	iPhone	iPhone 3GS +	Android	Blackberry OS 5.x	BlackBerry OS 6.x	WebOS	Windows Phone 7	Windows Phone 8	Windows 8	Symbian	Bada
Beschleunigungssensor	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Kamera	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Kompass	x	✓	✓	x	x	✓	✓	✓	✓	x	✓
Kontakte	✓	✓	✓	✓	✓	x	✓	✓	✓	✓	✓
Dateien	✓	✓	✓	✓	✓	x	✓	✓	✓	x	x
Geo-Position	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Medien	✓	✓	✓	x	x	x	✓	✓	✓	x	x
Netzwerk	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Benachrichtigung (Alarm)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Benachrichtigung (Sound)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Benachrichtigung (Vibration)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Speicher	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	x

Vorteile/Nachteile von nativen Apps

- Vorteile
  - Verbreitung über Marktplatz
  - Alle APIs stehen zur Verfügung
  - Performance
  - Natives Benutzererlebnis
- Nachteile
  - Kein HTML5/CSS/JS
  - Kenntnis für viele verschiedene Programmiersprachen nötig
  - Versionierung
  - Portierung für auf Plattformen sehr schlecht möglich

Vorteile/Nachteile Mobilen Webseiten

- Vorteile
  - Versionierung
  - Schnelle und einfache Programmierung
  - Meist nur eine Anpassung des Layouts auf mobile Geräte
  - Einfache Verbreitung (Browser aufrufen mit URL)
- Nachteile
  - Kein API Zugriff (HTML5 Aktionen ausgenommen)
  - Schlechte Performance
  - Online Zugriff zum Caching nötig
  - Keine Verbreitung über Marktplatz möglich

Vorteile/Nachteile von Offline Webseiten

- Vorteile
  - Verbreitung über Marktplatz
  - Fullscreen Browser möglich
  - Voll Offline möglich
  - Schnelle und einfache Programmierung
- Nachteile
  - Kein API Zugriff möglich (HTML5 Aktionen ausgenommen)
  - Schlechte Performance
  - Portierung für auf andere Plattformen nötig
  - Kein natives Benutzererlebnis

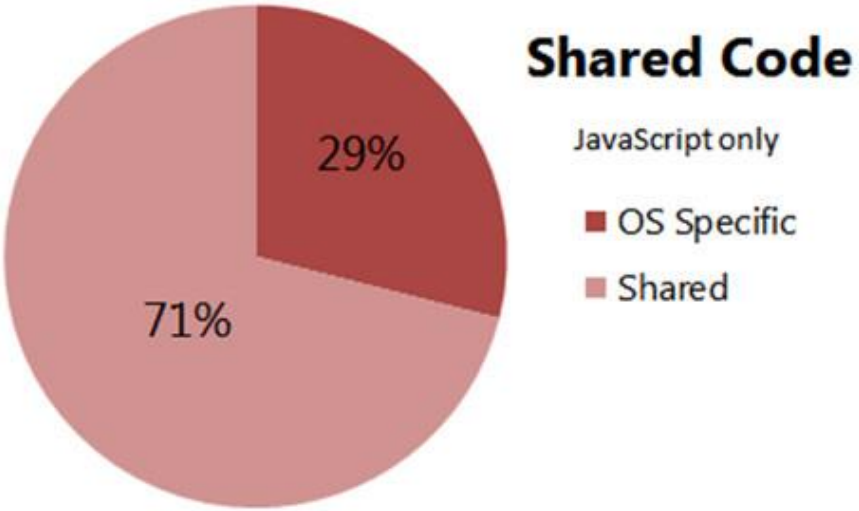
Vorteile/Nachteile von Cross Plattform Apps

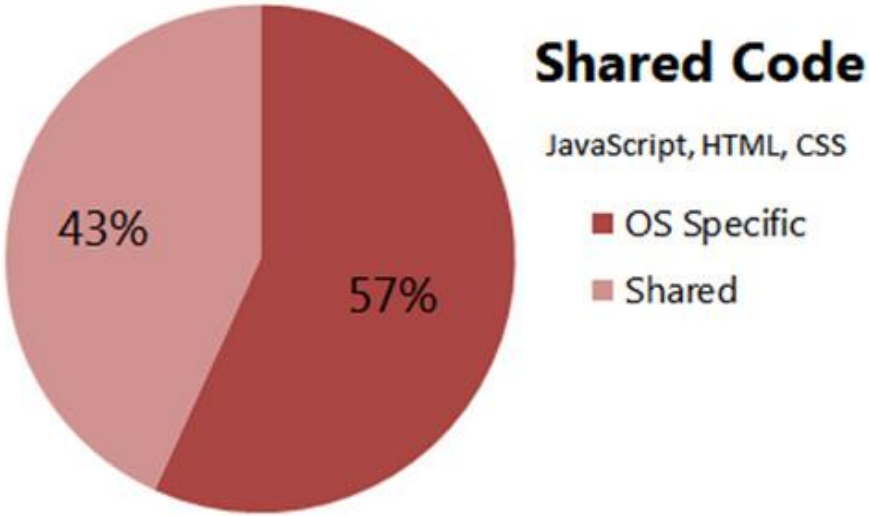
- Vorteile
  - Verbreitung über Marktplatz
  - Keine Browser Anwendung
  - Voll Offline möglich
  - Performance meist ganz gut
  - API Zugriff
- Nachteile
  - Meist kein HTML5 und CSS
  - Meist Programmiersprache erforderlich
  - Versionierung
  - Portierung für auf Plattformen nötig
  - Teilweise kostenpflichtig (Mono)

Vorteile/Nachteile von PhoneGap Apps

- Vorteile
  - Verbreitung über Marktplatz
  - APIs stehen zur Verfügung
  - Programmierung mit HTML5/CSS/JS
  - Sehr viele Betriebssysteme werden unterstützt
- Nachteile
  - Versionierung
  - Debugging sehr schwer möglich
  - Trotz einheitlicher API, unterschiedliches Verhalten auf den verschiedenen Betriebssystemen
  - Performance geht verloren
  - Browser Anwendung

Code Teilen - Beispiel





Zusatzfolien

- Eventuell nochmal genauer zur Implementierung
  - Mit schönen Screenshots
  - Weitere Hilfreiche Seiten für PhoneGap
- Genauere Ausführung zu jQueryMobile
- Andere Frameworks wie
  - Rhodes
  - Titanum
  - MonoTouch