



hochschule aschaffenburg  
university of applied sciences

# Generatives Modellieren

---

Spielmechaniken: Raum und Zeit

# Block I - Grundlagen

## 2. Einstieg in C#

- Theorie:
  - Wiederholung: Spielmechanik 1 + 2: Raum und Zeit
- Skripting in Unity mit C#
- Beispiel: „Generatives Modellieren“
- Theorie:
  - Spielmechanik 3: Objekte, Attribute, Statusangaben

- Wiederholung:
  - Spielraum
  - Spielzeit
- Proof of Concept

**Projekt**

---

# Spielraum

- diskret oder kontinuierlich
- Statische Screens oder dynamisch (bspw. Scrolling)
- eine oder mehrere Dimensionen (keine Dimensionen?)
- begrenzte Bereiche; open world vs. Schlauchlevel
- Verbindungen (oder auch keine)
- untergeordnete Spielräume

# Spielzeit

- diskret (rundenbasiert) oder kontinuierlich
- Spielzeit vs. reale Zeit
- Zeitmessung (UI), Zeitvorgabe
  - Zeit = Erfolg (Punktesystem)
  - Zeitlimit
- Zeitmanipulation
- „autonomes“ Vergehen von Zeit; Wartezeit; Bauzeit



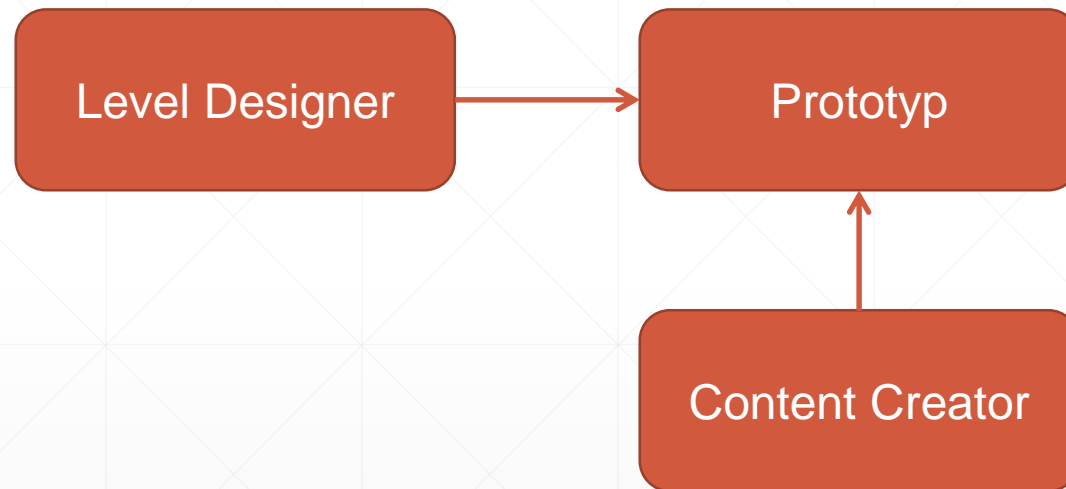
# Projektanfrage

- Diablo Clone
- Vorgaben:
  - Raum: 2,5D, kontinuierlich
  - Zeit: Echtzeit
- Größe Spielwelt:
  - 1.000 Level
  - pro Level 50 x 50 Felder

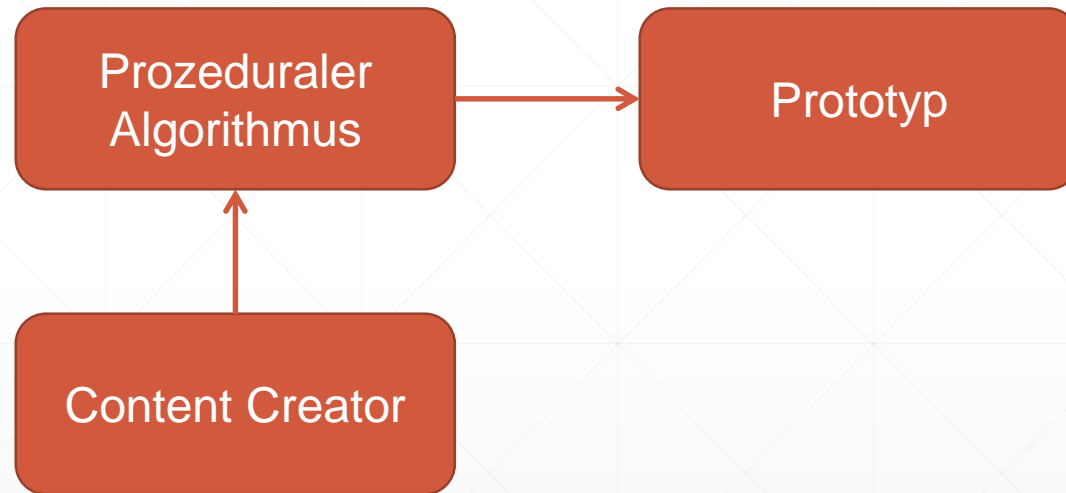


(Quelle: Wikipedia)

# Workflow



# Workflow





# Proof of Concept

- Aufgabe: Entwerfen Sie ein perfektes Labyrinth mit 5x5 Feldern.

- Labyrinth Generierung

**Theorie**

---

# Binary Tree Maze Generator

- Stateless
- Strong bias: Gegenüberliegende Ränder
- Startpunkt (0/0)
- Pro Zelle entweder Durchgang Rechts oder Durchgang Oben

# Recursive Backtracker Maze Generator

- Startpunkt beliebig
- Kein bias
- States („Visited“)
- Durchgang zu zufälligem, nicht besuchtem Nachbarn
- Von dort weiter (Rekursiv), bis keine unbesuchten Nachbarn
- Dann zurück, und zurück bis zum Startpunkt

# Weitere Maze Generatoren

- Eilers Algorithmus
- Kruskals Algorithmus
- Prims Algorithmus
- Recursive Division
- Aldous-Broder Algorithmus
- Wilsons Algorithmus
- Hunt and Kill Algorithmus

<https://www.jamisbuck.org/mazes/>

# Weitere Generatoren

- Was lässt sich noch generativ erstellen? Und warum?