
Zusammenfassung

In der folgenden Arbeit wird eine nebenläufige Architektur für die 3D-Spielebibliothek Blocklib entworfen und implementiert, die von Studierenden der Ostbayerischen Technischen Hochschule Regensburg entwickelt wird.

Dazu werden Anforderungen erörtert und basierend auf den in der Spielindustrie gängigen nebenläufigen Architekturen System on a Thread und Jobsystem ein hybrides Architekturmodell entworfen, das Elemente beider Architekturen nutzt. Dieses Modell wird schließlich implementiert und verwendet bekannte Techniken wie Double-Buffering, um Wettkampfbedingungen zu verhindern. Ein Großteil der bestehenden nebenläufigen Strukturen kann durch die neue nebenläufige Architektur ersetzt werden, die nun für die Nutzung durch zukünftige Studierende bereit steht, die die Blocklib weiterentwickeln.

Die Architektur teilt die Blocklib grundlegend in zwei nebenläufige Bereiche auf, ein System zur Simulation der Inhalte der Spielebibliothek und ein System, das für das Rendering der Grafik verantwortlich ist. Des weiteren wird nun eine Schnittstelle angeboten, die von der im Java-Umfeld bereitgestellten Schnittstelle `ScheduledExecutorService` abgeleitet und somit vertraut ist, die dort gebotene Funktionalität aber um die Möglichkeit der Komposition von Jobs mittels der Schnittstelle `CompletionStage` erweitert. Auf diese Weise lassen sich komplexe Aufgaben in Jobs zerlegen, die automatisch nebenläufig ausgeführt werden.

Über eine Performanceanalyse, die die Leistung der neuen Architektur mit der alten Architektur in fünf verschiedenen Szenarien untersucht und in der Messdaten zu den Metriken Bildwiederholrate, Auslastung der CPU und GPU sowie Speicherverbrauch erhoben werden, kann eine verbesserte Performance von durchschnittlich 170 % mehr Bildern pro Sekunde ermittelt werden. Die Daten zeigen zudem, dass durch das System eine höhere Auslastung von CPU und GPU erreicht werden.

Zuletzt lassen die Messwerte darauf schließen, dass sich eine weitere Steigerung der Auslastung mit erwarteter Erhöhung der Bildwiederholrate ermöglichen lässt. Das kann erreicht werden, indem die Nutzung von Nebenläufigkeit in der Simulation der Blocklib unter Verwendung der neuen nebenläufigen verstärkt wird, da die Auslastungswerte der CPU darauf schließen lassen, dass diese weiterhin durch mangelnde Nebenläufigkeit einen Flaschenhals darstellt, der das Leistungspotenzial der Blocklib einschränkt.