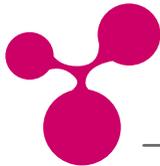


Technische Universität Dresden – Fakultät Informatik
Professur für Multimedialechnik, Privat-Dozentur für Angewandte Informatik

Prof. Dr.-Ing. Klaus Meißner
PD Dr.-Ing. habil. Martin Engelen
(Hrsg.)



GENeME '09

GEMEINSCHAFTEN IN NEUEN MEDIEN

an der
Fakultät Informatik der Technischen Universität Dresden

mit Unterstützung der

3m5. Media GmbH, Dresden
GI-Regionalgruppe, Dresden
Communardo Software GmbH, Dresden
Kontext E GmbH, Dresden
Medienzentrum der TU Dresden
nubix Software-Design GmbH, Dresden
objectFab GmbH, Dresden
SALT Solutions GmbH, Dresden
Saxonia Systems AG, Dresden
T-Systems Multimedia Solutions GmbH

am 01. und 02. Oktober 2009 in Dresden

<http://www-mmt.inf.tu-dresden.de/geneme/>
geneme@mail-mmt.inf.tu-dresden.de

B.12 3DIOS¹ – Konzept eines Internet-Operating-Systems

Hauke Coltzau, Herwig Unger
Fernuniversität in Hagen, Lehrgebiet Kommunikationsnetze

Abstract

Es wird ein Konzept für ein zukünftiges Betriebssystem vorgestellt, das den Anforderungen und Möglichkeiten der zunehmenden Vernetzung von Rechnern und anderen Endgeräten Rechnung trägt. Es erweitert den bekannten Grundgedanken, mit Hilfe von Peer-to-Peer Technologien explorierbare virtuelle 3D-Landschaften zu erzeugen, ohne dabei von einem zentralen Anbieter abhängig zu sein. Es soll Nutzern die Möglichkeit bieten, das Serviceangebot im Internet ohne Einschränkungen zu durchsuchen und dabei anhand eigener semantischer Kriterien die für sie interessanten Services hervorzuheben. Über Parameter der Darstellung sollen Netzwerkeigenschaften, die für den Nutzer von Belang sein können, intuitiv erfassbar sein. Das System beinhaltet darüber hinaus Möglichkeiten zur Last- und Verkehrsbalancierung. In diesem Artikel werden Anforderungen an die zu Grunde liegende Infrastruktur und die grafische Benutzeroberfläche definiert, erste Skizzen des zukünftigen Systems vorgestellt und die weiteren Schritte bis zur Verwirklichung eines Prototyps diskutiert.

Keywords: Internet Operating System, Peer-To-Peer, Collaboration

1 Motivation

Die grafische Benutzerschnittstelle aller modernen Betriebssysteme orientiert sich noch immer an der jahrzehntealten Analogie, Dokumente und Applikationen (Services) des Nutzers auf einem virtuellen Schreibtisch (Desktop) anzubieten. Durch aufwändigen Einsatz von 3D-Technologie (BumpTop²) wird dabei nur verschleiert, dass der Nutzer keine Möglichkeit hat, von seinem virtuellen Schreibtisch aufzustehen und wirklich „ins Netz“ zu gehen. Stattdessen wird behelfsmäßig der Internet-Browser verwendet, um entfernte Applikationen zu starten. Doch ohne zusätzliche (aktive) Komponenten wie JavaScript oder Shockwave-Flash können solche Web-Applikationen gar nicht oder nur sehr umständlich bedient werden. Durch die fehlenden Integrationsmöglichkeiten dieser Plug-Ins in ein rechner- oder netzwerkweites Sicherheitskonzept entstehen große Risiken, deren Management besondere Aufmerksamkeit erfordert [5].

1 3D-IOS = 3D Internet Operating System

2 BumpTop ist ein 3D-Desktop mit physikalisch basierter Animation der Desktopelemente.
Näheres unter: <http://www.bumptop.com/>

Es gibt selbst innerhalb der Grenzen eines lokalen Netzwerkes nur sehr enge Möglichkeiten, die im LAN zur Verfügung stehenden Services zu erforschen. Für die Suche von Services außerhalb des eigenen LANs stehen de facto nur kommerziell betriebene Suchmaschinen zur Verfügung, die eine intuitive Exploration der Inhalts- oder Servicelandschaft nicht ermöglichen. Ähnlichkeitssuchen oder komplexe Suchanfragen sind zurzeit nicht oder nur sehr eingeschränkt möglich. Der Nutzer kann also nie sicher sein, dass das Ergebnis einer Suche tatsächlich das für ihn beste Ergebnis darstellt.

Es existieren zwar bereits seit längerer Zeit Standards, die eine einheitliche Definition von Services ermöglichen (z.B. WSDL³). Allerdings können gewöhnliche Nutzer nicht intuitiv auf diese Services zugreifen oder eigene Services erstellen. An dem großen Erfolg von Blogs, Community-Portalen und Twitter⁴ lässt sich ablesen, dass ein großes Interesse der Nutzer an der inhaltlichen Mitgestaltung des WWW existiert. Zusammengefasst braucht es also ein zusammenhängendes Betriebssystemkonzept, dass folgende Forderungen erfüllt:

- 1) Das System bildet dynamisch ein virtuelles Netzwerk aus allen internetweit zur Verfügung stehenden Services und Inhalten und bietet dem Nutzer intuitiven Zugang zu diesem Netzwerk. Es berücksichtigt dabei, dass dem Nutzer nur solche Dienste und Inhalte präsentiert werden, für die er die entsprechenden vom Dienst-/Inhaltsanbieter vergebenen Berechtigungen besitzt.
- 2) Die Schnittstelle zu Services und Inhalten ist unabhängig von ihrem Ort. Demnach wird auch nicht notwendigerweise zwischen Services auf dem Rechner des Nutzers, denen in lokalen und denen in Weitverkehrsnetzwerken unterschieden.
- 3) Nutzer können das Netzwerk und damit die zur Verfügung stehenden Services ohne Stichwortsuche erforschen. Dabei können sie auch andere Nutzer (repräsentiert z.B. durch Avatare) sehen und mit ihnen interagieren.
- 4) Semantische Suchen und Filter erlauben kontextsensitive Exploration der für den Nutzer interessanten Services. Die Darstellung wird entsprechend auf die gesuchten Services konzentriert, andere Services rücken in den Hintergrund oder werden gar nicht dargestellt.
- 5) In der grafischen Repräsentation der Netzwerkumgebung kann der Nutzer bereits Netzwerkparameter (Auslastung, Wartezeit, Bandbreite, etc.) einzelner Knoten ablesen.
- 6) Eine geeignete Serviceschnittstelle (Vgl. SOA(P)) ermöglicht aus Applikationen heraus den Zugriff auf entfernte Ressourcen.

3 WSDL=WebService Description Language, s.a.: <http://www.w3.org/TR/wsdl20/>

4 <http://www.twitter.com>

Selbstverständlich müssen dabei gängige Eigenschaften moderner Betriebssysteme, wie z.B. Zugangskontrollen sowie Gewährleistung von Datenintegrität und -sicherheit, Berücksichtigung finden.

Im folgenden Kapitel werden drei zentrale Aspekte zukünftiger Betriebssysteme beleuchtet, die sich in logischer Konsequenz aus den genannten Forderungen ergeben. Kapitel 3 zeigt, wie diese Aspekte in einer grafischen Benutzeroberfläche dargestellt werden könnten und diskutiert verschiedene Umsetzungsvarianten. Kapitel 4 beleuchtet die Forschungsfragen, die auf dem Weg der Umsetzung zu beantworten sind. Verwandte Arbeiten werden – entgegen dem sonst üblichen Vorgehen – zum Schluss in Kapitel 5 beleuchtet.

2 Zentrale Aspekte

Wechsel der GUI-Metapher

Mit der zunehmenden Zahl an Internetzugängen hoher Bandbreite, durch die die Unterschiede zwischen lokaler und entfernter Ausführung von Applikationen in den Hintergrund gedrängt werden, wurde der Desktop nicht wesentlich weiterentwickelt. So ist es nicht oder nur sehr schwer möglich, alle dem Nutzer zur Verfügung stehenden Ressourcen übersichtlich darzustellen. Da der Mensch es gewohnt ist, sich im realen Leben in Städten zu bewegen und sich (ggf. mit Hilfe von Karten) in ihnen zurechtzufinden, liegt es nahe, eine solche Stadt-Metapher auch für grafische Benutzeroberflächen zu verwenden. Bei dem hier diskutierten Ansatz wird der einzelne Rechner (Peer) im Servicenetz durch ein virtuelles Gebäude repräsentiert, das von Nutzern betreten werden kann. Die Gebäude (Peers) sind durch Straßen miteinander verbunden (das Netzwerk), auf denen sich der Nutzer mit seinem Avatar fortbewegen und auch anderen Benutzern begegnen kann. Das äußere Erscheinungsbild eines Peers (die Fassade) gibt Aufschluss darüber, welches Serviceangebot bei diesem Peer in Anspruch genommen werden kann. Um die Navigation zu erleichtern, werden die Straßen rechtwinklig angeordnet. Im Gegensatz zur Arbeit von Schuster u.a. [6] wird die virtuelle Welt zusammenhängend und möglichst dicht besetzt dargestellt.

Der Nutzer hat die Möglichkeit, das Netz zu durchwandern und sich darin zu orientieren, ohne dabei zwangsläufig bereits ein konkretes Ziel vor Augen zu haben. Sämtliche Orientierungshilfen einer realen Stadt, also z.B. Wegweiser, markante Gebäude, etc. können eine Entsprechung in der virtuellen Welt finden und dadurch die Orientierung erheblich erleichtern.

Adaptive Ansichten

Mit wachsender Anzahl an Teilnehmern wird die virtuelle Umgebung schnell unübersichtlich. Es ist also für den Nutzer von entscheidender Bedeutung, wie aus seiner Sicht die Peers in seiner Umgebung arrangiert werden. Der Nutzer soll mit Hilfe von Stichworten und deren Kombination Peers ausblenden oder hervorheben können. Jeder Service wird im Rahmen einer geeigneten Ontologie durch den Anbieter beschrieben, so dass im System semantisch geordnete Karten der Servicelandschaft erzeugt werden können, in denen Peers benachbart sind, die ähnliche Services anbieten. So erhält der Nutzer einen kompakten Überblick über alle verfügbaren Services, die für ihn von Interesse sind. Es können also jederzeit Filter zu- und abgeschaltet werden, deren Anwendung die virtuelle Landschaft umgehend neu formieren. Insbesondere ist mit Hilfe der Metainformationen auch eine Ähnlichkeitssuche möglich, bei der der Nutzer eine Übersicht aller Services anfordert, die einem ihm bekannten Service in vom Nutzer bestimmten Aspekten ähnlich sind. Wenn die Services z.B. geographische Koordinaten in ihren Metainformationen vorhalten, kann der Nutzer eine Umgebungssuche initiieren.

Der Zugriff auf Services eines Peers kann natürlich weiterhin über eine URL, bestehend aus Protokoll(Service-)name, eindeutigen Bezeichner des Peers und weiteren Serviceparametern erfolgen. Zusätzlich ist aber auch eine Identifikation über Tags oder über angewandte Suchfilter möglich. Je stärker die Suche eingegrenzt wird, desto mehr wird ein einzelner Peer durch sie identifiziert.

Visualisierung von Netzwerk- und Systemparametern

Neben inhaltlichen Kriterien ist für den Nutzer auch die Verfügbarkeit des angebotenen Services von Bedeutung. Peers, deren Services eine hohe Auslastung haben, können ggf. weitere Anfragen nicht mehr in der gewünschten Qualität beantworten. Ebenso ist es möglich, dass die zur Verfügung stehende Bandbreite zwischen dem Rechner des Nutzers und dem jeweiligen Peer nicht den Anforderungen des Services oder denen des Nutzers entspricht. Für den Nutzer ist es also von Interesse, bereits vorab eine Abschätzung dieser Parameter zu erhalten. Dabei soll der Nutzer nicht mit numerischen Werten überflutet werden, sondern die Parameter in der Darstellung der virtuellen Objekte selbst erkennen können.

Ein eindrückliches Maß für die Auslastung eines Services ergibt sich implizit aus der Anzahl der Avatare, deren Nutzer den Service gerade verwenden. Da die Avatare von den anderen Nutzern wahrgenommen werden können, erhalten diese einen Eindruck von der Auslastung eines Services. Werden Avatare zusätzlich mit physikalischen Eigenschaften modelliert, kann eine „natürliche“ Begrenzung der Anzahl an Nutzern erfolgen, die den Service gleichzeitig verwenden können.

Es lassen sich weitere Beispiele für die Reflektion von Netzwerkeigenschaften im Erscheinungsbild finden: Die Breite einer Straße kann dynamisch derart variiert

werden, dass sie ein Maß für die Verbindungsqualität (Bandbreite, Latenz) darstellt. Die Auslastung eines Peers kann mit Hilfe der Farbsättigung modelliert werden, so dass stark ausgelastete Peers z.B. grau erscheinen würden. Derselbe Parameter ließe sich alternativ durch Transparenz abbilden, indem Peers mit hoher Auslastung nur noch schemenhaft erscheinen.

Führt man diesen Grundgedanken fort, lassen sich auch Parameter, die nicht in direktem Bezug zu Netzwerkeigenschaften stehen, intuitiv erfassbar machen. Tabelle 1 zeigt eine Auswahl möglicher Zuordnungen verschiedener Parameter auf die Darstellung der virtuellen Welt.

Tabelle 1: Darstellung von Netzwerkparametern

| Darstellung | Abgebildete Netzwerkparameter |
|--|--|
| Straßenbreite | Verfügbare Bandbreite Auslastung |
| Höhe eines Gebäudes | Verfügbarkeit/Auslastung Umfang des Serviceangebots Ähnlichkeitsmaß bei semantischer Suche |
| Transparenz Farbsättigung | Verfügbarkeit/Auslastung Ähnlichkeitsmaß bei semantischer Suche |
| Grundfläche des Gebäudes | Umfang des Serviceangebots |
| Elevation des Untergrundes (Gebirge, Täler) | Frequenzierungshäufigkeit/Beliebtheit Umfang des Serviceangebots |
| Ampeln, Schranken | Verkehrs- und Lastbalancierung |

Einige der Parameter können in das Arrangement der Peers einfließen. Peers, deren Auslastung zurzeit besonders hoch ist, könnten in der Karte ausgeblendet werden, transparent oder besonders weit entfernt erscheinen. Eine erschöpfende Diskussion aller Möglichkeiten ist an dieser Stelle kaum möglich und wird Gegenstand der weiteren Forschungsarbeit sein.

3 Darstellung

Die folgenden Grafiken geben einen ersten Eindruck dessen, wie sich die Netzwerkkumgebung aus Sicht eines Nutzers darstellen könnte. Mit Hilfe der Übersichtskarte (Abbildung 1) ist eine großräumige Orientierung in der Servicelandschaft möglich. Von den Gebäuden, die die einzelnen Peers darstellen, ist nur das „Dach“ zu sehen, also eine repräsentative Grafik des Serviceangebots. Die Straßen verlaufen zur Erleichterung der Orientierung, orthogonal. Es ist denkbar, die Flächen, die für jeden Peer zur Verfügung stehen, entweder genau gleich groß oder

mit variabler Größe darzustellen. Die Straßen können entweder durch die Peers selbst oder durch das System bereitgestellt werden (Abbildung 2). Ein Peer ist so entweder für jeweils eine Straßenkreuzung (Variante a) oder für ein von Straßen umrandetes Grundstück zuständig (Variante b), wobei die Straßen durch den Rechner des Nutzers hinzugefügt werden.



Abbildung 1: Übersichtskarte

Beide Varianten haben Vor- und Nachteile. Im ersten Fall sind die Gestaltungsmöglichkeiten für den einzelnen Peer größer und die Abbildung von Netzwerkparametern auf die Straßeneigenschaften ist gezielter möglich. Nachteilig ist allerdings, dass Peers auf diesem Wege den Durchgang zu anderen Peers verhindern und so einzelne Bereiche des Netzwerkes isolieren können. Die zweite Variante verhindert genau dieses Problem, verlangt im Gegenzug aber einen höheren Aufwand bei der Lokalisation und Zuordnung von Objekten (z.B. Avataren), die sich auf den Straßen selbst befinden.

Da das System an den Bedürfnissen der Nutzer ausgerichtet sein soll, wird das Zukaufen von Fläche seitens des Service-Providers (wie in SL) nicht möglich sein. Allerdings kann ein geeigneter systemweiter Ranking-Algorithmus Peers mit besonderer Bedeutung hervorheben und ihnen damit mehr Fläche zur Verfügung stellen. Ein solcher Algorithmus muss aber derart gestaltet sein, dass Ranking-Manipulationen analog zur sog. Suchmaschinenoptimierung (Sybill-Attacke, Linkfarmen, Brückenseiten, etc.) nicht oder nur sehr eingeschränkt möglich sind. Letztendlich muss es unter der Kontrolle des Nutzers bleiben, diese Hervorhebung nach Bedarf zu nutzen

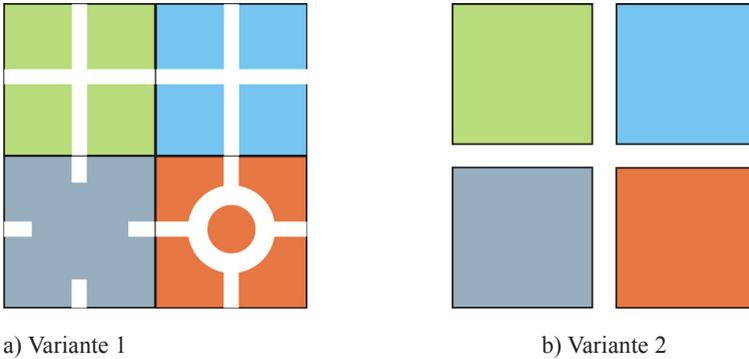


Abbildung 2: Zuständigkeiten der Peers

Abbildung 3 zeigt exemplarisch die Veränderung der Karte bei Anwendung der vom Nutzer gewählten Filter. Es ist zu sehen, dass Peers mit ähnlichem Serviceangebot in zusammenhängenden Bereichen der Karte dargestellt werden, so dass ein direkter Vergleich des Angebots erleichtert wird. Es ist ohne weiteres denkbar, mit Hilfe einer weiteren GUI-Komponente die Eigenschaften einiger Peers tabellarisch darzustellen und so die Services noch besser vergleichen zu können.



a) Perspektive=Drucken

b) Perspektive=Kommunikation

Abbildung 3: Ansichten/Perspektiven

Die Exploration des Netzwerkes ist in Abbildung 4 dargestellt. Der Nutzer kann seine Umgebung durchwandern und sieht dabei die durch Gebäude repräsentierte Peers. Findet er ein interessantes Serviceangebot, kann der Peer betreten werden. Die Innenansicht eines Peers bietet dann Möglichkeiten, auf die angebotenen Services zuzugreifen.

4 Forschungsfragen und Ausblick

Die Autoren gehen davon aus, dass ein System wie das hier vorgestellte nur dann Akzeptanz finden kann, wenn den Nutzern und Administratoren die Kontrolle über ihr eigene Hard- und Software erhalten bleibt. Daher sind zwar lokal Client-Server Architekturen denkbar, für die globale Servicelandschaft sind aber zwangsläufig verteilte, also P2P-Technologien notwendig. Neben dem Erhalt der administrativen Kontrolle beim Eigentümer spricht dafür auch die deutlich höhere Skalierbarkeit und Stabilität solcher verteilten Systeme.

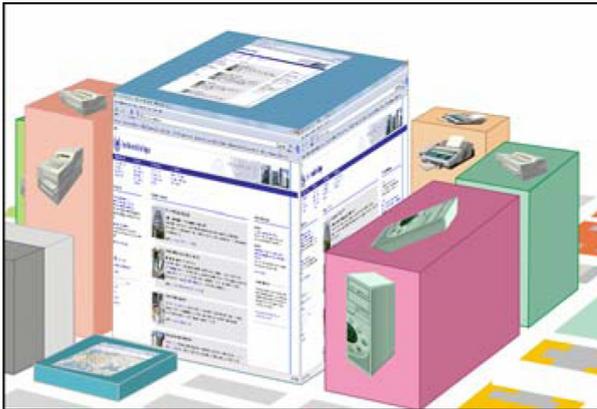


Abbildung 4: Ansicht der Netzwerkumgebung

Daher konzentrieren sich die zu beantwortenden Fragen darauf, wie auf verteiltem Wege die adaptiven Karten der virtuellen Servicelandschaft erzeugt werden können. Für den zentral gesteuerten Fall existieren bereits interessante Lösungen, die z.B. mit Hilfe von neuronalen Netzen (*Self-Organized-Maps, SOMs*) für die semantische Ordnung von Bildern und Musikstücken verwendet werden. Verteilte Lösungen ohne zentrale Instanzen existieren hingegen noch nicht. In diesem Zusammenhang ist insbesondere die Aktualität und Fehlerfreiheit der gebildeten Karten von Bedeutung. Das System muss robust genug sein, so dass temporäre Ausfälle von Peers nur lokal begrenzte Auswirkungen auf die dem Nutzer bekannte Umgebung haben. Hier müssen Mechanismen gefunden werden, die bei Bedarf einen Teil der Funktionalität des ausgefallenen Peers auf seine Nachbarn übertragen, bis dieser wieder online ist. Als interessanter Seiteneffekt können sich so auch Möglichkeiten zur Lastbalancierung ergeben.

Weiterhin stellt sich die Frage, wie unter Berücksichtigung begrenzter Bandbreiten die dreidimensionale Darstellung der aus einer bestimmten Perspektive sichtbaren Peers in akzeptabler Zeit von den Peers zu mehreren Nutzern übertragen werden soll. Hierzu könnten verteilte Caches eingerichtet werden, die dann aber semantic-aware

sein müssen und selbstverständlich nicht fehlerhafte oder veraltete Informationen weitergeben dürfen.

Im Gegensatz zu SecondLife soll der Diebstahl der eigenen virtuellen Identität nicht möglich sein, so dass ein geeignetes Trust-Management gefunden werden muss. Es existieren bereits einige auf Delegation basierende Lösungen (z.B. [11]), die ggf. für dieses System angepasst oder weiterentwickelt werden können.

5 Verwandte Arbeiten

Seit längerem gibt es Ansätze für verteilt arbeitende Betriebssysteme. Plaice u.a. stellten bereits prototypisch ein Web-Operating-System (WOS) [1][2] vor, das es ermöglichte, auf einfache Art und Weise Services im WWW zu veröffentlichen und auf diese zuzugreifen. Ähnlich wie WebOS [3] von der Berkeley University ist das Projekt allerdings in den letzten Jahren nicht weitergeführt worden.

SecondLife [4] ist eine zentralisierte Plattform einer virtuellen 3D-Welt, in der Nutzer eigene Grundstücke erwerben und bebauen können. Mittels einer eigens entwickelten Skriptsprache können interaktive Services angeboten und sogar Daten von externen Servern einbezogen werden. Interaktion mit anderen Nutzern ist möglich. Für den Handel existiert eine eigene Währung, der sog. Linden-Dollar, der gegen US-Dollar getauscht werden kann. Allerdings gibt es eine Reihe von Nachteilen des Systems: Regelmäßige Abstürze durch Überlastung der Server und begrenzte Bandbreite verhindern den Einsatz im operativen Betrieb. Identitätsdiebstahl und –missbrauch ist ebenso möglich wie der Raub des virtuellen Geldes. Ein zentraler Nachteil ist, dass Nutzer durch die gewählte Client-Server Architektur keinen Einfluss auf die Zuverlässigkeit und Verfügbarkeit ihres eigenen Service-Angebots haben. Zusätzlich bestehen Unklarheiten bzgl. der Eigentumsrechte an den durch Nutzer virtuell geschaffenen Umgebungen, die den kommerziellen Einsatz stark erschweren.

Schuster u.a. [6] stellen in ihrer Arbeit zum 3D-Web ein Konzept vor, das auf einer vollständig dezentralen Architektur aufbaut. Allerdings besteht die virtuelle Welt in diesem Ansatz aus nur lose gekoppelten Regionen, die über so genannte Tore miteinander verbunden sind. Auf diesem Wege kann das Netzwerk zwar exploriert werden, eine intuitive Orientierung scheint aber nur in Netzwerken mit wenigen Regionen möglich. Möglichkeiten zur Suche nach Services werden in der Arbeit nicht diskutiert.

Eine dichtbesetzte und zusammenhängende Darstellung der virtuellen Welt kann mittels eines verteilten Strukturalgorithmus von Sukjit und Berg [7][8] realisiert werden, der auf dem natürlichen Vorbild des Zellwachstums basiert. Der Algorithmus erzeugt basierend nur auf lokalem Wissen, eine reguläre Gitterstruktur und ordnet jedem Teilnehmer des Netzwerkes eine Position in diesem Gitter zu. Der Algorithmus ist robust gegenüber Ausfällen von Peers und korrigiert daraus entstehende Fehler selbstständig.

In [7] und [8] zeigt sich im Zusammenhang mit sog. MMOGs (*Massive-Multiplayer Online Games*), dass die Darstellung der für einen Nutzer relevanten Umgebung (*area of interest, AOI*) in Distributed Virtual Environments mit Hilfe von P2P-Architekturen signifikant besser skaliert als in Client-Server Architekturen. Sie ermitteln dazu mit Hilfe von Voronoi-Diagrammen die nächste Nachbarschaft eines (mobilen) Peers und können so gezielt nur diejenigen Teilnehmer über Veränderungen informieren, die davon betroffen sind. Allerdings beziehen sich diese Ansätze ausschließlich auf die Bewegung der Nutzer im Netz und die Verteilung der entstehenden Ereignisse. Sie beinhalten kein Konzept dafür, wie auch die virtuelle Umgebung selbst verteilt erzeugt werden kann.

Literatur

- [1] John Plaice, Herwig Unger: The Web Operating System – WOS. Distributed Communities on the Web, Springer-Verlag, LNCS 1830, Berlin Heidelberg New York, 2000
- [2] John Plaice, Peter Kropf: WOS Communities - Interactions And Relations Between Entities In Distributed Systems. Distributed Computing on the Web (DCW'99), Rostock, Germany, 1999
- [3] Amin Vahdat, Paul Eastham, Chad Yoshikawa, Eshwar Belani, Thomas Anderson, David Culler, Michael Dahlin: WebOS: Operating System Services for Wide Area Applications. Proc. 7th IEEE Symposium on High Performance Distributed Systems, 1997
- [4] <http://www.seconlife.com>, Stand vom 30. Mai 2009
- [5] A. A. Nouredine, M. Damodaran: Security in web 2.0 application development. Proceedings of the 10th international Conference on information integration and Web-Based Applications & Services, Linz, Österreich, 2008
- [6] Daniel Schuster, Moritz Biehl, Thomas Springer, Jörg Müller: Eine offene, skalierbare Architektur für 3D-Web als Informations- und Kommunikationsplattform der Zukunft. Gemeinschaften in neuen Medien (GeNeMe), 2008
- [7] S. Krause: A case for mutual notification: a survey of P2P protocols for massively multiplayer online games. In Proceedings of the 7th ACM SIGCOMM Workshop on Network and System Support For Games, Worcester, Massachusetts, 2008
- [8] S. Hu, G. Liao, G.: Scalable peer-to-peer networked virtual environment. In Proceedings of 3rd ACM SIGCOMM Workshop on Network and System Support For Games, Portland, Oregon, USA, 2004

- [9] Panchalee Sukjit, Daniel Berg, Herwig Unger: A New, Fully Decentralized Grid Generation Method. Proc. NCCIT Conference Thailand, 2009.
- [10] Daniel Berg, Panchalee Sukjit, Herwig Unger: Grid Generation in Decentralized Systems. Angenommen für IDNS Conference, Klagenfurt, Österreich, 2009
- [11] Abhishek Agrawal, Douglas J. Brown, Aditya Ojha, Stefan Savage: Bucking Free-Riders: Distributed Accounting and Settlement in Peer-to-Peer Networks. Technical Report, CS2003-0751, UCSD, 2003