

Bericht "World of the Fittest"

missing title page (author, date, dov. version)

Einführung

In diesem Bericht wird die Funktionsweise eines Simulationsprogrammes mithilfe von drei verschiedenen UML-Diagrammen erklärt: dem Klassendiagramm, dem Prozessdiagramm und dem Use-Case Diagramm. Mit dem geschilderten Programm soll ein Ökosystem mit Tieren wie Füchsen und Hasen sowie Futter (Gras) und deren Verhalten untereinander simuliert werden. Die Diagramme zeigen die Struktur, das Verhalten und die Interaktionen mit dem Benutzer der Software auf. Durch diese Analyse werden die wichtigsten Aspekte des Programmes identifiziert und ihre Bedeutung für das Erreichen der Ziele der Software hervorgehoben.

Methodik

In dieser ersten Phase wurde mithilfe vorgelegter Schritte die Funktionsweise und Struktur der Simulation erarbeitet. Die Schritte waren:

1. Das Klassendiagramm erstellen
2. Das Prozessdiagramm erstellen
3. Das Use Case Diagramm erstellen

Diese Diagramme werden im späteren Verlauf des Berichtes genauer erläutert. Die Diagramme wurden mittels einfachen Text-Dateien erstellt und mit UML (Unified Modelling Language) geschrieben. UML ermöglicht mittels einfachem Text-Coding Diagramme beschreibend zu erstellen und visualisieren. Diese Text-Dateien wurden anschliessend als .pu File abgespeichert und über das Programm PlantUML zu Diagrammen (als Bilddatei) übersetzt.

Resultate

Klassendiagramm

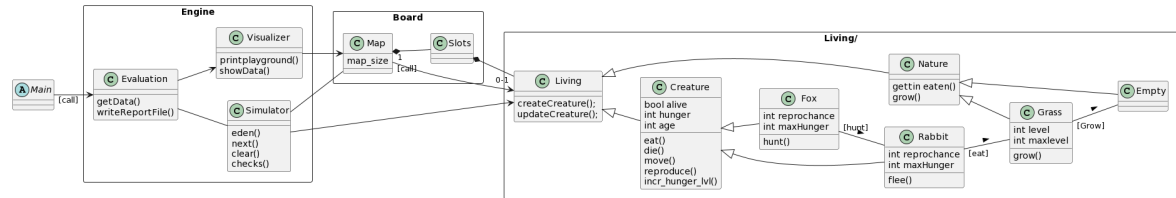


Figure 1: Klassendiagramm

What is the purpose of Nature? Should be explained in more detail.

Als erstes wurde das Klassendiagramm erstellt, welches die Struktur der einzelnen Komponenten des Programms aufzeigt und deren Interaktionen und Beziehungen untereinander. Im wesentlichen besteht das Diagramm aus den drei Komponenten Engine, Board und Living. Die Engine bildet den Kern des Programmes, da sie die ganzen Berechnungen der Simulation durchführt und die Interaktionen zwischen den verschiedenen Tieren steuert. Das Board stellt das "Spielbrett" dar, auf welchem die verschiedenen Einheiten interagieren. Die Living-Komponente enthält, welchen Zustand ein Feld des Spielbretts im Moment besitzt wie bspw. welches Tier momentan sich auf dem Feld befindet, in welche Richtung es sich bewegen wird oder wie alt es ist. Um das Diagramm nun zu lesen und verstehen werden die drei verschiedenen verwendeten Pfeilarten erklärt, da diese die Beziehungen zwischen den Kernkomponenten erläutern.

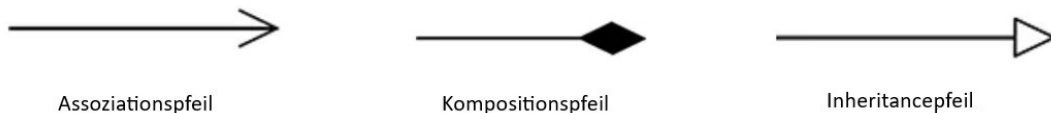


Figure 2: Pfeile Klassendiagramm

good!

Der Assoziationspfeil wird verwendet wenn zwei Klassen untereinander kommunizieren **müssen**. Er impliziert auch eine engere Verbindung zwischen den beiden Klassen. Der zweite Pfeil ist der Kompositionspfeil welcher zeigt, dass eine Klasse teil einer anderen Klasse ist oder eine andere enthält. Bspw. sieht man im Diagramm (Figure 1) beim Board, dass die Klasse Slots (Felder) Teil der Klasse Map (Karte) ist. Letztlich gibt es den Inheritancepfeil, um zu zeigen, welche Klasse von einer anderen Klasse ~~Attribute~~ erbt.

not only attributes

Prozessdiagramm

Anschliessend an das Klassendiagramm wurde das Prozessdiagramm erstellt, welches die wesentlichen Abläufe des Programms veranschaulicht. Die Schritte sind chronologisch von oben nach unten angeordnet. Zunächst startet der Benutzer die Simulation, woraufhin das System den Benutzer über den laufenden Prozess informiert. Anschliessend initiiert die Game Engine den Evaluator, der vom Benutzer die Startwerte für die verschiedenen Tierarten anfordert. Sobald diese vom Benutzer festgelegt wurden, werden die Startwerte vom Evaluator an den Simulator übermittelt. Der Simulator gibt die Startwerte wiederum an den Visualizer weiter. Als nächstes führt der Simulator die erforderlichen Iterationsschritte durch, um die Aktualisierungszyklen der Simulation zu berechnen. Die Anzahl der Aktualisierungszyklen wird sowohl an den Visualizer als auch an den Benutzer übergeben. Der Simulator liefert dann dem Evaluator die statistischen Werte, die wiederum über den Visualizer an den Benutzer weitergeleitet werden. Schliesslich beendet der Benutzer die Simulation über die Game Engine, die dem Benutzer eine abschliessende Mitteilung sendet, dass die Simulation gestoppt wurde.

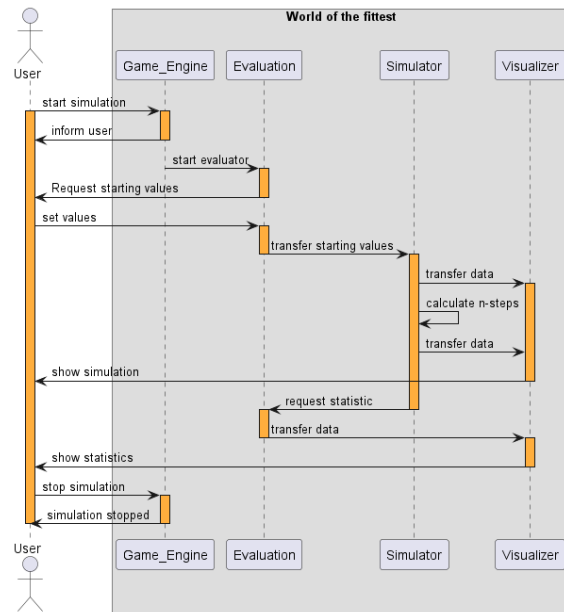


Figure 3: Prozessdiagramm

I like overview Sequence diagrams; you should, however, present a focused one (e.g., mating, killing) with the types/methods mentioned in the Class diagram.

Use Case Diagramm

Zuletzt wurde das Use Case Diagram erstellt welches die Interaktion des Benutzers mit dem Programm darstellt. Am Anfang hat der User mehrere Möglichkeiten mit dem Programm zu interagieren. Einerseits kann der Benutzer die Simulation direkt starten. Vor dem Start können aber auch die Startbedingungen geändert werden. Sobald der Start durchgeführt wurde simuliert das Programm das darausfolgende Resultat und zeigt die Ergebnisse dem User. Während der Simulation kann der Benutzer auch eingreifen indem er die Simulation stoppt oder die momentanen Daten anzeigen lässt. Als letztes kann der Benutzer auch während der Simulation mit speziellen Befehlen die Simulation beeinflussen.

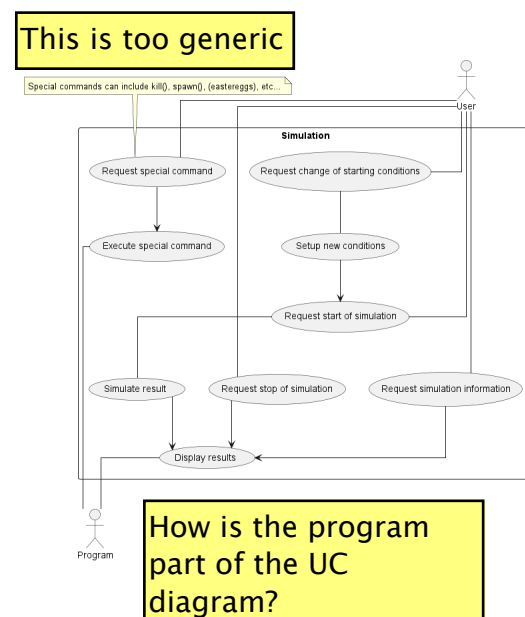


Figure 4: Use Case Diagramm

Diskussion und Fazit

Die Diagramme zeigen einen gut strukturierten Simulationsablauf, der den Datenfluss und die Interaktionen zwischen den verschiedenen Einheiten (Fuchs, Hase und Gras(Futter)) verbildlicht. Obwohl noch keine praktische Durchführung des Programmes gemacht wurde, legen die Diagramme einen umfassenden Ansatz zur Erfassung der Dynamik des simulierten Ökosystems nahe. Die Diagramme zeigen, wie die Simulation funktioniert, aber um genaue Ergebnisse zu erhalten und wirklich zu verstehen, wie das System "World of the Fittest" funktioniert, müssten erst noch praktische Tests durchgeführt werden.

List of Figures

1	Klassendiagramm	2
2	Pfeile Klassendiagramm	2
3	Prozessdiagramm	3
4	Use Case Diagramm	3