



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Ausarbeitung Seminar
Ringvorlesung - SoSe 2010
Andre Goldflam

Interest Management in
Massive Multiplayer Online Games

Inhaltsverzeichnis

Abbildungsverzeichnis	2
1. Einleitung	3
2. Hintergrund	3
2.1. Entwicklung von Massive Multiplayer Online Games (MMOGs)	3
2.2. Herausforderungen bei MMOGs	4
2.3. Verteilungskonzepte von MMOGs	5
2.4. Interest Management	6
2.5. Weitere Techniken	9
3. Masterarbeit	10
3.1. Ziel	10
3.2. Vorgehen	10
3.3. Konzeption	11
3.4. Implementierung	11
3.5. Vergleichbarkeit von Algorithmen	12
3.6. Risiken	13
4. Fazit	13
Literatur	14
A. Bilder	16

Abbildungsverzeichnis

1. Verteilungskonzept Zonen (aus Schönherr (2009))	16
2. Verteilungskonzept Instanzen (aus Schönherr (2009))	16
3. Verteilungskonzept Replikation (aus Schönherr (2009))	17
4. Dead Reckoning (aus Marcel Köhler (2007))	17
5. Regionsbasiertes Interest Management (aus Smed u. a. (2002))	18

1. Einleitung

Innerhalb der Ausarbeitung des Seminars Ringvorlesung sollen die Hintergründe der Masterarbeit mit dem Thema „Vergleich von Interest Management Algorithmen“ sowie das konkrete Vorgehen bei der späteren Umsetzung der Masterarbeit beschrieben werden.

2. Hintergrund

Interest Management ist ein Technik, welche bei sogenannten Massive-Multiplayer-Online-Games (MMOGs) eingesetzt wird, um die hohe Anzahl gleichzeitiger Nutzer zu erreichen. Anschließend wird zuerst auf die historische Entwicklung, dann auf die Herausforderungen bei der Implementierung, sowie die Verteilungskonzepte von MMOGs eingegangen. Darauffolgend soll auf das Interest Management selbst und abschließend auf weitere Techniken eingegangen werden, die zusätzlich zum Interest Management eingesetzt werden, um derart hohe Nutzerzahlen bei MMOGs zu ermöglichen.

2.1. Entwicklung von Massive Multiplayer Online Games (MMOGs)

Die folgende Zeitleiste soll einen kurzen Einblick in die historische Entwicklung von MMOGs geben und verdeutlichen, wie rasant die Entwicklung in diesem Bereich fortschreitet.

- 1985 Island of Kesmai: Erstes Kommerzielles MMOG, vertrieben über den Compuserve-Dienst. Das Spiel konnte von bis zu 100 Spielern gleichzeitig gespielt werden.
- 1997 Ultima Online (UO): Der Begriff Massive Multiplayer Online Role-Playing Games (MMORPG) wurde durch den Entwickler von Ultima Online(UO), Richard Garriot, geprägt. Im Jahre 2003 erreichte UO mit 250.000 Mitgliedern die höchste Anzahl an Abonnenten ([Wikipedia](#)).
- 1999 Everquest: Ebenfalls im Bereich der Rollenspiele angesiedelt, konnte Everquest bis zu 450.000 Abonnenten für sich gewinnen. Everquest gilt als das erste, finanziell wirklich erfolgreiche MMOG.
- 2004 World of Warcraft (WoW): Mit ca. 11.500.000 Abonnenten (im November 2008 ([Blizzard](#))) das bisher erfolgreichste MMORPG. Die reinen Entwicklungskosten bis zur ersten offiziellen Freigabe für die ersten Abonnenten beliefen sich auf ca. 50 ([Spiegel](#)) Mio USD. Die vollständigen Kosten seit der Inbetriebnahme von WoW 2004 wurde von Blizzard, dem Entwickler und Publisher von Wow, auf 200 Mio USD beziffert ([Wired](#)). Die Einnahmen werden dabei auf 50 - 150 Mio USD pro Monat geschätzt.

2.2. Herausforderungen bei MMOGs

Bei der Entwicklung von MMOGs müssen verschiedene Problematiken beachtet werden, welche physikalischer oder konzeptioneller Natur sein können. Folgend sollen zuerst die physikalischen, dann die konzeptionellen Herausforderungen beschrieben werden.

Physikalische Herausforderungen: Rechenkapazität Innerhalb eines Spieles, muss die Rechenleistung auf verschiedene Bereiche aufgeteilt werden, wie z.B. Rendering der Anzeige, Physik-Effekte oder Künstliche Intelligenz. Bei einem stark vernetzten Spiel, wie einem MMOG, muss auch der Kommunikationsaufwand für das Senden und Empfangen von Nachrichten über das Netzwerk von der CPU geleistet werden. Dementsprechend wirkt sich die Kommunikationsarchitektur sowie die Masse von Informationen, die über das Netzwerk gesendet werden, nicht nur auf die Latenz, sondern auch auf die Rechenkapazität aus.

Physikalische Herausforderungen: Latenz Viele Echtzeit-Multiplayer-Spiele setzen auf eine traditionelle Client-Server-Architektur. Steigt nun die Latenz zwischen dem Server und dem Client, wird die Ausgabe auf dem Computer des Spielers verzögert. In Abhängigkeit von der Spielart (z.B. reaktionsreiches Autorennen oder ein Aufbau-Strategiespiel) stört eine steigende Latenz unterschiedlich schnell die Spielerfahrung. Nach [Jehaes u. a. \(2003\)](#) können Latenzen dabei in drei Kategorien unterteilt werden:

- Latenz, die durch den Transport über das Netzwerk entsteht
- Latenz, die durch Hard- und Software entsteht, welche für die Anzeige sowie für die Berechnung der Zustandsänderung innerhalb dieser virtuellen Welt notwendig ist.
- Latenz, die durch das Empfangen und anschließende Verarbeiten zur Aktualisierung der virtuellen Welt entsteht

Es wurden bereits viele verschiedene Techniken entwickelt, um die Latenz zu verringern (siehe Abschnitt 2.5). Dabei ist das in Abschnitt 2.4 beschriebene Interest Management eines der effizientesten Verfahren.

Konzeptionelle Herausforderungen: Skalierung Ein Anstieg der Spielerzahlen kann bei einem MMOG auch in einem erhöhten Kommunikationsaufwand resultieren, da nun der Zustand des einen Spielers an mehrere Spieler gesendet werden muss. Ebenso muss auch der Server von verschiedenen Spielern ihre Aktualisierungen der Spielwelt empfangen. Daher wird mit Hilfe verschiedener Techniken die Verteilung der Informationen sowie der Informationsgehalt selbst minimiert, wobei der Spieler in seinem Spielerlebnis nicht negativ beeinflusst werden darf. Wenn nun ein Server eine bestimmte Anzahl von Spieler bedienen kann, sollte durch

das Hinzufügen eines weiteren Servers möglichst die doppelte Anzahl von Spielern verarbeitet werden können ([Schönherr \(2009\)](#)).

Konzeptionelle Herausforderungen: Verteilte Konsistenz Jeder Spieler in der gemeinsamen Spielwelt eines MMOG besitzt nur eine lokale Sicht auf den Zustand der Welt. Dies ist auf die Latenz zwischen Client und Server zurückzuführen. Wenn nun ein Spieler eine Aktion in der Spielwelt auslöst, muss diese erst von dem ausführenden Client an den Server und wieder an alle anderen Clients übertragen werden. Durch Synchronisationsmethoden werden alle Aktionen der Spieler in eine Reihenfolge gebracht und deren Auswirkungen auf die Spielwelt berechnet. Dabei unterscheidet man konservative Synchronisierungsmethoden (z.B. Bucket-synchronisierung aus [Gautier und Diot \(1998\)](#)), welche auf einer logischen Reihenfolge der Ereignisse aufbauen und optimistischen Verfahren (z.B. Time Warp, [Jefferson u. a. \(1987\)](#)), welche die Ereignisse in einen kausalen Zusammenhang setzen.

Konzeptionelle Herausforderungen: Cheating Wie schon in Abschnitt 2.1 angedeutet, ist für die Entwicklung eines MMOGs ein zweistelliges Millionenbudget nicht ungewöhnlich. Diese Entwicklungskosten werden erst im Laufe der Zeit, in der Regel durch die Abbonierungskosten, wieder eingespielt. Sollte sich ein Spieler durch einen Programm- oder Logikfehler einen nicht eingeplanten Vorteil erswindeln, kann dies den Spielspass der anderen Abonnenten stark negativ beeinflussen. Daher sollte bei MMOGs sehr auf die Vermeidung und Aufdeckung möglicher externer Manipulationen geachtet werden.

2.3. Verteilungskonzepte von MMOGs

Zonen

Die Spielwelt wird in unterschiedliche Zonen ([Kazem u. a. \(2007\)](#)) aufgeteilt. Dabei wird jede Zone von einem anderen Server betreut und spiegelt ein bestimmtes Gebiet der Spielwelt wieder. Der verantwortliche Server ist für sämtliche Berechnungen des Spielstandes in seiner Zone zuständig. Wenn der Spieler nun von einer Zone in eine Andere wechselt, wird sein Zustand zu dem Server des Zielgebietes übertragen. In der Regel wird der Spielfluss bei so einem Übergang unterbrochen und ein Ladebildschirm erscheint. Soll der Spieler den Serverwechsel nicht wahrnehmen, werden die Grenzgebiete der verantwortlichen Server übereinander geschoben und der Zustand in der Übergangszone wird von allen angrenzenden Servern redundant verarbeitet. Dieser nahtlose Übergang ist nur auf Kosten der Skalierbarkeit anwendbar. Das Zonenkonzept findet in der Regel bei vielen MMORPGs Anwendung. Abbildung 1 im Anhang verdeutlicht das Konzept.

Instanzen

Bei Instanzen wird ein Teil der Spielwelt kopiert und ist für eine bestimmte Anzahl von Spielern zuständig. Es können also verschiedene Instanzen eines Gebietes existieren, in dem die Welt innerhalb der Instanz einen anderen Zustand besitzt (Siehe [Abbildung 2](#) im Anhang). Eingesetzt wird diese Art der Verteilung oft für kleine Gebiete, in der eine Gruppe eine besondere Aufgabe gemeinsam lösen muss. Diese Technik findet bei vielen MMORPGs Anwendung. Bei dem Einsatz als ein allgemeines Verteilungskonzept, wobei nicht nur kleine Gebiete sondern ganze Zonen mehrfach instanziiert werden, kann der Spieler aktiv die Instanz einer Zone auswählen und dieser beitreten. Diese Technik gilt eher als unbeliebt, da die Spieler wahrnehmbar voneinander getrennt werden und so das Gefühl, gemeinsam eine riesige virtuelle Welt zu bevölkern, deutlich abgeschwächt wird (vgl. [Schönherr \(2009\)](#)).

Replikation

Bei der Replikation wird der Zustand einer einzelnen Zone auf mehreren Servern vorgehalten. Dabei ist ein Server nur für eine Teilmenge der Spieler zuständig. Die anderen Spieler, auch Schattenentitäten genannt, werden durch die anderen Server aktualisiert. Für diese Spieler berechnet der Server die Zustandsänderungen der Spielwelt und übermittelt diese dann an die anderen Replikationen, welche den alten Zustand der entsprechenden Schattenentität innerhalb ihrer Replikation überschreiben. Eine Replikation muss also nur die Daten der eigenen Spieler verarbeiten, da die Änderungen an den Schattenentitäten von den anderen Replikationen verwaltet werden. Eine grafische Darstellung hierzu findet sich im Anhang; [Abbildung 3](#). Dieses Konzept ist dem der Instanzen und Zonen bei MMOGs mit kleinen Spielwelten überlegen, wie z.B. bei einem Ego-Shooter üblich, da die Spieler nicht den Server in Abhängigkeit von ihrer Position in der Spielwelt wechseln müssen. Ebenso ist eine Gleichverteilung der Last zwischen den Replikationen möglich und eine Clusterbildung von Spielern in einem Bereich unproblematisch (vgl. [Schönherr \(2009\)](#) nach [Glinka u. a. \(2008\)](#)).

2.4. Interest Management

Interest Management kann nach [Morse u. a. \(2000\)](#) als der Prozess angesehen werden, der bestimmt, ob eine Information aus der virtuellen Umgebung für einen Teilnehmer von Interesse ist oder nicht. In diesem Abschnitt sollen die Hintergründe und Methoden vorgestellt werden, wie dieser Prozess durchgeführt werden kann.

Publish-Subscribe Abstrahierung

Interest Management kann über das Publish-Subscribe-Modell abstrahiert werden, wobei Publishers Ereignisse erzeugen und Subscriber diese verarbeiten (Caltagirone u. a. (2002)). In diesem Modell würde ein Spieler beide Rollen ausfüllen, da er selbst Ereignisse erzeugt und auch die Ereignisse der Umgebung aufnimmt. Innerhalb der Publish-Subscribe Abstrahierung ist es die Aufgabe des Interest Management zu bestimmen, wann ein Subscriber die Ereignisse eines Publishers abonniert und wann diese Abonniierung gestoppt wird. Interest Management kann dabei auf verschiedene Informationsbereiche angewendet werden, wobei der sichtbare Informationsbereich der gebräuchliste ist. Weiterhin werden häufig der Bereich für Umgebungsgeräusche sowie der Bereich für Funk-Kontakt (Text- oder Sprach-Chat) genutzt.

Kategorien von Interest Management

Interest Management Techniken können grob in zwei Oberklassen eingeteilt werden; Raumbasierte und Klassenbasierte Methoden (Morse u. a. (2000)). Raumbasiertes Interest Management nutzt dabei die relative Position des Objektes innerhalb der virtuellen Welt und klassenbasiertes die Eigenschaften des Objektes, um zu bestimmen, welche Ereignisse das Objekt erhalten soll. Innerhalb von MMOGs findet überwiegend das raumbasierte Interest Management Anwendung, da der größte Teil der für den Spieler relevanten Informationen in enger Beziehung zu dessen Position innerhalb der virtuellen Welt steht. Daher soll folgend nur auf das raumbasierte Interest Management eingegangen werden.

Raumbasiertes Interest Management

Das raumbasierte Interest Management bezieht immer den direkten Umgebungsbereich eines Objektes ein und kann als Aura-Nimbus Modell (Benford und Fahlén (1993)) interpretiert werden. Die Aura stellt ein Gebiet um ein Objekt dar, in welchem das Objekt von seiner Umgebung wahrgenommen werden kann. Der Nimbus beschreibt eine Fläche um das Objekt, in dem es andere Objekte wahrnehmen kann. In seiner einfachsten Form können Aura und Nimbus durch Kreise fester Größe um die Objekte repräsentiert sind. Dabei nimmt ein Objekt X ein anderes Objekt Y wahr, wenn der Nimbus von X die Aura von Y schneidet. Der Vorteil des reinen Aura-Nimbus-Modells ist es, dass es ein sehr exaktes Interest-Management ermöglicht und den einzelnen Subscribern nur die für sie relevante Ereignisse abonnieren lässt. Der Nachteil dieses Modells ist, dass es nicht gut skaliert, da die Berechnung der Überschneidungen von Aura und Nimbus für jedes einzelne Objekt durchgeführt werden muss und in einer Komplexität von O^2 resultiert, da die Überschneidungsberechnung von und mit jedem beliebigen Objekte durchgeführt werden muss.

Regionsbasiertes Interest Management Um die Nachteile des reinen Aura-Nimbus-Modells auszugleichen wird von vielen Interest-Management-Techniken ein regionsbasierter Ansatz verfolgt (z.B. [Funkhouser \(1995\)](#)). Dabei wird die virtuelle Umgebung in festgelegte Bereiche eingeteilt, bei denen die Subscriber angemeldet werden. Dabei werden alle Bereiche abonniert, die durch den Nimbus des Subscriber geschnitten werden. Die Fläche, die durch die abonnierten Bereiche gebildet wird, ist eine Annäherung an den eigentlichen Nimbus, wobei die Berechnung des Interest Management deutlich schneller ausgeführt werden kann als bei der exakten Berechnung des reinen Aura-Nimbus-Modells (Siehe Abbildung 5 im Anhang). Die Qualität der Annäherung hängt dabei stark von Form und Größe der Bereiche ab. Die Verwendung von Rechtecken, Sechsecken oder Splines als Grundform für die Bereiche birgt dabei verschiedene Vor- und Nachteile. Regionsbasiertes Interest Management kann sehr gut über Multicast realisiert werden, wobei jedem Bereich eine Multicast-Gruppe zugeordnet ist. Leider wird Multicast in der Regel von wenigen Internet Service Providern angeboten und ist somit für eine große Masse an Spielern nicht nutzbar. Regionsbasiertes Interest Management liefert die besten Ergebnisse, wenn die Objekte gleichmäßig über alle Bereiche verteilt sind. Falls sich viele Objekte in einem großen Bereich befinden, sollte zusätzliches Load Balancing innerhalb der Bereiche die Last verteilen. Ein Ansatz hierzu findet sich in [Abrams u. a. \(1998\)](#).

Sichtbarkeitsbasiertes Interest Management Die bisher vorgestellten Methoden basierten auf einer Fläche, die für den Subscriber den Wahrnehmungsbereich darstellt. Dabei vernachlässigen sie die Geographie der virtuellen Welt. Sichtbarkeitsbasierte Techniken basieren auf der Sichtweite des Subscribers anstelle einem festgelegten Radius. [Hosseini u. a. \(2002\)](#) beschreibt einen Prozess, wie die zu abonnierenden Publisher durch den Client identifiziert werden, indem während des grafischen Renderings gleichzeitig berechnet wird, welche Publisher für den Client sichtbar ist. Hierzu benötigt der Client allerdings immer die aktuelle Position aller Publisher der virtuellen Welt. Falls die Publisher sich dabei ständig bewegen und deren Position immer aktualisiert werden müssen, ist diese Herangehensweise nicht vorteilhaft.

Alternative Kriterien für Interest Management Unabhängig von der Bestimmung, welcher Subscriber welche Ausschnitte seiner Umgebung wahrnimmt, kann nach [Han u. a. \(2000\)](#) noch die Genauigkeit der Informationen unterschieden werden. Dabei wird eine Methode vorgestellt, bei der aus Nutzern, die sich für ein ähnliches Gebiet interessieren, eine Gruppe gebildet und aus dieser ein Repräsentant gewählt wird. Dieser Repräsentant ist dabei zuständig die Informationen mit geringer Genauigkeit an die anderen Gruppenteilnehmer weiterzuleiten. So können beispielsweise weit entfernte Regionen beobachtet werden, wobei durch eine Einschränkung der Informationsdaten die notwendige Bandbreite reduziert wird.

2.5. Weitere Techniken

Interest Management ist eine der geeigneten Techniken, um die Herausforderungen von MMOGs zu bewältigen. In der Regel muss das Interest Management noch mit weiteren Techniken kombiniert werden, um ein skalierbares System zu entwickeln. Folgend sollen diese Methoden kurz vorgestellt werden.

Dead Reckoning Dead Reckoning ([Singhal und Zyda \(1999\)](#)) ist eine Technik zur Reduktion der Anzahl der Nachrichten, die über das Netzwerk ausgetauscht werden müssen. Dabei wird versucht, die Position eines Objektes innerhalb einer Spielwelt vorauszuberechnen. Wenn sich beispielsweise zwei Objekte aufeinander zubewegen, wird in einer Simulation diese Bewegung diskretisiert und üblicherweise die Framerate (z.B. 30 Frames pro Sekunde) als kleinste Zeitspanne verwendet. Um eine Kollision zu ermitteln, müsste nun in jedem Frame der Zustand der beiden Objekte übermittelt werden; also 30 Nachrichten pro Sekunde. Stellt man sich nun ein Spiel mit vielen sich bewegenden Objekten vor, wächst der Kommunikationsaufwand enorm an. Dead Reckoning berechnet dabei die Positionsänderungen lokal voraus, wobei mögliche Abweichungen bei einer späteren Zustandsänderung durch einen Konvergenzalgorithmus angeglichen werden. Dead Reckoning erlaubt dabei leicht inkonsistente Spielzustände. Eine Abbildung des Dead Reckoning-Verfahrens ist im Anhang zu finden; [Abbildung 4](#). Diese Technik findet besonders im Bereich der Sportsimulationen, wie z.B. dem Autorennen, Anwendung, da hier in der Regel keine sprunghaften Richtungsänderungen möglich sind und somit Abweichungen von dem approximiert Zustand relativ gering ausfallen.

Message Compression Wie der Name schon sagt, wird der Inhalt der Nachrichten komprimiert und so die Größe der Nachrichten verringert. Das Packen bzw. Entpacken der Informationen reduziert auf Kosten von Rechenleistung die notwendige Bandbreite.

Message Aggregation Message Aggregation ([Smed u. a. \(2001\)](#)) reduziert die Frequenz, in der Zustandsänderungen versendet werden, indem die einzelnen Nachrichten in einer zusammengefassten, größeren Nachricht geschickt werden. Dabei werden redundante Informationen (wie z.B. die Header der einzelnen Pakete) eingespart. Auf der anderen Seite wirkt sich die Zusammenfassung von Nachrichten negativ auf die Latenz aus, da Nachrichten mit einer Verzögerung über das Netzwerk gesendet werden.

3. Masterarbeit

Im folgenden Kapitel soll der Rahmen der Masterarbeit vorgestellt werden. Dabei soll auf das Ziel der Masterarbeit, das Vorgehen zu Erreichung dieses Zieles, sowie potenzielle Risiken, die das beschriebene Vorgehen verhindern können, eingegangen werden.

3.1. Ziel

Als Resultat des praktischen Teils der Masterarbeit soll eine Komponente entstehen, welche Spiele-Entwickler bei der Umsetzung von MMOGs in Hinblick auf den Einsatz von Interest-Management-Techniken unterstützt. Da die einzusetzenden Interest Management Algorithmen voraussichtlicher Weise nicht universell, also auf jedes beliebige MMOG anwendbar sind, soll ein Vergleich von verschiedenen Algorithmen erfolgen. Der Vergleich dieser Algorithmen soll unter Berücksichtigung möglicher verfälschender Einflüsse durchgeführt werden, um ein aussagekräftiges Ergebnis zu erhalten.

Da eine Interest-Management-Komponente nur ein Teil dessen ist, was für eine Middleware für MMOG-Spiele-Entwickler notwendig ist, soll auf einer bereits in der Wirtschaft erfolgreich eingesetzten Lösung aufgebaut werden. Im Rahmen der Masterarbeit soll dabei zusammen mit der Firma ExitGames¹ auf Basis ihres Photon-Frameworks gearbeitet werden. Eine vorbereitende Zusammenarbeit wurde bereits im Rahmen des Master-Projekts 2 durchgeführt. Das Photon-Framework mit dem darunterliegenden C / C++ Socket-Server garantiert eine performante Kommunikation. Die Nebenläufigkeit der Serverkomponente wird über RetLang realisiert. Retlang ist eine auf dem Aktor-Modell basierende Threading-Bibliothek für C#. ExitGames setzt das Photon-Framework bereits für verschiedene MMOGs ein. Ein Beispiel hierzu ist Paradise Paintball, welches von über 500.000 Spielern genutzt wird. Die Verwendung des Photon-Frameworks ermöglicht es, sich mit dem Kernthema, dem Interest Management, zu beschäftigen.

3.2. Vorgehen

Das Vorgehen wird in drei unterschiedliche Bereiche gegliedert: Recherche, Konzeption und Implementierung. Diese Bereiche sollen folgend näher beschrieben.

¹<http://www.exitgames.com/>

Recherche

Neben den wissenschaftlichen Recherchen im Bereich des Interest Managements und den virtuellen Umgebungen soll der Hintergrund sowie die Entwicklungen im Bereich der MMOGs recherchiert werden. Während der Durchführung dieser Recherche, soll der Schwerpunkt auf den technischen Herausforderungen der MMOGs sowie deren Lösungen gelegt werden. Zudem soll der Stand der Technik von aktuellen MMOGs analysiert werden.

3.3. Konzeption

Die konzeptionelle Phase soll wie folgt durchgeführt werden:

1. Klassifizierung der häufigsten MMOG-Spieltypen
2. Klassifizierung von Interest Management Algorithmen
3. Begründete Auswahl eines oder mehrerer Spieltypen
4. Begründete Auswahl von zu den Spieltypen passenden Interest Management Algorithmen
5. Ausarbeitung unterschiedlicher Test-Szenarien

Die verschiedenen Test-Szenarien sollen die Stärken und Schwächen der Interest Management Algorithmen hervorheben. Ein mögliches Szenario könnte sein, dass mehrere hundert Spieler relativ unabhängig voneinander ein Gebiet erkunden und dann nach einiger Zeit zu einem bestimmten Punkt gezogen werden. Ein anderes Szenario könnte die Bildung von Spieler-Gruppen sein, z.B. 30 Spieler pro Gruppe, wobei manche durch das Gebiet ziehen und andere relativ fest in einem Teilgebiet bleiben.

3.4. Implementierung

Die Implementierung soll, wie Anfangs beschrieben, in Kooperation mit ExitGames auf Basis des Photon-Frameworks durchgeführt werden. Im Rahmen des Master Projekt 2 werden die Details des Photon-Frameworks vorgestellt, daher soll an dieser Stelle nicht näher auf dessen Aufbau eingegangen werden. Die Implementierung soll die folgenden Punkte umfassen:

- Umsetzung eines allgemeinen Interest Management-Frameworks als Entwicklungsgrundlage für alle zu vergleichenden Algorithmen
- Implementierung der Interest Management Algorithmen

- Aufzeichnung von Spielerverhalten
- Durchführung von Tests anhand der aufgezeichneten Spieldaten

Wenn möglich soll das Spielerverhalten nicht durch Bots, sondern durch menschliche Spieler generiert werden.

3.5. Vergleichbarkeit von Algorithmen

Um eine möglichst hohe Vergleichbarkeit unter den Algorithmen zu erreichen, sollen die einzelnen Clients, welche das aufgezeichnete Spielverhalten wiedergeben, in einer kontrollierten Umgebung installiert und gestartet werden. Seitens ExitGames wurden bereits Photon-Server auf der Amazon EC2-Cloud erfolgreich gehostet. Daher liegt es nahe, dass die Clients auch auf der Amazon EC2-Cloud gehostet werden sollen, um mögliche verfälschende Hardware- und Netzwerk-Einflüsse zu minimieren.

Da im Rahmen des Master Projekt 2 bereits Erfahrung mit Last-Tests in Bezug auf das Photon-Framework gewonnen werden konnten, soll an dieser Stelle eine erste vorläufige Liste von möglichen Performance-Indikatoren für den Vergleich von Interest Management Algorithmen vorgestellt werden. Dabei wird unter Hardware, Photon-Server und Interest Management Algorithmen unterschieden. Die Indikatoren zum Vergleich von Interest Management Algorithmen wurden in Anlehnung an die in [Boulanger u. a. \(2006\)](#) durchgeführten Tests gewählt.

- Hardware
 - Prozessorzeit
 - Zugesicherter Arbeitsspeicher in Bytes
- Photon-Server
 - Total-Peers
 - Total Commands in / sec
 - Total Commands out / sec
 - Commands Resent
 - Reliable Commands dropped / sec
 - Unreliable Commands dropped / sec
- Interest Management Algorithmen

- Inhalts-Nachrichten
- Aktualisierungs-Nachrichten

Die endgültigen Performance-Indikatoren können allerdings erst im Rahmen der Masterarbeit erarbeitet werden.

3.6. Risiken

Im Rahmen der Recherche ist davon auszugehen, dass aktuelle Technologien im Bereich der MMOGs als Firmengeheimnisse geschützt werden. Daher ist es unklar, wie erfolgreich die Nachforschungen in diesem Bereich sein werden. Daher wird das Risiko der Recherche von „State of the Art“-Technologien als hoch eingestuft.

Die Auswahl des Spieltyps während der Konzeptionierungsphase könnte möglicherweise nicht mit der Auswahl des Algorithmus und den Test-Szenarien zusammen passen. Dieses Risiko ist allerdings als gering einzustufen, da bereits auf Erfahrungen Dritter aufgebaut werden kann.

Bei der Implementierungsphase ist es ein potenzielles Risiko, dass die Entwicklung des Interest Management-Frameworks auf Basis des Photon-Framework nicht wie erwartet umzusetzen ist. Da bereits Erfahrungen mit Photon vorhanden sind, wird dieses Risiko als gering eingestuft. Weiterhin ist es möglich, dass die Amazon EC2-Cloud nicht das einheitliche Umfeld für die Testversuche bietet, wie vorher angenommen. Da auch bei ExitGames bisher noch keine Erfahrung mit einer Simulation dieser Größenordnung gesammelt worden ist, wird dieses Risiko als mittel eingestuft. Ebenso gilt das Aufzeichnen des Spielerverhaltens als ein mittleres Risiko, da es noch nicht geklärt ist, wie diese Daten erhoben werden sollen. Da das Aufzeichnen der Daten selbst auf dem Server durchgeführt werden kann, stellt dies kein grundlegendes Hindernis dar.

4. Fazit

In dieser im Rahmen des Seminars Ringvorlesung erstellten Ausarbeitung wurden zuerst die Hintergründe der Masterarbeit mit dem Thema „Vergleich von Interest Management Algorithmen“ vorgestellt. Anschließend wurde die Art und Weise der Umsetzung der Arbeit beschrieben. Anhand der dargelegten Vorgehensweise lässt sich auf eine zukünftig erfolgreiche Realisierung der Masterarbeit schließen.

Literatur

- [Abrams u. a. 1998] ABRAMS, Howard ; WATSEN, Kent ; ZYDA, Michael: Three-tiered interest management for large-scale virtual environments. In: *VRST '98: Proceedings of the ACM symposium on Virtual reality software and technology*. New York, NY, USA : ACM, 1998, S. 125–129. – ISBN 1-58113-019-8
- [Benford und Fahlén 1993] BENFORD, Steve ; FAHLÉN, Lennart: A spatial model of interaction in large virtual environments. In: *ECSCW'93: Proceedings of the third conference on European Conference on Computer-Supported Cooperative Work*. Norwell, MA, USA : Kluwer Academic Publishers, 1993, S. 109–124. – ISBN 0-7923-2447-1
- [Blizzard] BLIZZARD: *WORLD OF WARCRAFT® SUBSCRIBER BASE REACHES 11.5 MILLION WORLDWIDE*. – URL <http://us.blizzard.com/en-us/company/press/pressreleases.html?081121>
- [Boulanger u. a. 2006] BOULANGER, Jean-Sébastien ; KIENZLE, Jörg ; VERBRUGGE, Clark: Comparing interest management algorithms for massively multiplayer games. In: *NetGames '06: Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games*. New York, NY, USA : ACM, 2006, S. 6. – ISBN 1-59593-589-4
- [Caltagirone u. a. 2002] CALTAGIRONE, Sergio ; KEYS, Matthew ; SCHLIEF, Bryan ; WILLS-HIRE, Mary J.: Architecture for a massively multiplayer online role playing game engine. In: *J. Comput. Small Coll.* 18 (2002), Nr. 2, S. 105–116. – ISSN 1937-4771
- [Funkhouser 1995] FUNKHOUSER, Thomas A.: RING: a client-server system for multi-user virtual environments. In: *I3D '95: Proceedings of the 1995 symposium on Interactive 3D graphics*. New York, NY, USA : ACM, 1995, S. 85–ff.. – ISBN 0-89791-736-7
- [Gautier und Diot 1998] GAUTIER, L. ; DIOT, C.: Design and Evaluation of MiMaze, a Multi-Player Game on the Internet. In: *ICMCS '98: Proceedings of the IEEE International Conference on Multimedia Computing and Systems*. Washington, DC, USA : IEEE Computer Society, 1998, S. 233. – ISBN 0-8186-8557-3
- [Glinka u. a. 2008] GLINKA, Frank ; PLOSS, Alexander ; GORLATCH, Sergei ; MÜLLER-IDEN, Jens: High-level development of multiserver online games. In: *Int. J. Comput. Games Technol.* 2008 (2008), S. 1–16. – ISSN 1687-7047
- [Han u. a. 2000] HAN, Seunghyun ; LIM, Mingyu ; LEE, Dongman: Scalable interest management using interest group based filtering for large networked virtual environments. In: *VRST '00: Proceedings of the ACM symposium on Virtual reality software and technology*. New York, NY, USA : ACM, 2000, S. 103–108. – ISBN 1-58113-316-2

- [Hosseini u. a. 2002] HOSSEINI, Mojtaba ; PETTIFER, Steve ; GEORGANAS, Nicolas D.: Visibility-based interest management in collaborative virtual environments. In: *CVE '02: Proceedings of the 4th international conference on Collaborative virtual environments*. New York, NY, USA : ACM, 2002, S. 143–144. – ISBN 1-58113-489-4
- [Jefferson u. a. 1987] JEFFERSON, D. ; BECKMAN, B. ; WIELAND, F. ; BLUME, L. ; DILORETO, M.: Time warp operating system. In: *SOSP '87: Proceedings of the eleventh ACM Symposium on Operating systems principles*. New York, NY, USA : ACM, 1987, S. 77–93. – ISBN 0-89791-242-X
- [Jehaes u. a. 2003] JEHAES, Tom ; DE VLEESCHAUWER, Danny ; COPPENS, Toon ; VAN DOORSELAER, Bart ; DECKERS, Eva ; NAUDTS, W. ; SPRUYT, K. ; SMETS, R.: Access network delay in networked games. In: *NetGames '03: Proceedings of the 2nd workshop on Network and system support for games*. New York, NY, USA : ACM, 2003, S. 63–71. – ISBN 1-58113-734-6
- [Kazem u. a. 2007] KAZEM, Ihab ; AHMED, Dewan T. ; SHIRMOHAMMADI, Shervin: A zone based architecture for massively multi-user simulations. In: *SpringSim '07: Proceedings of the 2007 spring simulation multiconference*. San Diego, CA, USA : Society for Computer Simulation International, 2007, S. 149–156. – ISBN 1-56555-312-8
- [Marcel Köhler 2007] MARCEL KÖHLER, Andreas P.: *Weiterentwicklung einer Echtzeit-Kommunikationsbibliothek für handybasierte Multiplayerspiele*. 2007
- [Morse u. a. 2000] MORSE, Katherine L. ; BIC, Lubomir ; DILLEN COURT, Michael: Interest Management in Large-Scale Virtual Environments. In: *Presence: Teleoper. Virtual Environ.* 9 (2000), Nr. 1, S. 52–68. – ISSN 1054-7460
- [Schönherr 2009] SCHÖNHERR, Jan: *Message-Passing-Modell für die Entwicklung eines hochskalierbaren, multithreaded Gameservers*, Hamburg University of Applied Sciences, Diplomarbeit, 2009
- [Singhal und Zyda 1999] SINGHAL, Sandeep ; ZYDA, Michael: *Networked virtual environments: design and implementation*. New York, NY, USA : ACM Press/Addison-Wesley Publishing Co., 1999. – ISBN 0-201-32557-8
- [Smed u. a. 2001] SMED, Jouni ; KAUKORANTA, Timo ; HAKONEN, Harri: *Aspects of Networking in Multiplayer Computer Games*. 2001
- [Smed u. a. 2002] SMED, Jouni ; KAUKORANTA, Timo ; HAKONEN, Harri: A Review on Networking and Multiplayer Computer Games. In: *Turku Centre for Computer Science TUCS Technical Report No 454* (2002)
- [Spiegel] SPIEGEL: *Die Kapitalisten von Kalimdor*. – URL <http://www.spiegel.de/spiegel/0,1518,363363,00.html>

[Wikipedia] WIKIPEDIA: *History of massively multiplayer online games.* – URL http://en.wikipedia.org/wiki/History_of_massively_multiplayer_online_games

[Wired] WIRED: *Total Operating Costs for WoW: USD 200 Million.* – URL <http://www.wired.com/gamelifelife/2008/09/total-operating/>

A. Bilder

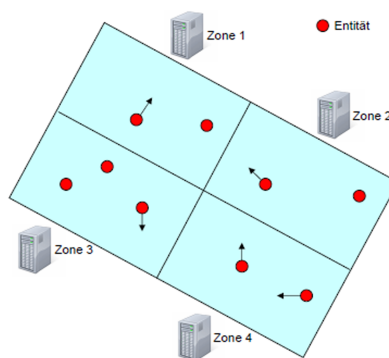


Abbildung 1: Verteilungskonzept Zonen (aus Schönherr (2009))

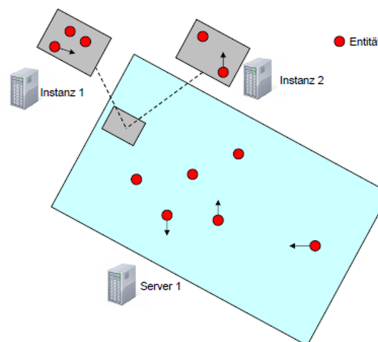


Abbildung 2: Verteilungskonzept Instanzen (aus Schönherr (2009))

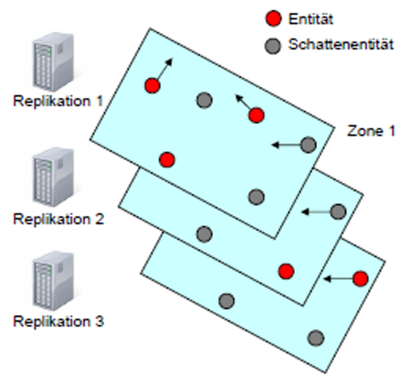


Abbildung 3: Verteilungskonzept Replikation (aus Schönherr (2009))

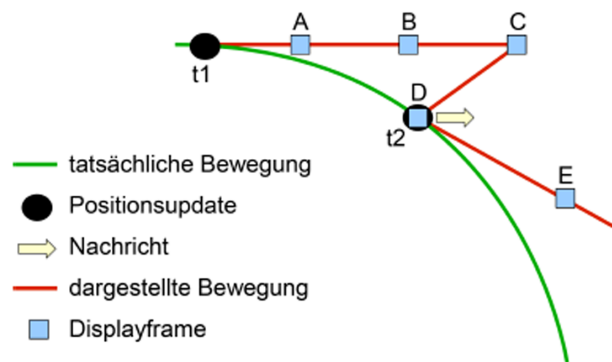


Abbildung 4: Dead Reckoning (aus Marcel Köhler (2007))

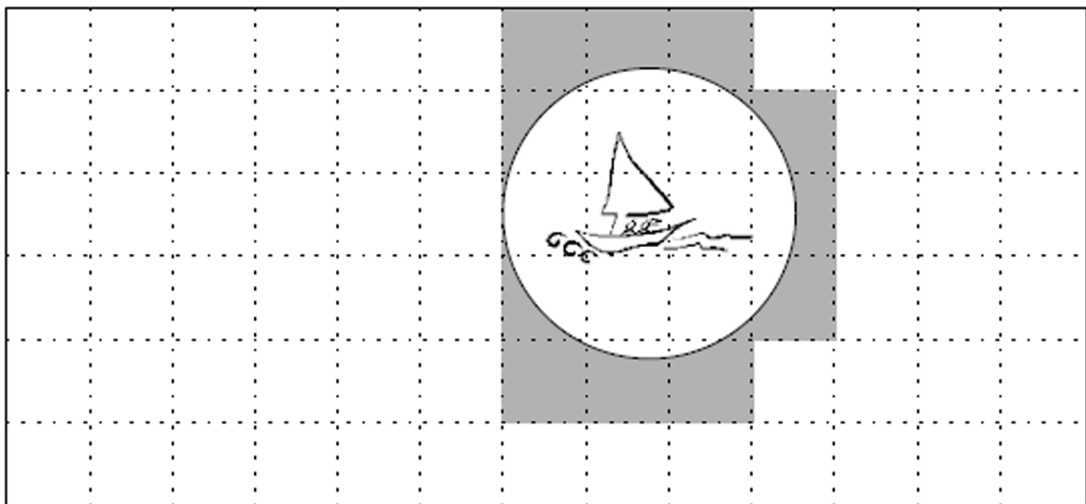


Abbildung 5: Regionsbasiertes Interest Management (aus [Smed u. a. \(2002\)](#))