

DATENINTEGRATION IN DIENSTBASIERTE

3D-GEOVISUALISIERUNGSSYSTEME FÜR MOBILE GERÄTE

JAN KLIMKE¹ & JÜRGEN DÖLLNER²

Zusammenfassung: 3D-geovirtuelle Umgebungen (3D GeoVE) als eine universelle Repräsentationsform für 3D-Geodaten stellen eine grundlegende technische Komponente von 3D-Geoinformationssystemen dar. Virtuelle 3D-Stadtmodelle bilden die Modellgrundlage für eine wachsende Zahl von Anwendungen in unterschiedlichen Bereichen, wie z.B. Navigation, Stadtplanung, Katastrophenmanagement u.a. In diesem Beitrag stellen wir einen serviceorientierten Ansatz für die Implementierung von interaktiven 3D-Geovisualisierungssystemen für massive, virtuelle 3D-Stadtmodelle auf mobilen Geräten vor. Das besondere Kennzeichen des Ansatzes ist, dass der den Umfang und die Modellkomplexität eines virtuellen 3D-Stadt- und Landschaftsmodells vom Umfang der übertragenen Geodaten und dem clientseitigen Renderingaufwand entkoppelt, so dass eine Nutzung von virtuellen 3D-Stadtmodellen auf mobilen Geräten möglich wird.

Um eine anwendungsgerechte Nutzung eines Visualisierungssystems für virtuelle 3D-Stadtmodelle zu ermöglichen, ist es nötig zusätzliche 2D- und 3D-Fachdaten in die Visualisierung zu integrieren. Die verschiedenen Möglichkeiten solche Datenquellen an ein solches serviceorientiertes Visualisierungssystem anzubinden werden im zweiten Teil des Beitrages erörtert.

1 Einleitung

3D-geovirtuelle Umgebungen (3D GeoVE) sind eine leistungsstarke, universelle Repräsentationsform für 3D-Geodaten; Sie stellen eine grundlegende technische Komponente für 3D-Anwendungen und 3D-Geoinformationssysteme dar. Virtuelle 3D-Stadtmodellen bilden dabei im Allgemeinen die Modellgrundlage für eine wachsende Zahl von Anwendungen und Systemen in unterschiedlichen Bereichen, wie z.B. Navigationssysteme, Simulationssysteme, CAD-Systeme oder Desastermanagementsysteme. Heutige 3D-Geovisualisierungssysteme beruhen im Allgemeinen auf clientseitigem 3D-Rendering und erfordern daher für jedes Nutzersystem leistungsstarke 3D-Grafikhardware und halten Modelldaten lokal vor, die z.B. über Streamingverfahren bereitgestellt werden. Die zunehmende Verbreitung von mobilen Geräten wie Smartphones oder Tablet-PCs hat die Art, in der digitale Inhalte heute verbreitet und genutzt werden, maßgeblich beeinflusst. Mehr und mehr Inhalte werden heute neben dem konventionellen Weg über Webbrowser oder Clientanwendungen auch speziell für mobile Geräte bereitgestellt. Mit Blick auf 3D-Geovisualisierungssysteme und virtuelle 3D-Stadtmodelle ist eine solche interaktive Bereitstellung von Visualisierungen auf mobilen Geräten eine besondere Herausforderung, da für diese Geräteklasse Beschränkungen hinsichtlich des verfügbaren Speichers sowie der verfügbaren Prozessor- und Grafikleistung bestehen.

In der Vergangenheit wurden Ansätze für die 3D-Visualisierung von Geodaten auf mobilen Geräten (NURMINEN, 2008, QUILLET 2006, JIANG ET AL. 2009) vorgestellt. Weitere Ansätze sind

1) Jan Klimke, Hasso-Plattner-Institut, Universität Potsdam, Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, jan.klimke@hpi.uni-potsdam.de

2) Jürgen Döllner, Hasso-Plattner-Institut, Universität Potsdam, Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, doellner@hpi.uni-potsdam.de

wie Google Earth¹, Paris3D² oder upNext³ sind kommerziell für mobile Plattformen verfügbar. Typische Beschränkungen von existierenden Lösungen sind fehlende Nutzung der semantischen Bezüge in virtuellen 3D-Stadtmodellen, die fehlende Nutzung von modernen computergrafischen Verfahren für das Rendering oder eine nicht ausreichende vollständige Interaktivität dieser Systeme. Insofern besteht eine spezielle Herausforderung darin, massive, virtuelle 3D-Stadtmodelle so auf mobilen Geräten bereitzustellen, dass a) dem Nutzer eine computergrafisch hochqualitative Visualisierung der 3D-Geodaten geboten werden kann, b) das Modell für den Nutzer vollständig interaktiv erfahrbar ist und c) die zugrundeliegenden Fachdaten gleichrangig zu den 3D-Geometriedaten visuell repräsentiert werden.

In diesem Beitrag stellen wir einen allgemeinen Ansatz für die Implementierung von 3D-Geovisualisierungssystemen vor, der wesentlich dadurch gekennzeichnet ist, dass er die Komplexität des Modells von der Komplexität der Geodatenübertragung entkoppelt. Das System basiert auf einem 3D-Portrayalservice, der bildbasierte Repräsentationen des virtuellen 3D-Stadt- und Landschaftsmodells in Form virtueller Panoramen erzeugt. Der Service kapselt die Modellhaltung und das 3D-Rendering für die im Allgemeinen massiven Daten eines solchen Modells auf Serverseite. Clientanwendungen werden dadurch von diesen speicher- und rechenintensiven Aufgaben entlastet, da sie ausschließlich auf die generierten virtuellen Panoramen operieren. Dadurch können komplexe virtuelle 3D-Stadtmodelle und 3D-Landschaftsmodelle auch auf mobilen Geräten, bzw. generell auf Geräten, mit stark eingeschränkten Ressourcen, bereitgestellt werden.

Für eine effektive Nutzung von 3D-Geovisualisierungssystemen zur Kommunikation von Geoinformationen ist es von entscheidender Bedeutung, ein solches Visualisierungssystem flexibel mit 2D- und 3D-Fachdaten anzureichern. In Abschnitt 3 gehen wir daher auf Möglichkeiten ein, solche Daten effizient, konfigurierbar und gezielt in die 3D-Visualisierung zu integrieren.

2 Servicebasierte Visualisierung für virtuelle 3D-Stadtmodelle auf mobilen Geräten

Serviceorientierte Architekturen stellen ein Paradigma für das Design und die Implementierung von Softwaresystemen dar. Der grundlegende Gedanke bei ist ein lose Kopplung der funktionalen Komponenten eines Softwaresystems, den sogenannten Services (PAPAZOGLU, ET AL., 2007). Die funktionale Abgeschlossenheit der Komponenten führt zu einer vereinfachten Wiederverwendung von bestehenden Services und ermöglicht somit eine effizientere Erstellung und einen robusteren Betrieb von Anwendungen auf dieser Basis. Voraussetzung für eine solche Service-Interoperabilität ist die Definition von Standards für deren Schnittstellen und Formate für den Datenaustausch.

¹<http://www.google.com/mobile/earth/>

²<http://www.mobile3dcity.com/>

³<http://www.upnext.com/>

Für den Bereich der Geoinformation wurden hierzu durch das *Open Geospatial Consortium* (OGC) als internationale Standardisierungsorganisation zahlreiche Standards für Schnittstellen und Formate zur Kodierung und zur Arbeit mit Geodaten festgelegt.

Viele durch das OGC standardisierte Services zur Verwaltung von Geodaten (Web Feature Service, WFS), deren Verarbeitung (Web Processing Service, WPS) und 2D-Rendering (Web Map Service, WMS) sind bereits weit verbreitet. Es existiert eine Vielzahl von Implementierungen dieser Service-Schnittstellen, die als Basis für Anwendungen genutzt werden können.

Für die Visualisierung von 3D-Geodaten fehlen solche international akzeptierten und implementierten Standards bislang. Aktuell befinden sich zwei Vorschläge für sogenannte 3D-Portrayal-Services in der Diskussion innerhalb des OGC: Zum einen der Web 3D Service (*W3DS*) (SCHILLING & KOLBE, 2010), der Geometriepäsentationen von 3D-Geodaten an Clients liefert und zum anderen der Web View Service (*WVS*) (HAGEDORN, 2010) der virtuelle Panoramen (Bildpräsentationen der unterliegenden Geodaten) liefert. Die beiden Services kapseln jeweils unterschiedliche Teile der Visualisierungspipeline (CARD ET AL. 1990) für Geodaten. Ein W3DS implementiert die Filterung von Geodaten und das Mapping auf renderbare computergrafische Primitive (z.B. Dreiecke und Texturen). Ein WVS übernimmt zusätzlich das Rendering solcher computergrafischen Primitive und entlastet einen Client somit von dieser ressourcen- und energieintensiven Aufgabe. Insbesondere bei massiven, detailreichen 3D-Stadt- und Landschaftmodellen verlangt ein visuell ansprechendes Rendering von 3D-Geodaten nach leistungsstarker Grafikhardware und -software, um die umfangreichen Textur- und Geometriedaten verarbeiten sowie spezialisierte Shaderprogramme ausführen zu können.

Klassische Ansätze beruhen im Allgemeinen auf einer monolithischen Softwarearchitektur; Die daraus resultierenden "schwergewichtigen" Clientanwendungen bringen hier entscheidende Nachteile mit sich:

- Massive 3D-Geodaten können nur mit hohem Aufwand auf Endbenutzersystemen übertragen, gespeichert und verwaltet werden.
- 3D-Renderingtechniken sind beschränkt auf die jeweils spezifischen 3D-Fähigkeiten von Client-Systemen; Sie können daher nicht für alle Nutzersysteme in gleicher Qualität garantiert werden.
- Implementierungen von 3D-Renderingtechniken müssen kompatibel zu einer sehr heterogenen Client-Hardware sein, um weite Verbreitung finden zu können bzw. robust betrieben werden zu können.
- Der Energieverbrauch von Endgeräten wird durch 3D-Rendering massiv erhöht. Dies beeinträchtigt insbesondere die Akkulaufzeit von mobilen Geräten.

Servicebasierte Geovisualisierung stellt deshalb eines der Kernkonzepte für die zukünftige Nutzung von virtuellen 3D-Stadtmodellen im Kontext von mobilen Geräten dar. Konzeptionell wird der Geovisualisierungsprozess so aufgeteilt, dass verschiedene Schritte zusammengefasst und von einem Service gekapselt werden. Auf diese Weise können Visualisierungsanwendungen erstellt werden, die optimal auf die Hardware, Software und Datenzugriffsrechte von verschiedenen Clientplattformen angepasst sind. Der Einsatz der bereits eingeführten 3D-Portrayal-Services bildet hier ein zentrales Element um 3D-Visualisierung von Geodaten effizient auf einer Vielzahl von Hardware- und Softwareplattformen anbieten zu können.

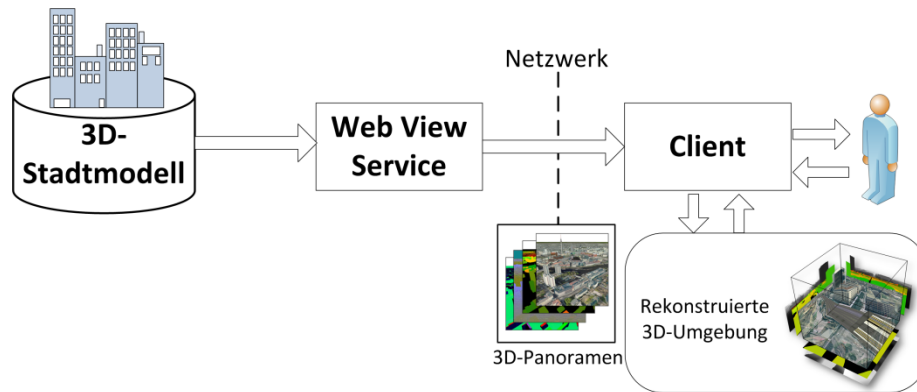


Abb. 1: Überblick über den Systemaufbau des servicebasierten 3D-Geovisualisierungssystems. Der Web View Service kapselt das Datenmanagement und das Rendering des 3D-Stadtmodells. Der Client fragt über ein Netzwerk 3D-Panoramen beim Service an und erstellt daraus eine Rekonstruktion der 3D-Umgebung.

Mobile Geräte wie Smartphones oder Tablet-PCs werden immer leistungsstärker in Hinblick auf ihre Rechen- und Grafikleistung. Dennoch sind viele der oben genannten Probleme gerade für die Visualisierung von sehr großen 3D-Geodaten auf mobilen Geräten ein großes Hemmnis. Bei geometriebasierten Ansätzen für das 3D-Rendering auf Clientseite stellt bei großen Modellen bereits die Übertragung von Geodaten an mobile Geräte stellt bereits eine große Herausforderung dar, da die Menge der übertragenen Geodaten sowie der Aufwand für das 3D-Rendering der Daten direkt abhängig von der Modellkomplexität (u. A. Umfang, Detailreichtum, Genauigkeit). Das heißt, je größer das virtuelle 3D-Stadtmodell, desto mehr Daten müssen auch auf ein Client-system übertragen, gespeichert und verwaltet werden. Dies stellt insbesondere bei kabellosen Netzwerken ein Problem hinsichtlich der Zuverlässigkeit und des Datendurchsatzes dar. Weiterhin ist das 3D-Rendering von großen Geometrie- und Texturdaten auf mobilen Geräten besonders energieintensiv und verkürzt damit die Akkulaufzeit der Geräte drastisch. Daher muss es das Ziel beim Entwurf eines Visualisierungssystems sein, die zu übertragende und zu verarbeitende Datenmenge so zu reduzieren, dass ein sehr effizientes 3D-Rendering möglich ist.

Im Folgenden wird ein System vorgestellt, das die oben genannten Probleme löst, indem ein WVS zum Rendern von Panoramaansichten von virtuellen 3D-Stadtmodellen in Kombination mit einer entsprechenden Client-Anwendung verwendet wird, um Nutzern einen vollständig interaktiven Zugang zu 3D-Stadtmodellen auch auf mobilen Geräten zu gewähren. Die Client-Anwendung nutzt hierbei eine aus den virtuellen Panoramen abgeleitete, stark vereinfachte lokale 3D-Repräsentationen des unterliegenden virtuellen 3D-Stadtmodells. Diese leichtgewichtige, bildbasierte Repräsentation des Modells lässt sich sehr effizient rendern. Der wesentliche Aufbau des Gesamtsystems ist in Abb. 1 dargestellt.

2.1 Aufbau und Eigenschaften des WVS-Servers

Der WVS als Basis des Gesamtsystems kapselt den Zugriff auf 3D-Geodaten, deren Abbildung auf computergrafische Primitive und deren Rendering; Er erzeugt Panoramen in Form von Cube-Maps. Diese können auf vielfältige Art und Weise konfiguriert werden. Zum einen lassen sich über so genannte *Layer* (Informationsschichten) die zugrunde liegenden Informationen gezielt auswählen. Zum anderen können virtuelle Panoramen, die der WVS berechnet, auf Grundlage

des G-Buffer-Konzepts (SAITO & TAKAHASHI, 1990), mit Zusatzinformationen, wie Tiefen-, Normalen- oder Objektidentitätsinformationen (Abb. 2), angereichert werden. Die Layer der Cube-Maps werden als Bild in Standardformaten, wie z.B. JPEG oder PNG, kodiert. Eine besondere Rolle besitzt der Objektidentitätslayer, da durch ihn im virtuellen Panorama jedes dargestellte Objekt rückverfolgt werden kann. Client-Anwendungen können auf dieser Basis z.B. Objekte interaktiv, z.B. durch Einfärbung, hervorheben.

Die Größe der durch den Renderingserver erzeugten kodierten Bilddaten hängt hauptsächlich von der Auflösung des angefragten Panoramas ab. Die Datenmenge ist jedoch unabhängig von der Komplexität des zugrundeliegenden Modells. Hierin liegt einer der zentralen Vorteile des serverseitigen Renderings: Es entkoppelt die Übertragungskomplexität von der Komplexität der unterliegenden Geodaten.

Die Software- sowie Hardwarekonfiguration des Renderingsservers kann durch den Service-Anbieter festgelegt werden. Dies ermöglicht es einem Anbieter, eine optimale Konfiguration des unterliegenden Systems in Hinblick auf die eingesetzten Renderingtechniken sicherzustellen. Auf diese Weise wird der Entwicklungsaufwand, den es im Allgemeinen bedarf, um eine komplexe Visualisierungstechnik für Endanwender bereitzustellen, stark verringert. Ebenfalls ist eine für den Service-Nutzer transparente Möglichkeit gegeben, weitere Renderingtechniken in ein Gesamtsystem zu integrieren und somit die Möglichkeiten zur Visualisierung zu erweitern, ohne dass die Endanwendung auf Clientseite verändert werden müsste.

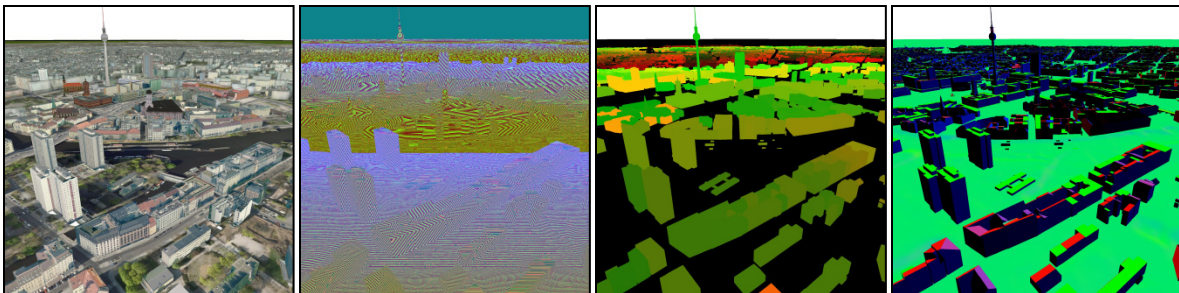


Abb. 2: Verschiedene Informationsschichten eines virtuellen Panoramas des Stadtmodells von Berlin kodiert als Farbinformation. Von links nach rechts: Bildlayer, Tiefenlayer (Entfernung von der virtuellen Kamera), Objektidentitätslayer, Normalen zu den einzelnen Bildpunkten

2.2 Aufbau und Eigenschaften des WVS-basierten Clients

Die Hauptaufgabe des zum WVS zugehörigen Clients besteht darin, die vom WVS gelieferten Panoramen interaktiv zu visualisieren. Der Client rekonstruiert bildbasiert die 3D-Szene aus einem im Client vorhandenen Panorama. In der einfachsten Variante stellt der Client ausschließlich die Cube-Map dar und ermöglicht die standortfixierte Interaktion in Panorama in Analogie zu handelsüblichen Panorama-Viewers. In unserem Ansatz wird eine erweiterte Variante implementiert, die pro Cube-Map-Seite aus den Farb- und Tiefeninformationen ein texturiertes Dreiecksnetz erstellt, sowie Objektidentitätsinformationen nutzt, um eine vollwertige Interaktion des Nutzers mit der bildbasierten Rekonstruktion zu ermöglichen.

Die Nutzerinteraktion des Clients ist weitgehend von der Antwortgeschwindigkeit des Rendering-Servers entkoppelt. Bewegt ein Nutzer die virtuelle Kamera, bewegt er sich in der entsprechend aufbereiteten 3D-Panoramageometrie; Es werden während dieser Bewegung keine weite-

ren Anfragen an den WVS gesendet. Die Darstellung während der Kameranavigation dient dem Nutzer vorrangig zur Orientierung, wobei Abstriche in der Darstellungsqualität hier in Kauf genommen werden, da die vom Server abgerufenen Bildinformationen jeweils nur für den angefragten Kamerastandpunkt optimal sind. Stoppt die Interaktion des Nutzers, wird ein neues Panorama vom WVS-Server angefragt. Sobald das Panorama übertragen wurde, wird die Darstellung aktualisiert. Das heißt, aus Nutzersicht wird die Darstellung nach wenigen Sekunden verfeinert. Dabei ist jedoch jederzeit eine Interaktion mit dem System möglich. Abb. 3 zeigt ein mobiles Gerät mit der Clientanwendung.



Abb. 3: Die interaktive Clientanwendung auf iPad und iPod touch ermöglicht dem Benutzer ein flüssiges Interagieren mit der leichtgewichtigen Rekonstruktion eines virtuellen 3D-Stadtmodells. Im Bild dargestellt ist das Pariser 3D-Stadtmodell (BATI3D©-Daten sind Eigentum des Institut Géographique National, France.)

3 Dienstbasierte Integration von Fachdaten in 3D GeoVEs

Für fachspezifische Anwendungen von 3D-GeoVEs ist es im Allgemeinen erforderlich, anwendungsspezifische Daten in die 3D-Visualisierung zu integrieren. Typische anwendungsspezifische Daten beinhalten:

- Kartenmaterial zur Texturierung des Geländemodells, bereitgestellt z.B. durch einen WMS;
- 3D-Featuredaten, z.B. Modelle von Planungsalternativen, bereitgestellt z.B. durch einen WFS oder W3DS;
- 2D-Featuredaten, z.B. Points-of-Interest, Flächen- oder Routeninformationen, bereitgestellt z.B. durch Open Location Services oder per WFS;
- Sensordaten, bereitgestellt z.B. durch einen OGC Sensor Observation Service

Die Integration von 2D- und 3D-Fachdaten in ein verteiltes Visualisierungssystem stellt eine Herausforderung dar: Einerseits sollen Fachdaten gezielt in die Visualisierung aufgenommen werden, andererseits müssen die heterogenen Datenquellen und die berücksichtigt werden. Clientanwendungen in unserem Ansatz verfügen nicht über vollständige Modellinformationen

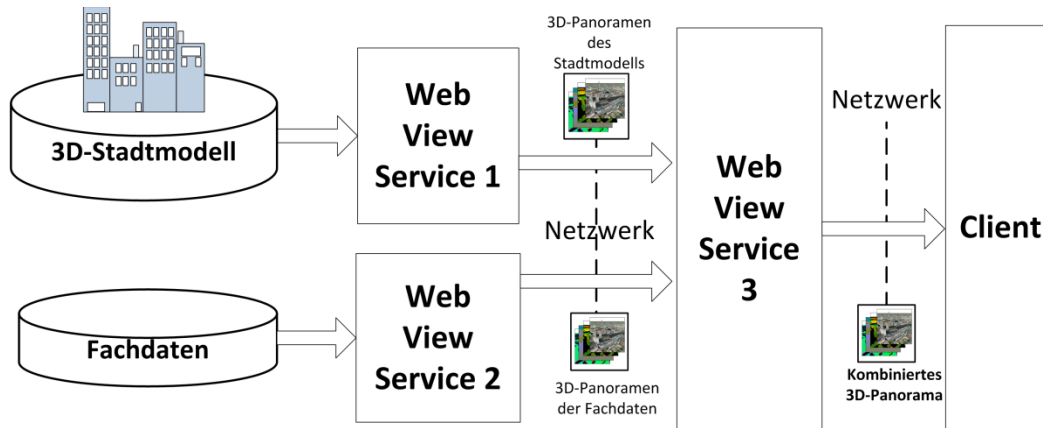


Abb. 4: Systemaufbau zur Datenintegration basierend auf der Nutzung mehrerer WVS-Instanzen und der Kombination von 3D-Panoramen

(u. a. Geometrie und Texturdaten), sondern basieren auf einer bildbasierten Modellrepräsentation. Dies beschränkt die Möglichkeiten z.B. für Berechnungen zur Platzierung von Beschriftungen oder Geometrierepräsentationen von Fachdaten.

3.1 Serverseitige Integration von Fachdaten in WVS

Die verteilte Architektur des hier vorgestellten Systems ermöglicht die Integration von Fachdaten in den einzelnen Komponenten. Eine Integration in den WVS-Server bietet sich dabei aus Effizienzgründen an. Bereitgestellte Fachdaten können direkt im serverbasierten Visualisierungsprozess für das Basis-Stadtmodell berücksichtigt werden. Da der Server in einer kontrollierten Umgebung operiert, sind Annahmen über die Speicher- und CPU-Ressourcen des Serversystems und über die Qualität der Anbindung an externe Services, die als Fachdaten zuliefern, in weit größerem Maße möglich als dies bei einem herkömmlichen Fat-Clientansatz der Fall ist.

Allgemein hat eine serverseitige Integration und Visualisierung von Fachdaten in die ausgelieferten Panoramen den Vorteil, dass existierende Clientanwendungen ohne Modifikationen auch zur Visualisierung dieser Daten genutzt werden können. Es wird für die Anzeige auf Clientseite weder mehr Speicher noch mehr Performance für das 3D-Rendering benötigt. In den Renderingprozess des Servers können so unterschiedliche Fachdaten eingebunden und auf Geometrie des Stadtmodells abgebildet werden; jedoch sind dynamische Daten, wie z.B. Fahrzeugpositionen, weniger geeignet, da bei jeder Wertänderung ein neues Panorama vom Server angefordert werden müsste um den aktuellen Stand zu visualisieren.

Neben der direkten Integration von Fachdaten in den Renderingprozess des Servers ist auch eine rein bildbasierte Komposition mehrerer, separat erzeugter, virtueller Panoramen möglich. Dazu rendert ein spezieller WVS-Server das Basis-Stadtmodell und ein anderer die Fachdaten (HAGEDORN ET AL., 2007). Da es sich um 3D-Panoramen dieser Daten handelt, ist jedoch das Problem der Verdeckung beim Verblenden der einzelnen Panoramen zu berücksichtigen. Dies ist möglich, indem zusätzlich zum Farbbild ein Tiefenbild in die Berechnung einbezogen wird: Anhand der jeweiligen Tiefenwerte pro Bildpunkt kann festgestellt werden, welcher der Bildpunkte der Quellbilder die Farbe des jeweiligen Bildpunktes im Zielbild bestimmen soll. Diese Art der Kombination von Daten in einem Panorama stellt eine sehr flexible Art der Datenintegration dar,

da weder eine Modifikation des WVS-Servers noch des Clients nötig ist. Jedoch ist diese Art der Kombination von virtuellen Panoramen weniger effizient, da die Einzelpanoramen erst zu einem Kombinationservice übertragen, dort dekodiert, kombiniert und anschließend wieder kodiert werden müssen (Abb. 4).

3.2 Clientseitige Integration von Fachdaten

Um Fachdaten in die Visualisierung von 3D-Stadtmodellen aufzunehmen kann auch die Clientanwendung mehrere Panoramen kombinieren, wobei die zu übertragende Datenmenge entsprechend anwächst, was insbesondere bei der Anwendung auf mobilen Geräten problematisch ist. Weiterhin können die Daten nur durch eine erneute Anfrage für die Fachdaten an den WVS aktualisiert werden. Dies behindert auch hier die Visualisierung von dynamischen Daten.

In unserem Ansatz erzeugt der Client eine eigenständige Geometrie, so dass eine approximative geometrische Repräsentation im Client zur Verfügung steht. Der Client kann so Fachdaten auch selbst anfragen und diese visuell repräsentieren. Der Client hat dementsprechend die Möglichkeit, ein eigenes Datenmodell für die Fachdaten zu halten. Diese können dann durch erneute Abfrage der Datenservices aktualisiert werden. Die Integration in die 3D-Visualisierung des Clients mittels clientseitig gerenderter Primitive ist die einzige Möglichkeit um Animationen von Datenrepräsentationen effizient zu implementieren. Dementsprechend eignet sich diese Variante insbesondere für dynamische Daten wie z.B. Fahrzeugpositionen oder Sensordaten. Ein solches hybrides Rendering, das teils auf Bilddaten und teils auf dem Mapping und Rendering von Rohdaten basiert bietet damit eine größere Flexibilität bezüglich Animation und Interaktion mit den Fachdaten.

4 Zusammenfassung und Schlussfolgerungen

In diesem Beitrag haben wir ein System für die interaktive 3D-Geovisualisierung auf Basis von serverseitigem Rendering vorgestellt. Dieses System mit seiner serviceorientierten, verteilten Architektur bietet Lösungsansätze für die Herausforderungen bei der Visualisierung von virtuellen 3D-Stadtmodellen auf mobilen Geräten. Durch die Beschränkung auf den Austausch von virtuellen Panoramen, die in Standard-Bildformaten kodiert sind, ist es möglich sehr flexible Lösungen für hochqualitative Visualisierung von virtuellen 3D-Stadtmodellen anzubieten, ohne dass Anforderungen an Clientsysteme gestellt werden müssen, die über die Fähigkeiten von Standardhardware hinausgehen. Der implementierte Prototyp für ein solches Visualisierungssystem zeigt, dass eine interaktive Nutzung auf mobilen Geräten möglich ist.

Für 3D-Geoinformationssysteme basierend auf solchen servicebasierten Ansätzen ist es für die Nutzbarkeit weiterhin entscheidend, flexibel Fachdaten aus verschiedenen Quellen zu integrieren. Es wurden Möglichkeiten aufgezeigt, die verschiedenen Arten von Fachdaten in ein solches Visualisierungssystem zu integrieren.

Die durch die verteilte Architektur des Gesamtsystems ermöglichte Nutzung komplexer Panoramen auf mobilen Geräten bietet zugleich einen Skalierungsansatz für 3D-Geovisualisierungsdienste. Der vorgestellte servicebasierte Ansatz kann dabei ein Grundpfeiler für künftige verteilte, skalierbare 3D-Geovisualisierungssysteme darstellen, da er für zentrale Probleme heutiger Geovisualisierungssysteme, wie die Verteilung und Verarbeitung von 3D-

Geodaten für die Visualisierung, die Diversität von Hard- und Software auf Clientseite, sowie die Verteilung und Wartung von Client-Anwendungen, Lösungen anbietet.

5 Literaturverzeichnis

CARD, S., MACKINLAY, J. D. & SCHNEIDERMAN, B., 1999. *Readings in Information Visualization: Using Vision to Think*. s.l.:Morgan Kaufmann.

DEL LA BEAUJARDIERE, J., 2004. *OGC Web Map Service Interface*. s.l.:Open Geospatial Consortium.

HAGEDORN, B., 2010. *Web view service discussion paper*. s.l.:Open Geospatial Consortium.

HAGEDORN, B., HILDEBRANDT, D. & DÖLLNER, J., 2009. *Towards Advanced and Interactive Web Perspective View Services*. s.l., Springer, pp. 33-51.

HAGEDORN, B., MASS, S. & DÖLLNER, J., 2007. *Chaining Geoinformation Services for the Visualization and Annotation of 3D Geovirtual Environments*. s.l., s.n.

HILDEBRANDT, D., KLIMKE, J., HAGEDORN, B. & DÖLLNER, J., 2011. *Service-Oriented Interactive 3D Visualization of Massive 3D City Models on Thin Clients*. New York, New York, USA, s.n.

JIANG, W., YUGUO, W., FAN, W.. 2009. An Approach for Navigation in 3D Models on Mobile Devices, *Location Based Services and TeleCartography (2009)* , p. 109-114.

NURMINEN, A., 2008. Mobile 3D City Maps. *IEEE Computer Graphics and Applications*, 28(4), pp. 20-31.

PAPAZOGLU, M. P., TRAVERSO, P., DUSTDAR, S. & LEYMAN, F., 2007. Service-Oriented Computing: State of the Art and Research Challenges. *Computer*, 40(11), pp. 38-45.

QUILLET, J., THOMAS, G., GRANIER, X., GUITTON, P, MARVIE, J., 2006, Using Expressive Rendering for Remote Visualization of Large City Models, Proceedings of the eleventh international conference on 3D web technology, ACM, New York, USA, pp 27 – 35.

SAITO, T. & TAKAHASHI, T., 1990. Comprehensible Rendering of 3-D Shapes. *Computer*, 24(4), pp. 197-206.

SCHILLING, A. & KOLBE, T., 2010. *Draft for Candidate OpenGIS® Web 3D Service Interface Standard*. s.l.:Open Geospatial Consortium.