

# Real-Time Strategie Spiel mit Data-Oriented Technology Stack

Studiengang: BSc in Informatik | Vertiefung: Computer Perception and Virtual Reality  
Betreuer: Prof. Marcus Hudritsch  
Experte: Harald Studer

Ein Real-Time Strategie(RTS) Spiel beinhaltet typischerweise eine Vielzahl an Objekten auf welche mindestens ein Pathfinding-Algorithmus angewendet wird. Dies bedeutet eine enorme Anforderung an die Rechenleistung des CPUs. Anstatt der herkömmlichen Objektorientierterprogrammierung soll die Datenorientierte Programmierweise DOTS (Data-Oriented Technology Stack) von Unity diese Anforderung enorm verringern und Multithreadedcoding einfacher gestalten.

## Problemstellung

Die grosse Anzahl an Objekten in einem RTS-Spiel auf welche viele verschiedene Verhaltens-Algorithmen angewendet werden ist eine hohe Belastung für den Prozessor. Da die reine Erhöhung von CPU-Leistung an einem Endpunkt ist, ist es umso wichtiger möglichst Maschinen-effizienten Code zu schreiben. Dies beinhaltet vor allem auch Multithreaded-Code. Unity ist hier in Entwicklung des Data-Oriented Tech Stack, welcher in dieser Arbeit untersucht wird.

## Umsetzung

In diesem Projekt wurden zwei Verhaltensalgorithmen verwendet. Separationbehaviour und Arrivalbehaviour. Diese Algorithmen wurden auf drei verschiedene Arten implementiert und deren Performance verglichen. Zuerst mit der herkömmlichen Objektorientierten Methode, danach mit der dem Entity Component System(ECS) von Unity, welches der Kern der Datenorientierten Programmierung darstellt und zuletzt

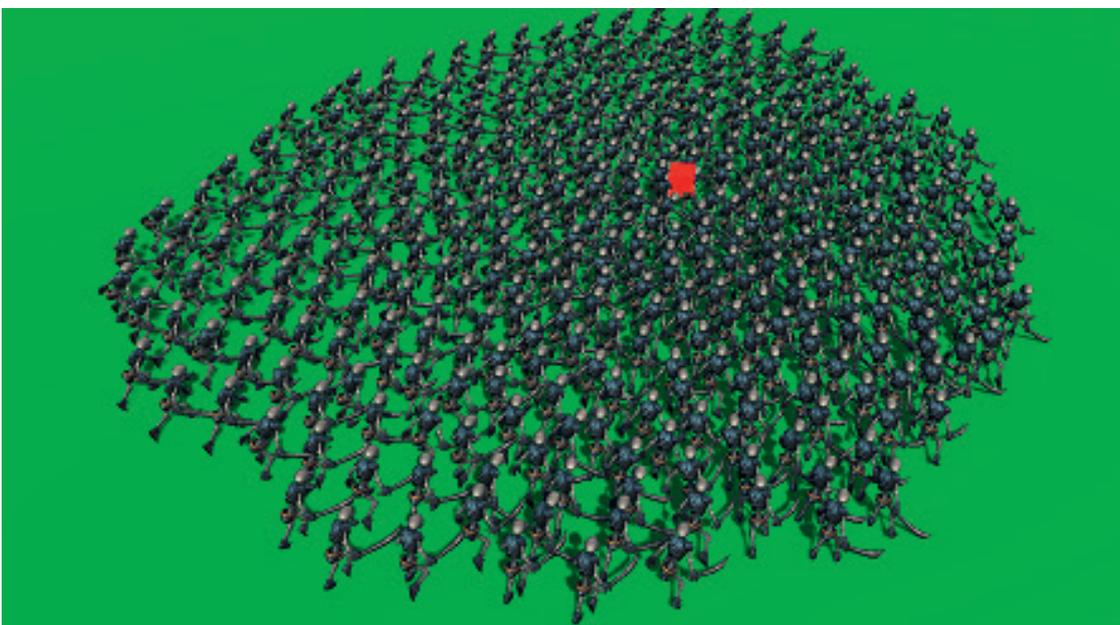
unter der Verwendung der gesamten DOTS Features welche das ECS, das C# Job System für Multithreaded Code und den Burst Compiler welcher .NET IL Code in hoch effizienten nativ Code übersetzt.

## Ergebnisse

Mit der herkömmlichen Objektorientiertenprogrammierung konnte auf einem herkömmlichen Computer eine Anzahl von 200 Objekten bei 30fps angesteuert werden. Im Vergleich mit der Verwendung der kompletten DOTS Features konnten nun statt 200 Objekten, 4500 Objekten bei 30fps angesteuert werden. Dies ist eine Performancesteigerung von 2200%.



Claudio Dano Polo



Bewegende Agents/Objekte mit rot markierter Zielposition