

Aufbau interaktiver 3D-Engines

Universität Osnabrück
Fachbereich Mathematik / Informatik

8. Übung

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

11.06.2013

Übersicht

1. Präsentation des letzten Übungsblattes
2. Vorstellung des nächsten Übungsblattes
 1. Lösung Demo
 2. PiratenBillard
 3. TestArena
 4. MarbleMadness

Präsentation des 4. Übungsblattes

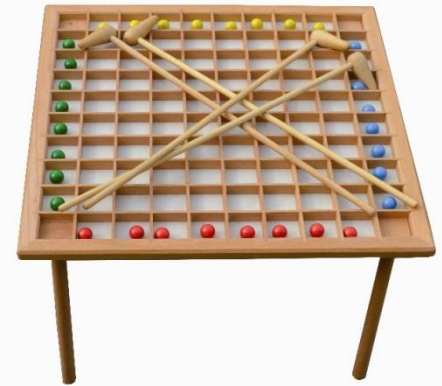
Übungsblatt 5

- Piratenbillard (Demo)



Übungsblatt 5

- Piratenbillard
 - 100 (-4) Fächer
 - 8 Kugeln pro Spieler
 - Einen Queue pro Spieler
 - Ziel ist es, die Kugeln in die gegenüberliegenden Fächer zu bekommen
 - Im Augenblick nur Testversion, Singleplayer



Übungsblatt 5

- Physik
 - Rigid Bodies
 - Kugeln (Density = 0.5)
 - Unsichtbarer Boden und Wände (Density = 0.0)
 - Queue ist ein Kinematic Actor -> er beeinflusst die Physik-Welt, aber wird nicht beeinflusst, sondern bewegt

Actor-XML:

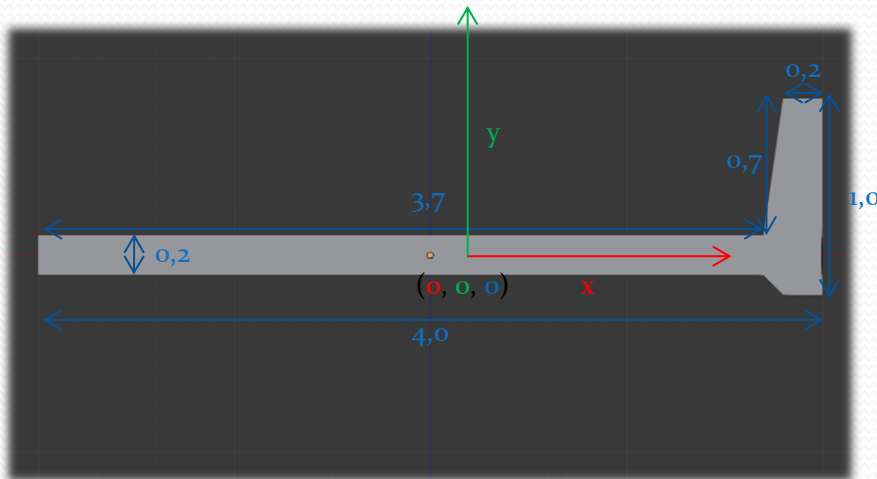
```
<Actor>
...
  <PhysicsComponent>
    ...
    <Property name="kinematic" value="true" />
  </PhysicsComponent>
...
</Actor>
```

PhysicsComponent.java - postInit()

```
if (this.kinematic) {
    this.body.setCollisionFlags(this.body.getCollisionFlags() | CollisionFlags.KINEMATIC_OBJECT);
    this.body.setActivationState(CollisionObject.DISABLE_DEACTIVATION);
}
```

Übungsblatt 5

- Queue



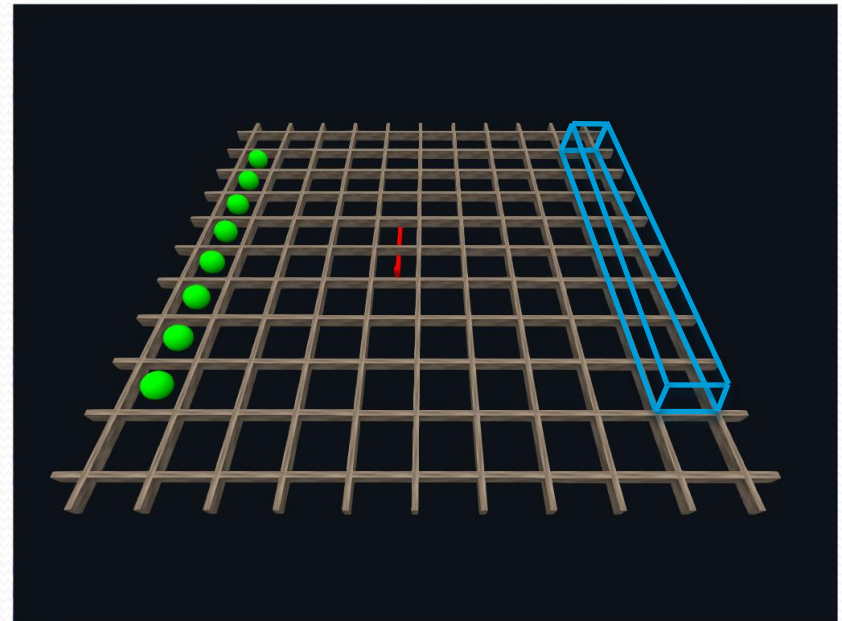
- Zusammengesetzter Physik-Actor (Beispiel):

- Tiefe: 0.2
- Kinematic Actor ist aus Cubes und Spheres zusammengesetzt

```
<PhysicsComponent>
  <Box>
    <Size x="1.0" y="1.5" z="4.8" />
    <Density value="0.0" />
  </Box>
  <Box>
    <Position x="5.0" y="-0.3" z="2.0" />
    <Size x="0.7" y="0.6" z="0.4" />
    <Density value="0.0" />
  </Box>
  <Property name="kinematic" value="true" />
</PhysicsComponent>
```

Übungsblatt 5

- Zielposition
 - CubeShape am Rande des Spielfelds
 - "GhostObject" ist durchlässig für andere Bodies
 - Dient zur effektiveren Kollisionserkennung
 - Alle Rigid Bodies, die mit dem GhostObject kollidieren, können abgerufen werden



Übungsblatt 5

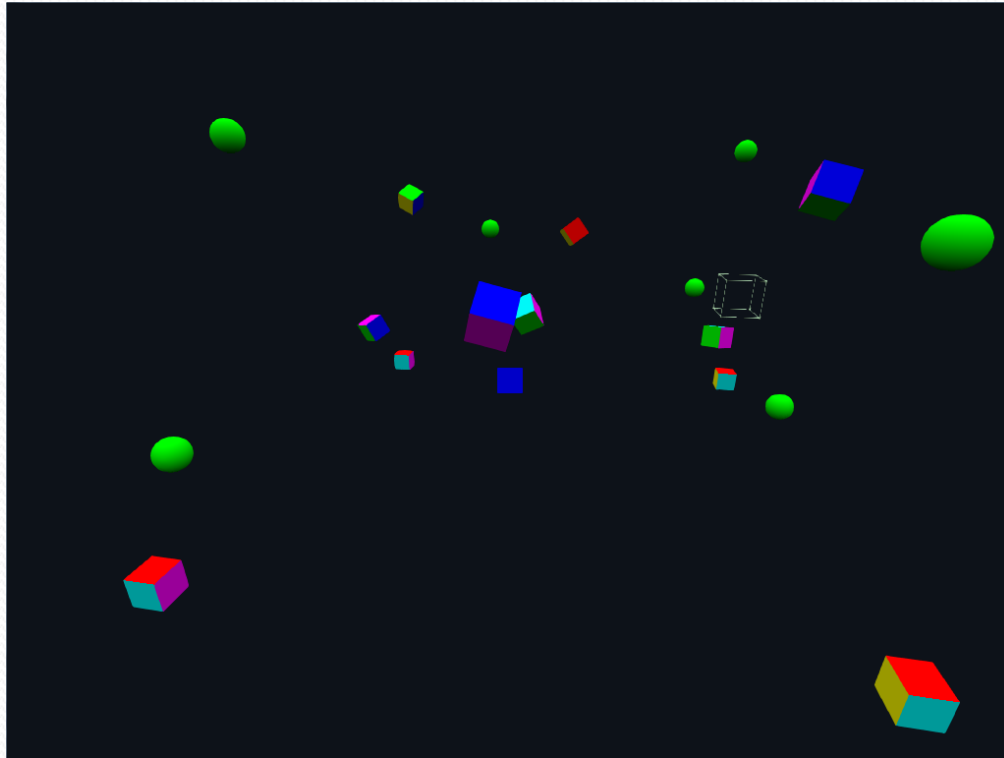
- Quest 1



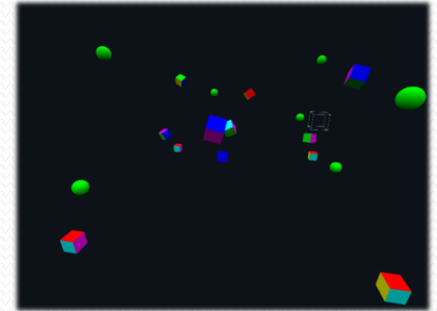
- Installation des Aufgaben-Projekts
- Den Queue aus verfügbaren PhysicShapes zusammensetzen
- Das Ziel-Volumen richtig setzen und Kugeln verschwinden lassen

Übungsblatt 5

- TestArena



Übungsblatt 5



- TestArena
 - Kamera ist Kinematic Actor
 - Besser: KinematicCharacterController (<http://bit.ly/13TwIDu>)
 - GhostObject wird mit Character verschoben und nächster Schritt auf Kollision überprüft
 - RayCast
 - StartPosition (Kamera)
 - Richtungsvektor mit gewisser Länge (?)
 - ClosestResultCallback ist schon implementiert und findet die nächste Kollision mit einem Rigid Body
 - Impuls
 - Braucht eine relative Position auf dem Body
 - Und einen Richtungsvektor mit Länge

Übungsblatt 5

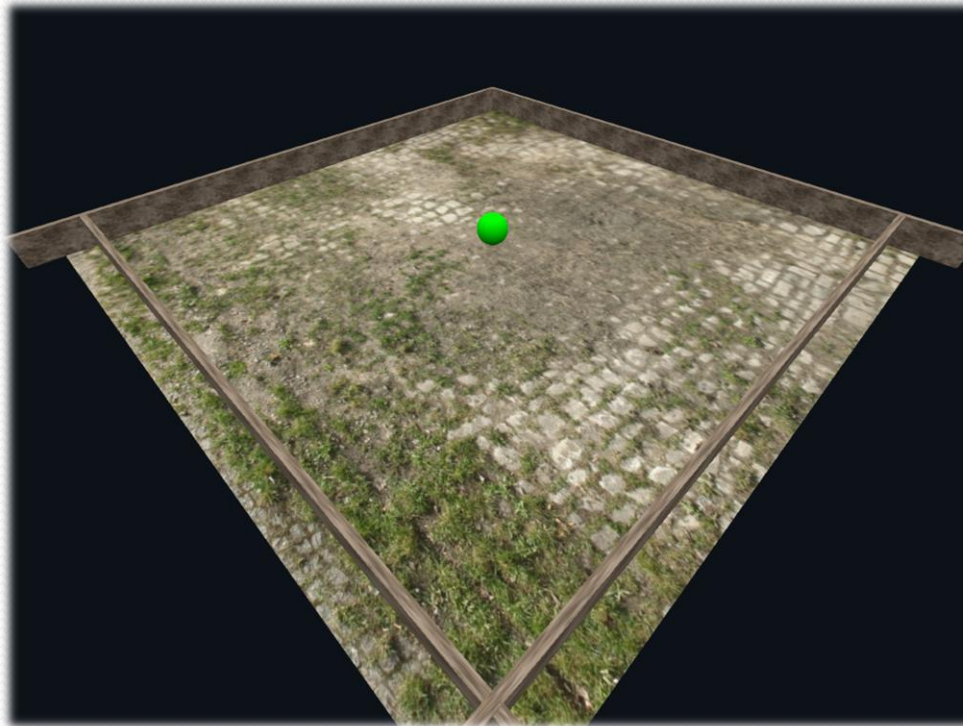
- Quest 2

- RayCast und Impuls komplettieren
- Sound abspielen, bei Impuls-Abgabe
- Mit Parametern und Setup herumspielen



Übungsblatt 5

- MarbleMadness



Übungsblatt 5

- SideQuest
 - Funktionierendes Maze erstellen
 - Zielposition für Kugel mit GhostObject erstellen



Nächste Woche vorraussichtlich:

- To be announced

Vielen Dank für die
Aufmerksamkeit 😊