Bremen

CG
VR

Virtual Reality & Physically-Based Simulation
Organization

G. Zachmann

University of Bremen, Germany

cgvr.cs.uni-bremen.de

Prerequisites

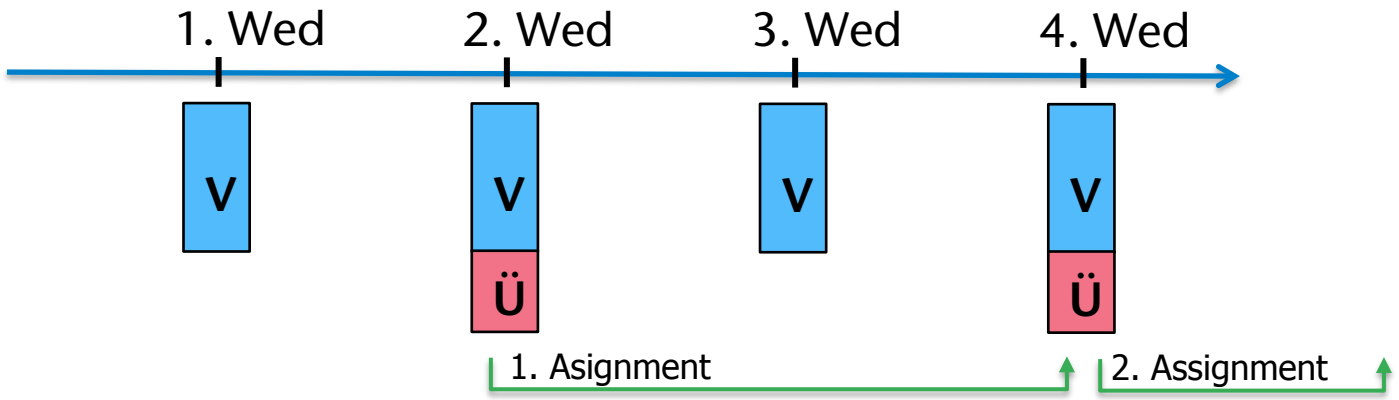
- A little bit of math (just 1. Semester)
  - Well, differential equations in the last few weeks ☺
- A little bit of programming (Java or Javascript or C/C++)

Where to Find Information for This Course

- The course's homepage:
  http://cgvr.cs.uni-bremen.de/
  → "Teaching" → "Virtual Reality"
- Slides (a.k.a. Script) & Assignments
- Hints for text books, online documentation
- Announcements (rarely)
- Please register in StudIP!

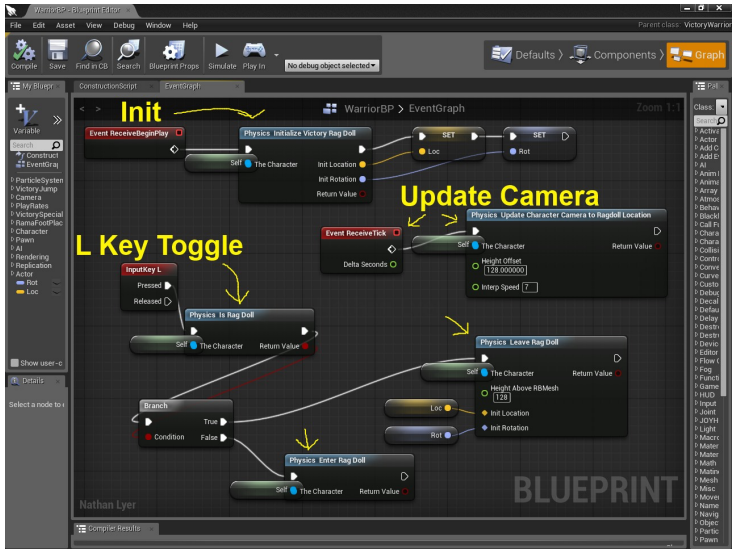
Time of the Class

- Wednesdays: alternatingly
  - 2 lectures in one row (8 ct — 12)
  - 1 lecture and 1 lab meting



Exercises

- Mix of theoretical and practical exercises
- Theoretical = pencil & paper
- Practical = Unreal engine
  - Using "blueprints" first
  - Later some C programming
- Unreal engine: version 4.9
  - Download: www.unrealengine.com
  - Tutorials at https://wiki.unrealengine.com/Videos
- Recommendation: work in groups of 2-4



The Exam

1. Either: long exam (= ½ hour per student)
2. Or: points from the assignments + short exam
   - Assignments → grade A , short exam → grade B
     - 95% of the points from all assignments → grade A = 1.0
     - 40% of the points from all assignments → grade A = 4.0
   - Overall grade = $0.5 \times A + 0.5 \times B$
   - Precondition: grade A ≥ 4.0 && grade B ≥ 4.0 !
     (Allgemeiner Teil der Bachelorprüfungsordnungen der Universität Bremen, 2010)

- Note: in *all* variants, *all* of the course material could be topics in the exam!

- Criteria for grading the practical assignments:
  1. Good (= speaking) variable and function names
  2. Sufficient in-line comments
  3. Documentation of the function and its parameters (in/out, pre-/post-condition, what does the function do, ...)
  4. Functionality (solves assignments? no bugs? ...)

Tentative Overview

1. Introduction, immersion/presence/fidelity
2. VR frameworks,
3. Devices
4. System overview
5. Stereo rendering
6. Some techniques for real-time rendering
7. Simple Interaction: gesture recognition, navigation, selection, ...
8. Complex interaction: WIM, action-at-a-distance, ..
9. Collision detection
10. Force feedback
11. Sound rendering
12. Particle Systems
13. Spring-mass systems