

2016, appPlant UG (haftungsbeschränkt)

[hrushchak@appplant.de](mailto:hrushchak@appplant.de)

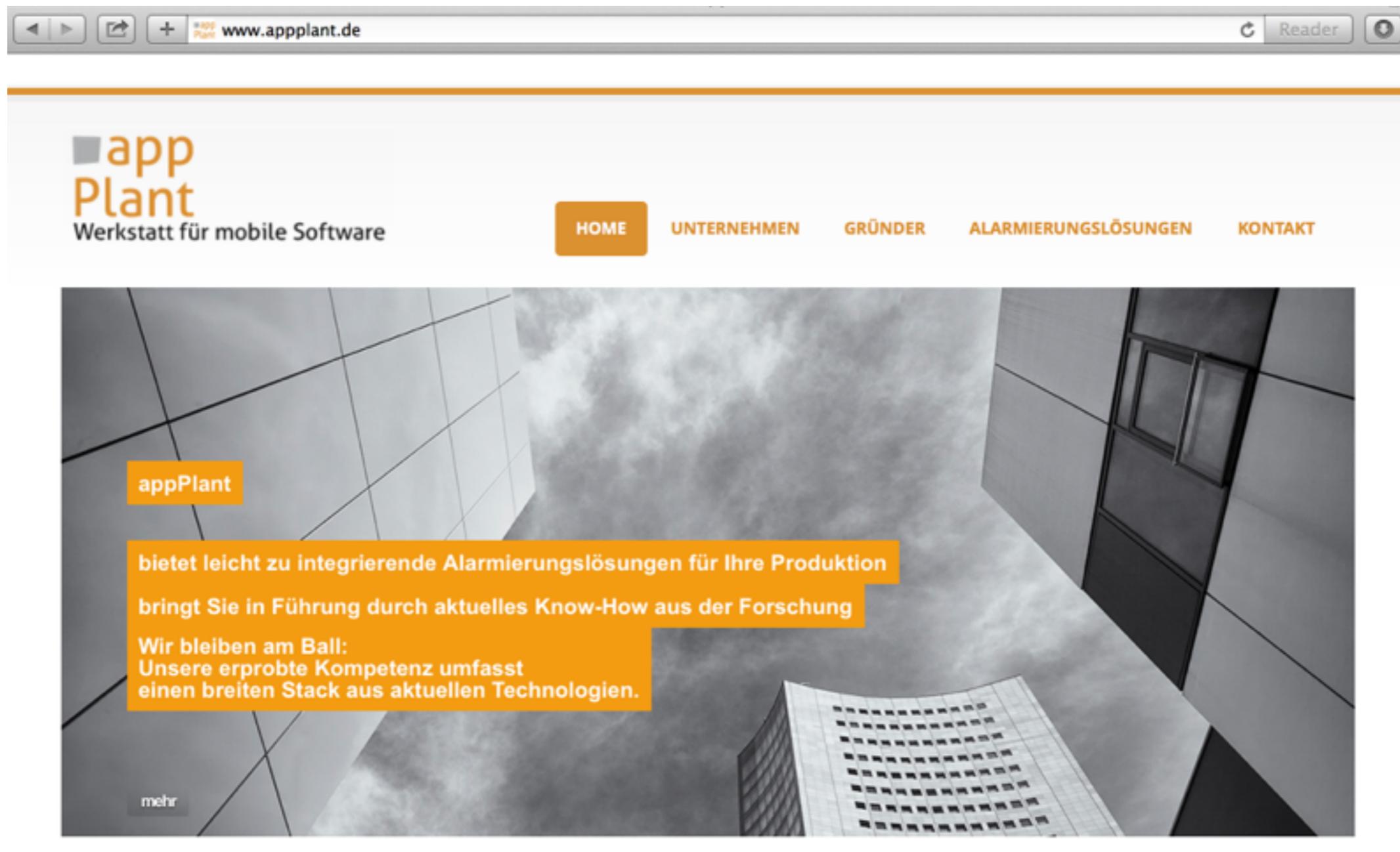


# appPlant

Werkstatt für  
mobile  
Software



# Wer wir sind ...



# Wer wir sind ...

## Leipziger Unternehmen

## Softwareentwicklung für die Industrie

- 2013 - Gründung, Stand 2016: 7 Mitarbeiter
- Seit 2013 offizieller Lieferant bei BMW
- Seit 2013 Projekte mit
  - BMW,
  - Univ. Leipzig,
  - Daimler Fleetboard,
  - MAN,
  - Zoccam (USA),
  - maaxi, perigee (GB),
  - papersmart,
  - IT SONIX ,
  - Wimdu,
  - vizzlo
- Spezialisierung auf
  - Mobile Architekturen, iOS, WP8, Android, BlackBerry, Apache Cordova
  - Enterprise Applications (JEE, .Net, Rails)
  - EAI, Prozess-Beratung, Rollout-Management
  - BI, Reporting

**Ca. 1/4 der Unternehmer gehen von einer Produktivitätssteigerung von 40 % aus, sollten alle Enterprise Applikationen mobil zur Verfügung gestellt werden**

Umfrage von Mobile Helix & Vanson Bourne

# Bedürfnisse der Unternehmen

- Sicherheit
- Netzwerkelastizität
- Intuitive und einheitliche UIs / innovative Benutzeroberflächen
- Hohe Integrationsfähigkeit mobiler Dienste
- Intelligente Konzepte zur Online- und Offline-Nutzung
- Integrierte Plattformen und intelligente Clients
- BYOD / Management von Devices, Applikationen und Content
- Kommunikation in Echtzeit
- Einfachheit der Entwicklung (write once, use across platforms)
- Performanz
- Personalisierung / Multi-User-Verwendung der Devices

# Herausforderungen bei der Umsetzung

- Fragmentierung
- Unterschiedliche Plattformen/Browser, Display-Auflösungen und Hardware-Ausstattungen
- Offline-Funktionalität
- Mangel an Standards
- Integration in und Verknüpfung mit bestehenden Systemumgebungen
- Mangel an Marktführern für Mobile App Tools

# Technologien in der Mobile-Software-Entwicklung



Werkstatt für mobile Software



# Kategorisierung der Mobile-Software-Technologien

## 1. Technologielieferant

- Native vs. Third-Party

## 2. Funktionsumfang

- in der Relation zu nativen APIs

## 3. Wie wird der Code geschrieben?

- Plattformagnostisch: 1 Technologie – N Plattformen
- Plattformgebunden: N Technologien – N Plattformen

## 4. Wie kommt der Code zum ausführbaren Zustand?

- Durch Code-Kompilierung: Ahead-of-Time vs. Just-in-Time
- Durch Code-Generierung
- Durch Interpretation zur Laufzeit

# Drei Klassen jenseits der Native-Entwicklung

## 1. Third-Party Plattformen

- Eine Third-Party-Technologie implementiert die nativen APIs
- m SDK zu n Mobile-Plattformen
- Beispiele: Mono

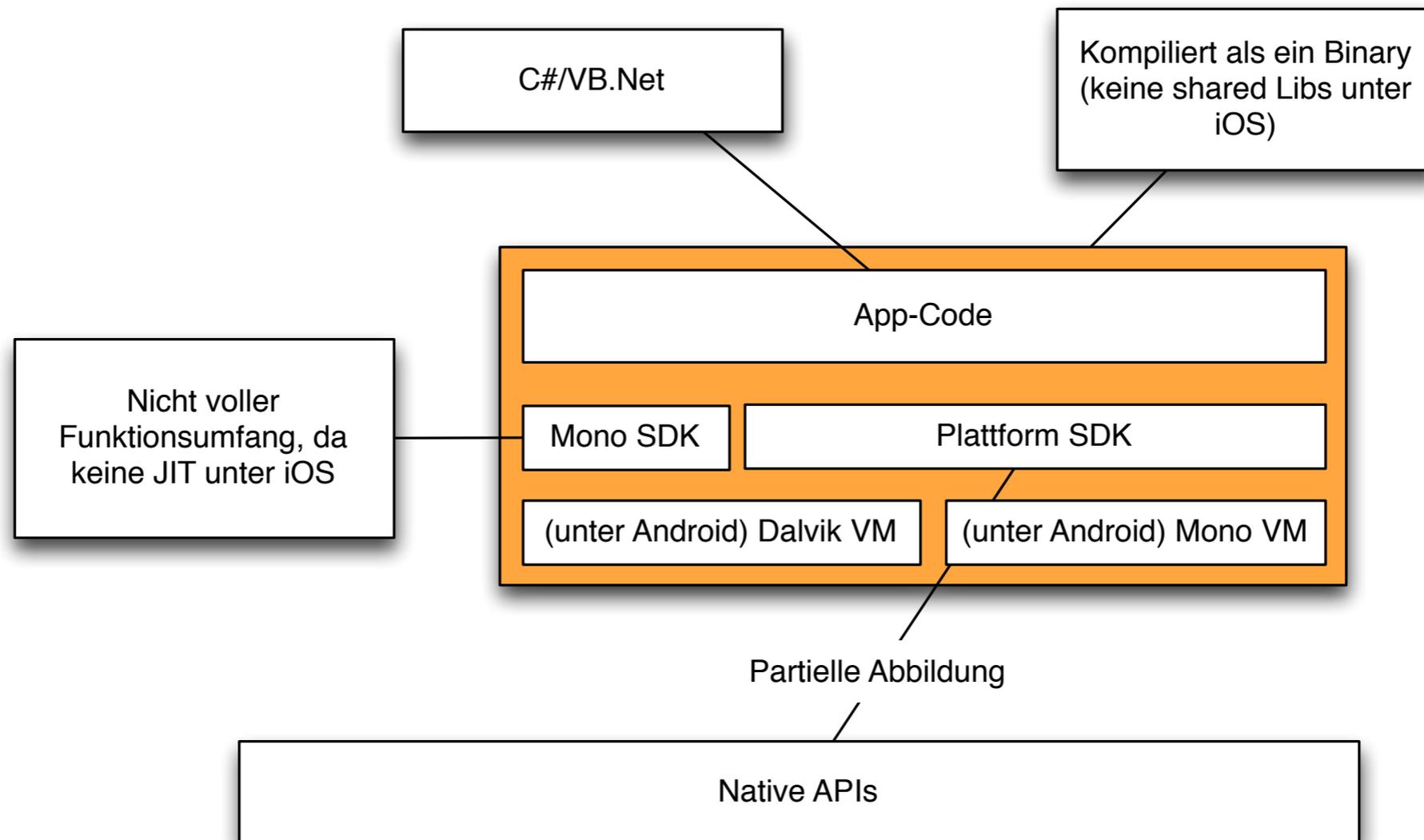
## 2. Cross-Plattform + Code-Generatoren

- Die App wird einmal mit einer plattformagnostischer Technologie entwickelt und durch einen speziellen Generatoren-Einsatz als native App ausgeliefert
- 1 SDK zu n Mobile-Plattformen
- Beispiele: Titanium, Marmalade, Rhomobile

## 3. WebApps

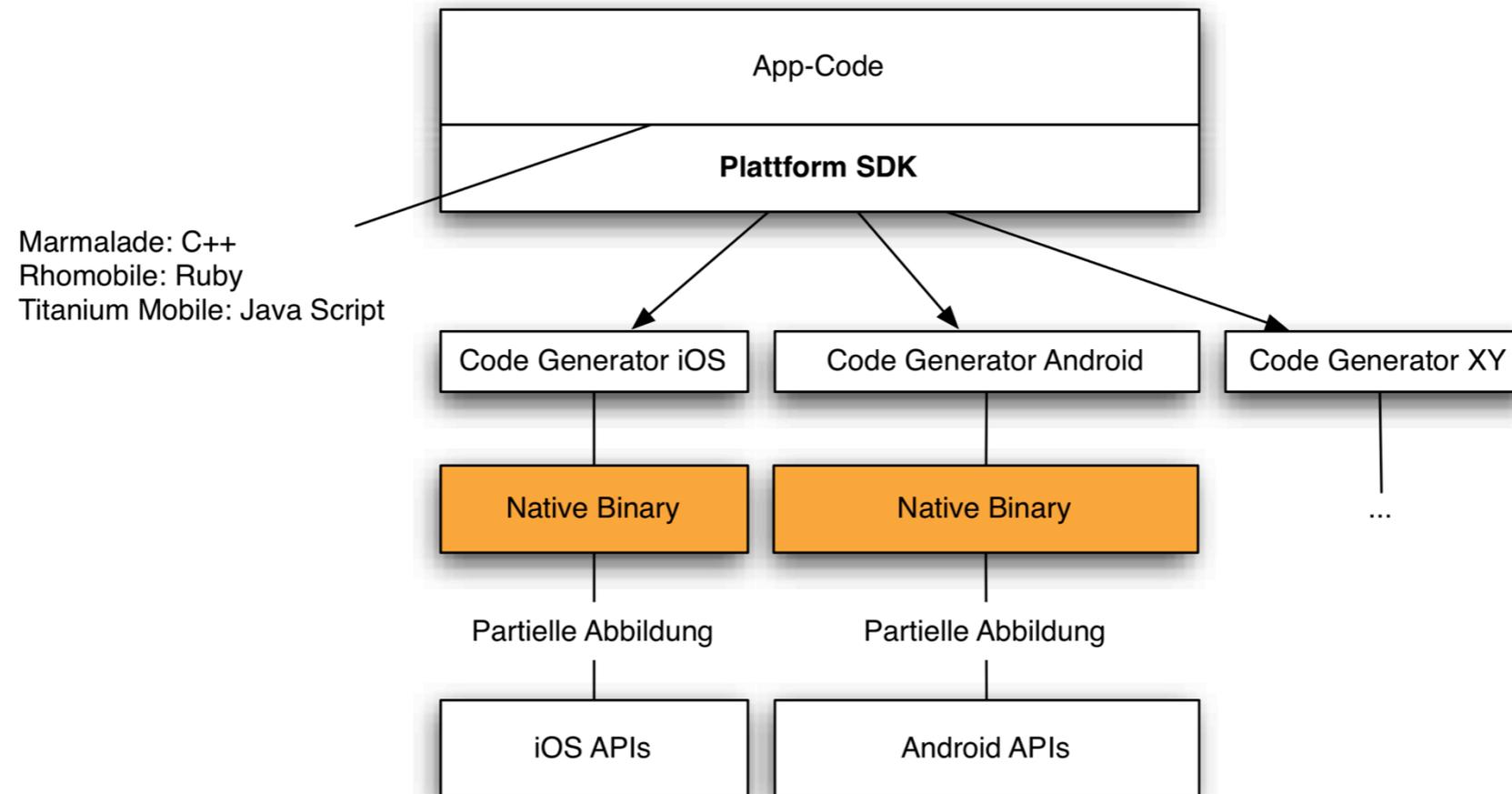
- Der überwiegende Teil einer App wird mit HTML 5 und JavaScript entwickelt
- (Viele) Plattform-Funktionen sind im nativen Teil zu finden
- Es existieren JavaScript-Bindings, um diese Funktionen aufrufen zu können
- 1 SDK zu n Mobile-Plattformen
- Beispiele: PhoneGap, Sencha

# MonoTouch / Mono for Android



- **Vorteile**
  - .NET bekannt
- **Nachteile**
  - Die Entwicklung (vor allem die UI Gestaltung) ist sehr plattformspezifisch
  - Grad Plattformunabhängigkeit ist schwer einzuschätzen
  - Technologieerweiterungen / Anpassungen NUR über den Hersteller
  - Somit Zukunftsfähigkeit des Technologie-Treibers unklar
- **Erwartung**
  - ich lerne EINE Technologie (eventuell kann ich die schon) und schreibe Programme für beide Plattformen
- **Realität**
  - zusätzlich zu den Plattformsprachen lerne ich noch Mono :)

# Cross-Plattform + Code-Generatoren



- **Vorteile**
  - Eine (bekannte) Entwicklungstechnologie
  
- **Nachteile**
  - Der Umfang der 3d-Party SDKs ist beschränkt
  - Individuelle Anpassungen schwer bis unmöglich

# Mobile Webseiten

- Die Apps laufen im Browser, Aufruf über URL
- **Pro**
  - Plattformunabhängigkeit
  - Einheitliches Design auf allen Plattformen
  - Schneller & weniger komplexer Entwicklungsaufwand
- **Contra**
  - Nicht jeder Browser ist kompatibel (Frameworks unterstützen meist nur WebKit Browser)
  - Aktive Internetverbindung wird vorausgesetzt
  - Eingeschränkter Funktionsumfang (kein Zugriff auf Kamera, Kalender, Kontakte, Dateisystem, Vibration,...)

# WebApps / Cordova / Sencha Touch

- **Framework zur Erstellung mobiler Webanwendungen**
  - iOS, Android, BlackBerry, Amazon Kindle
- **Konzepte**
  - Komplexe Komponenten und Layouts
  - objektorientiertes Klassensystem
  - Ajax, Proxies, History, Theming, Profile
  - Modelle, Assoziationen, Stores, Model-View-Bindings
  - Gestenerkennung
  - Eingeschränkter Zugriff auf native Api
  - Erweiterbar durch eigene Plugins (Cordova)
  - Sencha SDK Tools zum Erstellen von Themes und Deployment von Anwendungen (z.B. in den App Store)

# WebApps - Hybride Applikationen

- **Eine WebApp**
  - besteht aus einem nativen Code-Teil (Device-Funktionen) und einem Webbasierten Teil (UI, Logik)
  - wird mit Wrapper-Technik als natives Bundle ausgeliefert
- **Pro**
  - Weitgehende Plattformunabhängigkeit
  - Einheitliches Design auf verfügbaren Plattformen
  - Schneller & weniger komplexer Entwicklungsaufwand
  - Installierbar über den App Store
  - Zugriff auf die Funktionen des Betriebssystem über das native Framework-Teil möglich
- **Contra**
  - Eingeschränkter, unvollständiger bzw. umständlicher Zugriff auf Funktionen des Betriebssystems
  - (Nur) „oberflächliche“ Benutzeroberflächen

	nativ	hybrid	web
Performance	Sehr gut	Gut	Befriedigend
Funktionalität abhängig von	Nativer Plattform	Framework wie PhoneGap, Titanium	Browser (JavaScript/HTML5 Unterstützung)
Hardwarezugriffe und Funktionalitäten	Uneingeschränkt	Eingeschränkt	Stark eingeschränkt
Look-and-Feel	Entspricht der jeweiligen nativen Plattform	Angepasst an maximal eine native Plattform	Unterscheidung von nativen Plattformen
Aufwand für mehrere Plattformen	Erhöhter Aufwand je Plattform	Mehrheitlich unabhängig von Anzahl unterstützter Plattformen	Unabhängig von Anzahl unterstützter Plattformen
Veröffentlichung	Über App-Store	Über App-Store	Auf Web-Server
Bereitstellungsdauer/Updates	Verzögerung um einige Tage	Verzögerung um einige Tage	Aktualisierung auf Web-Server

# Was wir machen?

Auswahl (1)

## **Driver.app - Mobiles Frontend für das Fleetboard-System von Daimler FleetBoard GmbH, 2014 - 2015 (Android)**

- Verwaltung der Restlenkzeiten und Ruhezeiten von Fernkraftfahrern
- Benotung des Fahrstils durch das Fleetboard-System
- Chat-Funktion für die Fahrer untereinander
- Lokalisierung der Fahrzeugdaten und der Fahrzeugposition

# Was wir machen?

## Auswahl (1)

### Driver.app

Michael Fahrtüchtig  
on Tour  
Frankreich; Montrouge

2,5 h Restlenkzeit B. | 5 % Tankfüllstand | 8,9 Fahrweise

13:30 Uhr schaffe ich nicht, die E27/N1019 ist dicht.  
Heute; 11:25 Gerd Schnell

09. Juli 2014; 11:25 Max Fahrtüchtig

Rastplatz Chalon sur Saone hat noch freie Plätze. Duschen und WC auch in Ordnung.  
09. Juli 2014; 11:25 Gerd Schnellfahrer

Rastplatz Chalon sur Saone hat noch 20 weitere neue Ereignisse

TIME MANAGER  
Letzte Aktualisierung vor 00:05:00

Lenk- & Ruhezeiten | Logbuch

Donnerstag, 03. Oktober 2014

Uhrzeit	Zeitart	Dauer
07:00 - 10:00	Lenkzeit	3h
12:00 - 13:00	Ruhezeit	1h
13:00 - 16:30	Lenkzeit	3h 30min
16:30	Tanken	15min

MO DI MI DO FR SA SO  
< 30 01 02 03 04 05 06 >

TIME MANAGER  
Letzte Aktualisierung vor 00:05:00

Lenk- & Ruhezeiten | Logbuch

01:30:00  
Restlenkzeit Block  
in 01:25:00

07:00:00 Restlenkzeit Tag  
13:00:00 Restlenkzeit Woche  
00:45:00 Ruhezeit

MO DI MI DO FR SA SO  
< 30 01 02 03 04 05 06 >

FITNESS COACH

Aktivität | Verlauf | Videos

Rudergang 30 MB  
Sit ups 30 MB

Rudergang 30 MB  
Sit ups 30 MB

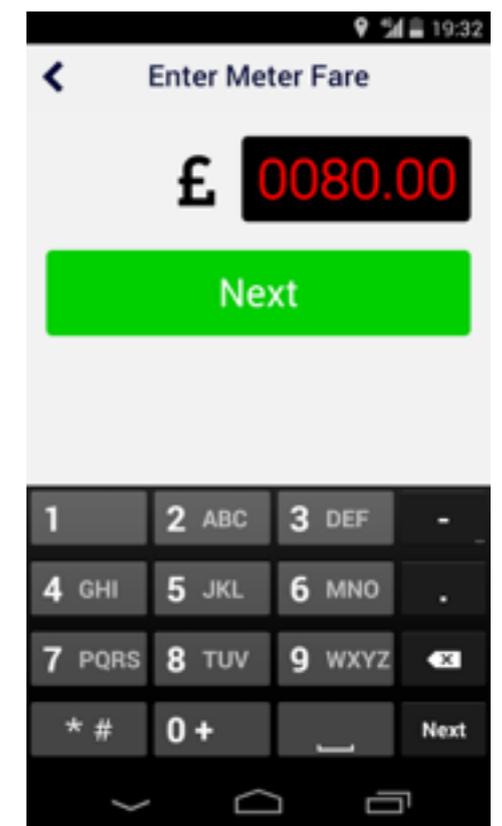
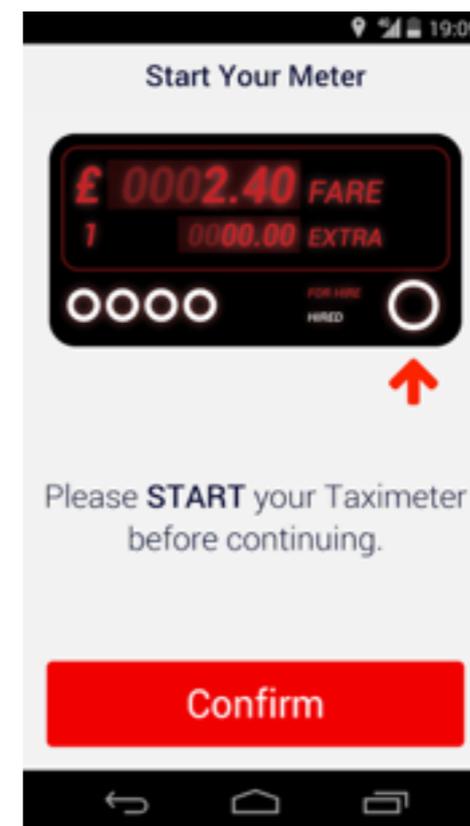
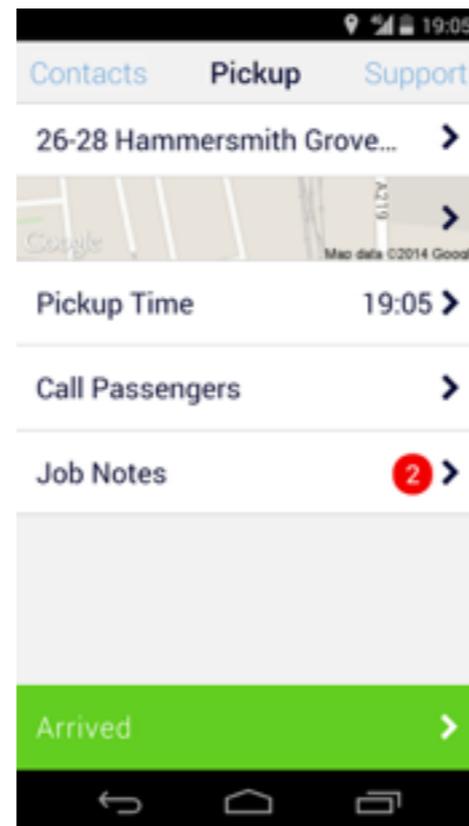
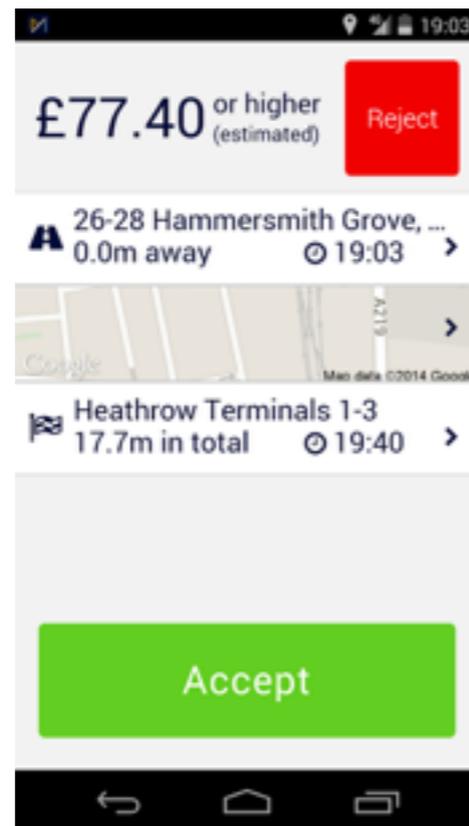
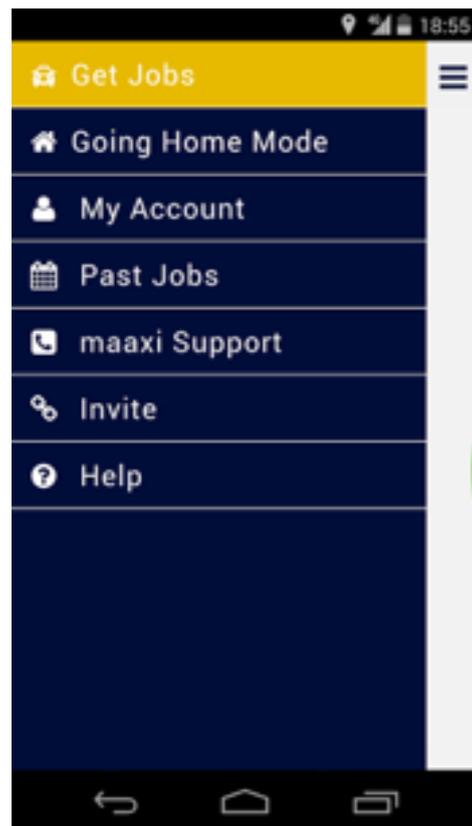
Rudergang 30 MB  
Sit ups 30 MB

# Was wir machen?

## Auswahl (2)

## Maaxitaxi - Taxi App für Londoner black cabs

## Maaxi, 2014 - 2015, (Android, iOS)



- [https://play.google.com/store/apps/details?id=com.maaxiservices.dapp&hl=en\\_GB](https://play.google.com/store/apps/details?id=com.maaxiservices.dapp&hl=en_GB)

# Was wir machen?

## Auswahl (3)

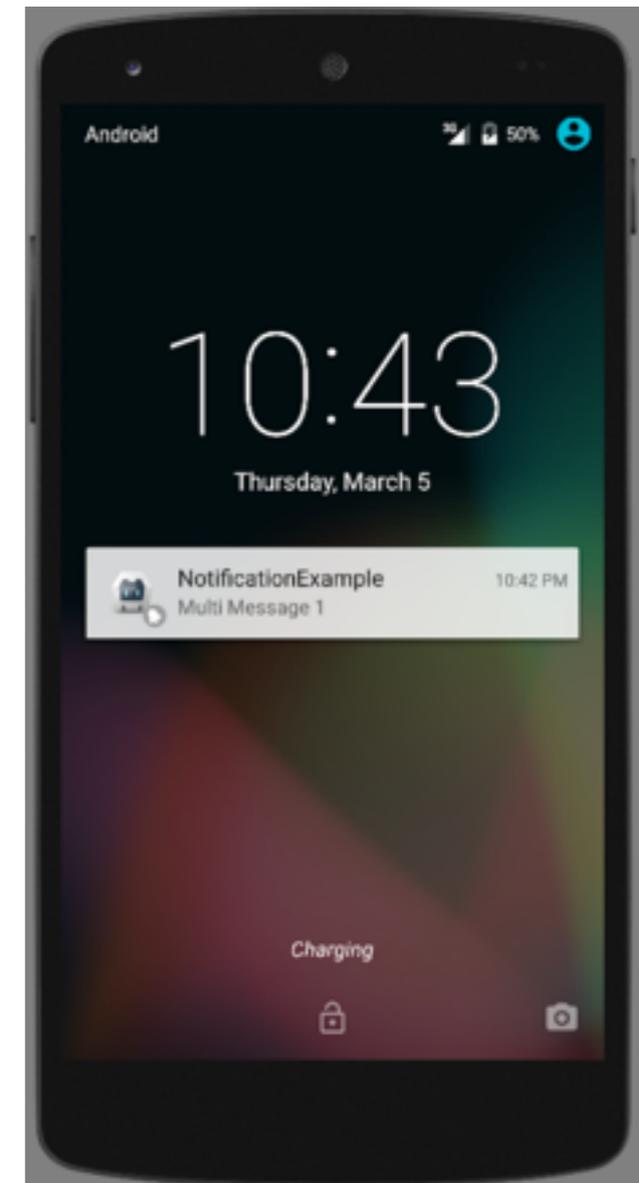
## cordova-plugin-local-notifications

## DAS Plugin für Web-Apps mit Cordova

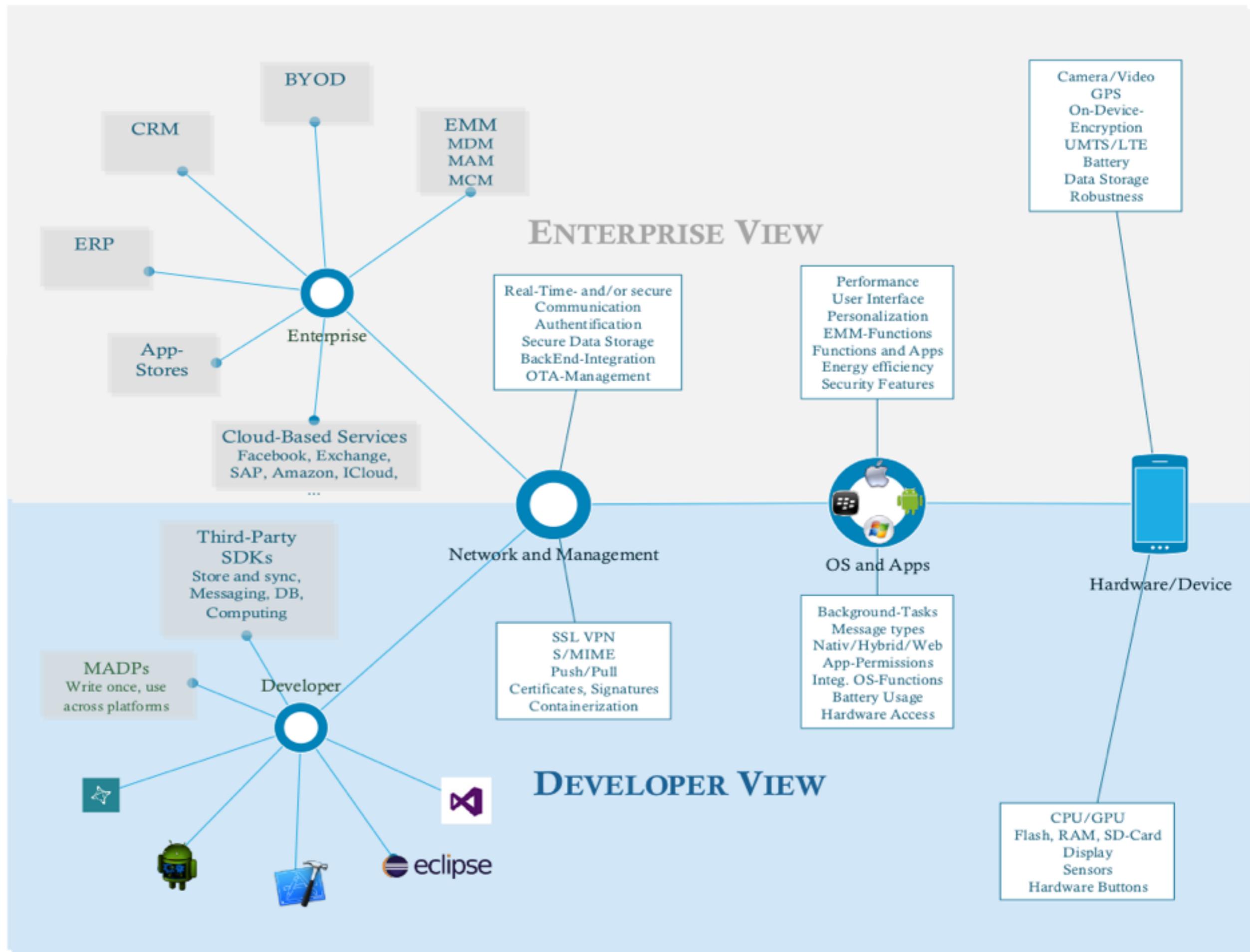
## appPlant, 2013-2016

- Open Source unter

<https://github.com/katzer/cordova-plugin-local-notifications>



# Das Ökosystem einer mobilen Enterprise Applikation



# Umsetzung der Anforderungen in großer Abhängigkeit zum mobilen Betriebssystem

# Wie kommt man zu einer passenden Mobile Enterprise Architektur?

- **Vorentwicklungsphase (Kriterien mobiler BSs)**
  - Sicherheit
  - Mobile Device Management
  - User Interface und Trennung von der Anwendungslogik
- **Entwicklungs- und Einsatzphase (Kriterien der App und ihres Einsatzes)**
  - Sicherheit
  - Benachrichtigungen
  - Mobile Cloud Computing
  - Mobile Device Management
  - User Interface und Trennung von der Anwendungslogik



1	Kriterien	Android 5 (enthält Andoid for Work)	Android 5 + Samsung Knox 2.4	iOS 8	Blackberry 10.3	Windows Phone 8.1	Gewicht (G)	Faktor (α)
31	<b>2. Mobile Device Management und Bring Your Own Device</b>							
32	MDM-Client	ja	ja	ja	ja	ja		1
33	Anzahl MDM-Regeln	Wenige Regeln (ca. 10) meist Sicherheitsfeatures, herstellerspezifische Erweiterungen	durch Samsung Knox erweiterte Features (über 600 Richtlinien, über 1400 APIs)	Große Anzahl an Regeln (über 100), die sich mit diversen Aspekten des Systems befassen	Große Anzahl an Regeln für Work Space, Eingeschränkte Regeln für privaten Bereich	Geringe Anzahl an Regeln; Obermenge der unter Exchange ActiveSync (EAS) verfügbaren Regeln		1
34	Entziehen der MDM-Kontrolle durch den Benutzer	ja, Deinstallation möglich, damit verbundene Einschränkungen sind vom Agenten festgelegt	ja, Deinstallation möglich, Löschen aller Daten des Knox-Containers	ja, Vorherige Spezifikation, ob Konfigurationsprofil entfernt werden darf	nein	ja, jedoch Deinstallation aller Unternehmensapplikationen und Unternehmenskontos		1
35	Integrierte BYOD-Lösung und Funktionsumfang	ja	ja, Erweiterung um zahlreiche APIs	nein	ja, große Anzahl an Regeln, s.o.	nein		1
36	Möglichkeit von Containerization	ja	ja	ja	ja	ja		1
37	OTA-Management (Regeln und Konfiguration, Firmware-Updates)	ja	ja	ja, Konfigurationsprofil, MDM, and Exchange ActiveSync	ja	ja, Updates über MDM oder WP Update Service		1
38	Einschränkung von Apps (Whitelist/Blacklist), Enterprise App Store	ja, im sicheren Bereich/ ja, Verwalten eines privaten Kanals über Admin-Konsole	ja/ja	nein, keine MDM-Regel, Möglichkeit: Liste von Apps abrufen + Restriktion durchführen / ja	ja, Application Restriction Rule IT/ja	ja, über App Allow/Deny-Listenfunktion oder Assigned Access / ja		1
39	Remote Wipe	ja	ja	ja	ja, remote und lokal	ja + Device Retirement (ausschließliches entfernen von Firmendaten) + Device Wipe Threshold		1
40	Remote Locate	ja, via Android Device Manager	ja, via Android Device Manager	ja, über "Find my iPhone" Service	ja	ja, Microsoft Dienst "Mein Handy finden"		1
41	Remote Lock	ja (Lock, Ring, Wipe)	ja (Lock, Ring, Wipe)	ja, über "Find my iPhone" Service	ja, über web service	ja (Lock, Ring, Reset Passwort)		1
42	Passwort und Authentifizierungskontrolle	ja	ja	ja	ja	ja		1
43	Lokale Verschlüsselung (native Unterstützung, über Software)	ja	ja	ja	ja	ja Bitlocker + removable Storage		1
44	Konfiguration der Kommunikation wie VPN	ja	ja	ja	ja	auto-triggered VPN		1
45	Konfiguration der Kommunikation wie WLAN	Liste erlaubter APs	ja	ja	ja	ja, - Wi-Fi Profile, Untersagen Wi-Fi Hotspot, kein Hinzufügen zusätzlicher Profile, Untersagen von Traffic über Wi-Fi-Verbindung		1
46	Backup/Restore von Gerätedaten	ja, Google Chrome Lesezeichen, WLAN-Passwörter, Wörterbücher und Geräteeinstellungen über Google-Zugang	ja, Google Chrome Lesezeichen, WLAN-Passwörter, Wörterbücher und Geräteeinstellungen über Google-Zugang	ja, über iCloud	ja, auto-Backup, BlackBerry über Desktop Software or BlackBerry Web Desktop Manager	keine vollständige Backup-Lösung, Synchronisieren von Kontakten, Nachrichten, Fotos, Systemeinstellungen etc. mit Microsoft-Konto		0,5
47	Backups auf externe Cloud-Services unterbinden	nein, nicht nativ	ja, Unterbinden der Synchronisation mit Google Account	Unterbinden von iCloud- und iTunes-Backups	Regel: Cloud Storage Access from Work Space; Unterbinden von z.B. Box oder Dropbox	nein, über Black- /Whitelisting von entsprechenden Apps möglich oder Assigned Access		1

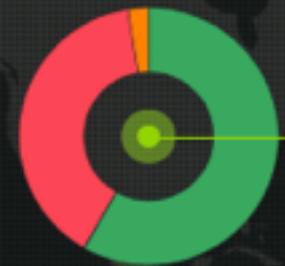
	A	B	C	D	E	F	G	H
1	<b>Kriterien</b>	<b>Android 5 (enthält Android for Work)</b>	<b>Android 5 + Samsung Knox 2.4</b>	<b>iOS 8</b>	<b>Blackberry 10.3</b>	<b>Windows Phone 8.1</b>	<b>Gewicht (G)</b>	<b>Faktor (α)</b>
47	Zustand von Geräten auslesbar	nein, nicht nativ	ja	ja	ja	Remote Inventory, z.B. installierte Apps, Telefonnummer, OS Version, MAC Adresse, ...	1	1
48	Aktivierung per App VPN, Always on	ja / n.a	ja/ ja	n.a/ ja	nicht verfügbar	ja/ ja	1	1
49	Sicheres Löschen von Daten (Daten nicht mehr auf Datenträger)	nein, Verschlüsseln + anschließendes Zurücksetzen möglich	nein, Verschlüsseln + anschließendes Zurücksetzen möglich	nein, jedoch Löschen des Block-Schlüssels = Dateien nicht zu entschlüsseln	ja, Security Wipe, außerdem Daten der Speicherkarte nicht mehr zugänglich	nein, Verschlüsseln + Zurücksetzen möglich	1	1
50	Assigned Access	nein, nicht nativ	ja	n.a	n.a	ja	1	1
51	Zwischenergebnis gemäß (5.5)	13,5 von 19	18 von 19,5	15 von 18	17,5 von 17,5	16,5 von 19,5	19,5	3/7
52		0,71	0,92	0,83	1	0,85		
53	<b>3. Benutzer Interface und Trennung der Anwendungslogik</b>							
54	Möglichkeiten der Plattformen	nativ, hybrid, web	nativ, hybrid, web	nativ, hybrid, web	nativ, hybrid, web	nativ, hybrid, web	1	1
55	Trennung von Layout, Design und Logik möglich	ja	ja	ja (.nib, .xib)	ja	ja	1	1
56	Beschreibungssprache für die UI	ja, deklarative Beschreibungssprache XML, programmatisch	ja, deklarative Beschreibungssprache XML, programmatisch	ja, deklarative Beschreibungssprache XML, programmatisch	ja, QML als vollständige deklarative Programmiersprache, programmatisch	ja, deklarative Beschreibungssprache XAML, programmatisch	1	1
57	UI Entwicklung über Designer (Nur Preview/ Manipulation)	ja	ja	ja	Preview möglich	ja	1	1
58	Möglichkeit Separation-Presentation-Pattern zu implementieren	ja	ja	ja	ja	ja	1	1
59	Sonstige Features zur Separation, Hilfen für Entwickler bei der Umsetzung bestimmter Pattern	Activities, Fragments	Activities, Fragments	Controller-Superklassen, Notifications	Slots, Signals	Commands, Notifications	1	1
60	Zwischenergebnis gemäß (5.5)	5 von 6	5 von 6	5,5 von 6	5 von 6	5,5 von 6	6	1/6
61		0,83	0,83	0,92	0,83	0,92		
62	Gesamtbewertung gemäß (5.6)	0,71	0,86	0,92	0,91	0,86		$\sum_{i=1}^3 \alpha_i = 1$
63	gemäß (5.7)	2,7 - Befriedigend	1,7 - Gut	1,3 - Sehr Gut	1,3 - Sehr Gut	1,7 - Gut		

70 **Quellen**  
71 Always on: <https://msdn.microsoft.com/en-us/library/dn499786.aspx>  
72 Windows Phone 8.1 security overview  
73 Windows Phone 8.1 Mobile Device Management overview  
74 <http://www.cso.com.au/article/574311/mobile-security-ios-vs-android-vs-blackberry-vs-windows-phone/>  
75 <https://www.samsungknox.com/de/products/knox-workspace/technical>

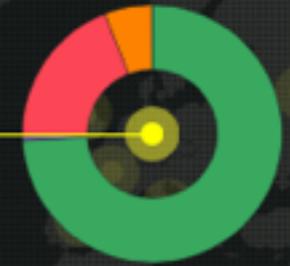
### Smartphone OS sales market share

Read our expert's latest commentary

iOS Growth Slows in China as Buyers Await New Year Deals



USA 3 M/E JAN 2016		
	Android	58.2%
	BlackBerry	0.1%
	iOS	39.1%
	Windows	2.6%
	Other	0.0%



GERMANY 3 M/E JAN 2016		
	Android	74.2%
	BlackBerry	0.6%
	iOS	19.3%
	Windows	5.9%
	Other	0.1%

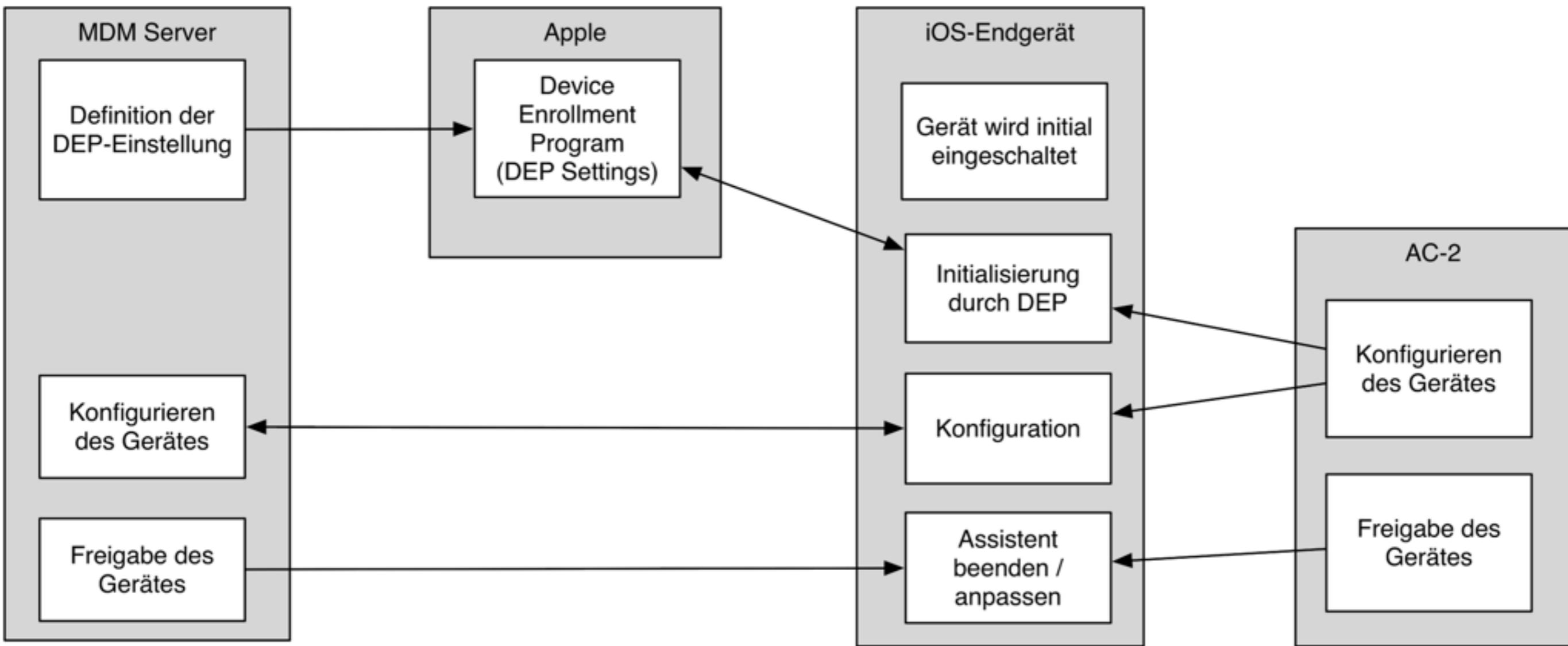
3 M/E JAN 2016



3 M/E JAN 2016

# Mobile Enterprise mit iOS9

- Ziel: Vereinfachung des sicheren Bereitstellen und des Betriebes mobiler Endgeräte im Unternehmen



# 1. Verteilen und Einrichten

- = Bereitstellen und Konfigurieren von Smartphones und Tablets im Büro
- Apple Geräte-Registrierungsprogramm (DEP Device Enrollment Program)
  - Anbindung an das MDM
  - Provisionierung "over the air" automatisch mit nötigen Profilen
    - Unternehmensrichtlinien
    - Konfigurationen
    - zu installierende Apps

# 1. Verteilen und Einrichten

- für Unternehmen unter 50 Geräte
- Auslieferung der Profile mit OS X Server
- Setup via Kabel mit Apple Configurator weiterhin möglich
  
- Einsatz im Umfeld
  - "Shared Use" - generische Aufbereitung und Nutzung der Geräte von mehreren Anwender
  - Multi-User-Support: Mit iOS 9.3 erlaubt Apple erstmals, mehrere Anwender auf einem iPad anzulegen. Diese Funktion steht aktuell aber ausschließlich nur Schulen und Bildungseinrichtungen zur Verfügung.
  - "One-to-one-Use" individuell an einen Mitarbeiter angepasste Geräte

## 2. Geräte konfigurieren

### Funktionen im "Supervised" Modus

Werkstatt für mobile Software

- AirDrop-Sperre - keine Daten aus gemanagten Apps übermitteln
- automatisch Apps installieren
- iCloud Photo Library deaktivieren
- Gerätekenwort erzwingen
- Apple Watch mit einem iPhone unterbinden
- Screen-Aufzeichnung unterbinden
- Konfigurationssperre (Name, Hintergrund, Passwort, Zurücksetzen des Gerätes etc.)
- Mobilfunkverbindung (mobile Daten, Freigabe/Sperren Roaming) einschränken
- AppStore sperren, Installation der Updates durch MDM oder AC-2
- Automatische stille Updates

## 3. Software auf Firmen-Geräten

- Früher: Personengebundener Account pro Gerät (wie Apple ID)
- Mit iOS 9 können Unternehmen die Apps auf Geräte lizenzieren, keine individuellen Apple IDs nötig

## 4. Optimieren der Netzwerkkommunikation

- Schlechtes WiFi, so wird parallel per Reliable Network Fallback eine Mobilfunk-Verbindung geprüft

# 4. Verteilen von Programmen und Lizenzen

## 1. VPP (Volume Purchase Program)

- Management von international gemischten Apps

## 2. Bereits installierte Anwendungen in den Bereich der Geschäftsanwendungen überführen

- ABER ein paralleler Einsatz von Apps im geschäftlichen und im privaten Kontext NICHT möglich

## 3. Vorgaben für sichere Kommunikation von Apps

- NE-HotSpotHelper APIs ermöglichen das Erstellen von Captive Network Apps - befindet sich das Smartphone in der Umgebung eines bekannten Hotspots, kann die App automatisch anmelden
- Entwickler-Info: die API soll im Xcode registriert werden (Entitlement)

# 4. Verteilen von Programmen und Lizenzen

## 4. Sicheres Verbinden via VPN

- Verbindungen für einzelne Programme möglich via Konfigurationsprofile des MDMd
- NETunnelProvider API oder NEAppProxyProvider API

## 5. Filtern des Zugriffes auf Internet-Seiten

- Früher: One-Site-Content oder aufwändige Global-Proxy
- dynamische Filter-Regeln auf dem Endgerät für WebKit-Datenverkehr und auch für Socker-Ebene!

# Was wir machen?

Auswahl (2)



app  
Plant

Werkstatt für mobile Software

**maintcall - mobile Alarmierungslösung**

**BMW Group, 2013 - 2014 (WindowsPhone, iOS)**



Information



- Entwicklung einer mobilen Alarmierungslösung zur Störmeldebenebenachrichtigung der Instandhaltung in der Fahrzeugmontage für BMW (Leipzig, Oxford)

# Was wir machen?

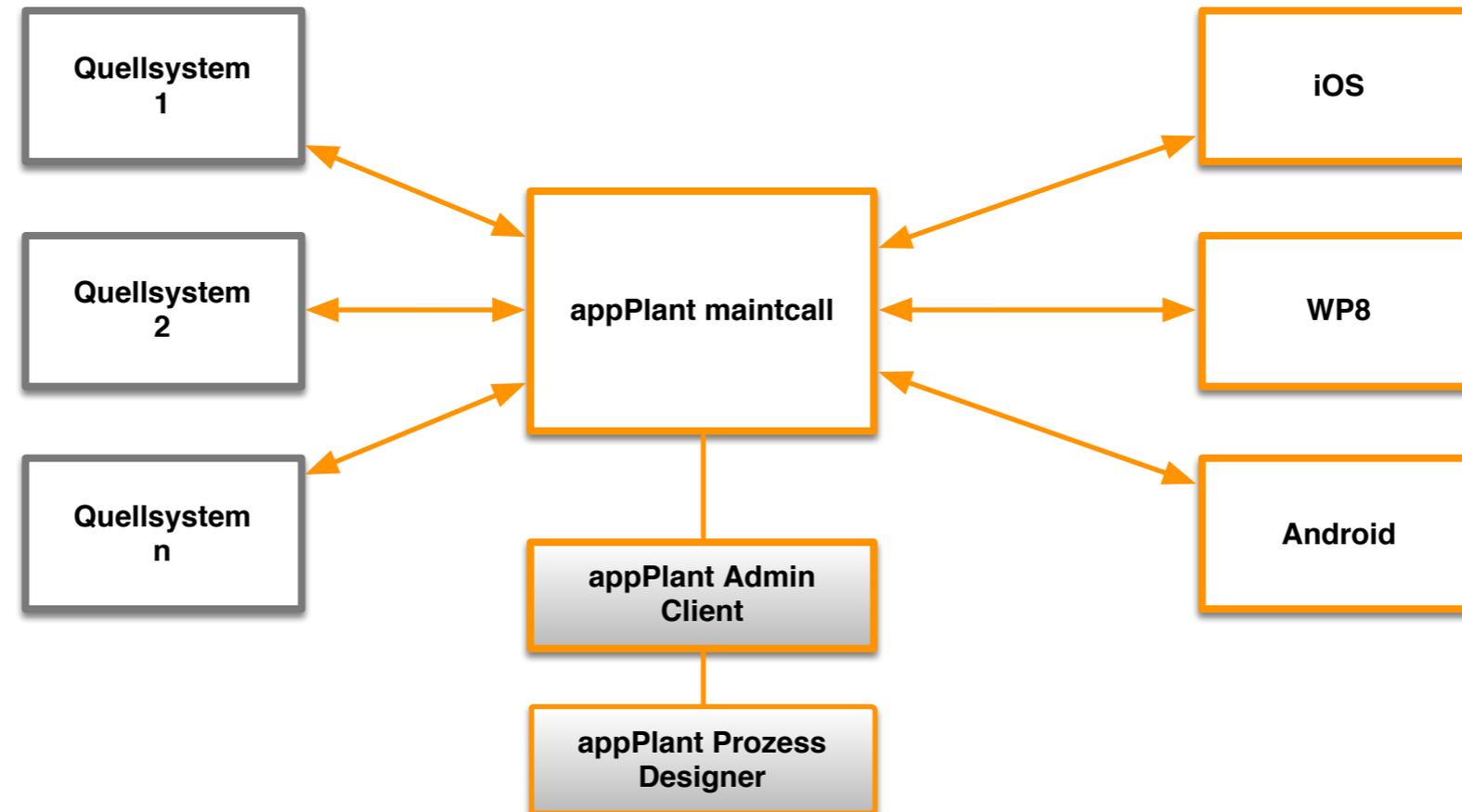
## Auswahl (2)



Werkstatt für mobile Software

## maintcall - mobile Alarmierungslösung

## BMW Group, 2013 - 2014 (WindowsPhone, iOS)



# Was wir machen?

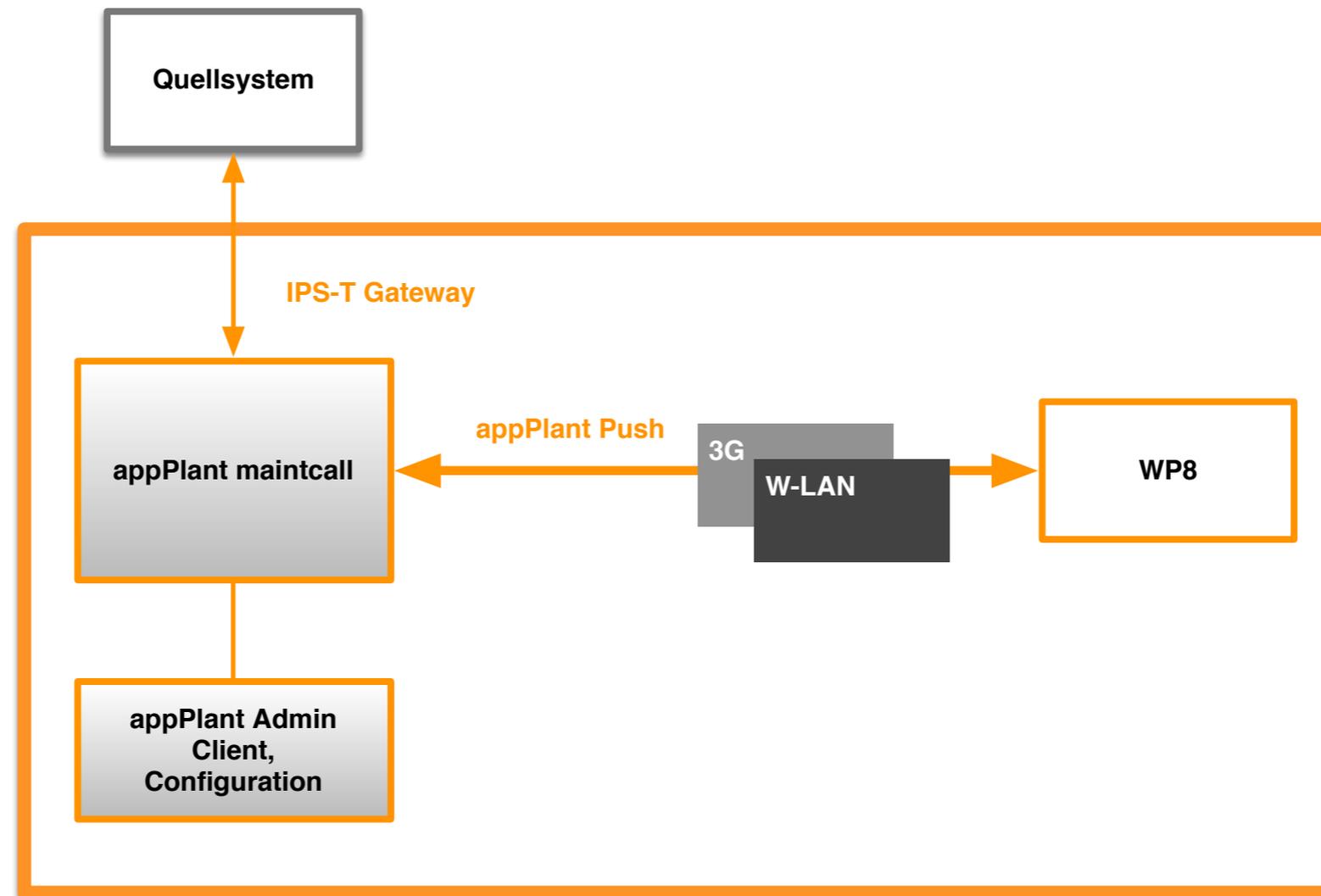
Auswahl (2)



Werkstatt für mobile Software

**maintcall - mobile Alarmierungs-lösung**

**BMW Group, 2013 - 2014 (WindowsPhone, iOS)**



# Was wir machen?

## Auswahl (2)



Werkstatt für mobile Software

## maintcall (2)

- Clients für mehrere mobile Plattformen verfügbar (iOs, WP8, Android)
- Push-Protokoll ohne Nutzung der Cloud-Services
- Flexibilität im Design/Konfiguration der Eskalationsprozesse
- Übertragung der Nachrichten über verschiedene Kanäle (3G, W-LAN)
- Unterstützung von Text-, Bild- und Ton-Nachrichten
- keine SMS-Kosten
- stabile Fallback-Logik für fragile Netzwerkkumgebungen
- Umschalt-Logik zwischen 3G und W-LAN

# Was wir machen?

## Auswahl (2)



app  
Plant

Werkstatt für mobile Software

## maintcall (3)

