

# Aufbau interaktiver 3D-Engines

Universität Osnabrück  
Fachbereich Mathematik / Informatik

## 3. Übung

Prof. Dr. rer. nat. Oliver Vornberger  
Nico Marniok, B. Sc.  
Erik Wittkorn, B. Sc.

23.04.2013

# Übersicht

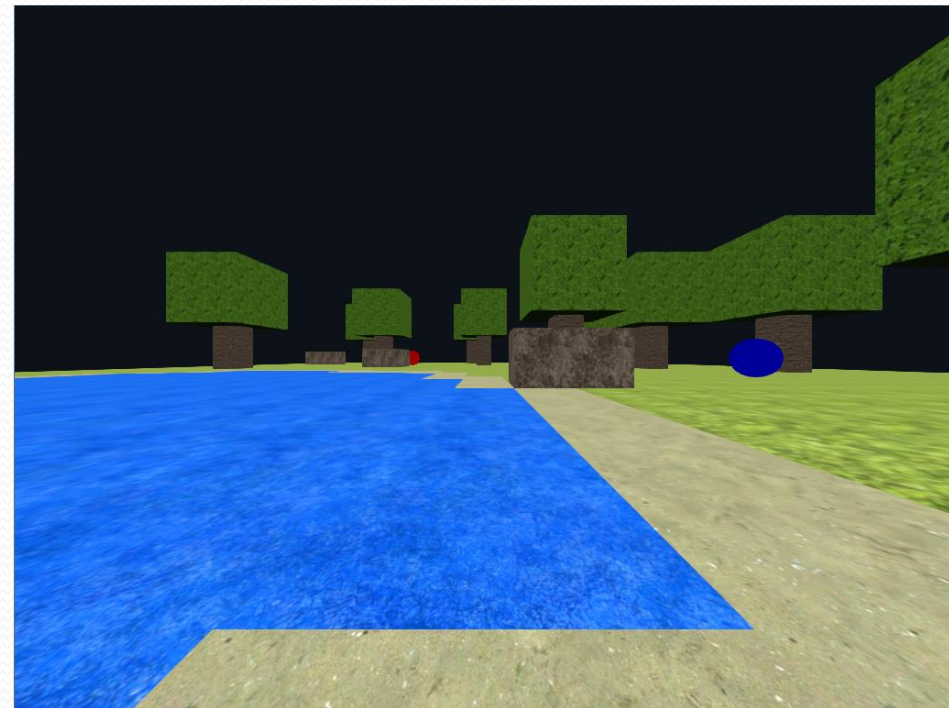
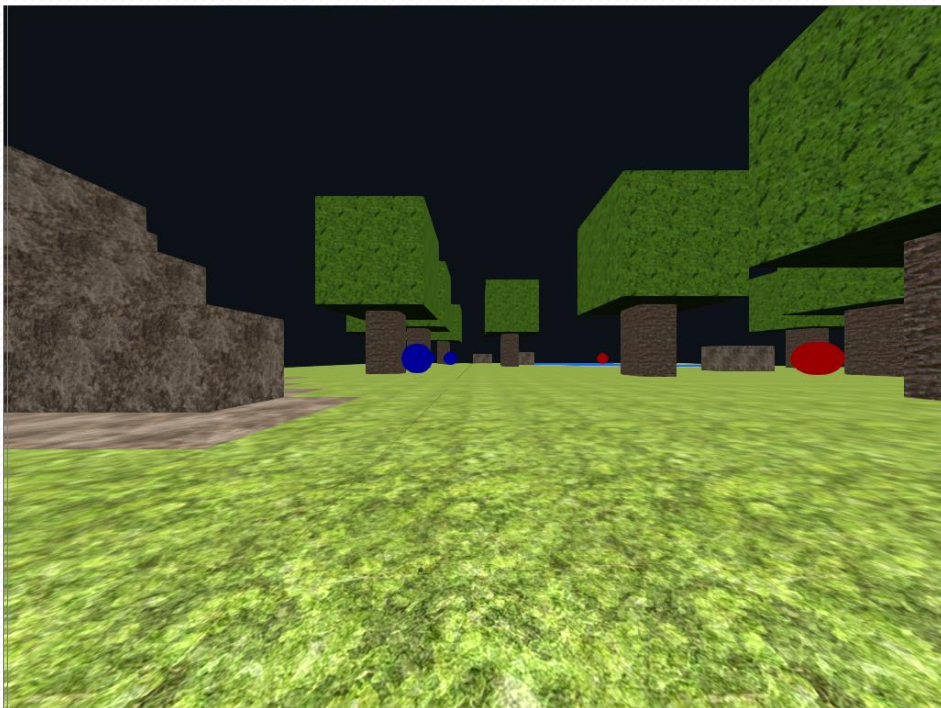
1. Opium Anmeldung!!
2. Präsentation des letzten Übungsblattes
3. Vorstellung des nächsten Übungsblattes
  1. Lösung Demo
  2. Installation
  3. Szenengenerierung
  4. Bewegung
  5. Mana Versorgung



# Präsentation des 1. Übungsblattes

# Übungsblatt 2

- Demo Lösung



# Übungsblatt 2

- Quest 1

- Installation des Projekts
- Neuste Engine laden
- Config anpassen
- LWJGL-Libraries einbinden



# Übungsblatt 2

- Quest 2
  - Baumkrone generieren
  - Koordinaten überlegen und die passenden Actornodes im Code generieren



# Übungsblatt 2

- Szenengenerierung
  - Obj- und mtl-Dateien

## cube.obj

```
mtllib meshes/cube.mtl
mtllib meshes/cube_1.mtl
mtllib meshes/cube_grass.mtl
...
```

Alle für dieses Mesh verfügbaren  
MaterialLibraries angeben

```
v      -0.500000    0.500000    0.500000
v      -0.500000    -0.500000   0.500000
v      0.500000    -0.500000   0.500000
v      0.500000    0.500000    0.500000
...
```

Vertices (Eckpunkte)  
Des Objekts

```
vn 1.000000  0.000000  0.000000
```

Normalen

```
vt 0.000000  0.000000
```

Texturkoordinaten

```
...
# 8 vertices
```

```
usemtl front
f 1/1/5 2/2/5 3/3/5 4/4/5
usemtl back
f 8/1/6 7/2/6 6/3/6 5/4/6
...
```

Erst Angabe des Materials, dann  
Angabe der Fläche mit vier  
Einträgen der Form v/vn/vt

## cube.mtl

```
newmtl front
Ka 0.100 0.100 0.100
Kd 1.000 0.000 0.000
Ks 1.000 0.200 0.200
```

Name des Materials

Angabe einer  
diffusen Farbe

```
newmtl back
Ka 0.100 0.100 0.100
Kd 0.000 1.000 0.000
Ks 0.200 1.000 0.200
...
```

## cube\_grass.mtl

```
newmtl front
Ka 0.100 0.100 0.100
Kd 1.000 0.000 0.000
Ks 1.000 0.200 0.200
map_Kd textures/grass/512x_foliage_coarse01_1.png
```

Angabe einer  
diffusen Textur

```
newmtl back
Ka 0.100 0.100 0.100
Kd 0.000 1.000 0.000
Ks 0.200 1.000 0.200
map_Kd textures/grass/512x_foliage_coarse01_1.png
...
```

# Übungsblatt 2

- Szenengenerierung
  - CubeActorNode

Referenzdatei zur Definition der Actorkomponenten



```
<Actor name="Cube_7.0_5.0_8.0" resource="actors/CubeActor.xml" >
```

```
  <TransformComponent>
```

```
    <Position x="7.0" y="5.0" z="8.0" />
```

```
  </TransformComponent>
```

```
  <RenderComponent>
```

```
    <ObjMesh customMaterialLibrary="meshes/cube_Leaves.mtl" resource="meshes/cube.obj" />
```

```
  </RenderComponent>
```

```
</Actor>
```

Spezielle MaterialLibrary



- actors/CubeActor.xml

```
<Actor>
```

```
  <TransformComponent />
```


```
  <RenderComponent>
```

```
    <ObjMesh resource="meshes/cube.obj" />
```

```
  </RenderComponent>
```

```
</Actor>
```

Alle möglichen Komponenten des Actors eingetragen, aber noch nicht alle ausdefiniert





# Übungsblatt 2

- Szenengenerierung
  - Dateien befinden sich im Levels Ordner

## MageQuestScene.xml

```
<Level xSize="8" zSize="8" ySize="3" tileSize="1" xOffset="0" yOffset="0" zOffset="0">  
  <SpawnPoint xPos="5" yPos="1" zPos="5" />  
  <Layer yPos="0" file="Levels/level1_00.mqll" />  
  <Layer yPos="1" file="Levels/level1_01.mqll" />  
  <Layer yPos="2" file="Levels/level1_02.mqll" />  
</Level>
```

← Benutzerdefiniertes XML-Format für ein Spiel-Level

## level1\_01.mqll

```
0 0 0 0 0 0 0 0  
0 0 0 2 2 0 0 0  
0 0 2 2 2 2 2 2  
0 2 2 2 2 2 2 0  
0 2 2 2 2 2 0 0  
0 2 2 2 2 2 0 0  
0 2 2 2 2 2 0 0  
0 2 2 2 2 2 0 0
```

← Jede ID definiert, an welcher Position ein Block eines bestimmten Typs gesetzt werden soll.

## mqll.txt

```
1 cube_grass  
2 cube_stone  
3 cube_sand  
4 cube_water  
5 cube_wood  
6 pickup  
7 leaves
```

# Übungsblatt 2

- Szenengenerierung
  - Level-Objekt (*Level.java*)
    - Container für geparste Informationen in einem 3D-Array [x][y][z]
    - Typen des Level-Elements und ActorIDs
  - MageQuestUtil (*MageQuestUtil.java*)
    - Generiert ActorNodes aus Level-Object
    - Ruft die `createActor(Node node)` Methode von MageQuestLogic auf

# Übungsblatt 2

- Quest 3



- Neue Events für Rotation erstellen
- Spieler um  $90^\circ$  ( $\text{PI}/2$ ) rotieren
- Bewegungsrichtung korrigieren
- Kollisionstest
- Rotationsinterpolation

# Übungsblatt 2

- Bewegung

- Events

```
public class MoveEvents {  
    public static class ForwardEvent extends ToggleEvent {  
        public static final int ID = 0x19975e7a;  
        public int getId() {  
            return ID;  
        }  
    }  
    public static class BackwardEvent extends ToggleEvent {  
        public static final int ID = 0x35d76bed;  
        public int getId() {  
            return ID;  
        }  
    }  
    public static class StrafeLeftEvent extends ToggleEvent {  
        public static final int ID = 0x66e04586;  
        public int getId() {  
            return ID;  
        }  
    }  
    public static class StrafeRightEvent extends ToggleEvent {  
        public static final int ID = 0xbecdda91;  
        public int getId() {  
            return ID;  
        }  
    }  
}
```

 Eindeutige ID

- config.ini

[CONTROLS]

events.MoveEvents\$ForwardEvent=W  
events.MoveEvents\$BackwardEvent=S  
events.MoveEvents\$StrafeLeftEvent=A  
events.MoveEvents\$StrafeRightEvent=D

- LWJGL Keyboard

- <http://www.lwjgl.org/javadoc/org/lwjgl/input/Keyboard.html>

static int	KEY_D
static int	KEY_DECIMAL
static int	KEY_DELETE
static int	KEY_DIVIDE
static int	KEY_DOWN
static int	KEY_E
static int	KEY_END
static int	KEY_EQUALS
static int	KEY_ESCAPE
static int	KEY_F

# Übungsblatt 2

- Bewegung
  - InputHandler
    - Ist ein Eventlistener für die InputEvents
    - Leitet Input an die GameLogic weiter

```
public void trigger(EventData data) {
    ToggleEvent controlEvent = (ToggleEvent)data;
    switch (controlEvent.getId()) {
        case MoveEvents.ForwardEvent.ID :
            if (controlEvent.isActivated()) {
                MageQuestLogic.movePlayer(PlayerController.DIR_FORWARD);
            }
            break;
        case MoveEvents.BackwardEvent.ID :
            if (controlEvent.isActivated()) {
                MageQuestLogic.movePlayer(PlayerController.DIR_BACKWARD);
            }
            break;
        case MoveEvents.StrafeLeftEvent.ID :
            if (controlEvent.isActivated()) {
                MageQuestLogic.movePlayer(PlayerController.DIR_STRAFELEFT);
            }
            break;
    }
}
```



Leitet Events an die GameLogic-Klasse weiter

# Übungsblatt 2

- Bewegung
  - PlayerController
    - Kann die Pose (Position, Rotation) des Spielers manipulieren

```
private void updatePose() {  
    if(this.playerPose != null) {  
        this.playerPose.setRotation(this.yprRotation.x, this.yprRotation.y, 0.0f);  
        this.playerPose.setPosition(this.position.x, this.position.y, this.position.z);  
    }  
}
```

- Ist ein Prozess, der zwischen Start- und Zielposition interpoliert

```
protected void onUpdate(long dMillis) {  
    if (this.moving) {  
        if ([im Ziel]) {  
            this.position = new Vector3f(this.goalPosition);  
            this.moving = false;  
        } else {  
            float speed = dMillis * moveSpeed;  
            this.position.x += this.diffPosGoalVector.x * speed;  
            this.position.y += this.diffPosGoalVector.y * speed;  
            this.position.z += this.diffPosGoalVector.z * speed;  
        }  
        this.updatePose();  
    }  
}
```

# Übungsblatt 2



- Quest4
  - PickupActor generieren
  - Zufällige Auswahl von Mana- oder Health-Pickup
  - PickupProcess verbraucht Pickup
  - LevelActorDeletedEvent erstellen und Pickup zerstören
- Sidequest
  - Prozess und Events erstellen, die ein neues Pickup erstellen

# Übungsblatt 2

- Mana Versorgung
- PlayerActor.xml

```
<Actor>  
  <TransformComponent />  
  <CameraComponent>  
    <Rotation phi="0.0" theta="0.0" />  
    <Perspective fov="120.0" aspect="auto" znear="0.1" zfar="100.0" />  
  </CameraComponent>  
  <AttributeContainerComponent />  
</Actor>
```

- MageQuestScene1.xml

```
<Actor resource="actors/PlayerActor.xml" name="pLayer">  
  <AttributeContainerComponent>  
    <Attribute name="Mana" value="0" />  
    <Attribute name="Health" value="0" />  
  </AttributeContainerComponent>  
</Actor>
```

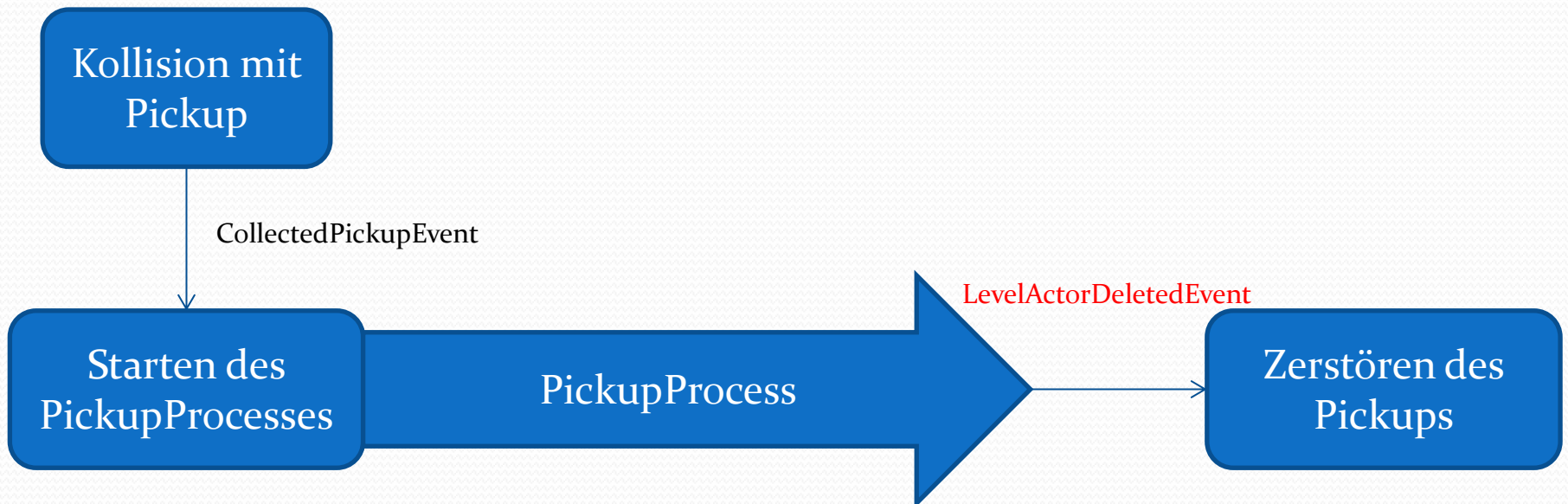
- PickupActor.xml

```
<Actor>  
  <TransformComponent />  
  <RenderComponent>  
    <ObjMesh resource="meshes/sphere.obj"/>  
  </RenderComponent>  
  <AttributeContainerComponent />  
  <PickupComponent />  
</Actor>
```



# Übungsblatt 2

- Mana Versorgung



Nächste Woche:

- Besprechung des aktuellen Übungsblattes
- Weiteres

**Vielen Dank für die  
Aufmerksamkeit 😊**