

# Entwicklung von Hybrid-Apps am Beispiel von Apache Cordova

Patrick Fehling

Hochschule für Technik und Wirtschaft Berlin

15. Januar 2018

- 1 Allgemeines
- 2 Installation & Build
- 3 Demo
- 4 Plugins
- 5 Nativ vs. Hybrid vs. Web

# Allgemeine Daten

- mobile application development framework
- Anfängliche Entwicklung durch Nitobi
- 2011 Übernahme durch Adobe
  - „PhoneGap“
  - „Apache Cordova“
- Basis für ähnliche Projekte z.B. Ionic, Monaca
- Unterstützung durch: Adobe, IBM, Intel, Microsoft
- Ziel-OSs: iOS, Android, Blackberry etc.



Pacifica

Clever Baby



# Installation - Cordova

Vorraussetzungen:

- Node.js
- (optional) Entwicklungsumgebungen für Plattformen

```
$ npm install -g cordova
$ cordova create MyApp; cd MyApp
$ cordova platform add <platform>
$ cordova run <platform>
#platform: browser, android, ios...

$ cordova requirements
```

# Installation - Plattformen

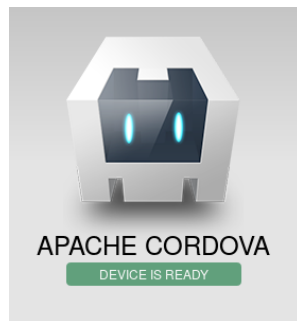
## Android:

- Java SDK (JAVA\_HOME setzen)
- Android SDK (ANDROID\_HOME setzen)
- gradle (z.B. über Android Studio)

installieren und der  
„PATH“-Umgebungsvariable hinzufügen

## iOS:

- auf OSX: [iOS Platform Guide](#)
- auf Windows, Linux: ☹ oder Adobe PhoneGap Build



# Demo

<https://github.com/pa-ling/apache-cordova>

# Plugins

# Hinzufügen

```
$ cordova plugin add \  
    <plugin>
```

# Liste

```
$ cordova plugin ls
```

→ Überschreiben von  
Standardfunktionalitäten

z.B. alert()

→ Erreichbar über globale  
Objekte

z.B. device, navigator

- Kamera
- GPS
- Vibration
- Dialoge
- Akkuladestand
- Bewegung und Ausrichtung  
des Geräts
- Dateiverwaltung
- Push-Notifications
- ...

# Bluetooth

```
1 ble.scan([], 10, function(device) {
2   if ("BLEEnv" === device.name) {
3     ble.connect(device.id, function() {
4       ble.startNotification(
5         device.id,
6         SERVICE_UUID,
7         CHARACTERISTIC_UUID,
8         function(arrayBuffer) {
9           var data = toString(arrayBuffer);
10          console.log("Received:" + data);
11        });
12    });
13  }
14 });
```



|                       | Native App | Hybrid App  | Web App |
|-----------------------|------------|-------------|---------|
| Programmiersprache:   | nativ      | nativ / web | web     |
| Portabilität:         | nein       | ja          | ja      |
| Gerätezugriff (APIs): | Hoch       | Hoch/Mittel | Niedrig |
| Wissensverfügbarkeit: | Niedrig    | Hoch        | Hoch    |
| Leistung:             | Hoch       | Mittel      | Mittel  |
| Updates:              | Langsam    | Mittel      | Schnell |

| Native App                             | Hybrid App                        | Web App               |
|--|-----------------------------------|-----------------------|
| Natives Know-How                       | Web Know-How                      | Web Know-How          |
| Nur eine Plattform                     | Kompromiss zwischen Nativ und Web | Schnelle Auslieferung |
| Viele/aktuelle native Funktionalitäten |                                   | Schnelle Updates      |
| Hohe Performance                       |                                   | mehrere Plattformen   |

Vielen Dank für Ihre Aufmerksamkeit!

Fragen?

# Quellen

- <https://cordova.apache.org/>
- <https://ccoenraets.github.io/cordova-tutorial/index.html>
- <https://www.thinkpacific.com/>
- <https://www.mycleverbaby.com/features.html>
- [https://en.wikipedia.org/wiki/Apache\\_Cordova](https://en.wikipedia.org/wiki/Apache_Cordova)
- <https://developer.ibm.com/dwblog/2017/what-is-apache-cordova-cross-platform-mobile-application-development/>
- <ftp://public.dhe.ibm.com/software/pdf/mobile-enterprise/WSW14182USEN.pdf>
- <https://www.npmjs.com/package/cordova-plugin-ble-central>