Architektur von Videospielen

Vincent Langenfeld, Daniel Dietsch

14. November 2019

Organisation



Beginnen Sie jetzt mit dem Programmieren

- /src/<projektname>.sln
- Arbeiten Sie an einem Projekt, integrieren Sie kontinuierlich
- Nehmen Sie die wöchentlichen Aufgaben als Chance sich zu verbessern
- MS01: Nächste Woche (Spielobjekt in der Welt bewegbar, bewegliche Ansichten, Level laden/speichern, Soundausgabe)

Ideenpräsentation

- ightharpoonup max 10~min (+ 5~min Fragen)
- Bereiten Sie sich vor:
 - Worum geht es?
 - Alleinstellungsmerkmal?
 - Spielablauf? (gewinnen, verlieren)
 - Warum macht es Spaß?

Architektur

Architektur • Aufgaben



Runtime

- Eingabe/Ausgabe
- Menüs und Popups
- Persistente Einstellungen
- Speichern/Laden
- Rendern und Kamera
- Spielobjekte und Eigenschaften
- Interaktion
- Sound und Musik
- KI
- Pathfinding
- Kollisionserkennung
- Netzwerk

Tools

- Modelle integrieren
- Texturen integrieren
- Sounds integrieren
- Level erstellen
- Debugging output

Architektur • Aufgaben

UNI

Runtime

- Eingabe/Ausgabe
- Menüs und Popups
- Persistente Einstellungen
- Speichern/Laden
- Rendern und Kamera
- Spielobjekte und Eigenschaften
- Interaktion
- Sound und Musik
- KI
- Pathfinding
- Kollisionserkennung
- Netzwerk

Tools

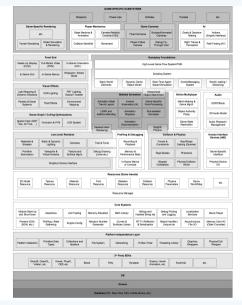
- Modelle integrieren
- Texturen integrieren
- Sounds integrieren
- Level erstellen
- Debugging output



A Man Ran (WS17/18)

Architektur • Referenz (Gregory 2019)





Gregory, Game Engine
Architecture, 2019

Architektur



Am Besten Top-Down anfangen:

- Zuerst Struktur aufbauen
- ► Funktionalität aufteilen
- ► SoC

MonoGame

Content

Game Logic

Engine

MonoGame und .net

MonoGame und .net

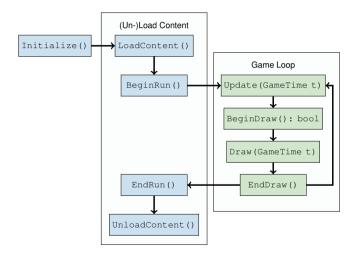


.net

- Mathematik
- Zufallszahlen
- ► (De-)Serialisierung in XML/Binär
- ► Datenstrukturen (Liste, Menge, ...)
- Debugging
- Profiling

MonoGame

- Abstraktion
 - Grafik
 - Sound
 - Input
- Content Pipeline
- Einfache Anzeigemethoden (BasicEffect, SpriteBatch)
- Datentypen
 - Vecto2, Vector3
 - Matrix
 - **.**.



Kochbuch

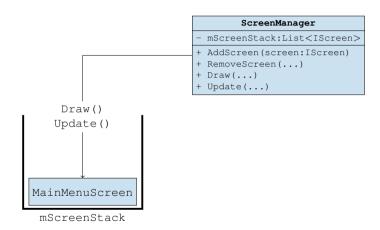


- Verschiedene Ansichten des Spiels in Screens unterteilen
 - Hauptmenü
 - Optionsmenü
 - Spielansicht
 - ► HUD
 - Ladebildschirm
 - **.**

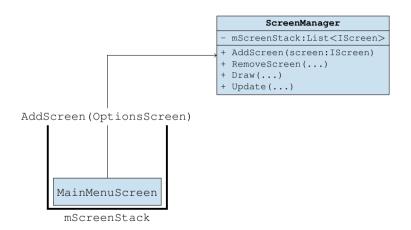


- Verschiedene Ansichten des Spiels in Screens unterteilen
 - Hauptmenü
 - Optionsmenü
 - Spielansicht
 - ► HUD
 - Ladebildschirm
 - ▶ ..
- ScreenManager verwaltet alle Screens
- ScreenManager verwaltet Übergänge zwischen Screens
- Verwaltung der Screens vom Inhalt des einzelnen Screens trennen

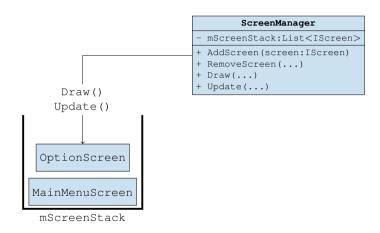




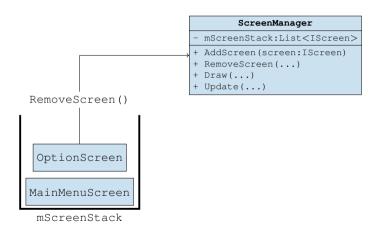




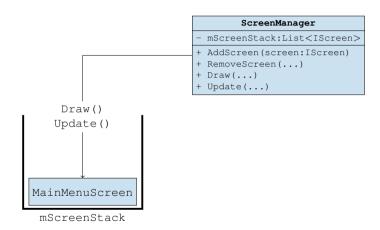




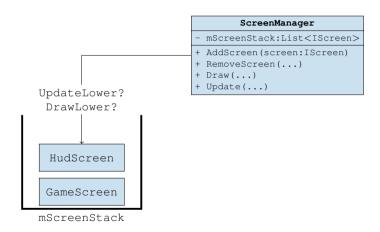




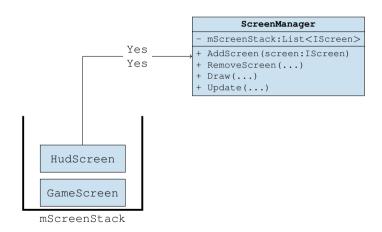








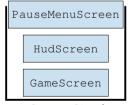






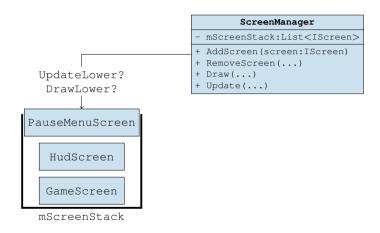
ScreenManager

- mScreenStack:List<IScreen>
- + AddScreen(screen:IScreen)
- + RemoveScreen(...)
- + Draw(...)
- + Update(...)

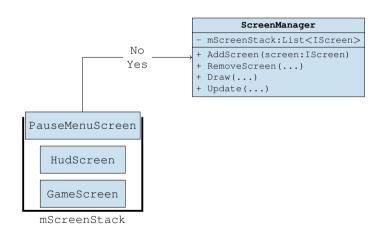


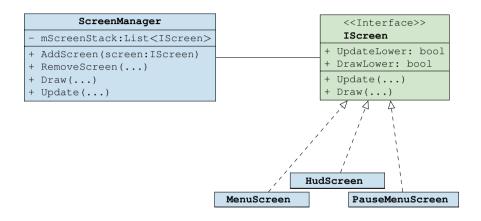
mScreenStack

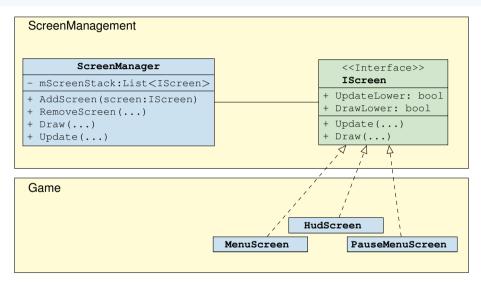


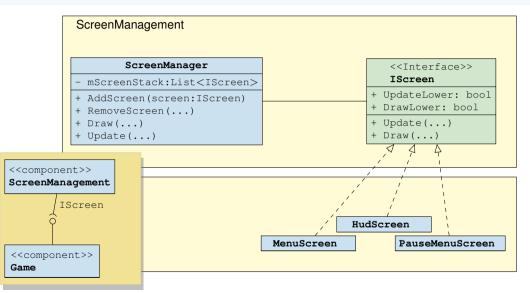










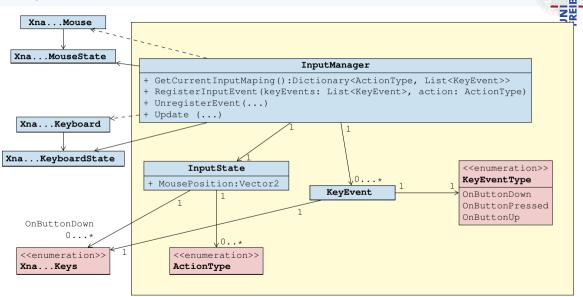


Rezept II • Input Management



- Tastatur, Maus, Gamepad, ...
- ▶ MonoGame Namespace Microsoft.Xna.Framework.Input
 - KeyboardState mKeyboardState = Keyboard.GetState();
 - Statusinformation per Frame: Zustand der Tasten, Mausposition
 - keine Historie
- Wir brauchen
 - Statusinformation in Abhängigkeit von Zeit (z.B.: A wurde diese Frame losgelassen)
 - Abstraktion von Tasten zu Aktionen

Rezept II • Input Management



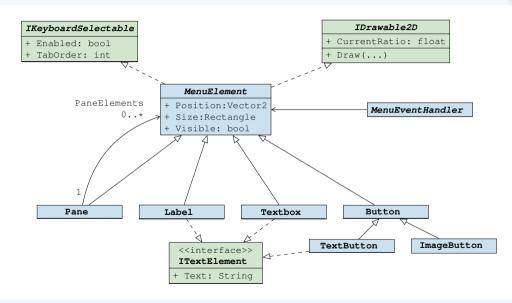
Rezept III • Menüs



- ► Viele Elemente die sich sehr ähnlich verhalten
 - Panels
 - Buttons
 - Labels
 - Textboxen
 - Checkboxen
- Auch im Hud verwendbar

Rezept III • Menüs





Rezept IV



Spielobjekte

Spielobjekte sind alle Dinge, die eine Repräsentation in der Spielwelt haben.

- Charaktere, Fahrzeuge, Bäume, Raketen, Gras, Steine, Trigger, Lichter, Sounds, ...
- Spielobjekte müssen manchmal
 - gezeichnet werden
 - sich bewegen
 - zerstörbar sein
 - miteinander kollidieren
 - ect.
- Wie verwaltet man so viele (verschiedene) Objekte effizient?

Rezept IV



Spielobjekte

Spielobjekte sind alle Dinge, die eine Repräsentation in der Spielwelt haben.

- Charaktere, Fahrzeuge, Bäume, Raketen, Gras, Steine, Trigger, Lichter, Sounds, ...
- Spielobjekte müssen manchmal
 - gezeichnet werden
 - sich bewegen
 - zerstörbar sein
 - miteinander kollidieren
 - ect.
- Wie verwaltet man so viele (verschiedene) Objekte effizient?
 - Zur Laufzeit?

Rezept IV



Spielobjekte

Spielobjekte sind alle Dinge, die eine Repräsentation in der Spielwelt haben.

- Charaktere, Fahrzeuge, Bäume, Raketen, Gras, Steine, Trigger, Lichter, Sounds, ...
- Spielobjekte müssen manchmal
 - gezeichnet werden
 - sich bewegen
 - zerstörbar sein
 - miteinander kollidieren
 - ect.
- Wie verwaltet man so viele (verschiedene) Objekte effizient?
 - Zur Laufzeit?
 - Im Softwaredesign?

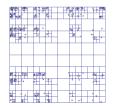
Rezept IV • Spielobjekte

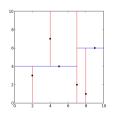


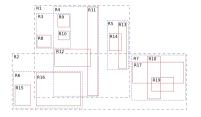
Szenengraph

Ein Szenengraph ist eine zentrale Datenstruktur, die der logischen und/oder räumlichen Verwaltung der Spielobjekte dient.

- ► Z.B.:
 - Antwort auf räumliche Fragen
 - Update- und Drawaufrufe an Spielobjekte weitergeben
- ▶ Beispiele: Liste, Heap, Quad-/ Octree, KD-Tree, R-Tree, ...







Rezept IV • Spielobjekte



Objektzentriert

- Spielobjekte sind Klassen mit
 - Eigenschaften des Spielobjekts
 - Verhalten des Spielobjekts
- Spielwelt ist eine Menge von Instanzen der Spielobjekte

Eigenschaftszentriert

- Jedes Spielobjekt ist nur eine ID
- Eigenschaften der Spielobjekte werden in Tabellen gespeichert
- Verhalten von Spielobjekten sind Operationen auf Tabellen
- Änhnlich zu Datenbanken

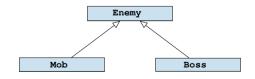
Rezept IV • Spielobjekte und Vererbung



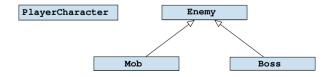
Mob

Boss

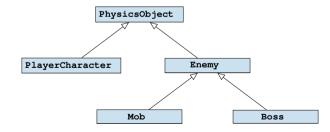




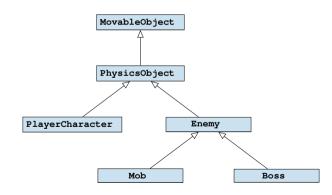




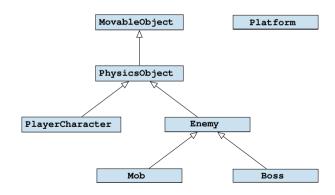




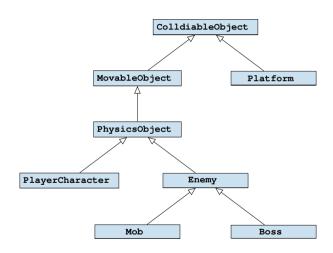




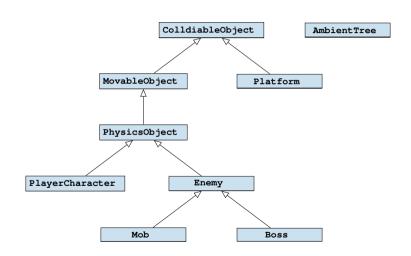




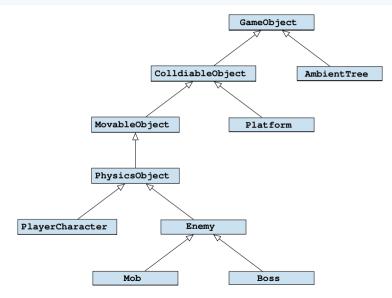




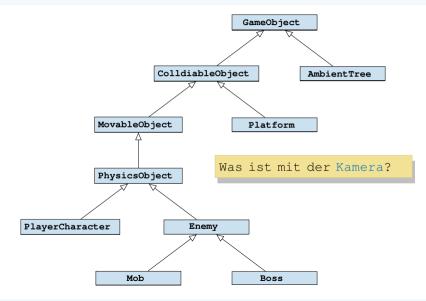






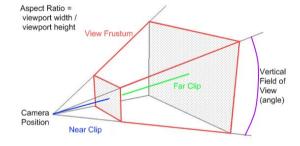






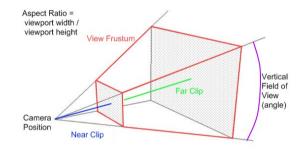
Eine einfache Kamera

- Speichert z.B.:
 - View Matrix
 - Projection Matrix
- Grundlage für Picking
- Definiert View Frustrum



Fine einfache Kamera

- Speichert z.B.:
 - View Matrix
 - Projection Matrix
- Grundlage für Picking
- Definiert View Frustrum
 - Teil der Spielwelt
 - Nicht kollidierend
 - Beweglich

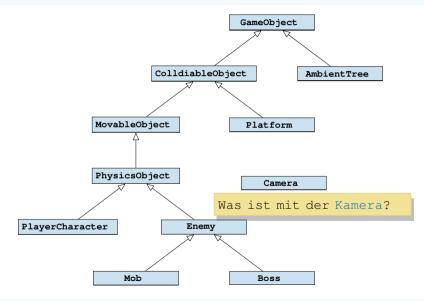


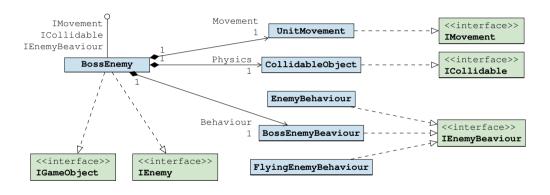
Rezept IV • Spielobjekte



	Tree	Platform	PlayerCharacter	Mob	BossMob	Camera
GameObject	✓	✓	✓	√	✓	\checkmark
Collidable		\checkmark	\checkmark	\checkmark	\checkmark	
Movable			\checkmark	\checkmark	\checkmark	\checkmark
Physics			\checkmark	\checkmark	\checkmark	
EnemyBehaviour				\checkmark	\checkmark	







Rezept V • Pathfinding



Offline

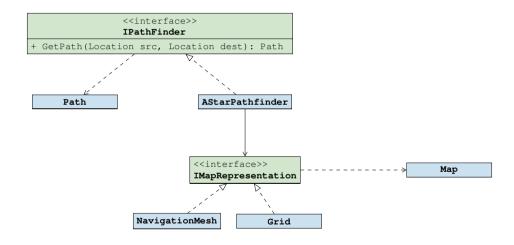
- Berücksichtigt Welt und unbewegliche Objekte
- Meistens A*
- Viele mögliche Weltrepräsentationen: Grid, Hirarchical Grid, Waypoint Graph, Navigation Mesh,...

Online

- Beweglichen Objekten ausweichen
- Verschiedene Verfahren: Steering, Flocking, Flow Fields, ...

Rezept V • Pathfinding





Was nun?

Was nun?



Die Libraryfalle

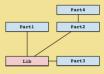
- ► Es existieren viele Libraries
- ► Pro:
 - Rad nicht neu erfinden
 - Optimierte Lösungen für bekannte Probleme
- Aber:
 - Problem muss verstanden sein
 - Library muss verstanden werden
 - Versteckte Kosten wenn Library fehlerhaft/schlecht/unpassend/...
- Vorher das Risiko abschätzen:
 - Wie zentral ist die Library?
 - Wie reif ist die Library?
 - Können wir das einfacher selbst machen?



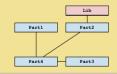
Die Libraryfalle

- ► Es existieren viele Libraries
- ► Pro:
 - Rad nicht neu erfinden
 - Optimierte Lösungen für bekannte Probleme
- Aber:
 - Problem muss verstanden sein
 - Library muss verstanden werden
 - Versteckte Kosten wenn Library fehlerhaft/schlecht/unpassend/...
- Vorher das Risiko abschätzen:
 - Wie zentral ist die Library?
 - Wie reif ist die Library?
 - Können wir das einfacher selbst machen?

Wie zentral ist die Library? -Viel potentieller Schaden:



- Weniger möglicher Schaden:



Was nun?



- Fangen Sie jetzt an zu programmieren
 - ▶ In einem Projekt
 - Arbeiten Sie gemeinsam
- ► GDD-Reviews lesen und Probleme beheben
 - Inhaltliche Probleme früh lösen
- Nicht vergessen:
 - Beta-Architektur
 - Ideenpräsentation

Was nun?



- Fangen Sie jetzt an zu programmieren
 - ► In einem Projekt
 - Arbeiten Sie gemeinsam
- GDD-Reviews lesen und Probleme beheben
 - Inhaltliche Probleme früh lösen
- Nicht vergessen:
 - Beta-Architektur
 - Ideenpräsentation

Für Fragen gibt es die Poolsprechstunde:
Donnerstags 14-18 Uhr

Fragen?