

HiQoS: High-Performance-Multimedia-Dienste mit Quality-of-Service-Garantien

P.Aldenbernd¹, F.Cortés², M.Holch³, J.Jensch⁴, O.Michel⁶, C.Moar⁵, T.Prill⁶, R.Lüling², K.Morisse⁴, I.Neumann⁷, T.Plachetka², M.Reith³, O.Schmidt², A.Schmitt⁵, A.Wabro⁸

Kurzbeschreibung: Hochperformante netzbasierte Multimedia-Dienste stehen im Zentrum der Aktivitäten des HiQoS-Projektes. Darin enthalten ist einerseits das Rendering in photo-realistischer Qualität sowie andererseits die Auslieferung von kontinuierlichen Medien mit garantiertem Quality-of-Service.

Keywords: HPC, Rendering, Multimedia, Quality-of-Service, Real-Time

1 EINLEITUNG

Ziel des HiQoS-Projektes ist die Realisierung von Plattformen zur Entwicklung hochperformanter netzbasierter Multimedia-Dienste, die prototypische Entwicklung dieser Dienste, sowie deren Evaluierung in konkreten industriellen Anwendungen. Die Dienste zeichnen sich vor allem dadurch aus, daß sie hochqualitative kontinuierliche Medien integrieren, photorealistische Animationen erlauben und auf standardisierten Internet-Technologien aufbauen. Zur Realisierung dieser Dienstleistungen werden in HiQoS insbesondere die zwei folgenden technischen Ansätze vertieft:

1. das Rendering von photorealistischen Animationen auf Parallelrechnern,
2. die Einhaltung von "Quality-of-Service-(QoS)"-Garantien in einem Netzwerk von Multimedia-Servern.

Das HiQoS-Konsortium umfaßt folgende Partner (inkl. Unterauftragnehmer) und deren Aufgaben:

- **Siemens AG (IC C-LAB + Corporate Technology):** Konsortialführung, Media-Server, QoS-Kommunikation

¹ Siemens IC C-LAB, Paderborn

² Universität GHS Paderborn

³ APE Ptacek Engineering GmbH, München

⁴ Axcent Paral. Informationstech. GmbH, Paderborn

⁵ Siemens ZT, München

⁶ Pixelpark Multimedia Agentur, Berlin

⁷ UPSTART! Filmproduktion GmbH, Wiesbaden

⁸ IEZ AG, Bensheim

- **Universität Paderborn:** Netzwerke von Media-Servern, Rendering-Server
- **Pixelpark Multimedia Agentur GmbH:** Anwendung von Media-Servern in der Multimedia-Produktion
- **Axcent Parallele Informationstechnik GmbH:** Netzwerke von Rendering-Servern, Netzwerke von Media-Servern
- **APE Ptacek Engineering GmbH:** Anwendung von Media-Servern in der Multimedia-Produktion
- **IEZ AG:** Anwendung von Rendering-Servern in der Architektur
- **UPSTART! Filmproduktion GmbH:** Anwendung Rendering-Servern in Filmproduktion

In den folgenden Abschnitten wird zunächst das HiQoS-Gesamtszenario (Abschnitt 2) erklärt, bevor auf die beiden o.g. Schwerpunkte in den Abschnitten 3 und 4 eingegangen wird. In Abschnitt 5 werden die in HiQoS enthaltenen Anwendungsszenarien erläutert und in Abschnitt 6 der Status des Projekts dargestellt.

2 DAS HIQOS-GESAMTSZENARIO

Das HiQoS-Gesamtszenario ergibt sich durch die Verbindung von Rendering-Services mit Multimedia-Servern. Dabei können Rendering-Aufträge vom Client aus an einen Rendering-Server über das öffentliche Internet vergeben werden. Ist das Rendering erfolgt, wird der entsprechende Datenstrom auf dem nächsten Multimedia-Server abgelegt. Dort kann das Ergebnis, wie gewöhnliche andere Datenströme z.B. bei Video-on-Demand (VoD), vom Client abgerufen werden. Bei einem Multimedia- bzw. Rendering-Server kann es sich u.U. um denselben Rechner handeln, bzw. können in einem Subnetz (Intranet) sowohl Rendering- als auch Multimedia-Server vorkommen. QoS-Garantien können nur innerhalb von Intranets gegeben werden, da realistischerweise gegenwärtig nur dort Bandweitenreservierungen möglich sind („QoS-Inseln“). Um also hochqualitatives Realzeit-Video und -Audio zu ermöglichen, müssen entsprechende Beiträge immer erst im lokalen Intranet des Clients vorliegen. Dazu werden in

HiQoS neuartige Caching/Mirroring-Techniken zum Einsatz gebracht.

3 RENDERING IN HIQOS

Zum Zwecke der Evaluierung stellt die Universität Paderborn den Zugang zu einem leistungsfähigen Parallelrechner zur Verfügung, auf den über das Internet zugegriffen werden kann. Auf diesem Parallelrechner wird der HiQoS-Rendering-Service sowohl für Einzelbilder als auch für Filme / Animationen ausgeführt.

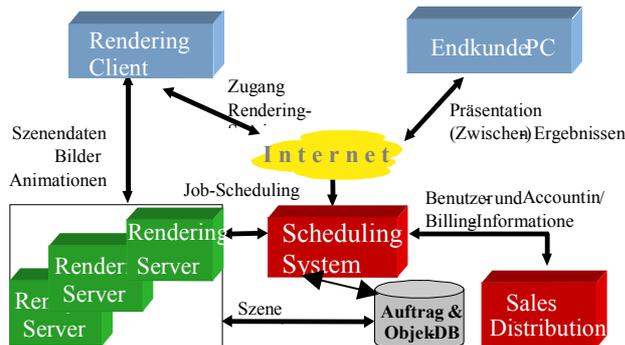


Abbildung 1: HiQoS-Rendering-Szenario

Das Rendering-System (siehe Abbildung 1) besteht aus den vier wesentlichen Komponenten Client-System, Service-Broker, Scheduling-System und Rendering-Netzwerk [7].

Bei dem **Client-System** handelt es sich um ein Konfigurations-Tool für den Anwender zur Erstellung von 3D-Modellen. Hier kommen insbesondere die Tools 3D Studio Max und Speedikon (von IEZ) zum Einsatz. Die Übergabe an den Rendering-Dienst erfolgt über ein IP-basiertes Netzwerk.

Der **Service-Broker** ist die Schnittstelle zwischen Client-Systemen und dem Rendering-Service. Seine Aufgabe liegt in der Administration des Rendering-Dienstes, z.B. die Validierung der Zugriffsberechtigung eines Benutzers. Außerdem fordert der Service-Broker beim Scheduling-System Ressourcen für die Bearbeitung eines Rendering-Auftrags an.

Das **Scheduling-System** übernimmt die Verwaltung des Rendering-Netzwerks, die Auftragsbearbeitung, das Job-Scheduling und die Ressourcen-Reservierung. Aus Gründen der Skalierbarkeit und zur Verwaltung von Rendering-Servern aus unterschiedlichen Rechenzentren besteht das System aus globalen und lokalen Schedulingern, die räumlich getrennt sein können. Ein lokaler Scheduler verwaltet alle Rendering-Server eines Rechenzentrums, während der globale Scheduler das gesamte Rendering-Netzwerk verwaltet.

Es ist geplant, ein ganzes **Netzwerk von Rendering-Servern** zur Verfügung zu stellen. Für die Bearbeitung eines Rendering-Auftrags wird

jeweils eine Menge von Rendering-Servern vom Scheduling-System ausgewählt. Es werden effiziente parallele Methoden zur globalen Beleuchtungsberechnung (Raytracing, Radiosity, TwoPass) verwendet.

Der geplante Rendering-Service konnte bereits als Prototyp mit Daten aus der Architektur (IEZ) und Filmproduktion (UPSTART!) getestet werden.

4 NETZWERKE VON MULTIMEDIA-SERVERN

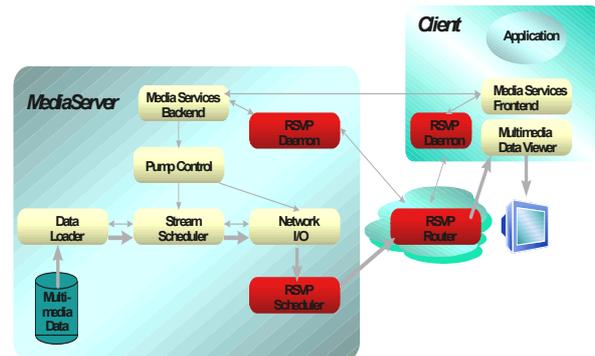


Abbildung 2: HiQoS-Media-Server

In einem heterogenen Netzwerk, das neben den multimedialen Daten auch andere Daten transportieren soll, müssen Mechanismen für die Einhaltung von Quality-of-Service (QoS) umgesetzt werden. Dieser Ansatz wird in HiQoS verfolgt und stützt sich auf RSVP (Resource Reservation Protocol) (vgl. Abbildung 2). RSVP erlaubt es, Ressourcen für Multicast oder Unicast Datenströme zu reservieren. Es handelt sich um ein Signalisierungsprotokoll, das dazu dient, auf jeder Komponente entlang eines Netzwerkpfades Zustandsinformationen über reservierte Ressourcen zu speichern.

Bei der HiQoS-Media-Server-Architektur [2] geschieht der Verbindungsaufbau zwischen Client und Server über einen erweiterten Mechanismus, bei dem Reservierungsnachrichten zwischen den RSVP-Daemons der beiden beteiligten Rechner ausgetauscht werden. Wenn neben Client und Server auch alle beteiligten Router einem bestimmten QoS zugesagt haben (admission control), so bedeutet dies, daß sie alle für die Verarbeitung noch genügend Ressourcen besitzen. Mit dieser Architektur können sogar Teilnehmer bedient werden, die keine Reservierungsanforderungen stellen, denn die Router verfügen über priorisierte Warteschlangen, bei denen Nicht-QoS-Daten entsprechend gering priorisiert werden.

In HiQoS erfolgt die erste Kommunikation mit dem Client auf Basis von RTSP (Real-Time Streaming Protocol). Sobald ein multimedialer Datenstrom angefordert wird, erhält der Client über RTSP die

nötige Ressourcen-Information. Nach erfolgter Reservierung über das RSVP-Protokoll wird über RTSP der Datenstrom gesteuert, die Datenpakete über RTP (Real-Time Transport Protocol) verschickt und Statusinformation mit RTCP (Real-Time Control Protocol) aufgefangen.

Im Rahmen eines lokalen Testnetzwerks soll in HiQoS ein neuartiger RSVP-Traffic-Scheduler (im Router) geschaffen werden. Hierbei sollen durch den Einsatz von klassischer Realzeit-Theorie [8] Reservierungen sinnvoll an die Netzauslastung angepaßt werden (bei gleichzeitiger QoS-Garantie).

Eine besondere hoch-performante Architektur läßt sich durch ein Media-Server-Cluster erreichen. Dabei werden unterschiedliche Aufgaben eines Servers von getrennten Prozessor-Einheiten bearbeitet. Diese Aufgabenteilung läßt sich mit einer SCI-Bus (scalable coherent interface) Architektur auf Basis von SMP-Rechnern, Multithreading und Shared-Memory kostengünstig und skalierbarer realisieren.

Eine wichtige Problemstellung des HiQoS-Projektes ist die Entwicklung von Algorithmen zum effizienten Management verteilter skalierbarer Netzwerke von Media-Servern [1]. Derartige Server-Netzwerke werden relevant, falls breitbandige Datenströme (Audio und Video hoher Qualität) an eine große Anzahl weit verteilter Client-Systeme ausgeliefert werden sollen.

Zahlreiche Anwendungsszenarien verlangen nach einer Live-Capturing-Lösung. Die Integration in den HiQoS-Media-Server erfolgt über eine Frame-Grabber-Karte, an die eine Kamera angeschlossen ist. Die digitalisierten Daten werden online nach MPEG1 komprimiert. Der Hardware-Encoder wird über das vom Hersteller zur Verfügung gestellte API direkt angesprochen und die Daten über einen auf PC-Plattform portierten Media-Server ins Netz gespeist.

5 ANWENDUNGEN IN HIQOS

Im Rahmen von HiQoS werden eine Reihe von Prototypen multimedialer Dienste entwickelt und in praktischen Anwendungen getestet. Dieses Vorgehen sichert einerseits eine bedarfsgerechte Entwicklung der Technologien, andererseits erlaubt es einen kurzen Migrationspfad hin zu einer späteren Produktentwicklung.

5.1 Computer-Based Training und Business TV

Computer-Based Training (CBT) und Business TV (BTV) (oder auch Distance Learning) sind in vielen Firmen wesentliche Anwendungsszenarien für Multimedia-Dienste mit QoS-Anforderungen. Sie

dienen der Mitarbeiterschulung, der Einführung in neue Produkte und der allgemeinen Weiterbildung. Jedoch ist die fehlende Unterstützung des Transports kontinuierlicher Medien ein wesentliches Hindernis für den Einsatz von CBT-Systemen in Intranets. Daher ist die Integration von kontinuierlichen Medien in CBT-Anwendungen heute nur selten anzutreffen.

Beim Projektpartner Pixelpark befindet sich ein entsprechender Trial zur Umsetzung von QoS gerade im Aufbau. Die dortige „Pixelvision“, dient den eigenen Mitarbeitern zu Schulungs- und Informationszwecken [4].

Bei APE soll in einer späteren Phase das HiQoS-Szenario eingesetzt werden, um im Rahmen von „Car TV“ in Autohausfilialen Werbefilme und Produktinformation auszustrahlen.

In Zusammenarbeit mit dem Paderborner Lehrstuhl für Didaktik der Informatik, Prof. Magenheimer wird auf Basis von HiQoS-Technologie eine Lehrerausbildung angeboten. Die Inhalte der Lehrerausbildung sind Teil des dortigen Projektes „ViLM“, dessen technische Umsetzung auf den Methoden von HiQoS beruhen wird [6]. Die Anwendung wird zunächst auf ein lokales Netz beschränkt sein, dann Campus-übergreifend arbeiten und schließlich überregional vertreten sein.

5.2 Rendering in der Architektur und der Filmproduktion

Das HiQoS-Rendering-Szenario wird in zwei verschiedenen Anwendungsbereichen evaluiert [3]. Zum einen wird dies bei der IEZ AG zu Präsentationszwecken von Bauprojekten gegenüber Kunden und Bauträgern eingesetzt. Als ein Anwendungsbeispiel wird die IEZ AG eine Rekonstruktion der 1933 niedergebrannten Mülheimer Synagoge beisteuern.

Zum anderen benutzt UPSTART! HiQoS für den Bereich Visual-Effects bei Filmproduktionen. Die Anforderungen an den Rendering-Service sind in diesen zwei Bereichen unterschiedlich. Für die IEZ AG ist die Berechnung fertiger Bilder oder Animationen mittels Raytracing interessant. Bei UPSTART! liegt das Hauptinteresse bei der Berechnung der globalen Beleuchtung in komplexen Szenen basierend auf der Radiosity-Methode. Die Beleuchtungsinformationen bilden die Grundlage für weitere Bearbeitungsschritte bei photorealistischen Animationserzeugung.

IEZ und UPSTART! können von einem hochwertigen parallelen Rendering-Server profitieren, sowohl was den qualitativen Anspruch an Bildqualität betrifft, als auch bei komplexen 3D-Szenen durch die Rechenleistung eines solchen Servers. In beiden Bereichen können damit Dinge visualisiert werden, die in-house aus Gründen der

Komplexität (große Datenmengen, lange Berechnungszeiten) nicht möglich wären.

5.3 Live-Streaming Demonstrator

Mithilfe einer Beispiel-Anwendung [5] aus dem Bereich der eingebetteten Systeme soll gezeigt werden, daß die in HiQoS entwickelten Verfahren zur QoS-gerechten Video-on-Demand-Abwicklung auch für Live-Streaming geeignet sind. Dazu wird eine Anzahl von mit Kameras versehenen Miniatur-Fahrzeugen von Benutzern über das QoS-Netz gesteuert. Die so gelieferten Video-Streams werden in einen Server eingespeist, durch den man Zugang zum HiQoS-Netzwerk hat.

Die mechanische Steuerung erfolgt benutzerseitig über ein Java-Applet, das gleichzeitig mit den Videobildern im Browser dargestellt wird. Die im Applet generierten Steuersignale werden über einen Steuer-Server und eine Funkübertragungsstrecke dem Fahrzeug übermittelt und dort ausgewertet. In diesem Trial gilt es, dem Benutzer unter zeitlichen Gesichtspunkten eine vernünftige Steuerung des Fahrzeugs zu gewährleisten.

6 STATUS DES PROJEKTS

Beim Rendering ist die Anforderungsanalyse bzgl. der Endanwender abgeschlossen, so daß das beschriebene Szenario spezifiziert werden konnte. Der Prototyp eines Services für ein Raytracing-Verfahren konnte bereits auf der Basis eines SCI-Clusters implementiert werden. Dabei werden die Standard-Fileformate VRML 2.0 und 3DS unterstützt.

Die nun folgenden Arbeiten werden zunächst die Integration der Architektensoftware Speedikon vom Projektpartner IEZ beinhalten. Auf Seiten des eigentlichen Renderings sollen eine dynamische Lastverteilung für heterogene parallele Systeme geschaffen und Raytracing sowie Radiosity integriert werden.

Im Bereich der Media-Server ist die Realisierung und des Integration Java Media Framework (JMF) zusammen mit Windows und Synchronized Multimedia Integration Language (SMIL) am Client abgeschlossen. Ein interaktiver HiQoS-Media-Server läuft auf Basis von Linux (PC-Architektur). Die geplante Integration der RSVP-Kommunikation in Client, Server und Router ist erfolgt. Das HiQoS-Client-System wird nun zunächst eingebracht in die Anwendungen bei Pixelpark und an der Universität Paderborn.

Die unmittelbaren Folgearbeiten bestehen aus der Entwicklung eines Systems zur Verwaltung von Media-Assets, der Entwicklung von Algorithmen zur Abbildung von Inhalten auf verschiedene Standorte und dem Einsatz von klassischem

Realzeit-Scheduling zur Effizienzsteigerung. Schließlich sollen die Verfahren Netzwerkübergreifend eingesetzt werden in den Trials bei Pixelpark, APE und der Universität Paderborn.

7 ZUSAMMENFASSUNG

Das HiQoS-Projekt besteht aus zwei wesentlichen Schwerpunkten: einem netzbasierten Rendering-Service für photorealistische Animationen und der QoS-gerechten Auslieferung von Media-Assets. Beide Bereiche werden durch Anwendungen aus Multimedia-Produktion, CBT/BTV, Live-Streaming, Architektur und Filmproduktion evaluiert. Weitere Information zum Projekt ist erhältlich unter:

<http://www.uni-paderborn.de/fachbereich/AG/monien/PROJECTS/HIQOS/>

8 REFERENZEN

- [1] R. Lüling: *Communication scheduling in a distributed memory parallel interactive continuous media server*, International Conference on Parallel Processing (ICPP), August 1998, Minneapolis, pp. 58-65
- [2] C. Moar, A. Schmitt: HiQoS-Anforderungsanalyse und technische Spezifikation, HiQoS Report
- [3] I. Neumann, A. Wabro: Anforderungsanalyse der Firmen IEZ und UPSTART!, HiQoS Report
- [4] O. Michel, T. Prill: Die Pixelvision und Report Netzwerkerkausbau bei der Pixelpark GmbH, HiQoS Report
- [5] H. Patecki: Beschreibung einer Livestreaming-Anwendung innerhalb des Forschungsprojektes HiQoS, HiQoS Report
- [6] A. W. Bartels, H. Patecki, P. Altenbernd: Beschreibung eines Computer-Based-Training-Prototypen innerhalb des Forschungsprojektes HiQoS, HiQoS Report
- [7] K. Morisse, J. Jensch: Task 2.1: Spezifikation der Gesamtarchitektur, HiQoS Report
- [8] M. Sjödin, Response-Time Analysis for ATM Networks, Licentiate Thesis (DoCS 97/92), Uppsala University
- [9] Olaf Schmidt, Jan Rathert, Ludger Reeker, Tomas Plachetka: Parallel Simulation of Global Illumination, Proc. of the 1998 Int. Conf. on Parallel and Distributed Processing Techniques and Applications (PDPTA'98), pp. 289-1296