

**Macoun**



# iBeacon, HealthKit & Co.

Matthias Krauß · Tim Becker · Press Any Key UG





# Ablauf

- Motivation
- BLE I: GAP
- iBeacon
- BLE II : GATT
- Healthkit
- Ausblick: HomeKit, ApplePay

# Motivation



# Motivation

- Apple öffnet sich!
- iOS wird zu universellen Fernbedienung fürs Leben
- “Digital Hub”



iBeacon HomeKit HealthKit  
ApplePay HandOff

**BLE I: GAP**



# BLE: Was ist daran so toll?

1 Becher Joghurt = 727GB!

Der große Bauer  
Heidelbeere-Cassis

- Sehr geringer Energiebedarf ( $\sim 0.153 \mu\text{J}/\text{bit}$ )
- Günstig in der Herstellung
- Verhältnismäßig einfach (für ein drahtloses Übertragungsprotokoll)
- Breite Unterstützung
- Kein MFI

# Protokollschichten

GAP

GATT

SMP

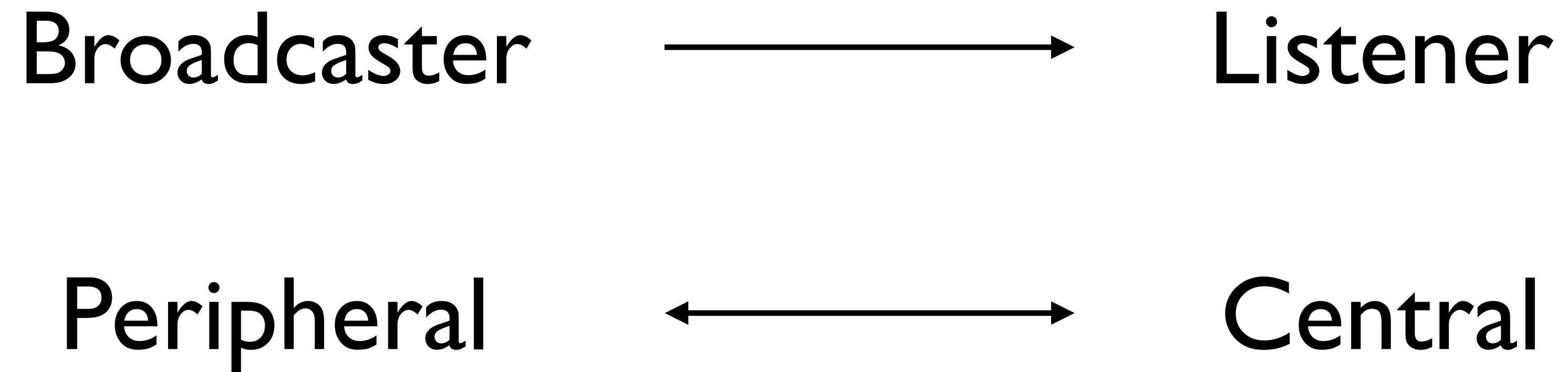
ATT

L2CAP

Link Layer

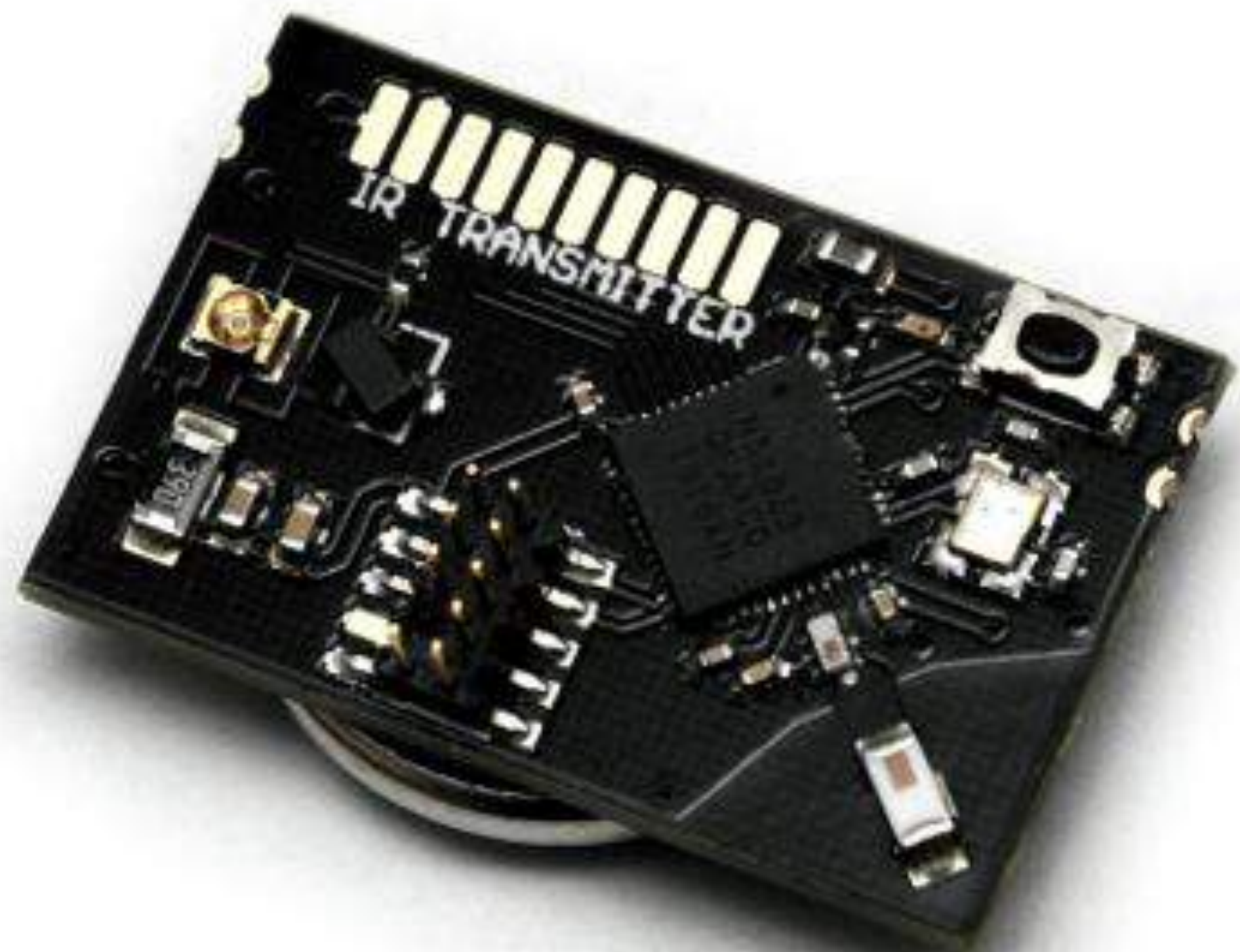
BLE PHY

# GAP Rollen





# iBeacon



# iBeacon Advertisement

02	01	04	1a	ff	4c	00	02
15	16	a2	23	0e	0b	8b	4d
3d	bf	83	36	da	12	00	19
90	00	00	00	04	ba		

02										Field length: 2 (1 byte type + 1 byte payload)
01										Advertising data type: Flags
04										Flag: BR/EDR not supported (BLE only)
1a										Field length: 26 bytes
ff										Advertising data type: Manufacturer-specific data
4c	00									Manufacturer ID: 0x004c = Apple
02										Apple payload ID: 2 = iBeacon
15										Apple payload length: 21 bytes
16	a2	23	0e	0b	8b	4d	3d			16 byte
bf	83	36	da	12	00	19	90			iBeacon UUID
00	00									Major 0
00	04									Minor 4
ba										TX level: RSSI -70dB at 1m



# 0. Import

```
//iOS 7+, OSX 10.9+ :  
CoreLocation.framework  
  
#import <CoreLocation/CoreLocation.h>
```

# I. Location Manager Erzeugen

```
@property (strong) CLLocationManager* locationManager;

self.locationManager = [[CLLocationManager alloc] init];
if (!self.locationManager) {
    if (![CLLocationManager authorizationStatus]) {
        NSLog(@"Location Services not allowed");
    } else {
        NSLog(@"Location Services init failed");
    }
    return;
}
self.locationManager.delegate = self;
```

## 2. Beacon Region Erzeugen

```
@property (strong) NSUUID* beaconUUID;  
@property (strong) CLBeaconRegion* region;  
  
NSString* uuidString = @"16A2230E-0B8B-4D3D-BF83-36DA12001990";  
self.beaconUUID = [[NSUUID alloc] initWithUUIDString:uuidString];  
self.region = [[CLBeaconRegion alloc]  
               initWithProximityUUID:self.beaconUUID  
               identifier:uuidString];  
  
[self.locationManager startRangingBeaconsInRegion:self.region];
```



# 3. An Beacon Ranging erfreuen

```
- (void) locationManager: (CLLocationManager*) manager
    didRangeBeacons: (NSArray*) beacons
    inRegion: (CLBeaconRegion*) region {

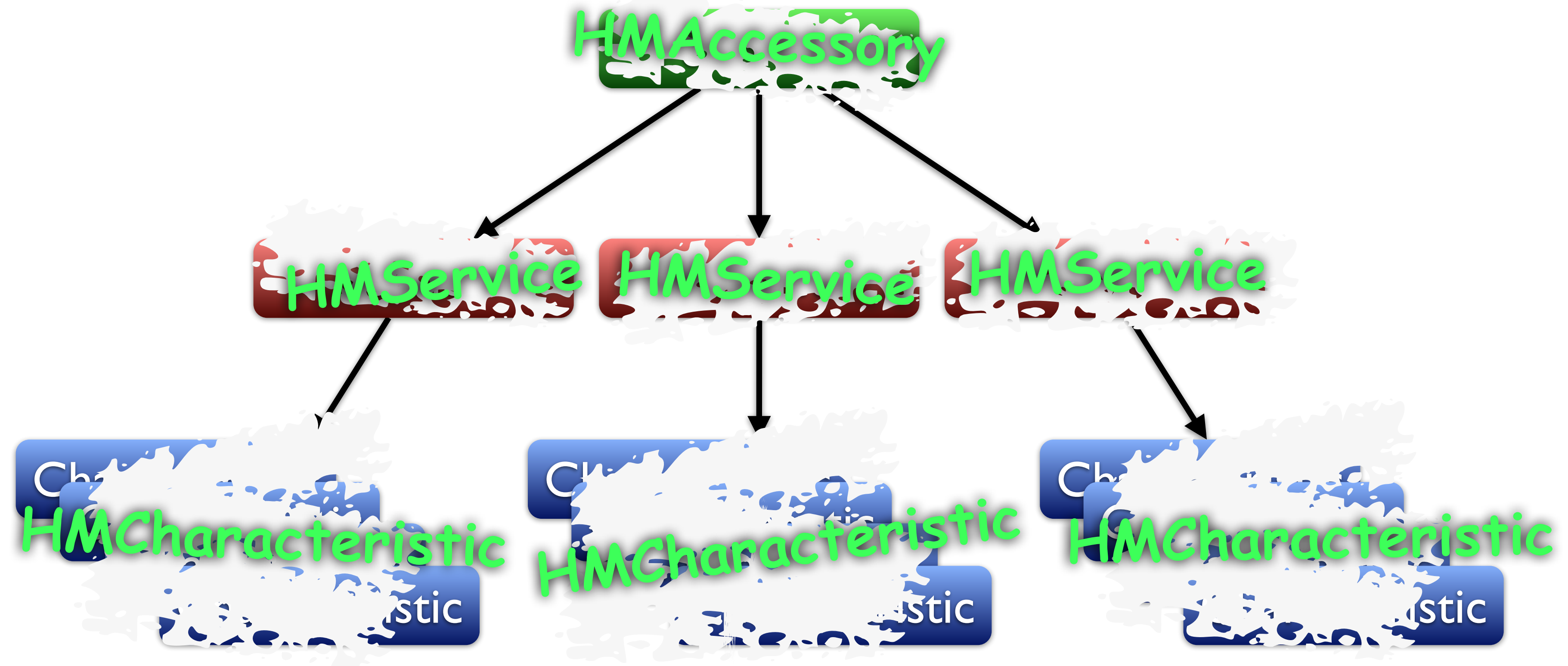
    if ([beacons count] > 0) {
        CLBeacon* strongestBeacon = [beacons objectAtIndex:0];
        NSLog(@"Found major: %i minor: %i prox: %i rssi: %i",
                strongestBeacon.major.intValue,
                strongestBeacon.minor.intValue,
                strongestBeacon.proximity,
                (int)(strongestBeacon.rssi));
    } else {
        NSLog(@"No beacon in range");
    }
}
```

DEMO

**BLE II : GATT**



# HomeKit



DEMO

# HealthKit

# HealthKit Features

- Repräsentationen für körperspezifische Daten
- Gemeinsame Datenbasis
- Datenzugriff
- Datenschutz
- Konvertierung, Lokalisierung

# HealthKit Datenschutz

[...]

27.3 Apps using the HealthKit framework that store users' health information in iCloud will be rejected

27.4 Apps may not use user data gathered from the HealthKit API for advertising or other use-based data mining purposes other than improving health, medical, and fitness management, or for the purpose of medical research

27.5 Apps that share user data acquired via the HealthKit API with third parties without user consent will be rejected

[...]

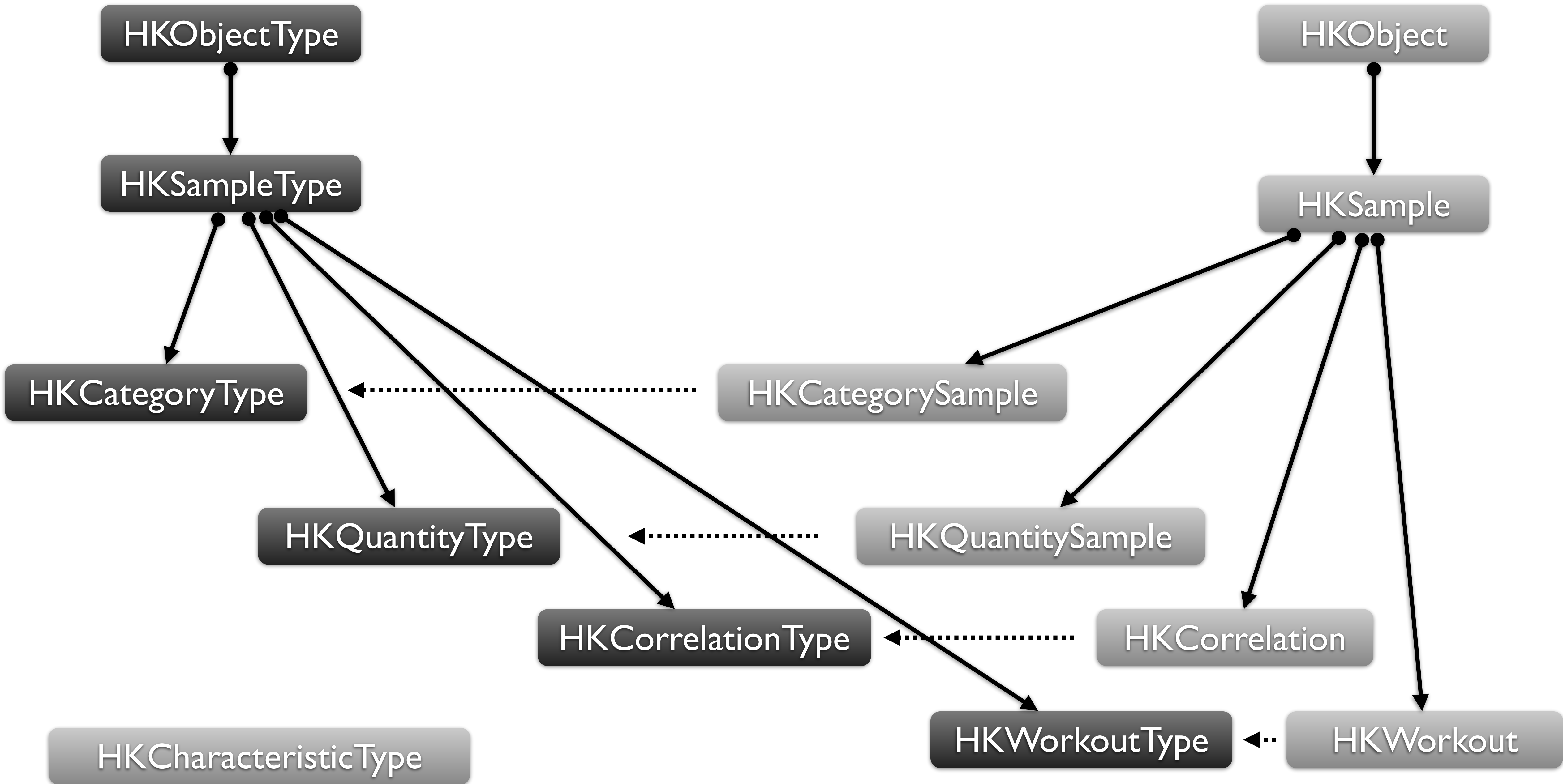
<https://developer.apple.com/app-store/review/guidelines/#healthkit>

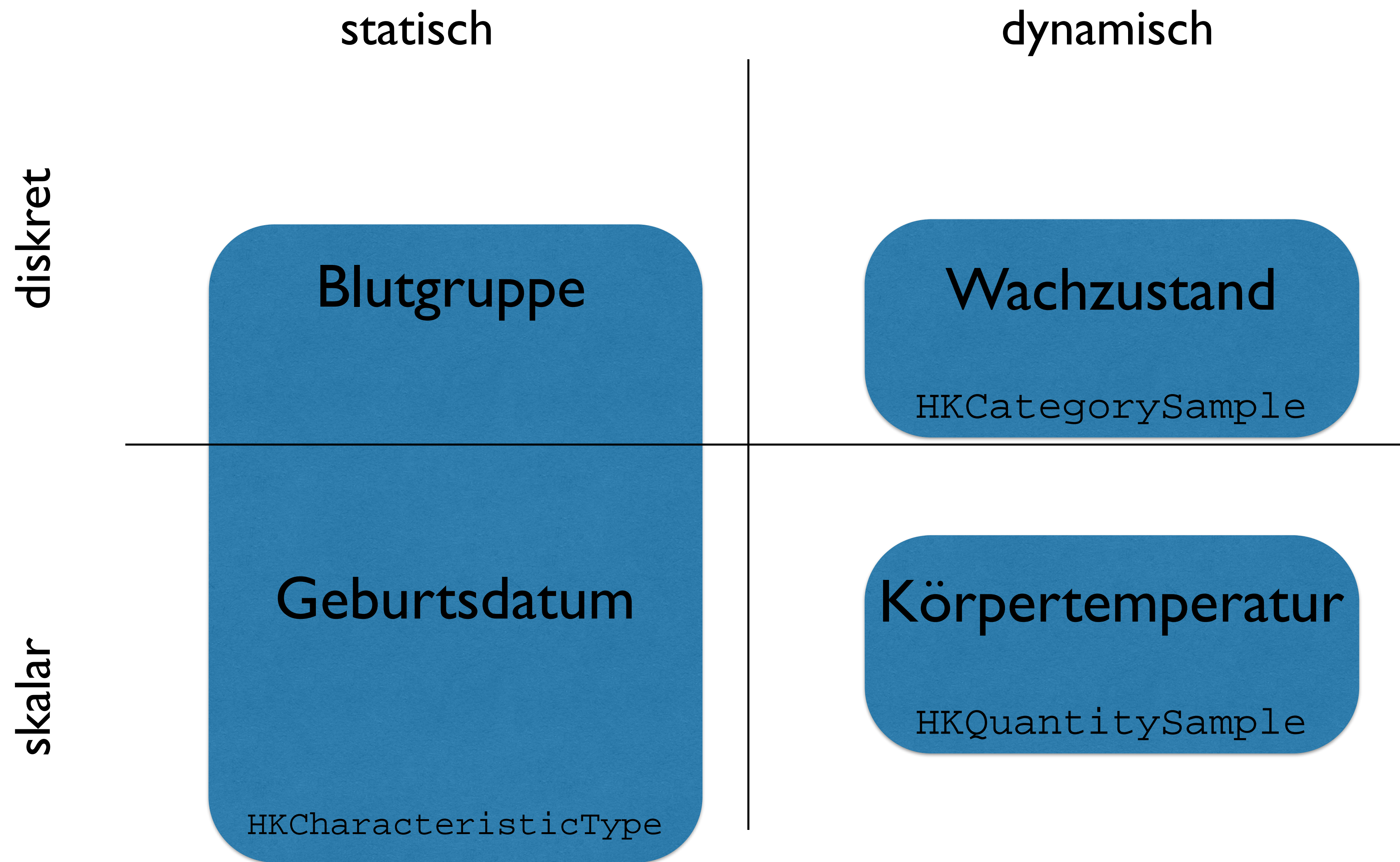


HKHealthStore

HKQuery

HKObject





und `HKCorrelation` und `HKWorkout`

DEMO

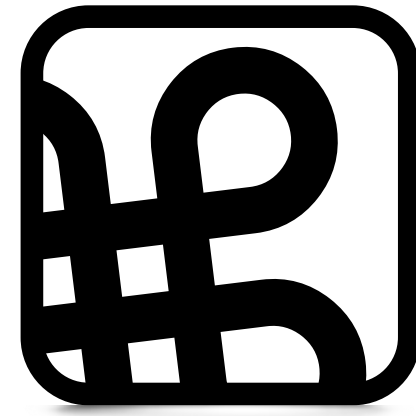
# Fragen?

matthias@press-any-key.de  
tim@press-any-key.de

@presseverykey



**Vielen Dank**



**Macoun**