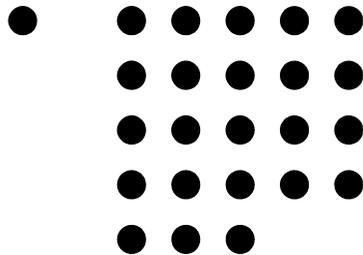




# Android: Property Animation – Einfache Techniken –

Prof. Dr. Carsten Vogt



Fachhochschule Köln  
Cologne University of  
Applied Sciences



## Android: Property Animation

### 1.) Zielsetzung und Möglichkeiten

### 2.) Animation durch animate()

### 3.) Animation durch ObjectAnimator

### 4.) Animation durch XML-Code

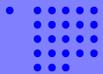
### 5.) Nichtlineare Animation

### 6.) Weitere Informationen



## Zielsetzung und Möglichkeiten

- **generell** möglich:
  - **Animation beliebiger Eigenschaften beliebiger Objekte**
    - Eigenschaft = “Property” = Attributwert
    - Animation = kontinuierliche Wertänderung
  - Voraussetzung:  
Objekt hat entsprechende `set`-Methode
- **hauptsächlich**es Anwendungsgebiet:
  - **Animation von Views auf dem Display**
  - **Position, Größe, Rotation, Farbe, ...**



## Animierbare View-Eigenschaften

- in Android 3.0: **neue View-Eigenschaften**
  - `x`, `y`, `translationX`, `translationY`
  - `rotation`, `rotationX`, `rotationY`
  - `scaleX`, `scaleY`
  - `pivotX`, `pivotY`
  - `alpha`
- zur **Unterstützung der Property Animation** von Views



## Android: Property Animation

- 1.) Zielsetzung und Möglichkeiten
- 2.) Animation durch `animate()`
- 3.) Animation durch `ObjectAnimator`
- 4.) Animation durch XML-Code
- 5.) Nichtlineare Animation
- 6.) Weitere Informationen



## Animation durch `animate()`

- einfachster Ansatz zur Animation
- **View-Methode `animate()`**
  - liefert einen **`ViewPropertyAnimator`**
  - zur Definition und Ausführung der Animation
- Methoden von `ViewPropertyAnimator`:
  - **`setDuration()`**: Dauer der Animation
  - **`x()`**, **`y()`**: Zielposition des Views
  - ... weitere zur Skalierung, Rotation, ...
  - **`start()`**: Starten der Animation



## Android: Property Animation

- 1.) Zielsetzung und Möglichkeiten
- 2.) Animation durch `animate()`
- 3.) Animation durch `ObjectAnimator`
- 4.) Animation durch XML-Code
- 5.) Nichtlineare Animation
- 6.) Weitere Informationen



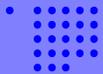
## Animation durch ObjectAnimator

- Klasse `ObjectAnimator`
  - zur Definition und Steuerung einer Animation
- **Factory-Methoden** zur Erzeugung legen fest:
  - Objekt, das animiert wird
  - seine Eigenschaft, die animiert wird
  - Start- und Zielwert der Animation
- **Methoden** auf `ObjectAnimator`-Objekt u.a.:
  - Dauer der Animation
  - Anzahl Wiederholungen, vorwärts/rückwärts?



## Animation mehrerer Eigenschaften

- Animation mehrerer Eigenschaften:
  - mehrere unabhängige ObjectAnimators oder
  - ObjectAnimators in **AnimatorSet**
- Klasse **AnimatorSet** mit Methoden
  - **playTogether()**
  - **playSequentially()**
  - ...



## Android: Property Animation

- 1.) Zielsetzung und Möglichkeiten
- 2.) Animation durch `animate()`
- 3.) Animation durch `ObjectAnimator`
- 4.) Animation durch XML-Code
- 5.) Nichtlineare Animation
- 6.) Weitere Informationen



## Animation durch XML-Code

- XML-Datei in `res/animator`
- mögliche Elemente:
  - `<objectAnimator>`
  - `<animatorSet>`
  - `<valueAnimator>`
    - ValueAnimator:  
Oberklasse von ObjectAnimator
- Details siehe Folge-Video



## Android: Property Animation

- 1.) Zielsetzung und Möglichkeiten
- 2.) Animation durch `animate()`
- 3.) Animation durch `ObjectAnimator`
- 4.) Animation durch XML-Code
- 5.) Nichtlineare Animation
- 6.) Weitere Informationen



## Nichtlineare Animation

- insbesondere: **nichtlineares Zeitverhalten**
  - z.B. Beschleunigung und/oder Verzögerung
  - z.B. Überschießen des Ziels
  - z.B. Abprallen am Ziel
  - z.B. Durchlauf mehrerer Zyklen
- durch Setzen eines **“Time Interpolators”** beim ObjectAnimator
  - hier: Nutzung vorgegebener Interpolatoren
  - Folge-Video: Def. eigener Interpolatoren



## Android: Property Animation

- 1.) Zielsetzung und Möglichkeiten
- 2.) Animation durch `animate()`
- 3.) Animation durch `ObjectAnimator`
- 4.) Animation durch XML-Code
- 5.) Nichtlineare Animation
- 6.) Weitere Informationen



## Weitere Details und Beispiel-Apps

Programmcode zu dieser Präsentation:

- <http://www.nt.fh.koeln.de/vogt/vma/videos.html>

Android-Online-Dokumentation:

- <http://developer.android.com/guide/topics/graphics/prop-animation.html>
- API-Dokumentation der jeweiligen Klassen

Folge-Video zu fortgeschrittenen Techniken:

- <http://www.nt.fh.koeln.de/vogt/vma/videos.html>