

Praktikum Entwicklung von Mediensystemen mit iOS

SS 2011

Michael Rohs

michael.rohs@ifi.lmu.de

MHCI Lab, LMU München

Timeline

Date	Topic/Activity
5.5.2011	Introduction and Overview of the iOS Platform
12.5.2011	Implementing a User Interface
19.5.2011	App Architecture, Touch Input, Saving Data
26.5.2011	Location, Sensors; Brainstorming
2.6.2011	no class (Christi Himmelfahrt)
9.6.2011	Networking, Camera API; Milestones
16.6.2011	
23.6.2011	no class (Fronleichnam)
30.6.2011	
07.07.2011	
14.07.2011	
21.07.2011	Project Presentation
28.07.2011	Evaluation
	Paper Writing

Today

- Location
- Sensors
- Exercise 3
- Brainstorming

Location

- Location manager provides location information

```
CLLocationManager *locationManager;
```

- Configuration

```
locationManager = [[CLLocationManager alloc] init];
```

```
locationManager.delegate = self;
```

```
locationManager.desiredAccuracy = [[setupInfo  
objectForKey:@"SetupInfoKeyAccuracy"] doubleValue];
```

```
[locationManager startUpdatingLocation];
```

- Update location

```
(void)locationManager:(CLLocationManager *)manager  
didUpdateToLocation:(CLLocation *)newLocation fromLocation:  
(CLLocation *)oldLocation { ... }
```

- Details: LocateMe example application

Latitude and Longitude

- Location manager state
manager.`locationServicesEnabled`
- Location manager `desiredAccuracy` constants
 - `kCLLocationAccuracyBestForNavigation`
 - `kCLLocationAccuracyBest`
 - `kCLLocationAccuracyNearestTenMeters`
 - `kCLLocationAccuracyHundredMeters`
 - `kCLLocationAccuracyKilometer`
 - `kCLLocationAccuracyThreeKilometers`
- `CLLocation`
- `CLLocationDegrees`
- `CLLocationDistance` (from location ... to location)

Accelerometer (since iOS 2.0)

- UIAccelerometer

```
UIAccelerometer *acc;
```

- Configure accelerometer

```
acc = [UIAccelerometer sharedAccelerometer];
```

```
acc.updateInterval = 1.0; // sec
```

```
acc.delegate = self;
```

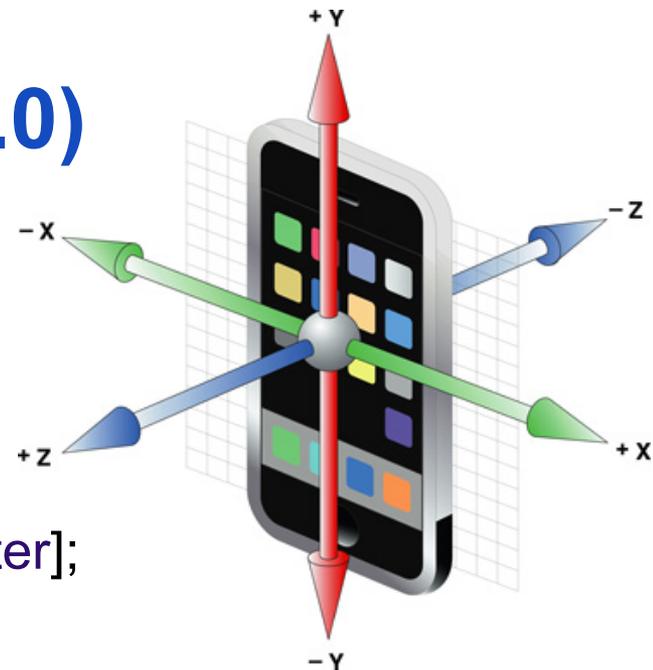
- Receive updates

```
- (void)accelerometer:(UIAccelerometer *)a  
    didAccelerate:(UIAcceleration *)acceleration
```

```
{
```

```
    NSLog(@"%.3f, %.3f, %.3f, %f",  
          acceleration.x, acceleration.y,  
          acceleration.z, acceleration.timestamp);
```

```
}
```



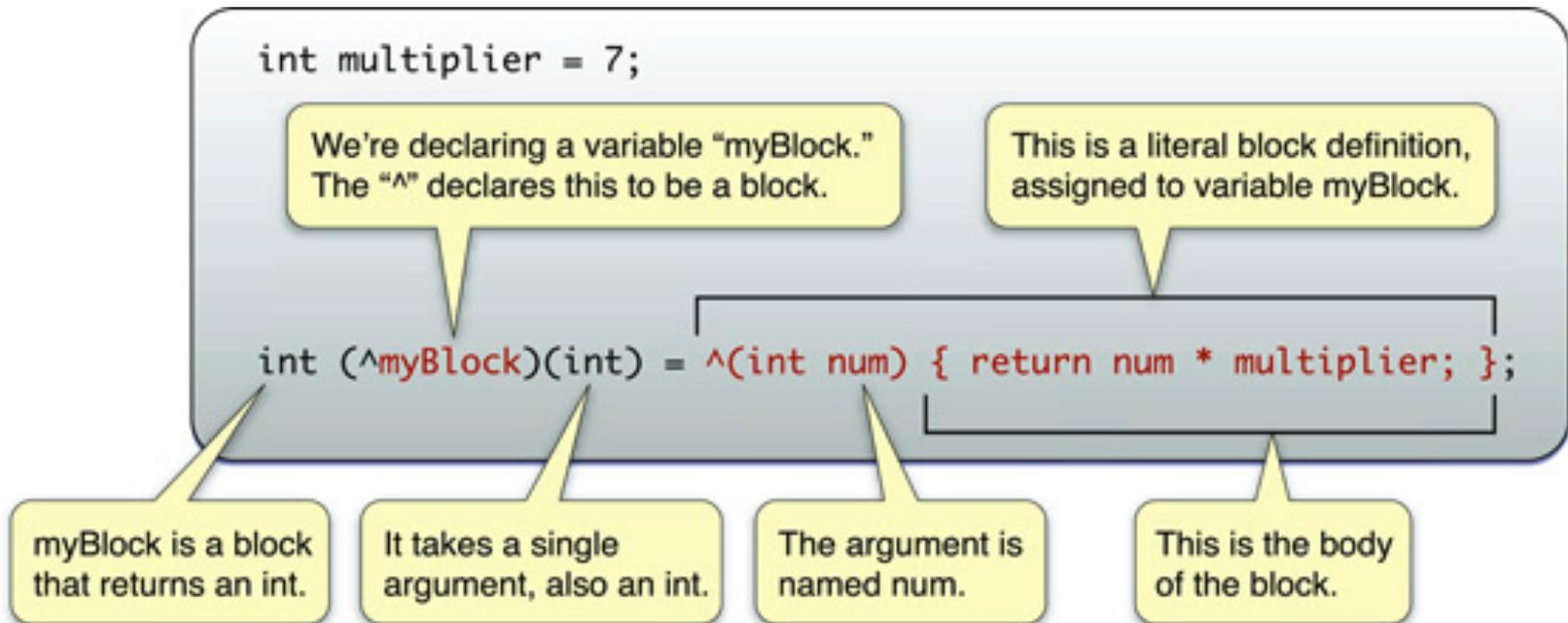
Source: iOS Documentation, Apple

Motion Sensors (since iOS 4.0)

- Core Motion Framework
- CMMotionManager provides access to
 - raw accelerometer data (accelerometer)
 - raw rotation-rate data (gyroscope)
 - processed device-motion data (fused sensor data)
- Receiving sensor data
 - Specific intervals
 - accelerometerUpdateInterval
 - startAccelerometerUpdatesToQueue:withHandler:
 - On demand
 - startAccelerometerUpdates
 - accelerometerData

Code Blocks

- Blocks: conceptually similar to function pointers in C
 - Can access variables from function in which they are defined
- Passing a code block as argument



Source: iOS Documentation, Apple

- Calling: [someObject someFunction:myBlock];

Getting Accelerometer Values

- Create and configure motion manager

```
motionManager = [[CMMotionManager alloc] init];  
motionManager.accelerometerUpdateInterval = 1.0; // sec
```

- Register handler and start updates

```
void (^accHandler)(CMAccelerometerData*, NSError*) =  
    ^(CMAccelerometerData *accData, NSError *error)  
    {  
        NSLog(@"%f, %f, %f", accData.acceleration.x,  
            accData.acceleration.y, accData.acceleration.z);  
    };
```

```
[motionManager startAccelerometerUpdatesToQueue:  
    [NSOperationQueue mainQueue] withHandler:accHandler];
```

- Stop updates

```
[motionManager stopAccelerometerUpdates];
```

should not use main queue

Getting Accelerometer Values

- Create new queue

```
queue = [[NSOperationQueue alloc] init];
```

- Register handler and start updates

```
void (^accHandler)(CMAccelerometerData*, NSError*) =  
    ^(CMAccelerometerData *accData, NSError *error)  
    {  
        NSLog(@"%f, %f, %f", accData.acceleration.x,  
            accData.acceleration.y, accData.acceleration.z);  
    };
```

```
[motionManager startAccelerometerUpdatesToQueue:queue  
                withHandler:accHandler];
```

- Need to use main queue to update UI

Update UI on Main Queue

```
void (^accHandler)(CMAccelerometerData*, NSError*) =  
  
^(CMAccelerometerData *accData, NSError *error)  
{  
    dispatch_async(dispatch_get_main_queue(), ^{  
        NSString *s;  
        s = [NSString stringWithFormat:@"%%.3f", accData.acceleration.x];  
        labelX.text = s;  
        s = [NSString stringWithFormat:@"%%.3f", accData.acceleration.y];  
        labelY.text = s;  
        s = [NSString stringWithFormat:@"%%.3f", accData.acceleration.z];  
        labelZ.text = s;  
    });  
};
```

Operation Queues

- NSOperationQueue objects store and execute NSOperation objects
- Operation queues
 - Store and execute operation objects
 - Provide their own threads to execute operations
 - Encapsulate threads
- Operations
 - Abstract base classes for tasks to perform
 - Have priorities
 - One operation may depend on completion of other operations

Exercise 3

Project Ideas

- Strengths of mobile devices?
- Usage situations?
- Target users?
- Application domains?
- Technological possibilities on the device (sensors...)?
- Combination with external hardware (Arduino...)?

Brainwriting

- Repeat 5 times
 - 3 minutes: On paper, fill one row with 3 ideas
 - Pass on paper clockwise
 - Read other ideas, fill next line with 3 more ideas
- Select the 3 best ideas
 - 10 minutes
 - Present selected ideas

User Profiles or “Personas” (Cooper, 1998)

- Short profiles of typical users
 - Prototypical user for a specific user group
 - Fictitious individual with concrete characteristics
- Building personas
 - Often built from interview results
 - Synthesize fictitious users from real user characteristics
 - Develop multiple personas for different types of users
- Bring them to life
 - With a name, characteristics, experience, personal background, environment they are located in, goals, tasks, skill levels, etc.
- Base design decisions on the needs of the personas

Personas Example

(Cooper, About Face, Chapter 5)

Building a car that pleases everyone

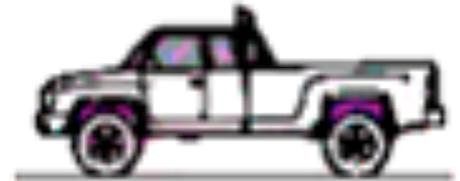


Building a car based on three personas (representing larger groups)



Marge, *mother of three*

Marge wants safety and room for many passengers. A minivan meets her needs.



Jim, *construction worker*

Jim wants cargo space and the ability to carry heavy load. A pickup truck meets his needs.



Alesandro, *software engineer*

Alesandro wants sporty looks and speed. A two-door sports car meets his needs.

Personas Help to...

- Communicate user characteristics to developers
 - Consistent, coherent understanding of target group
 - Features prioritized by how well they match personas' needs
 - Evoke empathy for the target users represented by personas
- Avoid elastic users
 - Constantly changing target user to justify differing design choices
 - Clear focus towards the intended users
- Avoid self referential design
 - Developers / designers designing for themselves
- Avoid designing for “edge cases”
 - Should not be central to design (prioritize!)

Example Persona: Bob



Bob is **52 years old** and works as a **mechanic** with an organisation offering road service to customers when their car breaks down. He has worked in the job for the past 12 years and knows it well. Many of the younger mechanics ask Bob for advice when they meet up in the depot as he always knows the answer to tricky mechanical problems. Bob **likes sharing his knowledge with the younger guys**, as it makes him feel a **valued part of the team**.

Bob works rolling day and night shifts and spends his shifts attending breakdowns and lockouts (when customers lock their keys in the car). About 20% of the jobs he attends are complex and he occasionally needs to refer to his standard issue manuals. Bob tries to avoid using the manuals in front of customers as he thinks it gives the impression he doesn't know what he's doing.

Bob has seen many changes over the years with the company and has tried his best to move with the times. However he found it a bit **daunting** when a new **computer** was installed in his van several years ago, and now he has heard rumors that the computer is going to be updated to one with a bigger screen that's meant to be faster and better.

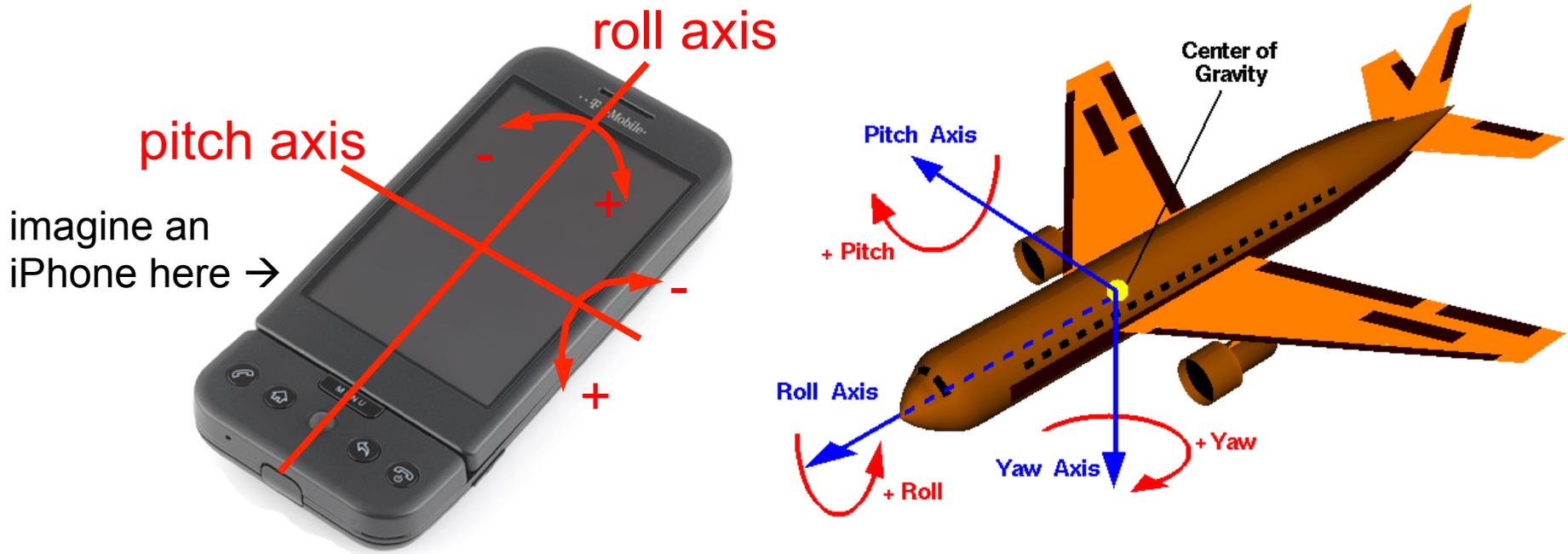
Bob's been told that he will be able to access the **intranet** on the new computer. He has heard about the intranet and saw once in an early version on his manager's computer. He **wonders if he will be able to find out what's going on in the company more easily**, especially as customers seem to know more about the latest company news than he does when he turns up at a job. This can be embarrassing and has been a source of frustration for Bob throughout his time with the company.

Bob wonders if he will be able to **cope with the new computer system**. He doesn't mind asking his grandchildren for help when he wants to send an email to his brother overseas, but asking the guys at work for help is another story.

Source: http://www.steptwo.com.au/papers/kmc_personas/

Sensor Coordinate Systems

“Orientation Sensor” = Accelerometer + Magnetometer



Axis	Zero position	Range	Details
Yaw	north	$-\pi.. \pi$	0=north, $-\pi/2$ =east, $\pm\pi$ =south, $\pi/2$ =west
Pitch	horizontal	$-\pi/2.. \pi/2$	0=horizontal, >0: up, <0: down
Roll	horizontal	$-\pi.. \pi$	0=horizontal, >0: right, <0: left