TECHNISCHE UNIVERSITÄT
CHEMNITZ

Fakultät für Informatik

CSR-12-01

Studentensymposium Informatik Chemnitz 2012

Tagungsband zum 1. Studentensymposium Chemnitz vom 4. Juli 2012

Juni 2012

Chemnitzer Informatik-Berichte

Herausgeber

Technische Universität Chemnitz
Fakultät für Informatik
Straße der Nationen 62
09111 Chemnitz

Weitere Informationen zum Studentensymposium Informatik Chemnitz befinden sich im Internet unter der Adresse
http://www.tu-chemnitz.de/informatik/studsym/

ISSN 0947-5125

Organisations- und Programmkomitee

Sören Auer, General Chair
Arne Berger
Olexiy Chudnovskyy
Jörg Dümmler, Proceedings Chair
Thomas Kanzok, Review Coordinator
Vanessa Kretzschmar
Andreas Müller, Sponsoring Chair
René Oertel, Publicity Chair
Wolfgang Rehm
Michael Reißner
Paul Rosenthal
Johannes Steinmüller

Zusätzliche Reviewer

Jonathan Fischer
Walentin Heft
Michael Heidt
Tom Kühnert
Jens Kürsten
Vera Obländer
Jochen Strunk
Libor Vasa

Sponsoren des Studentensymposiums

Wir bedanken uns herzlich für die Unterstützung bei den folgenden Sponsoren.

IAV GmbH

chemmedia AG

Unister GmbH

IT Bündnis Chemnitz

ARC Solutions GmbH

AMS Gesellschaft für angewandte Mess- und Systemtechnik mbH

e-dox GmbH

SIGMA Chemnitz GmbH

msg systems ag

Stiftung IBS

Inhaltsverzeichnis

Poster Abstracts

Vorwort zum Tagungsband

In diesem Jahr fand das erste Studentensymposium Informatik Chemnitz (TUCSIS StudSym 2012) statt. Wir freuen uns, Ihnen in diesem Tagungsband studentische Beiträge präsentieren zu können.

Das Studentensymposium der Fakultät für Informatik der TU Chemnitz richtet sich an alle Studierende und Doktoranden der Informatik sowie angrenzender Disziplinen mit Schwerpunkt Informatik aus dem Raum Chemnitz. Das Symposium hat das Ziel, den Studierenden eine Plattform zu geben, ihre Projekte, Studienarbeiten und Forschungsvorhaben vorzustellen. Im Mittelpunkt des Symposiums stehen studentische Projekte aus Seminaren, Praktika, Abschlussarbeiten oder extracurricularen Aktivitäten. Das Symposium bietet die Möglichkeit, vor einem akademischen Publikum Ideen, Pläne und Ergebnisse zu präsentieren und zu diskutieren. Darüber hinaus sind Doktoranden eingeladen ihre Promotionsprojekte mit einem Poster zu präsentieren um dadurch Feedback von anderen jungen Wissenschaftlern und Professoren für ihre wissenschaftliche Arbeit zu erhalten.

In Anbetracht stagnierender oder sogar zurückgehender Studierendenzahlen in MINT Studiengängen im Allgemeinen und Informatik im Besonderen ist eine umfassende Nachwuchsförderung von besonderer Bedeutung. Studentensymposien sind eine Möglichkeit die Identifikation mit dem Studienfach Informatik und die Begeisterung für IT-Themen allgemein bei Studenten zu wecken. Bei einer Studentenkonferenz reichen Studierende kurze Artikel über Studien-, Abschlussarbeiten oder in der Freizeit absolvierte Informatik-relevante Projekte ein. Andere Studierende, Doktoranden, wissenschaftliche Mitarbeiter und Professoren bewerten und diskutieren die eingereichten Arbeiten. Interessante und gut ausgearbeitete Einreichungen werden zur Präsentation auf dem Symposium angenommen und die besten Arbeiten mit Preisen prämiert. Eine Studentenkonferenz unterscheidet sich damit kaum von einer anderen wissenschaftlichen Konferenz. Die Themenvielfalt kann allerdings durch die Breite der vertretenen Themen größer sein und die wissenschaftliche Innovation ist bei der Bewertung der Arbeiten nicht immer das primäre Kriterium. Ein Studentensymposium hilft das kreative Potential von Studierenden besser sichtbar zu machen, Studierende für die Informatik und die Forschung zu begeistern, den Austausch zwischen verschiedenen Disziplinen innerhalb der Informatik zu stärken und insbesondere das gegenseitige Verständnis von Lehrkräften und Studierenden zu fördern. Darüber hinaus ist das Studentensymposium eine wichtige Plattform zu Vorstellung von und Kontaktaufnahme mit Unternehmen der Region.

Diese erste Version des Studentensymposiums an der TU Chemnitz hat 18 Einreichungen angezogen. Die Qualitätssicherung und Differenzierung wurde allerdings nicht durch eine hohe Ablehnungsquote, sondern durch eine Einordnung der Beiträge in lange Artikel, kurze Artikel mit entsprechenden Vorträgen sowie Poster und Demobeiträge erreicht. Das Orgnisationskomitee, welches auch die Begutachtung der Beiträge organisierte, setzte sich aus Vertretern fast aller Lehrstühle der Fakultät zusammen. Mit dem Jahresbeginn 2012 hat sich das Organisationskomitee regelmäßig zusammengefunden, um alle Aktivitäten rund um das Studentensymposium zu koordinieren und organisieren. Wir freuen uns besonders, dass viele regionale Unternehmen die Bedeutung des Studentensymposiums erkannt haben und die Veranstaltung als Sponsoren sowie mit ihrer Anwesenheit und kurzen Vorstellung ihrer Unternehmen unterstützen. Nicht zuletzt soll das Studentensymposium auch als Netzwerkplattform für Studie-

rende, junge Forscher und Unternehmen der Region fungieren. Das Programm des Studentensymposiums wird durch einen Vortrag von Karsten Schulze vom Unternehmen IAV eingeleitet. Besonders freut uns, dass der neue Rektor Arnold van Zyl am Tag des Studentensymposiums die Fakultät besucht und plant sich in der Postersession unter die Teilnehmer des Studentensymposiums zu mischen. Das Studentensymposium schließt mit der Verleihung von Preisen für den besten Artikel und das beste Poster. An dieser Stelle möchten wir uns daher auch bei den Hauptsponsoren IAV und Chemmedia, sowie den Unternehmen des IT-Bündnisses und der Stiftung IBS herzlich für die Unterstützung bedanken.

Wir wünschen allen Teilnehmern des Studentensymposiums einen fruchtbaren Tag, mit vielen Gesprächen, interessanten Vorträgen und einer Menge neuer Eindrücke.

Das TUCSIS Studsym Organisationskomitee

Sören Auer, Arne Berger, Olexiy Chudnovskyy, Jörg Dümmler, Thomas Kanzok, Vanessa Kretzschmar, Andreas Müller, René Oertel, Wolfgang Rehm, Michael Reißner, Paul Rosenthal und Johannes Steinmüller

Verfahren zur Optimierung der Energieeffizienz in drahtlosen, energieautarken Sensornetzen

René Bergelt
berre@informatik.tu-chemnitz.de

Technische Universität Chemnitz
Fakultät für Informatik, Professur Technische Informatik
Betreuer: Prof. Dr. Wolfram Hardt, Dr.-Ing. Matthias Vodel

Abstract. Drahtlose Sensornetze werden heute an vielen Stellen in Wirtschaft und Forschung eingesetzt. Dabei sollen die einzelnen Sensorknoten in drahtlosen Kommunikationsszenarien autark agieren, das heißt ohne direkte Anbindung an externe Systeme, wie Stromversorgung oder ähnliches, was zur Verknappung der für die Funktionalität zur Verfügung stehenden Energie führt. Ein zentrales Forschungsziel ist deshalb die Bestimmung und Verbesserung der Energieeffizienz von Sensornetzen und deren einzelnen Knoten. In diesem Paper wird eine Definition für die Energieeffizienz von Sensorknoten aus dem allgemeinen Begriff der Effizienz hergeleitet und es wird gezeigt, wie der objektive Nutzen eines Knotens bestimmt werden kann. Darauf aufbauend werden Metriken herausgestellt, die die Optimierung der Energieeffizienz der komplexen Systeme Sensorknoten und Sensornetz vereinfachen sollen. Ebenso werden einige Optimierungsverfahren beschrieben, die sich diese Metriken zu Nutze machen. Abschließend wird auf die Problematik heterogener Sensornetze eingegangen und relevante Forschungspunkte der Professur für Technische Informatik an der Technischen Universität Chemnitz vorgestellt.

1 Zielsetzung

Sensornetze mit drahtloser Kommunikationstechnologie bieten vielfältige Anwendungsmöglichkeiten und kommen heute an vielen Stellen in Wirtschaft und Forschung zum Einsatz. Die drahtlose Kommunikation zwischen den Knoten ermöglicht es, diese ungeachtet der natürlichen Hindernisse und Unebenheiten der Umgebung anzubringen, wo kabelgebundene Lösungen nicht praktikabel sind [BAP03]. Die großflächige Verteilung begünstigt die Erfassung von physikalischen Phänomen und das Erkennen von Ereignissen, wenn ihr Auftreten nicht auf ein enges Gebiet eingeschränkt werden kann. Anwendung findet dies unter anderem bei der Erkennung von Naturkatastrophen wie Tsunamis oder Waldbränden. Ebenso kommen Sensornetze bei der Wartung von Brücken zum Einsatz, die das Verhalten und eventuelle Veränderungen in der Stabilität (zum Beispiel durch Materialermüdung) überwachen und übertragen. Aber auch Langzeitstudien, bei denen der Verlauf bestimmter Parameter in einem Gebiet über die Zeit untersucht werden soll, sind mit Sensornetzen

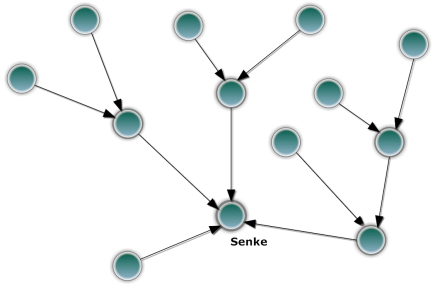


Abbildung 1: Ein exemplarisches multi-hop Sensornetz

realisierbar. Ein exemplarisches Sensornetz mit Sensorknoten und Verbindungstopologie ist in Abbildung 1 dargestellt. Der als Senke bezeichnete Knoten erhält die von den Sensorknoten gemessenen Daten und leitet diese an externe Systeme weiter, verarbeitet oder archiviert sie. All diesen Systemen ist gleich, dass die einzelnen Sensorknoten Energie benötigen, um ihre Funktion im Sensornetz erfüllen zu können. Da die einzelnen Sensorknoten in drahtlosen Kommunikationsszenarien autark agieren sollen, das heißt ohne direkte Anbindung an externe Systeme, wie Stromversorgung oder ähnliches, ist ein zentrales Forschungsziel, die Lebenszeit eines Sensornetzes so lang wie möglich zu gestalten, um die Funktion desselben so lang wie möglich aufrecht zu erhalten. Aufgrund der Knappheit der Ressourcen von Sensorknoten, vor allem an Rechenleistung, Datenspeicher und Ladungskapazität der Stromspeicher, sollen die einzelnen Sensorknoten und das gesamte Netz so effizient wie möglich mit diesen arbeiten [MM11].
Deshalb soll zunächst der Begriff der Energieeffizienz für Sensorknoten definiert werden, um feststellen zu können, ob ein Sensorknoten effizient ist beziehungsweise, ob Veränderungen an dessen Funktionsweise die Energieeffizienz positiv beeinflussen. Des Weiteren müssen Metriken gefunden werden, die eine entsprechende Abschätzung bezüglich der Energieeffizienz erlauben und folglich Optimierungen ermöglichen. Darüber hinaus sollen ausgewählte Verfahren vorgestellt werden, die die beschriebenen Metriken nutzen, um die Energieeffizienz von Sensorknoten und -netzen zu steigern.

2 Effizienzbegriff

Die allgemeine Definition der Effizienz eines Vorgangs (engl. *action*) ist das Verhältnis des erbrachten Nutzens (engl. *gain*) zu den dafür aufgewandten Kosten (engl. *cost*) wie in Formel (1) gezeigt. Ein gesetztes Ziel soll dabei mit so wenig Ressourcenaufwand beziehungsweise -verlust erreicht werden.

$$E_{\text{action}} = \frac{\text{gain}_{\text{action}}}{\text{cost}_{\text{action}}} \tag{1}$$

Zusätzlich zur Energieeffizienz existiert der Begriff der Energiekonservierung, die als direktes Ziel die Einsparung von Energie besitzt. Im Gegensatz zur Effizienzdefinition wird

dabei die Erhaltung des Nutzen eines Vorgangs nicht zwingend gefordert, so dass die maximale Energiekonservierung darin besteht, eine Aktion gar nicht erst auszuführen. Es kann somit nicht generell davon ausgegangen werden, dass ein Vorgang automatisch effizienter ist, wenn sein Energieverbrauch[1] reduziert wird.
In vielen technischen Anwendungen wird die Energieeffizienz als Wirkungsgrad bezeichnet. Der Wirkungsgrad ist dabei das Verhältnis von Nutzleistung zu zugeführter Leistung, und kann für einzelne Bauelemente vergleichsweise einfach ermittelt werden (beispielsweise über die Verlustleistung) [Sch12]. Im Gegensatz dazu stehen Sensorknoten eines Sensornetzes als komplexere Systeme, die aus einer Vielzahl verschiedener Bauteile bestehen, was die rechnerische Ermittlung erschwert. Darüber hinaus wird die Funktion eines Knotens einerseits durch seine Hardwarekomponenten und andererseits durch seine Softwareroutinen realisiert. Die Bestimmung der Energieeffizienz eines solchen Knotens und damit auch der eines Verbunds dieser auf Basis des Wirkungsgrades aller verschiedenen Bauteile ist jedoch zu aufwendig und lässt darüber hinaus die Softwareanteile der Funktionalität völlig außer Acht.

3 Energieeffizienz von Sensorknoten

Es erscheint zweckmäßig die Kosten für die Funktionalität eines Sensorknotens hauptsächlich von dessen Energieverbrauch abhängig zu machen, da die Energieknappheit bei autarken Sensorknoten die kritischste Beschränkung ist (keine Restenergie ist gleichbedeutend mit keiner Funktion und somit einem Totalausfall des Knotens). Der Nutzen eines Sensorknotens oder eines ganzen Netzes kann in der Regel hingegen nicht wertmäßig ausgedrückt werden, da es sich dabei vor allem um funktionale Eigenschaften handelt. Der Nutzen eines Knotens (engl. gain, G) soll deshalb vereinfacht bezüglich dessen Spezifikation betrachtet werden. Es ergibt sich damit folgende einfache Nutzenfunktion:

$$G_{\text{node}} = \begin{cases} 1, & \text{wenn Knoten Spezifikation erfüllt} \\ 0, & \text{sonst} \end{cases} \tag{2}$$

Dies bedeutet, dass es eine zwingende Voraussetzung für die Energieeffizienzeinschätzung eines Knotens ist, dass dieser seine Spezifikation erfüllt. Ist dies nicht der Fall, so besitzt er im Sinne dieser Definition keinen Nutzen und seine Energieeffizienz ist demzufolge nach Formel (1) gleich null. Für die Kosten eines Knotens wird eine streng monoton wachsende Kostenfunktion $c(x) : [0, \infty] \rightarrow [1, \infty]$ eingesetzt, die den Energieverbrauch des Knotens für eine Aktion auf die Gesamtkosten dieser Aktion abbildet (da je nach Anwendung beispielsweise auch zeitliche Einschränkungen in die Kostenrechnung einbezogen werden müssen). Somit kann nun die Energieeffizienz eines Knotens wie folgt definiert werden:

$$E_{\text{node}} = \frac{G_{\text{node}}}{c(W(t))} \tag{3}$$

[1] Energieverbrauch soll hier für die umgewandelte bzw. transportierte elektrische Energie stehen, die zur korrekten Funktion eines Knotens über eine bestimmte Zeit benötigt wird

Wobei $W(t)$ die elektrische Arbeit bezeichnet, die über einen betrachteten Zeitraum, von der Stromversorgung des Sensorknotens (meist einem Akkumulator) entnommen wird. Da die Kostenfunktion für einen Knoten abhängig von der genauen Implementierung und sehr vielen Rahmenbedingungen des konkreten Sensornetzes ist, ist es nicht praktikabel die absolute Energieeffizienz eines Knotens zu bestimmen. Jedoch kann mit Hilfe der Definition in (3) eine Tendenz zur Optimierung der Energieeffizienz eines Knotens angegeben werden:

$$\Delta E_{\text{node}} \sim \frac{1}{\Delta W(t)} \tag{4}$$

Dies bedeutet, dass die Energieeffizienz eines Knotens gesteigert werden kann, wenn sein Energieverbrauch gesenkt wird, unter der Voraussetzung, dass sein Nutzen erhalten bleibt. Dieser Ansatz entspricht damit dem Minimalprinzip (auch: ökonomisches Prinzip), das versucht einen gegebenen Nutzen mit einem so geringen Einsatz der (knappen) Ressourcen wie möglich zu erreichen. Kann die Lebenszeit eines Knotens gesteigert werden, ohne dass sich sein Nutzen verändert oder sein Energieverbrauch unverhältnismäßig ansteigt, so gilt auch durch Ersetzung von t durch die Lebenszeit des Knotens T_{life} in (4):

$$\Delta E_{\text{node}} \sim \Delta T_{\text{life}} \tag{5}$$

4 Metriken und Verfahren zur Steigerung der Energieeffizienz

Aufbauend auf der in Abschnitt 3 gegebenen Definition können nun Metriken[2] und darauf basierende Verfahren für die Steigerung der Energieeffizienz abgeleitet werden, die es einfacher gestalten Sensorknoten und den Sensornetzaufbau zu optimieren. Dabei ist es zweckmäßig zwischen verschiedenen Abstraktionsebenen zu unterscheiden, auf denen Optimierungspotentiale vorhanden sind.

4.1 System-Ebene

Als System-Ebene soll die Betrachtung des Sensornetzes beziehungsweise eines Knotens als Gesamtheit bezeichnet werden, es handelt sich also um die globale Sicht auf die Energieeffizienz. Die Optimierung der Energieeffizienz auf dieser Ebene erfolgt dann über die Gesamtleistungsaufnahme wie in Formel (4). Auf dieser Ebene kann nun relativ einfach festgestellt werden, ob Veränderungen am Knoten sich positiv auf seine Energieeffizienz auswirken, zur Optimierung ist sie aber unter Umständen zu komplex. Aus diesem Grund sollen im folgenden Metriken für Einzelkomponenten des Gesamtsystems vorgestellt werden, die schlussendlich alle zur Effizienzeinschätzung auf Systemebene beitragen. Da ein Sensorknoten selbst aus einem Hardware-Anteil, nämlich den physischen Komponenten aus denen er gebaut wurde, und einem Softwareteil, der mit diesen Komponenten arbeitet

[2] Die Bezeichnung wurde auf Basis von [TBP08, Beu06] (engl. metric) gewählt und soll im Sinne von Maß, Effizienzmaß verstanden werden (vgl. auch: Softwaremetriken)

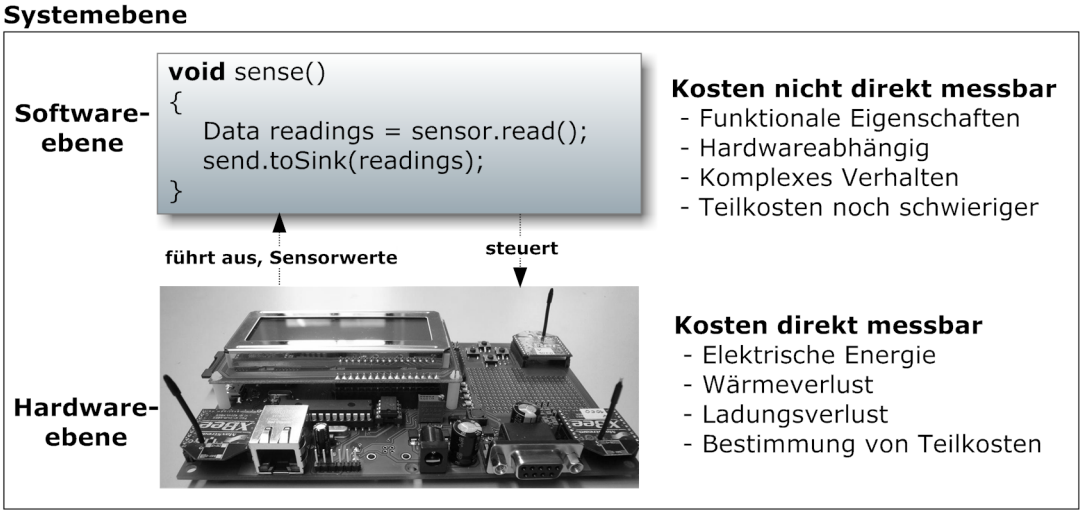


Abbildung 2: Kostenbestimmung auf verschiedenen Ebenen eines Sensorknotens

und die eigentliche Funktion diktiert, aufgebaut ist, bietet es sich an, die Effizienzmetriken für diese Teilbereiche getrennt zu betrachten. Darüber hinaus ist die Kostenbestimmung auf Hardwareebene in der Regel einfacher als auf Softwareebene wie in Abbildung 2 dargestellt. Die angegebenen Metriken beschreiben somit eine eher lokale Sicht auf die Energieeffizienz, da sie sich nur auf Teilaspekte eines einzelnen Knotens beziehen.

4.2 Hardware-Ebene

Auf Hardware-Ebene werden die einzelnen Hardware-Komponenten betrachtet, aus denen ein Sensorknoten aufgebaut ist, wie CPU, Funkmodul und die Sensoren, die Messwerte aufnehmen. Die Leistungsaufnahme der CPU kann aus Hardwaresicht jedoch nur gering beeinflusst werden, sieht man vom sogenannten Undervolting[3] ab, das aber für Sensornetze nicht praktikabel ist, da die Stabilität des entstehenden Systems nicht ausreichend garantiert werden kann. Da die CPU-Leistung jedoch auch von der Last abhängig ist, soll diese auf Softwareebene indirekt bestimmt werden (s. Abschnitt 4.3). Bei der Optimierung des Energieverbrauchs einzelner Komponenten ist es zweckmäßig, sich auf diejenigen Hardware-Module zu beschränken, die den Hauptanteil der Leistungsaufnahme eines Sensorknotens ausmachen. In drahtlosen Kommunikationsszenarien ist dies vor allem das Funkmodul, das für die Kommunikation der Knoten untereinander zuständig ist [BCS05, MSFC02, TGS07]. Die geläufigsten Metriken für die Energieeffizienz dieses Moduls sind nach [TBP08] die *Energie pro Datenpaket und Hop* (in $\frac{J}{packet*hop}$) sowie die *Energie pro Datenpaket und Meter* (in $\frac{J}{packet*m}$), also die mittlere Energie, die vom Funkmodul aufgebracht werden muss, um ein Datenpaket zu senden. Ein Hop ist dabei die

[3] Gezieltes Absenken der Versorgungsspannung der CPU unter Herstellerangaben, um die Leistungsaufnahme und Wärmeentwicklung zu senken

Übertragung von Daten von einem Sensorknoten zu einem seiner Nachbarn. Da sich die durchgeführte Aktion *Senden eines Pakets* nicht verändert, kann durch die Senkung der dafür benötigten Energie die Energieeffizienz des Knotens gesteigert werden. Zusätzlich erlaubt dies, die Energieeffizienz der Funkmodule unterschiedlicher Sensorknoten zu bewerten und zu vergleichen.
Auch die Senkung der Sendeleistung auf Basis der Verbindungsqualität zu benachbarten Knoten ermöglicht eine Energieeinsparung [BCS05]. In statischen Sensornetzen[4] führt dies nahezu immer zur Steigerung der Energieeffizienz. Die Knoten müssen dabei einmalig die Sendeleistung an die Entfernung ihrer benachbarten Knoten anpassen (auf Basis der Empfangsqualität) und nutzen von da an die niedrigste Sendeleistung, mit der die Nachbarknoten noch erreichbar sind. In dynamischen Sensornetzen hingegen muss die Sendeleistung während der Laufzeit unter Umständen korrigiert werden, wenn beispielsweise Sensorknoten ausfallen oder neu hinzukommen. Eine Möglichkeit zur Erkennung neuer Knoten ist zum Beispiel ein Broadcast-Signal mit voller Sendeleistung in bestimmten Intervallen, um neue, weiter entfernte Knoten zu entdecken. Danach kann die benötigte Sendeleistung wieder an die tatsächlich Entfernung angepasst werden. Ist in der Spezifikation eines Sensornetzes zusätzlich verzeichnet, dass Sensorknoten eine maximale Entfernung zu einander haben dürfen, die unter der maximalen Reichweite der Kommunikationsmodule liegt, so kann eine permanente Einsparung erzielt werden, wenn die Sendeleistung nur soweit angehoben wird, wie eine Verbindung in dieser Entfernung benötigt.
Ein weiterer Ansatzpunkt ist die Nutzung von Energiesparfunktionen oder sogar das vollständige Deaktivieren von einzelnen Komponenten, wenn diese nicht benötigt werden [BCS05]. In diesen Fällen muss jedoch sichergestellt werden, dass der Nutzen des Sensornetzes erhalten bleibt. Ein weiteres Problem besteht in der Entscheidung, wann eine Komponente deaktiviert werden kann, da ein zu schnelles Wiederaktivieren unter Umständen mehr Energie verbraucht als wenn die Komponente nicht in den Energiesparmodus versetzt worden wäre. Dies kann beispielsweise durch energiebewusste Kommunikationsprotokolle erreicht werden, die feste Zeitvorgaben bieten, wann eine Kommunikation stattfinden darf und wann nicht. Ein Problem dieses Ansatzes besteht in der Tatsache, dass der Knoten, dessen Funkmodul im Rahmen der Energieeinsparung deaktiviert ist, nach außen hin „taub“ und „stumm“ ist und nicht auf Anfragen von anderen Knoten oder der Senke reagieren kann. Im Zuge der aktuellen Forschung wird deshalb an sogenannten Wake-Up-Receivern gearbeitet, bei denen es sich um Funkempfänger mit äußerst geringem Energieverbrauch handelt. Ihre einzige Aufgabe besteht darin, auf ein bestimmtes, drahtlos übertragenes Steuersignal zu warten, um bei dessen Empfang das voll funktionsfähige Funkmodul des Knotens zu aktivieren (quasi aufzuwecken) und eine Kommunikation mit dem Knoten, der den Weckruf gesendet hat, zu ermöglichen. Somit kann das Hauptfunkmodul eines Sensorknotens in der Zeit, in der keine Kommunikation angefragt wird, deaktiviert werden und genau in dem Zeitpunkt, in dem Kommunikation erwünscht wird, wieder aktiviert werden, was je nach Länge der Kommunikationspausen einen mehr oder weniger großen Energieeinspareffekt zur Folge hat. Da der Sensorknoten jedoch auch in Kommunikationspausen von außen angesprochen werden kann (per Weckruf), ist es möglich, asynchrone Anfragen an diesen zu stellen. Da sich der Nutzen des Knotens, nämlich jederzeit erreichbar zu sein, nicht ändert, kann die Verwendung von

[4] Sensornetze, die eine festgelegte Topologie besitzen und deren Knotenmenge sich zur Laufzeit nicht ändert

Wake-Up-Receivern zu einer Steigerung der Energieeffizienz führen.

4.3 Software-Ebene

Die Effizienzmetriken auf Software-Ebene wirken sich natürlich in letzter Instanz auf die vorgestellten Metriken auf Hardware-Ebene aus, da die Software letztendlich mit und auf der Hardware arbeitet. Sie stellen jedoch abstraktere und unter Umständen für Entwickler verständlichere Richtlinien zur Optimierung von Sensorknoten dar. Auch kann die Optimierung auf Software-Ebene meist einfacher als auf Hardware-Ebene erfolgen, da ein Firmware-Update bei den meisten Sensorknoten ohne Weiteres möglich ist und meist schneller vollzogen werden kann als ein Modultausch. Betrachtet man die Kommunikation der Knoten untereinander im Sensornetz, dann ergeben sich als Effizienzkriterien beispielsweise die *Paketanzahl pro Zeiteinheit*, also die Menge an Paketen, die ein Knoten in einer bestimmten Zeit verschickt und die *Paketanzahl pro Aktion*, das heißt die Menge an Paketen, die im Sensornetz verschickt werden muss, um eine bestimmte Aktion, wie das Hinzufügen und Entfernen von Knoten, erfolgreich auszuführen. Können die dafür jeweils benötigten Paketmengen reduziert werden, und kann die Aktion dennoch weiterhin der Spezifikation entsprechend ausgeführt werden, senkt sich auf Basis der Hardware-Metriken *Energie pro Datenpaket* der aufgebrachte Energieaufwand und somit steigt die Energieeffizienz des Knotens beziehungsweise Sensornetzes beim Ausführen seiner Funktion.
Dieser Sachverhalt wird zum Beispiel bei Aggregationsverfahren[5] wie dem sogenannten In-Netzwerk-Verfahren verwendet, bei dem das Übermitteln der gemessenen Daten von den Sensorknoten nicht mehr einzeln pro Knoten erfolgt [MSFC02]. Jeder übergeordnete Knoten sammelt dabei die Daten seiner Kindknoten und fasst diese und seine eigenen in einem Paket zusammen, das wiederum an seinen eigenen übergeordneten Knoten gesandt wird bis die Daten die Senke erreichen. Dadurch kann die Gesamtpaketmenge im Sensornetz effektiv verringert werden. In Abbildung 3 ist dieses Verfahren exemplarisch für ein Sensornetz dargestellt. Im linken Sensornetz bezeichnen die Kantenbeschriftungen dabei, die benötigte Anzahl an Paketen, die über die entsprechende Knotenverbindung gesendet werden muss, wenn jeder Knoten seine Daten einzeln an die Senke sendet. Im rechten Bild (Anwendung des In-Netzwerk-Verfahrens) steht die Funktion $f(x, y)$ für ein Paket, das aus einer Zusammenfassung der Pakete x und y besteht. Wenn die durch das Zusammenfassen der Daten entstehende Verzögerung im Rahmen der Spezifikation des Sensornetzes bleibt, kann mit Hilfe dieses Verfahrens die Energieeffizienz gesteigert werden. Die Senkung der Paketmenge beziehungsweise des Datenverkehrs innerhalb des Sensornetzes führt dabei generell immer dann zu einer Steigerung der Energieeffizienz, wenn die spezifizierten funktionalen Eigenschaften der Aktion weiterhin eingehalten werden.

Ein weiterer Aggregationsansatz sind datenbank-orientierte Aggregationsverfahren, wie beispielsweise TinyDB und COUGAR. Bei diesen Verfahren werden die Messdaten der

[5] Aggregationsverfahren beschreiben die Vorgänge, die notwendig sind, damit die Messdaten von Knoten innerhalb des Sensornetzes zur Senke gelangen.

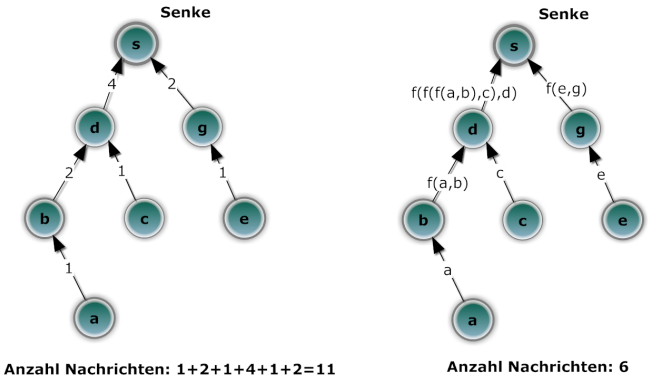


Abbildung 3: Verringerung der Paketmenge durch das In-Netzwerk-Verfahren (rechts)

Knoten nicht kontinuierlich an die Datensenke übertragen. Vielmehr kann ein Nutzer beziehungsweise eine Anwendung Abfragen (engl. *queries*) an das Sensornetz stellen [AV10]. Zu diesem Zweck betrachtet das Verfahren das Sensornetz als virtuelle Tabelle einer virtuellen Datenbank, in der jeder Knoten einen Datensatz und jede Spalte einen Sensorwert repräsentiert. Dies ist in Abbildung 4 dargestellt. Die Verfahren TinyDB und COUGAR verwenden eine SQL-ähnliche Sprache, die es dem Anwender erlaubt, vergleichsweise intuitiv Abfragen an das Sensornetz zu stellen. Eine beispielhafte Abfrage im Dialekt von TinyDB ist in Abbildung 5 dargestellt. Das datenbank-orientierte Verfahren übernimmt dann die Propagierung der Abfrage im Sensornetz, die Verarbeitung von Zwischenergebnissen für erweiterte Funktionen auf den Knoten (Maximalwert, Gruppierung etc.) und die Rücksendung der Ergebnismenge an die Senke. Die Definition von Abfragen und die Darstellung der Ergebnisse erfolgen meist über eine Client-Anwendung eines Computers, der mit der Datensenke kommunizieren kann. Durch die automatische Fusion von Zwischenergebnissen und die Beschränkung des Datenverkehrs im Sensornetz (nur Abfragen und Antworten auf Abfragen) kann durch diese Verfahren eine Steigerung der Energieeffizienz im Sensornetz erreicht werden. Darüber hinaus bietet sich für datenbank-orientierte Verfahren die Verwendung von Wake-Up-Receivern (vgl. Abschnitt 4.2) an, da die Abfragen hochgradig asynchron sind und die Sensorknoten in der Zeit, in der keine Abfragebearbeitung durchgeführt werden muss, keine Kommunikation zur Senke benötigen. Wird eine neue Abfrage gestellt, wird die Senke zuerst ihre Kindknoten per Weckruf aktivieren, die danach wiederum ihre Kindknoten aktivieren und die Abfrage im Sensornetz verteilen. Kann auf Basis der Abfrage eine Vorauswahl der Knoten getroffen werden, zu denen die Abfrage nicht gesendet werden muss, da einige Eigenschaften von diesen nicht erfüllt werden können[6], müssen diese nicht aktiviert werden, was zu einer weiteren Energieeinsparung im Sensornetz führt. Einerseits, da die deaktivierten Knoten nicht aktiviert werden müssen und andererseits da der Datenverkehr im Sensornetz verringert wird.
Wenn auf den Sensorknoten ein Betriebssystem zum Einsatz kommt, welches Multitasking unterstützt, ist es möglich die Prozessorauslastung als weitere Metrik zu betrachten. Da in diesem Fall unter anderem das Messen und die Kommunikation in verschiedene

[6]Zum Beispiel muss eine Abfrage zu Temperaturmessungen nicht an Knoten gesendet werden, die keine Temperatursensoren besitzen

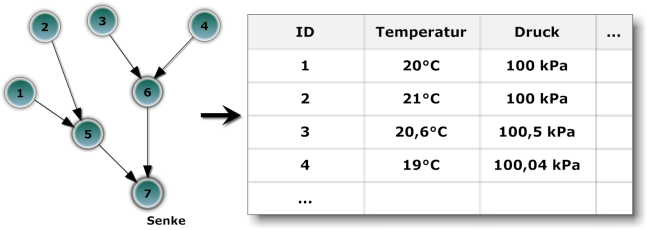


Abbildung 4: Sicht eines datenbank-orientierten Verfahrens auf ein Sensornetz

```
SELECT id, temp
FROM sensors
WHERE temp > 20
TRIGGER ACTION SetSnd(400)
SAMPLE PERIOD 60
```

Abbildung 5: Abfragebefehl im TinyDB-Dialekt

(quasi-)parallele Tasks ausgelagert werden, kann über die Zeit, die sich der Prozessor im Leerlaufprozess befindet, eine Effizienzaussage getroffen werden. Da es möglich ist, bestimmte Energiesparfunktionen im Leerlaufprozess auszuführen, ist es erstrebenswert die verschiedenen Tasks des Knotens so kurz wie möglich laufen zu lassen. Bei modernen Prozessoren, die sich bei geringerer Auslastung beispielsweise selbst niedriger takten, ist dies auch gleichbedeutend mit einer geringeren Wärmeentwicklung und somit einem geringeren Energieverlust.

5 Analyse und Optimierung

5.1 Heterogene Sensornetze

Die Optimierung eines Sensornetzes auf Basis der vorgestellten Metriken verkompliziert sich, wenn die Betrachtung nicht für homogene sondern heterogene Sensornetze durchgeführt wird. Heterogene Sensornetze liegen vor, wenn die Sensorknoten nicht von der gleichen Bauart und Beschaffenheit sind. Dies kann der Fall sein, wenn unterschiedliche Versionen der selben Firmware oder unterschiedliche Firmware-Implementationen an sich verwendet werden. Des Weiteren lässt sich der Umstand der Heterogenität in einem langlebigen Sensornetz mit großer Knotenanzahl nur schwer vermeiden, auf Grund finanzieller Beschränkungen beim Kauf von Sensoren oder dem Erweitern bestehender Systeme um neue Sensoren. Entweder weil neue Sensoren bessere Eigenschaften als die bisher genutzten liefern oder ausgefallene, zu ersetzende Sensoren nicht mehr lieferbar sind. Eine weitere Problematik besteht in drahtlosen Ad-hoc-Sensornetzen, bei denen sich die Knotenmenge dynamisch verändern kann. So können in Car2Car-Szenarien[7] jederzeit neue Automobile zum Sensornetz hinzukommen, aber auch entfernt werden, da sie sich aus dem Zielbereich bewegt haben. Zusätzlich wird die genaue Funktionalität der Sensorknoten (Autos) hochgradig herstellerabhängig sein. Nichtsdestotrotz bieten solche heterogenen Sensornetze aber auch weitere Optimierungsmöglichkeiten, da beispielsweise nicht alle Sensorknoten die gleichen Sensoren besitzen müssen. So kann einerseits eine Daten-

[7] Informationsaustausch zwischen Kraftfahrzeugen mit dem Ziel, Verkehrsbehinderungen und Verkehrseinschränkungen schnell zwischen den Verkehrsteilnehmern zu propagieren

fusion der Informationen unterschiedlicher Sensoren verschiedener Knoten helfen, weitere Informationen über das beobachtete System zu liefern und andererseits eine Vorauswahl bei der Datenaggregation auf Basis der vorhandenen Sensoren getroffen werden. So kann der Datenverkehr innerhalb des Sensornetzes gering gehalten werden.

5.2 Verfahren zur Optimierung der Kommunikation in ressourcenbeschränkten Systemen

Das Ziel einer momentan an der Technischen Universität Chemnitz durchgeführten Arbeit ist es, aus den in diesem Paper vorgestellten Sachverhalten und Metriken ein universelles Sensornetz-Aggregationsverfahren zu entwickeln, das die Energieeffizienz in heterogenen Kommunikationsszenarien mit stark ressourcenbeschränkten Knoten verbessert. Darüber hinaus erfolgt in diesem Rahmen die Betrachtung von Sensor-Aktor-Systemen, bei denen zusätzlich zur passiven Funktionalität klassischer Sensorknoten (Messen) aktive Knoten in das Sensornetz eingebracht werden. Diese können auf Basis der Messungen Einfluss auf die beobachtete Umgebung nehmen. Die Bestimmung exemplarischer Metrikwerte zur Optimierung wird dazu in einem heterogenen Sensornetz bestehend aus verschiedenen, teils an der Technischen Universität Chemnitz entwickelten Sensorknotentypen vorgenommen. Ein beispielhafter Aufbau ist in Abbildung 6 dargestellt, wobei verschiedene Formen verschiedene Sensorknotentypen repräsentieren. Der Zweck des Verfahrens ist die Laufzeitverlängerung des Sensornetzes und der einzelnen Knoten durch erhöhte Energieeffizienz. Dazu soll auf Basis eines gegebenen (statischen) Sensornetzes eine stufenweise Integration verschiedener Optimierungsverfahren erfolgen und deren Auswirkung auf die Energieeffizienz bewertet werden. Das Ziel ist dabei die Entwicklung und Erweiterung eines energiebewussten datenbank-orientierten Verfahrens, das sich sowohl für asynchrone als auch synchrone Szenarien verwenden lässt. Im Zuge dieser Entwicklung soll der tatsächliche Vorteil bei der Verwendung von Wake-Up-Receivern in hochgradig asynchronen Aggregationsszenarien ermittelt werden. Ausgehend von einem heterogenen Sensornetz mit passiver Senke[8], soll zu einem mit aktiver Senke übergegangen werden. Der Übergang zwischen Ausgangs- und Zielszenario ist in Abbildung 6 dargestellt. Der Nutzer des Verfahrens (Benutzer oder Programm) kann Abfragen über die Senke in das Sensornetz eingeben. Die Senke entscheidet, an welche Knoten die Abfrage zwingend gesendet werden muss, um sie zu beantworten. Anschließend werden diese Knoten aus dem Energiesparmodus erweckt. Jeder Knoten, der die Abfrage erhält wird sie ggf. an seine Kindknoten senden und deren Antwort mit seiner eigenen zusammenfassen, bis die Ergebnismenge die Senke erreicht. Dadurch erledigt jeder Knoten nur einen kleinen Teil der Arbeit, die notwendig ist, um die Abfrage zu beantworten. Alle nicht betroffenen Knoten (im Bild grau) müssen den Energiesparmodus nicht verlassen und können effektiv Energie einsparen. Durch die Aufgabenteilung der Knoten kann das Verfahren trotz Ressourcenbeschränkung in Rechenleistung und Speicher auch komplexe Abfragen beantworten. Dieser Ansatz entspricht somit auch dem verteilten Rechnen. Ein weiterer wichtiger Aspekt

[8] Senke, die keine Daten an die Knoten sendet, sondern nur Daten empfängt

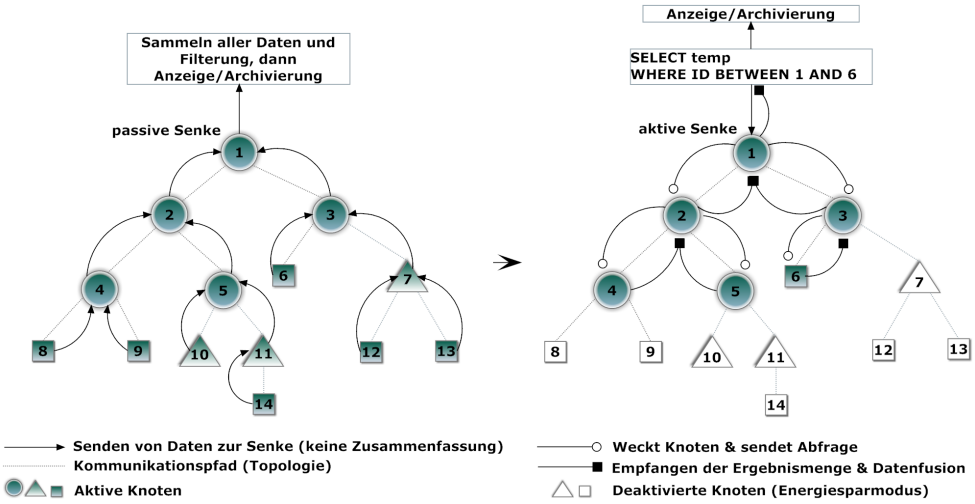


Abbildung 6: Übergang eines Sensornetzes mit passiver zu einem Netz mit aktiver Senke

ist, dass jeder Knoten die gleiche Implementation des Verfahrens verwenden können soll und sich somit der Konfigurationsaufwand des Sensornetzes vor dessen Betrieb im erträglichen Bereich befindet. Darüber hinaus wird die Datenlast im Sensornetz reduziert, da nur tatsächlich benötigte Daten übertragen werden. Ein wichtiges, vom Verfahren zu lösendes Problem, ist die Entscheidung, wann ein Knoten in den Energiesparmodus versetzt werden soll. Sind die Abfrageintervalle zu klein, so ist das Deaktivieren und das anschließende sofortigen Reaktivieren unter Umständen kostspieliger als ein durchgängig aktiver Knoten. Dazu soll eine entsprechende Heuristik gefunden werden, die es erlaubt, diese Entscheidung auf Basis des vorangegangen Verhaltens im Betrieb des Sensornetzes zu treffen. In anschließenden Forschungen soll das Aggregationsverfahren für die Anwendung in dynamischen Ad-hoc-Sensornetzen erweitert werden. Dazu müssen energiebewusste, robuste Explorations- und Routingverfahren gefunden werden.

6 Abschließende Bemerkungen

In diesem Paper wurde die Energieeffizienz für Sensorknoten definiert und es wurde begründet, dass Energieeinsparung nicht automatisch die Steigerung der Energieeffizienz zur Folge hat. Die Wichtigkeit der Feststellung des Nutzens eines Knotens wurde herausgestellt und über die Erfüllung der Spezifikation des Knotens definiert, da dieser Wert vergleichsweise einfach und objektiv ermittelt werden kann. Die vorgestellten Metriken und Verfahren können genutzt werden, um Sensornetz-Anwendungen energieeffizienter zu gestalten. In der Praxis handelt es sich aber meist um hochgradig applikationsspezifische Netze, deren Optimierungspotential mühsam untersucht werden muss. Aus diesem Grund und mit Blick auf dynamische Ad-hoc-Sensornetze wird an der Professur für Technische Informatik der Technischen Universität Chemnitz ein Aggregationsverfahren ent-

wickelt, das die Energieeffizienz eines Sensornetzes auf Basis der beschriebenen Metriken und des Laufzeitverhaltens zu verbessern sucht. Es wurde gezeigt, dass sich mit der Entwicklung von Wake-Up-Receivern ein enormes Energiesparpotential für Sensorknoten ergibt, vor allem in asynchronen Kommunikationsszenarien. Aus diesem Grund soll die Wake-Up-Funktionalität fester Bestandteil des entwickelten Verfahrens sein. Im Rahmen der Entwicklung soll jedoch auch untersucht werden, unter welchen Bedingungen dieses Energiesparpotential ausgeschöpft werden kann. Darüber hinaus erfolgt die Betrachtung explizit für heterogene Sensornetze, da in Zukunft eine Vielfalt von Sensorknotentypen in dynamischen Sensornetzen erwarten werden kann und aus den genannten Gründen auch erwünscht ist. Vor allem auch bei der Entwicklung der dynamischen Ad-hoc-Sensornetze der Zukunft, die aus tausenden winzigen, energieautarken Sensorknoten mit beschränkten Ressourcen bestehen[9], sind Energieeffizienz und Verfahren der Arbeitsteilung wichtiger denn je.

Literatur

[AV10] Ian Fuat Akyildiz und Mehmet Can Vuran. *Wireless Sensor Networks*. John Wiley and Sons, 2010.

[BAP03] Archana Bharathidasan, Vijay Anand und Sai Ponduru. Sensor Networks: An Overview. *IEEE Potentials*, 22(2):20–23, April 2003.

[BCS05] Joel W. Branch, Gilbert G. Chen und Boleslaw K. Szymanski. ESCORT: Energy-efficient Sensor Network Communal Routing Topology Using Signal Quality Metrics. In *Proceedings International Conference on Networking*, Jgg. 3420, Seiten 438–448. Rensselaer Polytechnic Institute, Troy, New York, U.S.A., Springer-Verlag, April 2005.

[Beu06] Jan Beutel. Metrics for Sensor Network Platforms. In *Proceedings ACM Workshop on Real-World Wireless Sensor Networks*, Seiten 26–30. Swiss Federal Institute of Technology (ETH), Zurich, ACM Press, New York, June 2006.

[MM11] Shilpa Mahajan und Jyoteesh Malhotra. Energy Efficient Path Determination in Wireless Sensor Network Using BFS Approach. 2011.

[MSFC02] Samuel Madden, Robert Szewczyk, Michael J Franklin und David Culler. Supporting Aggregate Queries Over Ad-Hoc Wireless Sensor Networks. *Proceedings Fourth IEEE Workshop on Mobile Computing Systems and Applications*, Seiten 49–58, 2002.

[Sch12] Adolf J. Schwab. *Elektroenergiesysteme – Erzeugung, Transport, Übertragung und Verteilung elektrischer Energie.* Springer Berlin Heidelberg, 3. Auflage, 2012.

[TBP08] Jonathan Tate, Iain Bate und Simon Poulding. Tuning Protocols to Improve the Energy Efficiency of Sensornets. 2008.

[TGS07] Niki Trigoni, Alexandre Guitton und Antonios Skordylis. Querying of Sensor Data. In João Gama und Mohamed Medhat Gaber, Hrsg., *Learning from Data Streams - Processing Techniques in Sensor Networks*, Seiten 73–86. Springer-Verlag Berlin Heidelberg, 2007.

[9] oft als „Smart Dust“ bezeichnet

Konzepte zur Integration des Widget-Nachrichtenaustauschs in User-Interface-Mashups

Christian Fischer, Olexiy Chudnovskyy[1], Martin Gaedke[2]

Technische Universität Chemnitz
christian.fischer@cs.tu-chemnitz.de
olexiy.chudnovskyy@cs.tu-chemnitz.de
martin.gaedke@cs.tu-chemnitz.de

Abstract: Widgets findet man heutzutage auf immer mehr Webseiten oder Webportalen. Durch die individuelle Kombination verschiedener Widgets in Form eines Mashups ist es für einen Nutzer auf einfache Art und Weise möglich, die gewünschten Informationen zu aggregieren. Das Konzept der Inter-Widget Kommunikation ermöglicht einen Nachrichtentransfer zwischen verschiedenen Widgets und kann damit die Endbenutzer-Entwicklung signifikant vereinfachen. In diesem Paper wird ein Überblick über die Thematik gegeben und es werden mehrere Konzepte erläutert wie eine benutzerfreundliche Integration der Inter-Widget Kommunikation mit bereits vorhandenen Widgets möglich ist. Zudem werden auch andere Kommunikations-formen über die Inter-Widget Kommunikation hinaus angeschnitten.

Keywords: IWC, Widget, Gadget, Mashup, Web 2.0

1 Einleitung

Wenn man die heutige Nutzung des Internets mit der von vor einigen Jahren vergleicht, bemerkt man besonders in Zeiten des Web 2.0 ständige Veränderungen der Nutzungsformen. Die Benutzer aggregieren ihre benötigten Informationen nicht mehr ausschließlich durch das Besuchen einzelner Webseiten, sondern mehr und mehr durch Mashups. Die Anzahl der Mashups auf programmableweb.com belegt dies eindrucksvoll. Zum Zeitpunkt der Erstellung des Papers existieren auf diesem Portal 6655 Mashups mit einer Wachstumsrate von durchschnittlich 2,7 Mashups pro Tag[3].

Ein bekanntes Beispiel für ein Mashup ist das im Jahre 2007 entstandene iGoogle, ein freies Portal der Firma Google. Indem der Nutzer die sog. Google-Gadgets individuell arrangiert, wird für den jeweiligen Nutzer und Anwendungsfall die ideale Umgebung erstellt. Die Bedienung ist zudem intuitiv und einfach gestaltet.

[1] Wissenschaftlicher Betreuer
[2] Wissenschaftlicher Betreuer
[3] www.programmableweb.com/mashups , ausgehend vom 1.6.2012 über die letzten 6 Monate betrachtet

Die Kombination der Informationen für ein Mashup kann auf unterschiedlichen Wege realisiert werden. Es wird dabei zwischen Daten-Mashups und Benutzerschnittstellen-Mashups (UI-Mashups) unterschieden. [1] Bei den Daten-Mashups liegt der Fokus auf der Integration und Verarbeitung von Daten verschiedenster Quellen. RSS Feeds, XML und CSV Dateien sind nur ein paar Beispiele dafür. Yahoo!Pipes[4], JackBe Presto[5] sind beispielhafte Vertreter der Entwicklungsumgebungen dieser Art von Mashups. UI-Mashups legen dagegen den Fokus auf die Integration bereits vorhandener Benutzerschnittstellen aus dem Internet. Dies können HTML-Inhalte anderer Webseiten ebenso wie bereits existierende Widgets sein. [1] UI-Mashups verbergen die Komplexität der Technologie hinter den Widgets. Sie sind daher für die Endbenutzer-Entwicklung von besonderen Interesse und Grundlage für diese Arbeit.

Momentan erfordern die UI-Mashups eine manuelle Zustands-Synchronisation zwischen den Widgets. Dies ist jedoch ein entscheidender Nachteil für die Unterstützung der Endbenutzer-Entwicklung. Um diese bestmöglich zu unterstützen ist es notwendig, dass die Widgets Ihren Zustand selbstständig synchronisieren können. An dieser Stelle setzt die Inter-Widget-Kommunikation (IWC) an.

Mit Hilfe der IWC ist es möglich, einen Nachrichtenaustausch zwischen verschiedenen Widgets zu realisieren, sodass komplexere Szenarien implementiert werden können. Bspw. lässt sich der Zustand zwischen einem Wetter-Widget und einem Karten-Widget synchronisieren, wenn der Nutzer den Positionsmarker auf der Karte ändert. Je nach Wetterlage könnte bei diesem Beispiel durch die Rückmeldung des Wetter-Widgets auch ein Regenradar-Overlay auf der Karte angezeigt werden.

In diesem Paper werden Konzepte erarbeitet, die es ermöglichen Widgets eines Mashups mit der IWC-Funktionalität zu erweitern. Des Weiteren wird aus Sicht des Endbenutzers auf die Benutzerfreundlichkeit dieser Konzepte eingegangen. Das Ziel ist es ferner, die Endbenutzer-Entwicklung mittels UI-Mashups und Inter-Widget-Kommunikation zu unterstützen.

In Kapitel 2 werden zunächst die vorhandenen Konzepte und Grundlagen analysiert. Eigene Konzepte für eine Plattform zur Unterstützung der Endbenutzer-Entwicklung mittels UI-Mashups und IWC werden in Kapitel 3 erläutert, wobei jeweils auch auf die Benutzerfreundlichkeit der genannten Konzepte eingegangen wird. Abschließend folgt eine Zusammenfassung und der Ausblick.

2 Endbenutzer-Entwicklung mittels UI-Mashups und IWC

Im Folgenden werden die Grundlagen der Endbenutzer-Entwicklung mittels UI-Mashups vorgestellt. Zusätzlich wird ein kurzer Überblick bezüglich bereits vorhandener Ansätze gegeben.

[4] http://pipes.yahoo.com/pipes/
[5] http://www.jackbe.com

Der Begriff Mashup kommt ursprünglich aus dem musikalischen Bereich und beschreibt dort eine Zusammenführung mehrerer Lieder unterschiedlichen Stils zu einem neuen Werk. [2] Im Zusammenhang mit Widgets und dem World Wide Web beschreibt der Begriff die Aggregation und Aufbereitung verschiedenster Daten und Quellen zu einer zentralen Instanz, um damit einen Mehrwert zu bieten. [2] In diesem Kontext ist es demnach eine Webseite, die Informationen unterschiedlichen Ursprungs vereint und dem Nutzer bereitstellt.

Da in diesem Kontext Widgets eine entscheidende Rolle spielen, muss zunächst definiert werden, was genau mit dem Begriff Widget gemeint ist: Gemäß dem World Wide Web Consortium (W3C) ist ein Widget eine für Endbenutzer konzeptualisierte interaktive Minianwendung bezüglich eines speziellen Anwendungsfalls, um Daten aus dem Internet oder lokale Daten anzuzeigen bzw. zu aktualisieren. [3] Dabei muss die Minianwendung so gepackt sein, dass diese problemlos heruntergeladen und auf einem Computer, einem Mobiltelefon oder einem anderen internetfähigen Gerät installiert werden kann. [3]

Widgets werden innerhalb bestimmter Container, bspw. Apache Wookie[6], ausgeführt und können via Schnittstellen auf die Funktionalitäten dieser zugreifen. Diese Container können wiederum von Mashup-Plattformen wie Apache Rave[7] genutzt werden. Innerhalb der W3C Spezifikation gibt es keine Beschreibung dazu, wie Widgets miteinander kommunizieren können. Es ist daher notwendig, entweder andere Widget-Formate, welche IWC bereits beinhalten, als Grundlage für die weitere Arbeit zu nutzen oder die W3C Widgets entsprechend zu erweitern. Widgets werden in anderen Domänen auch als Gadgets bezeichnet.

Wenn mehrere Widgets miteinander kommunizieren sollen, so gibt es dafür mehrere Ansätze. Zum einen ist es relevant, welche Techniken für die Nachrichtenübertragung in Frage kommen, zum anderen ist es wichtig wie die Nachrichtenverteilung geschehen soll. Bezüglich der Kommunikation zwischen den Widgets können die Bereiche orchestrierte-, choreografierte- und hybride UI-Mashups klassifiziert werden. [1] Bei der orchestrierten Variante wird die gesamte Kommunikation über ein zentrales Element gesteuert. Dies ist vergleichbar mit einem Dirigenten eines Orchesters, daher auch der Name. In der choreografierten Variante kommunizieren hingegen die Widgets miteinander und tauschen die Informationen selbst aus. Vergleichbar mit einem Chor, der nur durch Zusammenarbeit ein gutes Ergebnis erhält. Die hybride Variante vereint die ersten beiden. Die Widgets dürfen demnach miteinander kommunizieren, es kann aber auch der Nachrichtenfluss zentral gesteuert werden.

Technisch gesehen können die auszutauschenden Nachrichten über mehrere Methoden transferiert werden. Eine Liste von Ivan Zuzak [4] gibt Aufschluss über die derzeit vorhandenen Projekte und Methoden bezüglich der Kommunikationsmöglichkeiten. Eine Klassifikation über die verschiedenen Möglichkeiten kann durch [4] wie folgt festgelegt werden:

[6] http://incubator.apache.org/wookie/
[7] http://rave.apache.org/

	Inter-Widget	Intra-Widget
Same-Browser	1	3
Cross-Browser	2	4

Tabelle 1: Matrix zur Klassifizierung der Widget-zu-Widget Kommunikation [5]

Mit Hilfe dieser Klassifikation und der genannten Liste lässt sich eindeutig eine Tendenz zu Kategorie 1 feststellen. Dies ist auch nicht verwunderlich, da dies einem üblichen Mashup entspricht. Eine Webseite beinhaltet viele Widgets und diese sollen ihre Nachrichten und Daten untereinander austauschen. In dieser Arbeit wird diese Kategorie daher näher betrachtet. Der Nachrichtenaustausch geschieht auf unterster Ebene meist mit dem HTML Standard Postmessage, welcher bspw. einen Nachrichtenaustausch innerhalb des Browserfensters zwischen mehreren iframes zulässt. [4] Zusätzlich sind sog. Publish-Subscribe (pubsub) Techniken etabliert. [4] In vielen Projekten von der Liste wird bspw. der OpenAjax Hub [5] genutzt. Bei den OpenSocial Gadgets und bei Apache Rave kann damit problemlos das pubsub-Konzept angewendet werden. [6] Wegen Google+ wurden diese Funktionen bei den iGoogle-Gadgets wieder entfernt. [7] Bei den W3C Widgets ist die Funktionalität der IWC derzeit nicht vorgesehen. [8] Das pubsub-Verfahren basiert auf Nachrichtenthemen (Topics) bzw. Nachrichtenkanäle (Channels). Eine Nachricht, die über einen Kanal veröffentlicht (published) wird, empfangen alle Abonnenten (subscriber). Auf diese Art ist eine lose Kopplung der beteiligten Komponenten möglich. Es kümmert also einen publisher nicht, ob sich jemand für die zu sendende Nachricht interessiert. Das ist im Kontext von Mashups und Widgets insoweit von Vorteil, als dass dadurch problemlos Widgets ausgetauscht werden können.

Plattformen wie bspw. scrapplet.com oder das Apache Rave Projekt besitzen eine Implementierung des pubsub-Verfahrens durch den OpenAjax Hub. Es muss jedoch manuell JavaScript Code innerhalb der Widgets ergänzt werden, damit diese erweiterte Funktionalität auch genutzt werden kann. Das ist ein klarer Nachteil dieser Plattformen. Die Nutzbarkeit für den Endbenutzer hinsichtlich der IWC ist damit eingeschränkt. Derzeit ist es für einen Endbenutzer nicht ohne weiteres möglich bereits existierende Widgets, die keine IWC Funktionalität aufweisen, IWC-fähig zu machen. Anders als für die Endbenutzer-Entwicklung, stellt es für einen Entwickler kein Problem dar, da dieser seine eigenen Widgets mit dem erweiterten Funktionsumfang neu erstellen kann. Dabei gibt es bereits Ansätze um die Integration der IWC in einem Mashup für den Anwender benutzerfreundlicher zu gestalten: Bspw. gibt es Bemühungen und ein erstes Framework um Drag & Drop Funktionalitäten in Mashups zu integrieren. [9] Generell sind aber die vorhanden Lösungen sehr domänenspezifisch und daher für die allgemeine Endbenutzer-Entwicklung kaum zu gebrauchen. Bei iGoogle oder den Windows-Minianwendungen sucht man IWC derzeit vergebens.

Ein anderer Ansatz [10] bestrebt vorhandene Webstandards-basierte Widgets in Richtung IWC zu erweitern. Dies funktioniert sehr gut, jedoch erfordert diese Methode Hintergrundwissen von IWC-Konzepten und Semantic-Web Technologien. Ein

Systemverwalter kann damit, ohne selbst den Quellcode anzupassen, IWC-fähige Widgets für die Nutzer bereitstellen. Für die Endbenutzer-Entwicklung können diese Widgets entsprechend unkompliziert genutzt werden. Ohne das entsprechende Hintergrundwissen kann der Nutzer jedoch selbst keine Widgets bereitstellen. Da bei dieser Methode der Container angepasst wurde, ist eine Änderung des Quelltextes nicht notwendig. Das ist ein wichtiger Schritt in die richtige Richtung.

3 Erweiterung der UI-Mashups zur Unterstützung von IWC

Im Folgenden werden verschiedene Konzepte für die Unterstützung der Endbenutzer-Entwicklung mittels IWC genannt. Es werden jeweils die nötigen Voraussetzungen und die beabsichtigte Wirkung des Konzepts erläutert.

3.1 Containerkonzept

Besteht die Möglichkeit den Widget-Container zu beeinflussen oder durch Plug-Ins zu erweitern, steht diese Herangehensweise zur Verfügung. Der Container, in dem die Widgets gerendert und ausgeführt werden, stellt bei diesem Konzept eine zentrale JavaScript Instanz (im folgenden JS-Helper genannt) zur Verfügung. Sofern es durch den Container technisch möglich ist, durchsucht der JS-Helper alle Widgets innerhalb des Containers nach Formularelementen. Um auf bestimmte Ereignisse und Aktionen innerhalb der Widgets reagieren zu können, werden jeweils Event-Listener hinzugefügt. Die ausgeführten Aktionen werden vom JS-Helper gespeichert. Er lernt von den gewählten Aktionen des Nutzers, bei denen jeweils auch eine Kategorie durch ein Dialogsystem abgefragt wird. Die Kategorien wie bspw. Websuche, Wetter, Standort, usw. sollten thematisch abgegrenzt sein. Zusammen mit einem Wörterbuch ist es bei einer weiteren Abfrage durch die Kategorisierung möglich, mit gleichen oder ähnlichen Eingaben die hinterlegten Aktionen durchzuführen. Das Wörterbuch sollte dabei bereits einen gewissen Wortstamm mit zugehöriger Kategorisierung aufweisen. Dies kann auch durch einen zentralen Webdienst realisiert werden.

Die gespeicherten Aktionen sollen auch beim nächsten Besuch des Mashups wieder verfügbar sein. Cookies oder ein Loginsystem des Containers sind denkbare Speichermöglichkeiten. Der Nutzer muss zudem jederzeit die Möglichkeit haben, das Unterstützungssystem zu deaktivieren, falls dies nicht benötigt wird oder als störend empfunden wird. Es soll auch möglich sein, die ausgewählten Aktionen für die nächsten x Minuten auszuführen, damit der Nutzer in dieser Zeit nicht weiter gestört wird (siehe Abbildung 2).

Es ist dadurch möglich dem Nutzer ein System zur aktiven Unterstützung bei der Endbenutzer-Entwicklung anzubieten. Durch die zentrale Logik lässt sich dieses Konzept in die orchestrierte Kategorie einordnen. Die Widgets können ohne die zentrale JavaScript-Instanz nicht selbstständig miteinander kommunizieren. Ist das System einmal eingelernt, so ist eine automatisierte, indirekte Kommunikation zwischen den Widgets möglich – der Informationsfluss wird jedoch zentral vom JS-Helper gesteuert.

Abbildung 1 zeigt beispielhaft ein Wireframe eines Mashups. Der rote Rahmen soll dabei den eingebetteten JS-Helper darstellen. Durch eine Eingabe im Such-Widget oben links (Abbildung 1) soll eine vom Benutzer definierte Aktion ausgeführt werden. Bei der ersten Eingabe und dem anschließenden Senden der Nachricht mittels der Submit-Methode des Formulars kann sich der JS-Helper melden und Nachfragen, welche Aktion ausgeführt werden soll. Das Problem hierbei ist nicht die Herkunft der Daten zu bestimmen, sondern diese zum richtigen Ziel zu leiten und dort die richtige Aktion auszuführen. Dem Benutzer muss daher eine intuitive Eingabemöglichkeit gegeben werden, dies auf einfache Art und Weise zu bewerkstelligen. Abbildung 2 zeigt ein Wireframe einer möglichen Implementierung eines dialogbasierten Unterstützungssystems.

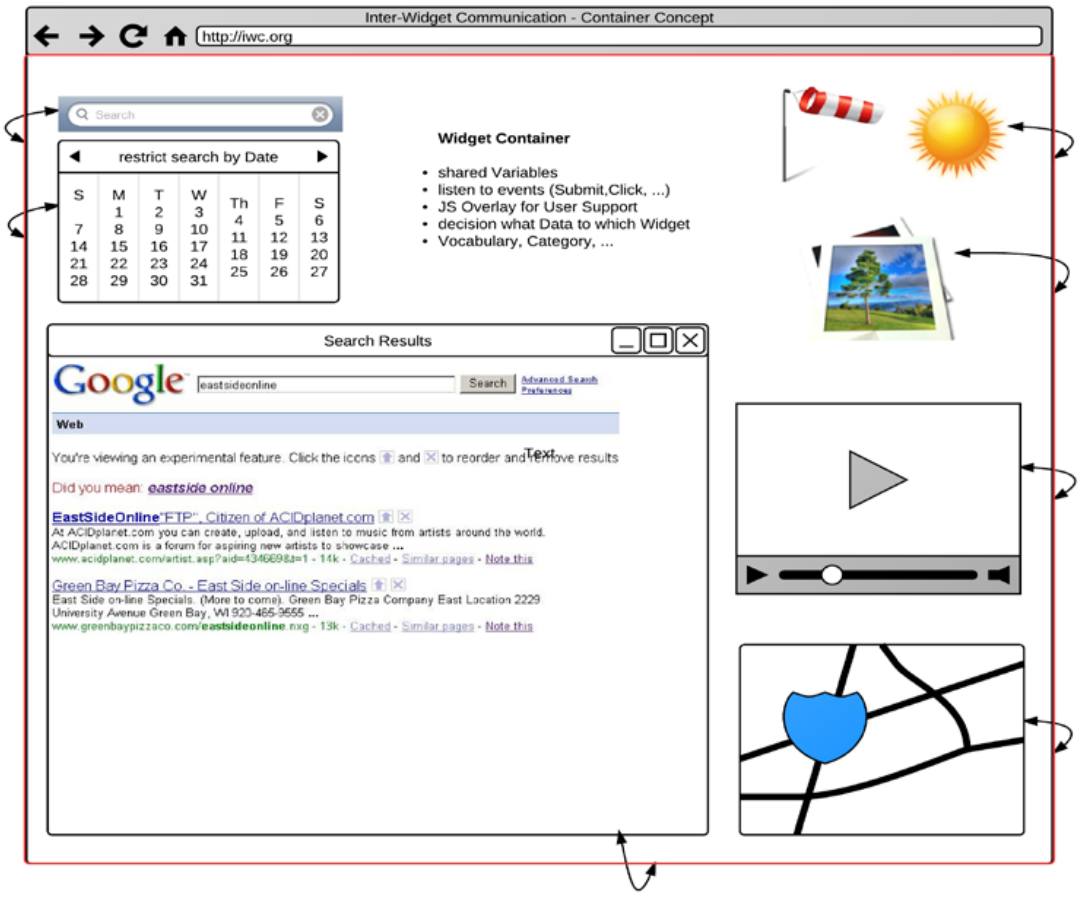


Abbildung 1: Wireframe des Containerkonzepts

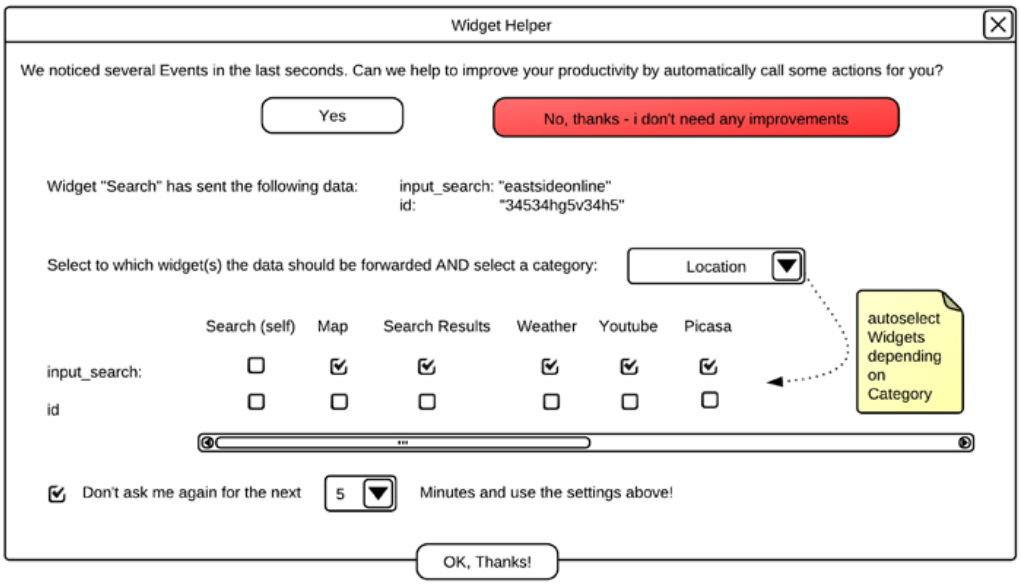


Abbildung 2: Dialogbasiertes JavaScript-Overlay des Unterstützungssystems

Eine Erweiterung des vorgestellten Ansatzes stellt die Identifikation von Quell- und Ziel-Widgets mittels Drag & Drop dar. Für den Nutzer ist dies intuitiver und damit einfacher zu handhaben. Entweder zieht der Nutzer die Daten direkt ins gewünschte Widget, indem er diese markiert und an die richtige Stelle im Ziel-Widget ablegt oder es wird mittels JavaScript ein Overlay angezeigt, wo der Nutzer von der Quelle zum Ziel einen Pfeil zieht und damit das Ziel eindeutig bestimmt. Damit der Inhalt im Ziel-Widget auch aktualisiert wird, muss bei den verschiedenen Aktionen jeweils die richtige Methode aufgerufen werden. In den meisten Fällen wird es eine Submit-Methode sein, es kann jedoch auch komplexere Ausmaße annehmen. Bei Änderungen von Radio- oder Checkboxen wird ggf. eine erweiterte dialogbasierte Methode nötig sein, da dies meist nicht sinnvoll via Drag & Drop abzubilden ist.

Um die nötigen Benutzereingaben weiter zu reduzieren, kann der JS-Helper beobachten welche Daten der Nutzer in welchem Widget eingibt und welche Aktionen ausgeführt werden. Gibt der Nutzer bspw. den Suchbegriff „eastsideonline“ im Such-Widget ein und kurz drauf wird der Begriff „eastsideonline“ auch im Wetter-Widget und im Map-Widget registriert, so kann der Nutzer, bspw. durch einen Balloon-Tipp, gefragt werden, ob die letzten Aktionen seit der ersten Eingabe des Begriffs in Zukunft automatisch ausgeführt werden sollen.

3.2 Helper-Widget-Konzept

Ist keine Möglichkeit zur Erweiterung der Plattform bzw. des Containers gegeben, so ist das gerade beschriebene Containerkonzept nicht anwendbar. Das Helper-Widget-Konzept basiert hingegen auf dem üblichen Publish-Subscribe (pubsub)-Verfahren. Der Container bzw. die Plattform muss dafür die Unterstützung des pubsub-Verfahrens anbieten. Anders als beim ersten Konzept müssen nun die Widgets durch das Einbinden

einer JavaScript-Datei (JS-Helper) ergänzt werden. Dieser JS-Helper sorgt dafür, dass sich ein Widget mit einem eindeutigen Topic am pubsub Hub anmeldet und alle Events und Daten des Widgets über dieses Topic sendet. Als Kodierung der zu sendenden Daten ist JSON zu nutzen. Für dieses Konzept ist es wichtig, dass es keine Widgets mit dem gleichen Topic gibt. Alle Widgets müssen zwingend unterschiedliche Topic-Namen besitzen. Dies ist für ein pubsub-Verfahren unüblich, aber für diesen Verwendungszweck notwendig.

Kernkomponente bei diesem Konzept ist ein Helper-Widget. Dies ist ein Widget, welches alle Topics abhört (subscribe: *), ein Unterstützungssystem anbietet und ähnlich dem JS-Helper aus dem ersten Konzept agiert. Durch das Abhören aller Topics bekommt das Helper-Widget alle Informationen der anderen Widgets und kann zentral entscheiden, an welche Topics die eingehenden Nachrichten weitergeleitet werden sollen. Auch hier ist wieder eine Lernphase notwendig bis eine automatisierte Weiterleitung der Nachrichten möglich ist. Da dieses Konzept Techniken der choreografischen Kategorie (pubsub-Methodik) und eine zentrale Komponente für die Weiterleitung der Nachrichten nutzt, ist dieses Konzept in die Kategorie der hybriden Mashups einzuordnen.

Abbildung 3 zeigt ein mögliches Mashup mit einem Helper-Widget. Am Rand jeden Widgets sind beispielhaft die Topic-Namen notiert. Das Helper-Widget empfängt bspw. eine Nachricht zu dem Topic „org.iwc.search“ vom Such-Widget. Es schaut in seinem Datenspeicher, ob bereits eine Regel dafür vorhanden ist und führt diese aus, falls dem so ist. Wenn keine Regel vorhanden ist, wird ein Dialogsystem angeboten, damit der Nutzer entscheiden kann, an welche registrierten Kanäle die Informationen gesendet werden sollen. Abbildung 4 zeigt einen möglichen Unterstützungsdialog.

Auch hier ist es notwendig, dass der Nutzer die Möglichkeit hat das System einfach zu deaktivieren. Durch die lose Kopplung beim pubsub-Verfahren, kann entweder das Widget aus dem Mashup entfernt werden oder es existiert im Widget selbst eine Funktion zur Deaktivierung (siehe Abbildung 3). Das Widget sollte außerdem verschiedene Ansichten haben, damit der Nutzer nicht zu viel Platz in seinem Mashup dafür bereitstellen muss. Denkbar ist eine Detailansicht mit den vergangenen Aktionen wie in Abbildung 3 zu sehen und eine normale Ansicht, wo nur ein Ein- bzw. Ausschalter zu sehen ist. Im Beispiel ist die Detailansicht der bereits ausgeführten Aktionen von unten nach oben zu lesen. Wie in Abbildung 4 zu sehen ist, ist auch eine Funktion zu integrieren, wo das System in den nächsten x Minuten die gleiche Aktion ausführen soll. Die Dialoge sollen zudem die Topic-Namen nutzerfreundlich darstellen. Dazu kann bspw. der umgekehrte Domänenname „org.iwc“ bei der Darstellung entfernt werden. Die gespeicherten Aktionen sollen ebenso bei der nächsten Verwendung des Mashups zur Verfügung stehen. Dies kann auch hier durch Cookies oder ein Loginsystem realisiert werden.

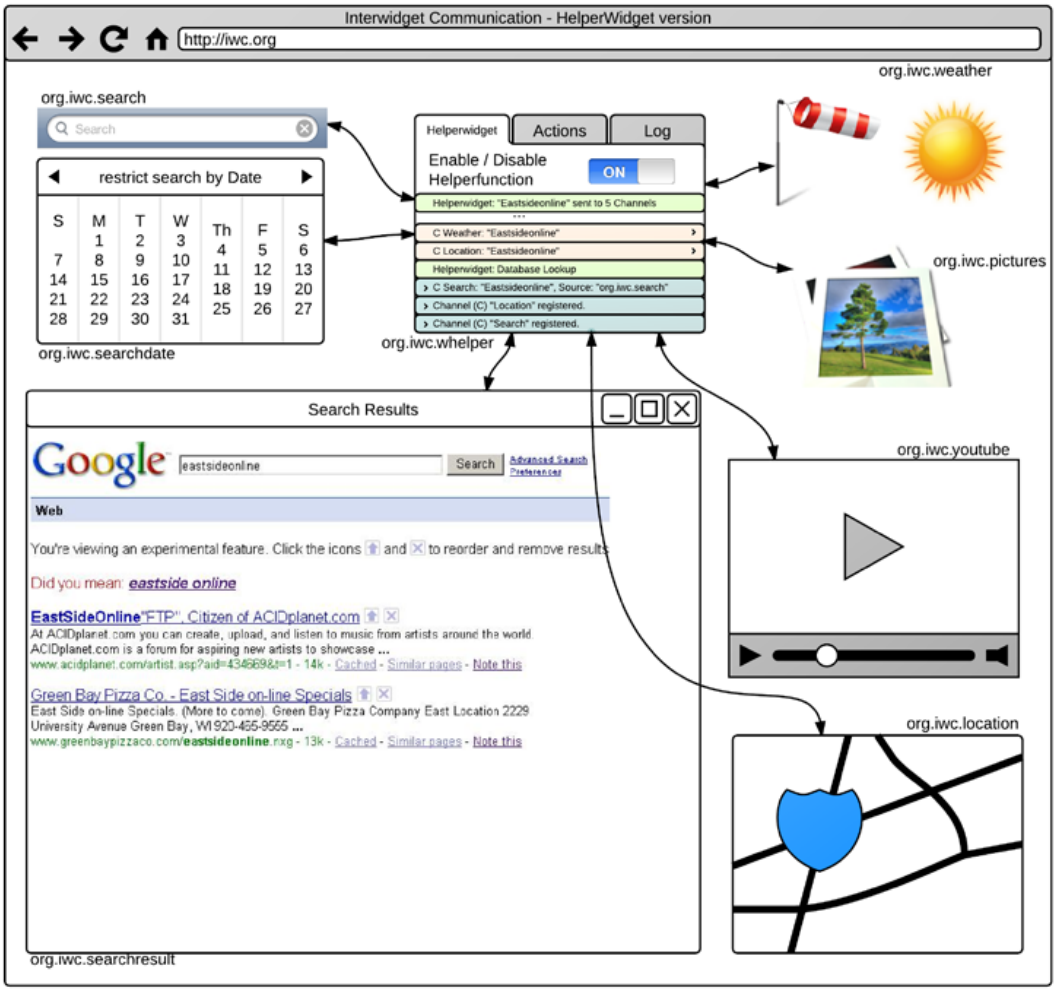


Abbildung 3: Wireframe eines Mashups mit einem Helper-Widget

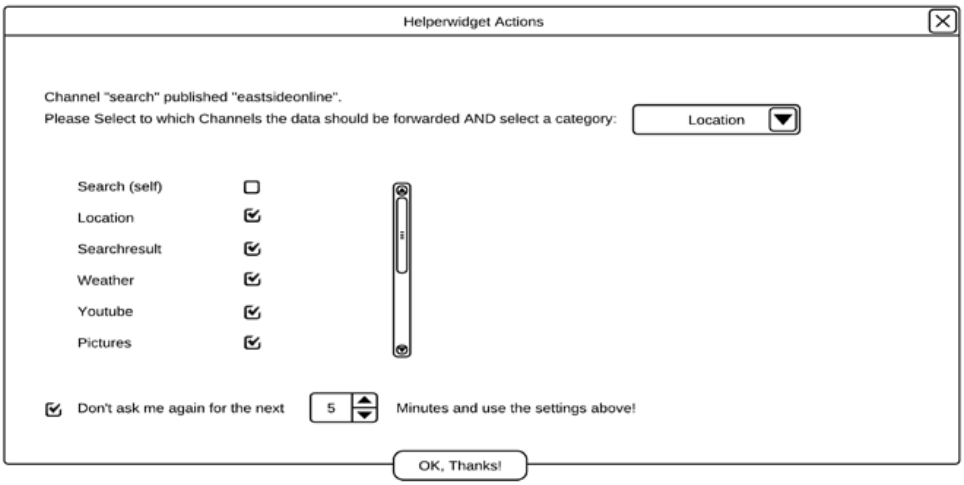


Abbildung 4: Unterstützungsdialog für das Helper-Widget

Um die Anzahl der nötigen Benutzereingaben zu minimieren, kann mit diesem Konzept auch ein passiveres Verhalten realisiert werden. Gibt der Nutzer in zwei Widgets die gleichen Daten ein, so kann dies auch vom Helper-Widget registriert werden und diese Zuordnung entweder stillschweigend übernommen werden oder es wird, wie in Abbildung 5 zu sehen, der Nutzer gefragt. Für manche Nutzer ist diese Variante besser geeignet als für andere, daher sollte in den Optionen des Helper-Widgets das Umschalten zwischen diesen Funktionsweisen möglich sein. Drag & Drop Operationen sind ebenso durch das Helper-Widget feststellbar. Dies wird durch die jeweiligen Events in den anderen Widgets und dem anschließenden Senden dieser Informationen via pubsub zum Helper-Widget realisiert. Damit können die Drag & Drop Aktionen auf die gleiche Weise behandelt werden (Abbildung 5).

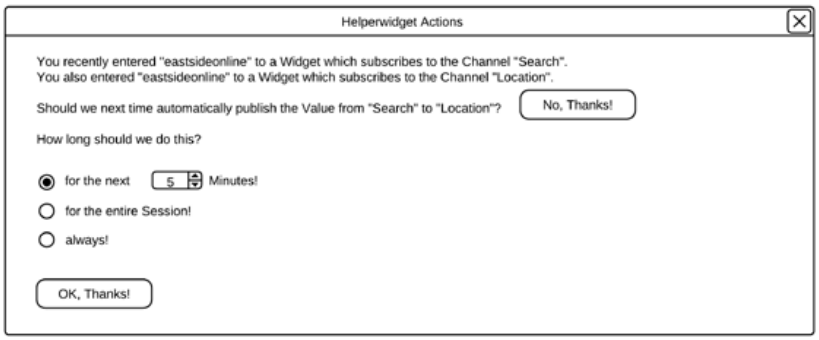


Abbildung 5: Unterstützungsdialog für das Helper-Widget bei doppelter Eingabe

Der Vorteil des Helper-Widget-Konzepts liegt bei der Nutzung vorhandener, etablierter Techniken. Der OpenAjax Hub bietet eine sichere Möglichkeit des Informationsaustausches innerhalb eines Browserfensters mit verschiedenen iframe- oder CSS-Widgets. Die Sicherheit des Gesamtsystems kann damit von vornherein auf einem hohen Niveau stattfinden. Ein Nachteil ist ganz klar die nötige Anpassung der Widgets durch das Laden der JavaScript-Erweiterung.

3.3 Mischformen und Cross-Browser / Intra-Widget Kommunikation

Je nach Anwendungsfall sind auch Mischformen der beiden genannten Konzepte möglich. Hat man bspw. die Möglichkeit den Container beliebig zu erweitern und es wird das pubsub-Verfahren genutzt, so können dadurch ggf. positive Ergebnisse erzielt werden. Man muss die Widgets unter Umständen nicht anpassen und gleichzeitig nutzt man ein bereits bewährtes pubsub-Verfahren.

Bietet eine Umgebung keine Möglichkeit der Anpassung des Containers und auch keine Funktionalität des pubsub-Verfahrens, kann über eine serverseitige Verteilung der Daten nachgedacht werden. Die Widgets müssten dann allerdings so erweitert werden, dass diese ihre Daten zu einem festgelegten Server senden. Dies ist wie im Helper-Widget-Konzept durch das Laden einer JavaScript-Erweiterung möglich. Dienste wie Firebase[8]

[8] http://www.firebase.com/

oder Pusher[9] bieten u. a. eine Möglichkeit die Daten auf schnellem Wege via Websockets ohne Long-Polling [11] an ihr Ziel zu leiten. Durch die Verteilung der Daten über einen Server sind alle genannten Bereiche der Widget-zu-Widget Kommunikation abdeckbar. So kann die IWC gleichermaßen wie die Intra-Widget und Cross-Browser Kommunikation realisiert werden. Im Allgemeinen müssen dafür allerdings die Widgets angepasst werden.

Gelingt es jedoch bspw., dass ein Container für jedes Widget standardmäßig eine Option für die Angabe eines Servers in Form einer URL hinzufügt und bei jedem Widget eine global definierte Erweiterung injiziert (ähnlich dem JS-Helper beim Helper-Widget-Konzept), so kann der Datenaustausch über einen zentralen Server erfolgen ohne dass die originalen Widgets angepasst werden müssen. Der Nutzer kann in diesem Fall bei jedem Widget einen Zielserver über die Widget-Optionen definieren. Es bietet sich dafür ein serverseitiges pubsub-Verfahren an.

4 Fazit und Ausblick

In dieser Arbeit wurden mehrere Ansätze zur Unterstützung der Endbenutzer-Entwicklung mittels UI-Mashups und IWC vorgestellt. Dabei wurden verschiedene Fälle zur Anpassung des Containers, der Widgets und ebenso Mischformen betrachtet. Es wurden mehrere konzeptuelle Ideen entworfen, welche für die Ansätze weiterer Arbeiten dienen können. Es fehlt eine prototypische Implementierung dieser Ansätze um deren Funktionstüchtigkeit im Alltag zu erproben. Es ist daher geplant, eine Implementierung auf Basis von Apache Rave, Apache Wookie und dem OpenAjax Hub zu realisieren.

Die vorgestellten Konzepte zielen auf eine benutzerfreundliche Lösung damit ein Endanwender die Widget-zu-Widget Kommunikation einfacher in ein von ihm zusammengestelltes Mashup integrieren kann. Es wird dafür ein aktives Hilfesystem angeboten. Ist die Integration der IWC erst einmal erfolgt, kann der Nutzer die Produktivität aufgrund der weniger anfallenden Benutzereingaben erhöhen. Die manuellen Zustandssynchronisierungen zwischen den Widgets werden auf ein Minimum reduziert.

Kommerzielle Dienste wie bspw. Firebase.com oder Pusher.com ermöglichen eine schnelle Übertragung von Informationen über einen Server. Diese Möglichkeit kann auch für die Widget-zu-Widget Kommunikation genutzt werden. Da diese Dienste jedoch kommerziell vermarktet werden, ist die Zukunft der Dienste ungewiss und die Sicherheit der abgelegten Daten ist auch zu hinterfragen. Um die volle Kontrolle des Systems zu behalten, ist es notwendig einen eigenen Server aufsetzen zu können. Sinnvoll wäre bspw. ein pubsub-Server mit einer Websocket-Anbindung um unnötige Latenz zu vermeiden. Um eine sichere Umgebung zu gewährleisten, muss bei den gezeigten Konzepten die Sicherheit weiter analysiert werden.

[9] http://pusher.com/

5 Literaturverzeichnis

[1] S. Wilson, F. Daniel, und U. Jugel, "Orchestrated User Interface Mashups Using W3C Widgets," *Current Trends in Web Engineering*, S. 49-61, 2012.

[2] "Mashup Definition." [Online]. Verfügbar: http://www.techterms.com/definition/mashup. [Letzter Zugriff: 24.05.2012].

[3] "Widget Packaging and XML Configuration." [Online]. Verfügbar: http://www.w3.org/TR/widgets/#definitions. [Letzter Zugriff: 24.05.2012].

[4] I. Zuzak, M. Ivankovic, und I. Budiselic, "A Classification Framework for Web Browser Cross-Context Communication," *Arxiv preprint arXiv:1108.4770*, 2011.

[5] OpenAjax Alliance, "OpenAjax Hub 2.0 Specification," 2009. [Online]. Verfügbar: http://www.openajax.org/member/wiki/OpenAjax_Hub_2.0_Specification. [Letzter Zugriff: 01.06.2012].

[6] "OpenSocial Core Gadget Specification 2.0.1." [Online]. Verfügbar: http://opensocial-resources.googlecode.com/svn/spec/2.0.1/Core-Gadget.xml#interGadgetEventing. [Letzter Zugriff: 24.05.2012].

[7] Google Inc, "Removing social functionality from iGoogle gadgets - iGoogle — Google Developers," 2012. [Online]. Verfügbar: https://developers.google.com/igoogle/docs/removing_social. [Letzter Zugriff: 01.06.2012].

[8] W3C, "Widget Packaging and XML Configuration," 2011. [Online]. Verfügbar: http://www.w3.org/TR/widgets/. [Letzter Zugriff: 01.06.2012].

[9] S. Sire, M. Paquier, und A. Vagner, "A Messaging API for Inter-Widgets Communication," *Proceedings of the 18th international conference on World wide web*, S. 1115-1116, 2009.

[10] O. Chudnovskyy, S. Müller, und M. Gaedke, "Extending Web Standards-based Widgets towards Inter-Widget Communications (to be published)," *ComposableWeb 2012*, 2012.

[11] I. E. T. Force, "RFC 6202," 2011. [Online]. Verfügbar: http://tools.ietf.org/pdf/rfc6202.pdf. [Letzter Zugriff: 01.06.2012].

Ein Personalisierungskonzept für Dataset-Repositorys am Beispiel von CKAN

Sven R. Kunze

kunsv@hrz.tu-chemnitz.de

Abstract: Mit dem Aufkommen großer sozialer Netzwerke wie Facebook sind Personalisierungsfunktionen immer wichtiger geworden. Die bereits in sozialen digitalen Gemeinschaften eingesetzten Techniken der Personalisierung können auch für fachspezifische Portale gewinnbringend eingebracht werden. Als ein solches Portal wird hier CKAN eingeführt. Es ist ein von der Open-Knowledge-Foundation entwickeltes Dataset-Repository mit dem Ziel, den Linking-Open-Data-Gedanken voranzutreiben. Dieser Artikel untersucht am Beispiel von CKAN Strategien zur Personalisierung des Community-Erlebnisses. Der Nutzer soll in die Lage versetzt werden, seinen Interessen auf CKAN zeitsparender und gezielter nachzugehen. Besonderer Fokus liegt hierbei darauf, welche Möglichkeiten der Nutzer hat, seine Interessen der Plattform mitzuteilen, und wie die Plattform den Nutzer über Ereignisse, die ihn interessieren, informieren kann.

1 Einleitung

Soziale Netzwerke wie Facebook[1], Twitter[2], Google+[3] u.v.a. haben in den letzten Jahren eine enorme Anziehungskraft entwickelt. Im April 2012 stellten bereits über 900 Millionen aktive Nutzer Beiträge, Meinungen und Bewertungen zu diesen, Fotos und andere persönliche Daten auf ihr Facebook-Profil [Hac12]. Die Nutzer des Kurznachrichtendienstes Twitter „twittern“ die neusten persönlichen Nachrichten oder auch Nachrichten von allgemeinen Interesse in alle Welt [Twi12a] und zeigen so, für welche Themengebiete sie sich begeistern lassen.

Dieser Artikel arbeitet die Thematik um Personalisierung von *Social Communitys* im Allgemeinen und im Speziellen am Dataset-Repository CKAN aus. Es ist wie folgt strukturiert. Der nachfolgende Abschnitt 2 behandelt allgemein die Vorteile von Personalisierung im Kontext von sozialen digitalen Gemeinschaften. Der Abschnitt 3 stellt die Open-Source-Software CKAN vor und im darauf folgenden Abschnitt 4 wird ein Konzept zur Personalisierung von CKAN präsentiert. Um die Anschaulichkeit des Konzepts zu erhöhen, zeigt der vorletzte Abschnitt 5 an zwei Beispielen konkrete Personalisierungen. Der letzte Abschnitt gibt eine kurze Zusammenfassung der Problembeschreibung und die wesentlichen Punkte des Personalisierungskonzeptes wieder.

[1] http://www.facebook.com/
[2] https://twitter.com/
[3] https://plus.google.com/

2 Personalisierung von Communitys

Neben dem Einstellen von Daten teilen die Nutzer ihre ganz persönlichen Vorlieben den Systemen mit. Aus diesen *Personalisierungen* ergeben sich eine Reihe von Vorteilen sowohl für die Nutzer als auch für die Plattformen [SK02].

Die Nutzer bauen durch das Preisgeben ihrer Interessen eine Beziehung zu der Plattform, die es zu erhalten gilt, auf [SK02]. Durch diese Beziehung erhält der Nutzer einen umfassend auf ihn zugeschnittenen Zugang zu den Daten in einer Community [SK02, GNOT92], die er sonst in ihrer vollen Form nicht erfassen könnte [Gro00].

Ganz explizit wird dieses Konzept bei Twitter benannt [Twi12b]. Durch das „Folgen“ bestimmter Nutzer erscheinen ausschließlich deren Beiträge auf der Hauptseite des „Folgenden“.

Neben dem Filtern aus dem Datenüberfluss wird das Community-Erlebnis durch das gezielte Hinzufügen von genau an die Bedürfnisse und Interessen der Nutzer angepasste Informationen bereichert [Har07]. Hierbei handelt es sich unter anderem um Verweise auf Profile weiterer Nutzer oder Gruppen, die zum Interessenbereich des jeweiligen Nutzer gehören [Har09]. Auch entsprechende kommerzielle Angebote in Form von Werbung lassen sich mithilfe von persönlichen Präferenzen punktgenau platzieren.

Hierbei treten nun die entsprechenden Vorteile für die Plattformen zutage. Neben geeigneten Werbeangeboten, die in der Tat einen erheblichen Umfang an den Einkünften vieler sozialer Plattformen darstellen [WDO09], erzielen auch personenbezogene Daten bzw. statistische Erhebungen aufgrund ihres Wertes für das Marketing einen sehr guten Preis auf dem Markt [Gol11, Adr12].

Wenn darüber hinaus der Zweck einer Community der Verkauf von Produkten oder Dienstleistungen ist, z.B. Amazon[4], so ist eben schon deswegen eine starke Kundenbindung unerlässlich. Diese lässt sich ebenfalls mit Personalisierung erreichen bzw. fördern [SK02, PR97].

Über die finanziellen Aspekte hinaus erreichen die Plattformbetreiber einen Zufluss an neuen Mitgliedern. Dies wirkt sich indirekt vorteilhaft auf beiden Seiten – Nutzer und Plattform – aus. Die Plattform wird durch die Beiträge neuer Mitglieder bereichert, was den Nutzern zugute kommt, und diesen neuen Nutzern kann man gezielte Werbung – basierend auf ihren persönlichen Einstellungen und Beziehungen zu anderen Nutzern – einblenden.

3 Das Dataset-Repository CKAN

CKAN[5] bezeichnet eine Open-Source-Software, gepflegt von der Open-Knowledge-Foundation [OKF12b], mit deren Hilfe man Daten in komponentenbasierter Form verwalten kann [OKF11a]. Diese Komponenten lehnen sich an den Gedanken des Packages in Debi-

[4] http://www.amazon.com/

[5] Comprehensive Knowledge Foundation Archive Network; siehe auch `http://ckan.org/`

an an und werden als Dataset bezeichnet.

Nutzer können sich auf dieser Plattform registrieren und dort die Daten, die sie mit anderen teilen wollen, veröffentlichen [OKF12c]. Dies kann durch Verlinkung mithilfe einer Web-Adresse oder durch direktes Hochladen der Ressourcen geschehen [Pol11]. Eine Vielzahl von Ressourcenformaten, z.B. bekannte Dateiformate, SPARQL-Endpunkte, Google Docs Spreadsheet etc., werden von CKAN unterstützt [OKF11b].

Neben der bloßen Bereitstellung der Datasets, ist es von Vorteil, diese mit weiteren Metadaten auszustatten [OKF11b], um eine leichtere Auffindbarkeit durch andere gewährleisten zu können.

In Abbildung 1 ist die Startseite von „the Data Hub“ (CKAN-Version 1.7) dargestellt. Sie bietet einen Überblick über die Funktionalitäten in CKAN, die ein Nutzer wahrnehmen kann. Über die Menüleiste wird sichergestellt, dass der Nutzer zu jeder Zeit neue Datasets hinzufügen oder suchen und sein Nutzerkonto aufrufen und bearbeiten kann.

Eine Übersicht über den in CKAN vorhandenen Funktionsumfang finden sich in Abbildung 2. Hieraus wird ersichtlich, das CKAN auch auf administrativer Seite viele Konfigurationsmöglichkeiten mit Themes, APIs etc. zulässt. Zusätzlich besitzt CKAN eine Extension-Infrastruktur zur modularen Erweiterung des Funktionalitätsumfangs.

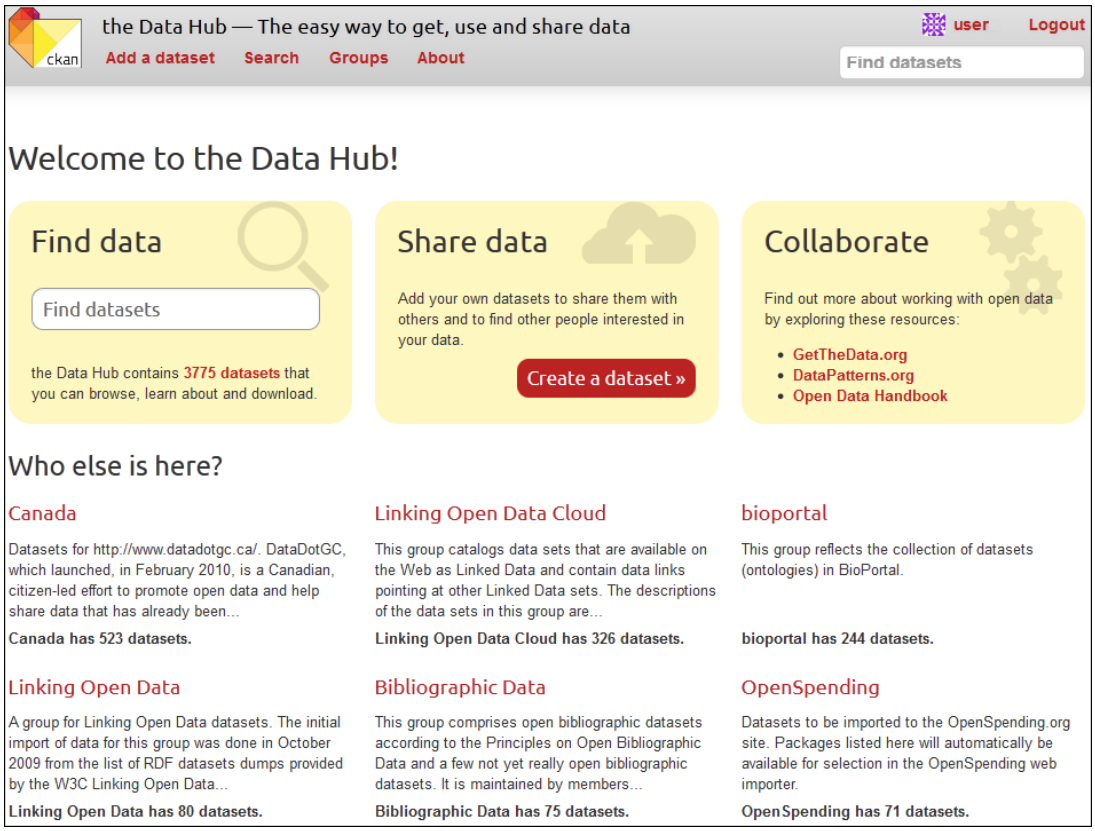


Abbildung 1: Die populäre CKAN-Instanz „the Data Hub“ bot im Mai 2012 über 3700 Datasets an. Die Nutzerschnittstelle bietet einen komfortablen Weg zum Bereitstellen, Suchen und Gruppieren von Datasets an.

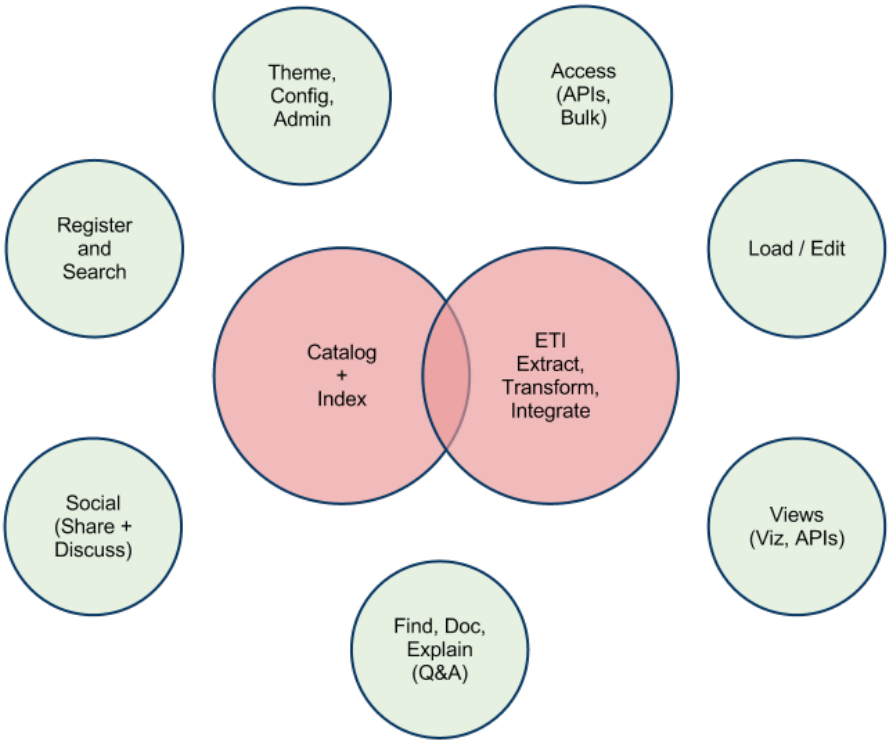


Abbildung 2: CKAN-Features[6]. Von größerem Interesse ist hier der untere linke Bereich „Social“, der im Rahmen einer Masterarbeit ausgebaut werden soll.

CKAN trägt somit wesentlich zur Stärkung des Web-of-Data-Gedankens bei, in dem es die hierfür erforderliche Verknüpfung von Daten ermöglicht [OKF12a, W3C07].

4 Problemstellung und Lösungskonzept

CKAN verfügt gegenwärtig über ein sehr minimalistisches Personalisierungskonzept, das kaum über die Registrierung, Anmelden und das Eintragen der hierfür erforderlichen Daten hinausgeht. Dies wird auch in der Abbildung 2 ersichtlich. Der Schwerpunkt wurde eindeutig auf die Verwaltung von Datasets gelegt.

Die Problemstellung ist es, die in Abschnitt 2 erläuterten Ideen auf das Dataset-Repository CKAN anzuwenden. Dies soll wie beschrieben eine tiefere Integration und Identifizierung der Nutzer mit der Plattform ermöglichen, andere ermutigen an dem Projekt teilzuhaben, Daten mit anderen zu teilen und somit die Entwicklung von LOD (Linking Open Data) weiter voranzubringen. Hierbei stehen einige Funktionalitäten, die für ein umfassendes

[6] `http://wiki.ckan.org/Design_and_Architecture`

Personalisierungskonzept weiter ausgebaut und zusammengebracht werden müssen, bereits zur Verfügung.

4.1 Activity-Streams

Nach dem Anmelden des Nutzers soll dieser auf eine Seite, die ihm einen Überblick über die aktuellen Vorkommnisse in der Community informiert, weitergeleitet werden. Dieser Strom an Informationen wird als *Activity-Stream* bezeichnet. Gegenwärtig wird ausschließlich der des Nutzer, auf dessen Profilseite man sich gegenwärtig befindet, angezeigt.

Der vereinheitlichende Activity-Stream auf der Hauptseite soll nun nicht nur die Aktivitäten des jeweils angemeldeten Nutzers, sondern noch weitere für ihn relevante Ereignisse in der Community anzeigen. Dem Nutzer wird somit eine Auf-einen-Blick-Übersicht über die neusten Vorkommnisse, die ihn betreffen oder ihn interessieren, informiert.

4.2 Personalisierung

Um den Nutzer geeignete Beiträge im Activity-Stream anzeigen zu können, ist es notwendig, seine Interessen genau zu kennen. Hierbei wurde in der neusten Version von CKAN, 1.8a, die Funktionalität des *Folgens von Datasets und Nutzern* nach dem Vorbild von Twitter implementiert.

Neben dem Folgen von Datasets und Nutzern sind weitere Formen der Personalisierung möglich. So bietet CKAN zur Verwaltung zusammengehöriger Datasets „Gruppen“, die von einzelnen Mitgliedern administriert werden, an. Um Datasets mit bestimmten inhaltlichen Eigenschaften zu markieren, steht so genannten „Tags“ zur Verfügung. Diese Entitäten (Gruppen und Tags) sollen ebenso in die Personalisierung zur Anreicherung des Activity-Streams aufgenommen werden.

Neben den durch die Autoren/Administratoren von Datasets gepflegten und festen Strukturen kann der Nutzer die Ad-Hoc-Suchfunktion nutzen, um Datasets aufzufinden. Ein weiterer Schritt in Richtung eines umfassenden Personalisierungskonzeptes soll mit der permanenten Überwachung solcher Suchergebnisse gegangen werden.

Über zwei Arten können Nutzer beliebte Suchanfragen permanent abspeichern, um im Falle einer Änderung des Suchergebnisses, z.B. wenn ein neues Dataset hinzugefügt wurde, oder bei Änderungen an den Metadaten einzelner Datasets eine aussagekräftige Meldung zu erhalten.

Zum einen kann der Nutzer direkt eine bereits durchgeführte Suche speichern; ähnlich der Follow-Funktionalität. Die Suchanfrage wird auf einer separaten Seite „Interessen“ angezeigt und kann dort entsprechend geändert oder gelöscht werden.

Zum anderen bietet die Interessen-Seite verschiedene Rubriken, wie „Orte“, „Sprachen“, „Sport“, „Finanzen“ etc., in denen der Nutzer seine Interessen spezifizieren kann, an. Die

Idee hierbei ist, dem Nutzer begriffliche Anker zu geben, an denen er sich orientieren kann. Hier sei besonders die „Orte“-Rubrik, die neben der bloßen Angabe geographischer Begriffe auch die von Geo-Koordinaten ermöglicht, hervorgehoben.

Als speziellste Form der Personalisierung sind vom Nutzer angegebene SPARQL-Anfragen[7] an einzelne Datasets denkbar, um so auch tiefere Veränderungen in Datasets, wie das Entfernen oder Hinzukommen einzelner RDF[8]-Tripeln, zu realisieren. Eine wichtige Aufgabe hierbei ist das Bereitstellen geeigneter Vorlagen für einen schnellen Einstieg des Nutzers.

4.3 Benachrichtigungen

Zwar stellt ein einheitlicher Activity-Stream eine sehr gute Möglichkeit, die neusten Ereignisse geeignet zusammenzufassen, dar, allerdings kann man von den wenigsten Nutzern ständige Präsenz auf einer Plattform verlangen. Eine gute Möglichkeit, den Nutzer dennoch über Änderungen zu informieren, ist der Email-Versand.

Diese Emails sollten in ebenso konziser Form wie der Activity-Stream die neusten Begebenheiten zusammenfassen und einen einfachen Zugang zum Dataset-Repository mithilfe von Web-Adressen ermöglichen.

Um den Nachrichtenverkehr möglichst klein zu halten, ist es sinnvoll nicht sofort nach jedem Ereignis auf der Plattform eine einzelne Email zu versenden. Anstelle dessen kann das Warten auf andere Ereignisse innerhalb eines bestimmten Zeitfenster und die Aggregation dieser Ereignisse in einer Email das Fluten von Email-Postfächern vermeiden.

4.4 Weitere Personalisierungswerkzeuge

„Folgen“ ist nur eine der möglichen Verbindungen, die ein Nutzer zu einem Dataset aufbauen kann. Daneben kann ein Nutzer bestimmte Datasets explizit weiterempfehlen oder sie bewerten. Auch wäre es möglich, sich Datasets zu bestimmten Themenbereichen zu wünschen und so andere Nutzer zu motivieren, ebensolche Datasets einzustellen.

Eine weitere Möglichkeit wäre die gegenseitige Integration der Activity-Streams von Facebook, Twitter, Google+ etc. Auf diese Art und Weise würde es dem Nutzer ermöglicht, unabhängig davon, auf welcher Plattform er sich gerade aufhält, über die neusten Ereignisse informiert zu werden.

[7] SPARQL Protocol And RDF Query Language; eine standardisierte Abfragesprache, entwickelt an RDF-Daten Anfragen stellen zu können; siehe http://www.w3.org/TR/rdf-sparql-query/

[8] Ein Framework zur Beschreibung von Ressourcen; siehe http://www.w3.org/TR/REC-rdf-syntax/

1	Personalisierung
1.1	Folgen von Nutzern
1.2	Folgen von Datasets/Gruppen/Tags
1.3	Empfehlen von Datasets/Gruppen/Tags
1.4	Interessen-Seite zur Verwaltung der Vorlieben der Nutzer
1.5	„Wunschzettel“ für Datasets zu bestimmten Themenbereichen
1.6	Integration in die Activity-Streams verschiedener sozialer Netzwerke
2	Benachrichtigungen
2.1	Einheitlicher Activity-Stream für jeden Nutzer
2.2	Email-Versand
3	Hilfestellungen
3.1	Einblenden von Empfehlungen
3.2	Anbieten verschiedener Rubriken auf der Interessen-Seite
3.3	Bereitstellen verschiedener SPARQL-Vorlagen
4	Konfiguration
4.1	Möglichkeit zur Bearbeitung und Aufhebung jeglicher Personalisierung
4.2	Möglichkeit zur Einschränkung der Sichtbarkeit für andere

Tabelle 1: Personalisierungskonzept im Überblick.

4.5 Übersicht über die Personalisierungsfeatures

Zur besseren Übersichtlichkeit werden die einzelnen Punkte des Personalisierungskonzepts für CKAN geeignet gruppiert in Tabelle 1 zusammengetragen werden.

5 Anwendungsfälle

Als mögliche Zielgruppe kommen insbesondere Menschen, die ein hohes Informationsbedürfnis in speziellen Bereichen haben, in Frage; z.B. Journalisten, Politiker oder Unternehmensberater. Aber auch interessierte Bürger, die schnell und umfassend informiert sein wollen, können die Plattform zur Informationsgewinnung nutzen.

Darüber hinaus sind die Personalisierungsfunktionen auch für Firmen interessant. Nicht nur können sie über bestimmte rechtliche und politische Entscheidungen auf dem Laufenden gehalten werden. Ebenso können sie mithilfe von Personalisierungen CKAN als Präsentationsplattform nutzen.

Dieser Abschnitt illustriert an zwei Beispielen die konkreten Einsatzmöglichkeiten der vorgestellten Personalisierungsoptionen in CKAN.

5.1 Journalistische Tätigkeit

Der Journalist X verfasst Artikel für eine regionale Zeitschrift. Um seine Arbeit, zu der neben dem Schreiben auch die Recherche gehört, zu beschleunigen, hat er sich vor einiger Zeit an einem CKAN-getriebenen Portal angemeldet und seine Interessen („ÖPNV“[9], „Kindertagesstätten“, „Medizin“) eingepflegt. In der Rubrik „Sport“ gab er an, Neuigkeiten zum Thema „Fußball“ erhalten zu wollen. Ebenso folgt er dem regionalen Sportverein Y.

Durch die Angabe seiner Präferenzen erhielt X nun vor Kurzem eine Email mit der Nachricht, dass die Kindertagesstätte K in seinem Ort renoviert wird. Da diese Email einen Web-Link auf die entsprechende CKAN-Seite enthält, ist es dem Journalisten möglich, schnell und einfach zusätzliche Daten zu dem Ereignis „Renovierung der Kindertagesstätte K“, wie Zeitraum, Kosten, Verantwortlicher etc., abzurufen.

Nach einiger Zeit stellt X fest, dass er überregionale Nachrichten zum Thema „Fußball“ erhält. Allerdings ist er nur am regionalen Ballsport interessiert, so dass er seine Einstellungen anpasst.

Er entfernt den Eintrag „Fußball“ aus der Sport-Rubrik und legt stattdessen eine SPARQL-Anfrage, ähnlich der in Listing 1, an. So ist er nun in der Lage, alle die Datasets, die in wirklich interessieren, abzufragen und wird über Änderungen an diesen durch seinen Activity-Stream und per Email benachrichtigt.

Listing 1: Mithilfe dieser SPARQL-Anfrage kann man alle Datasets, welche die Ontologie http://example.org/fussball verwenden und deren Entitäten in einem bestimmten geographische Gebiet liegen, abfragen. Die Anfrage verwendet hierbei das von LODStats[10] eingesetzte VoiD[11]-Vokabular. LODStats stellt neben den von den Autoren auf CKAN eingetragenen Metadaten weitere Daten über die LOD-Datasets bereit.

```
prefix void: <http://rdfs.org/ns/void#>
prefix location: <http://www.w3.org/2003/01/geo/wgs84_pos#>
select ?dataset
where
{
    ?dataset a void:Dataset.
    ?dataset void:vocabulary <http://example.org/fussball>.
    ?dataset void:propertyPartition
        [
            void:property location:long;
            void:min ?longmin;
            void:max ?longmax.
            filter(?longmin > XX && ?longmax < YY)
        ].
    ?dataset void:propertyPartition
        [
```

[9] Öffentlicher Personennahverkehr

[9] Vocabulary of interlinked Datasets; siehe auch http://vocab.deri.ie/void

[10] Ein Statistik-Framework für RDF-Daten; siehe auch http://stats.lod2.eu/

```
            void:property location:lat;
            void:min ?latmin;
            void:max ?latmax.
            filter(?latmin > XXX && ?latmax < YYY)
        ].
}
```

5.2 Selbstporträt eines Unternehmens

Die Verkehrsbetriebe Z entschieden sich ihre Geschäftszahlen, Kontaktdaten und Bekanntmachungen auf einer CKAN-Instanz zu veröffentlichen. Hierdurch werden nun potenziellen Kunden, Lieferanten oder auch der Journalist aus Unterabschnitt 5.1 über Änderungen der Geschäftsbedingungen, Aktionen etc. informiert.

Neben einer reinen Datenpublikation gewinnt Z ebenfalls Informationen darüber, wer sich alles für das Unternehmen interessiert (Liste der „Folgenden“) und so für eine eventuelle Zusammenarbeit bereit wäre.

Zusätzlich bietet CKAN auch für Z die Möglichkeit über firmenspezifische Themengebiete informiert zu werden. Dies könnten beispielsweise „StVO“[12], „Schulferien“ oder „EU-Abgasverordnung“ sein.

6 Zusammenfassung

Dieser Artikel hat die grundlegenden Eigenschaften eines Personalisierungskonzeptes untersucht und dies auf ein konkretes Beispiel, CKAN, übertragen.

Viele soziale Netzwerke haben bereits Techniken, die den Aufbau und Erhalt der Nutzerbasis fördern und von denen auch fachlich spezialisierte Communitys, wie CKAN, profitieren können, entwickelt [Har09]. Hierzu zählen im Wesentlichen das Einpflegen von Interessen und der Aufbau von sozialen Bindungen.

Die Interessen, die Nutzer auf CKAN nachgehen können, beziehen sich auf verlinkte veröffentliche Daten (Datasets). Hier wird es entscheidend sein, wie man aus den Nutzerpräferenzen die wichtigen Datasets herausfiltern und den Nutzer über Änderungen dieser in prägnanter Form benachrichtigen kann. Für ersteres soll auf die meist verbreitetsten Personalisierungsmittel, wie „Folgen“ oder „Empfehlen“, zurückgegriffen werden. Für letzteres sind Activity-Streams und Emails ein bereits gut erprobtes Mittel.

[12] Straßenverkehrsordnung

Literatur

[Adr12] Adressen, Branchenadressen, Adressbroker, Adressen online. http://www.adressen.de/, Juni 2012.

[GNOT92] David Goldberg, David Nichols, Brian M. Oki und Douglas Terry. Using collaborative filtering to weave an information tapestry. *Commun. ACM*, 35(12):61–70, Dezember 1992.

[Gol11] David Goldman. Your phone company is selling your personal data. http://money.cnn.com/2011/11/01/technology/verizon_att_sprint_tmobile_privacy/index.htm, November 2011.

[Gro00] Peter Gross. *Die Multioptionsgesellschaft: Edition Suhrkamp ; 1917 = N.F., 917*. Suhrkamp, Frankfurt am Main, 8. Auflage, 2000.

[Hac12] Mark Hachman. Facebook Now Totals 901 Million Users, Profits Slip. http://www.pcmag.com/article2/0,2817,2403410,00.asp, April 2012.

[Har07] F. Maxwell Harper. Encouraging Contributions to Online Communities with Personalization and Incentives. In *Proceedings of the 11th international conference on User Modeling*, UM '07, Seiten 460–464, Berlin, Heidelberg, 2007. Springer-Verlag.

[Har09] F. M. Harper. *The impact of social design on user contributions to online communities*. Dissertation, University of Minnesota, 2009.

[OKF11a] OKFN. Purpose - CKAN. http://wiki.ckan.org/Purpose, Dezember 2011.

[OKF11b] OKFN. Resource – CKAN Data Management System Documentation 1.8a documentation. http://docs.ckan.org/en/latest/domain-model-resource.html, Mai 2011.

[OKF12a] OKFN. Linked Data - CKAN. http://wiki.ckan.org/Linked_Data, Januar 2012.

[OKF12b] OKFN. Projects — Open Knowledge Foundation. http://okfn.org/projects/#ckan, Juni 2012.

[OKF12c] OKFN. User Stories - CKAN. http://wiki.ckan.org/User_Stories, Marz 2012.

[Pol11] Rufus Pollock. Storage Extension for CKAN. http://ckan.org/2011/05/16/storage-extension-for-ckan/, Mai 2011.

[PR97] Don. Peppers und Martha Rogers. *Enterprise one to one : tools for competing in the interactive age / Don Peppers and Martha Rogers*. Currency Doubleday, New York :, 1st ed.. Auflage, 1997.

[SK02] Petra Schubert und Michael Koch. The Power of Personalization: Customer Collaboration and Virtual Communities. In *Proc. Americas Conf. on Information Systems (AMCIS2002)*, Seiten 1953–1965, Dallas, TX, August 2002.

[Twi12a] Twitter. Verschiedene Arten von Tweets und wo sie zu sehen sind. https://support.twitter.com/groups/31-twitter-basics/topics/109-tweets-messages/articles/313052-verschiedene-arten-von-tweets-und-wo-sie-zu-sehen-sind#, Juni 2012.

[Twi12b] Twitter. Verschiedene Arten von Tweets und wo sie zu sehen sind. https://support.twitter.com/groups/31-twitter-basics/topics/104-welcome-to-twitter-support/articles/324311-twitter-101-wie-beginne-ich-mit-twitter#, Juni 2012.

[W3C07] W3C. Linking Open Data. http://esw.w3.org/SweoIG/TaskForces/CommunityProjects/LinkingOpenData, August 2007.

[WDO09] Lea M Wakolbinger, Michaela Denk und Klaus Oberecker. The Effectiveness of Combining Online and Print Advertisements. *Journal of Advertising Research*, 49(3):360, 2009.

Sprachmodelladaption von CMU Sphinx für den Einsatz in der Medizin

Christina Lohr

christina.lohr@informatik.tu-chemnitz.de

Abstract: Man stelle sich folgendes Szenario vor: Ein Patient kommt in ein Krankenhaus und wird operiert. Jeder Behandlungsschritt, der am Patienten durchgeführt wird, wird aus u.a. Versicherungs und Abrechnungsgründen dokumentiert. Der Arzt, der die Operation durchgeführt hat, verfasst dazu einen Bericht und schildert dabei in Sätzen sein Vorgehen vom Anfang bis zum Ende. In vielen Einrichtungen läuft dieser Vorgang so ab, dass ein Arzt den verbalen Operationsbericht per Diktiergerät einspricht und eine Person aus der Verwaltung hört diesen ab und arbeitet die Daten in das elektroische Verwaltungssytem ein. Automatische Spracherkennungssoftware könnte diesen Schritt enorm vereinfachen. Aus verschiedenen Gründen wird bereits existierende Spracherkennungssoftware kaum einsetzt. Spracherkennungssoftware muss bei sehr spezifischen Wörtern und großen Wortschätzen, was bei medizinischen Fachvokabularen der Fall ist, an den Nutzer angepasst werden. Dieser Vorgang ist sehr aufwendig und auch wenn eine kostenintensive Software aufwendig an den Nutzer und zusätzliche Wörter angepasst wurde, heißt das nicht, dass Erkennungsquoten von bis zu 95-100 Prozent bzw. einer Fehlerrate von maximal fünf Prozent erkannt werden können. CMU Sphinx ist ein Framework, mit dem man selbst Modelle aufbauen und anschließend Sprache erkennen kann. In meiner Bachelorabreit habe ich mich gefragt, wie viel Wörter ich mit einem kleinen Testkorpus erkennen kann, wenn ich dieses selbst zusammenstelle. Im Zusammenarbeit mit vier Ärzten der Klinik für Allgemein- und Viszeralchirurgie vom Klinikum Chemnitz habe ich verschiedene Sprachmodelle trainiert und anschließend ausgewertet. Die Sprachmodelle wurden aus drei verschiedenen häufig durchgeführten OP-Bereichen der Allgemein- und Viszeralchirurgie zusammengestellt. Die Ergebnisse und Fehlerraten, die dabei enstanden sind, sind dabei zwar nicht im gewünschten Bereich von 95-100 Prozent und können bisher so nicht eingesetzt werden. Aber für einen Sprecher lag die Erkennungsrate knapp über 85 Prozent und zeigt, dass ein zielgerichtetes Training für einen spezifischen Anwendungsbereich sinnvoll ist. Die hier entstandenen Modelle bilden somit für eine mögliche Weiterentwicklung eine Grundlage.

1 Einleitung

Im Gesundheitswesen ist es aus rechtlichen Gründen nicht möglich, auf *Dokumentation* zu verzichten. [mus] Über verschiedene Wege und Möglichkeiten werden die unterschiedlichsten Daten gesammelt, gespeichert, geordnet und wieder zugänglich gemacht, da das menschliche Erinnerungsvermögen und auch das von Ärzten, Krankenschwestern und anderen Personen, die im Gesundheitswesen arbeiten, begrezt ist. Ein Schichtbetrieb eines Krankenhauses sowie die Kommunikation zwischen Ärzten verschiedener Einrichtungen

und Praxen wie auch Behandlungsabrechnungen wären ebenfalls ohne ausreichend dokumentierte Befunde nicht möglich. Durch *Dokumentation* entsteht jedoch im Gesundheitswesen ein sehr großer Verwaltungsaufwand, der mit enormen Kosten verbunden ist. Bei Dokumenten, die das Ärztepersonal selbst verfassen muss, geht dabei viel Zeit verloren. Ärzte müssen im Verlauf von Behandlungen Briefe, Operationsberichte und Schreiben jeglicher Art immer wieder neu verfassen. So müssen zum Beispiel Operationsberichte zeitnah nach jeder Operation (OP) vom Ärztepersonal erstellt werden. In vielen Kliniken ist es derzeit üblich, dass Ärzte mithilfe eines Diktiergerätes den Verlauf einer Operation aufzeichnen. Später wird das Diktat von einer Schreibkraft abgehört, getippt und in das elektronische Verwaltungssystem der Einrichtung eingearbeitet. Dabei kommt es nicht selten vor, dass zwischen dem Diktat und dem Vorliegen des endgültigen Operationsberichtes sehr viel Zeit verloren geht. Maschinelle Erkennung von Sprache würde die Erstellung von wichtigen Dokumenten im Gesundheitswesen, zum Beispiel Operationsberichte oder Arztbriefe, erheblich beschleunigen. Dadurch könnten Arbeitsaufwände verringert und Kosten gespart werden. Allerdings wird bereits exisitierende Spracherkennungssoftware in Krankenhäusern kaum eingesetzt. Gründe dafür sind hohe Lizenzkosten digitaler Aufnahmesysteme und für Spracherkennungssoftware. Wortschätze, die den Softwaresystemen zur Verfügung standen, waren in den letzten Jahren noch nicht umfangreich genug, und somit setzen viele Einrichtungen Spracherkennungssoftware in den meisten Bereichen bis heute kaum ein.

2 Aufgaben und Ziele

Derzeit existieren viele unterschiedliche Softwareprodukte zur Erkennung gesprochener Sprache - auch für die Medizin. Im Klinikum Chemnitz wurde die Software *Dragon* vor einigen Jahren getestet und die Resultate waren nicht zufriedenstellend. Aus diesem Grund wird in den meisten Stationen bis heute keine Spracherkennungssoftware eingesetzt. *CMU Sphinx* ist ein Framework zur Erstellung von Sprachmodellen und Erkennung von Sprache. Für viele Sprachen existieren bereits *gute* Modelle, jedoch nicht für die deutsche Sprache und nicht im Zusammenhang mit medizinischen Wörtern. Mithilfe von CMU Sphinx wurde ein Sprachmodell für einen sehr kleinen spezifischen Wortschatz aus dem Bereich der Allgemein- und Viszeralchirugie erstellt. Grundlage bildete eine Projektarbeit an der TU Chemnitz aus dem WS 2010/2011. [FTRL11] Die entstandenen Erkennungsquoten wurden im Anschluss getestet und mit einer kommerziellen Spracherkennungssoftware für medizinische Anwendungen, *Dragon Medical* von *Nuance Communications*, verglichen.

3 Einige Grundlagen der Spracherkennung

Die grundlegenden Zusammenhänge der Sprachverarbeitung und die zur Erkennung von Sprache sind sehr komplex, da viele Aspekte aus unterschiedlichen Disziplinen zusammenspielen. Ausgesprochene Laute können aufgrund vieler Faktoren, wie die Anatomie

des menschlichen Sprechapprates und des Hörempfinden des Individuums, das gesprochene Sprache aufnehmen soll, sehr unterschiedlich ausgesprochen und wahrgenommen werden. Deshalb ist es schwierig, Sprache durch Computer zu erkennen. Die Hintergründe der Spracherkennung werden hier nur kurz betrachtet. - Alle notwendigen Bestandteile und Informationen, die ein System zur Spracherkennung benötigt, fasst man unter dem Begriff *Sprachmodell* zusammen. Ein Sprachmodell besteht aus Teilmodellen und ein Teilmodell repräsentiert alle zur Verfügung stehenden akustischen Merkmale. Es wird als *Akustisches Modell* bzw. *Acoustic Model* oder *Speech Model* bezeichnet. Neben den akustischen Elementen befasst sich ein zweites Modell mit Sprache im *abstrakten Sinne*. Dieser Teil wird in der Literatur ebenfalls oft mit *Sprachmodell* bezeichnet. Um Missverständnisse auszuschließen, wird das Modell, das abstrakte Beobachtungen der jeweiligen Sprache beinhaltet, hier als *Language Model* definiert und das Akustische Modell als *Acoustic Model*. Der Begriff *Sprachmodell* wird hier als Überbegriff für die beiden Modelle betrachtet. Die grundlegenden Komponenten des Sprachmodells sind in Abbildung 1 dargestellt und werden in den folgenden Abschnitten näher beschrieben. [PK08] [Ste12] [Wen04]

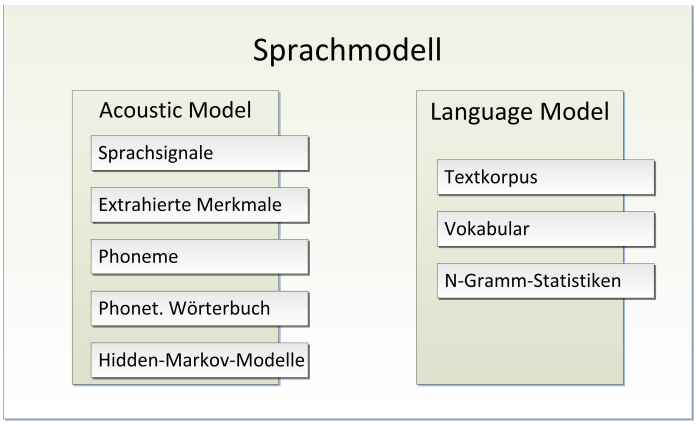


Abbildung 1: Überblick über Bestandteile eines Sprachmodells

3.1 Das Acoustic Model

Im *Acoustic Model* werden alle akustischen Informationen eines Spracherkennungssystems verarbeitet. Dazu gehören die kleinsten Bestandteile einer gesprochenen Sprache, *Phoneme*, und alle Wörter in phonetischer Beschreibung, die im *phonetischen Wörterbuch* gesammelt werden. Nur dort hinterlegte Wörter können erkannt werden. Alle zur Verfügung stehenden Audiosignale mit dazugehörigen Merkmalsextraktionen gehören mit dazu. Ein Acoustic Model schätzt von einer gegebenen Merkmalssequenz eine Wortfolge mit der höchsten A-posteriori-Wahrscheinlichkeit aus einer Folge von Wörtern und kann entscheiden, welche Wörter erkannt werden. Für eine Wortfolge kann mit Hidden-Markov-Modellen (HMM) die Variabilität von entsprechenden Merkmalssequenzen beschrieben werden. Neben HHMs für einzelne Wörter müssen Modelle für *stille Ereignisse* festgelegt sein und es ist möglich, lange Pausen zwischen Wörtern zu erkennen. [PK08, 12] [Wen04]

3.2 Das Language Model

Ein *Language Model* beinhaltet einen großen Textkorpus, sammelt ohne Wissen über akustische Merkmale *A-priori-Kenntnisse* über Sprache und beobachtet Wortreihenfolgen sowie deren Vorkommnisse im *abstrakten Sinne*. Ein Language Model gibt an, welche Wörter oft und wie häufig sie verwendet werden. Unter zwei Gesichtspunkten werden Language Models aufgebaut: *statistisch* und *wissensbasiert*. Statistische Modelle ermitteln ihre *Erfahrungen* durch Messungen und das Zählen von Wörtern aus einem relativ *großen* Textkorpus. Dabei wird Sprache so betrachtet, als sei sie *zufällig* entstanden. Wissensbasierte Modelle fundieren auf Beobachtungen aus der Linguistik. Da da Language Modell mit medizinischen Texten nach statistischen Verfahren aufgebaut wird, wird das wissensbasierte Modell hier nicht näher betrachtet. Statistische Verfahren verfolgen den Ansatz, dass Sequenzen von Wörtern Größe N, sog. *N-Gramme*, in ihrer Häufigkeit betrachtet werden und somit Vorhersagewahrscheinlichkeiten ermitteln. Es hat sich gezeigt, dass es für relativ gut aussagekräftige Vorhersagen ausreicht, Wortgruppen mit drei zusammenhängenden Wörtern bzw. *Trigrammen* zu betrachten - Über zusätzliche Schlüsselwörter, wie $Start$ und $Ende$, kann in einer N-Gramm-Statistik registriert werden, ob ein Wort am Anfang oder am Ende eines Satzes steht. Dadurch ist es möglich, einzelne Wörter, die zum Beispiel häufig am Satzanfang vorkommen, mit syntaktischen Strukturen in Verbindung zu bringen. [PK08, 13] [Fin03, 6] [Wen04, 3]

4 Erstellung eines Sprachmodells für einen ausgewählten medizinischen Wortschatz

In den folgenden Abschnitten wird beschrieben, wie unter der Verwendung der Werkzeuge von *CMU Sphinx* ein Sprachmodell für einen ausgewählten Wortschatz erstellt wird. Es werden nur die Elemente betrachtet, die im Verlauf der Abarbeitung notwendig sind.

4.1 Die Werkzeuge von CMU Sphinx

An der US-Universität Carnegie Mellon (CMU) entstand durch das Projekt *CMU Sphinx* eine gleichnamige Programmbibliothek, mit der Spracherkennungssysteme auf der Basis von Hidden-Markov-Modellen entwickelt werden können. Die Schnittstellen, die seit 2000 als Open Source zur Verfügung stehen, gliedern sich in verschiedene Komponenten auf, wovon jede für eine bestimmte Aufgabe entwickelt wurde. [Uni12] [Ger06, 4.2]

4.1.1 Das CMU - Cambridge Statistical Language Modeling Toolkit (cmuclmtk)

Das *CMU-Cambridge Statistical Language Modeling Toolkit* (Version *v2*) (CMUlmtk) besteht aus einigen ausführbaren Dateien, die per Kommandozeile angesteuert werden. Zur Erstellung eines Language Models wird nur ein Teil der Funktionen aus dem Toolkit

verwendet. Aus einem großen Textkorpus in einer *.txt*-Datei lässt sich nun ein Language Model erstellen. Markiert man vor der Abarbeitung der einzelnen Programmdateien im Ausgangstext mit XML-Tags Sätze (Bsp. <s> *Das ist ein Satz.* </s>), Abschnitte oder Paragraphen, so können Grammatikstrukturen in den N-Gramm-Statistiken berücksichtigt werden. Allerdings müssen die Struktur-Tags in einem *context cue File* bzw. in einer *ccs*-Datei festgelegt sein. In den ersten Schritten wird eine Häufigkeitstabelle und eine Vokabelliste des vorliegenden Textes ausgegeben. Mit diesen Ausgangsdateien wird dann eine N-Gramm-Statistik erstellt, wobei das N vom Nutzer aus festgelegt werden muss. Standardmäßig ist $N = 3$ voreingestellt. Zum Schluss wird aus der N-Gramm-Statistik, der Vokabelliste und den Context Cues das Language Model (*.lm*-Datei) ausgegeben. [CR97]

4.1.2 SphinxBase und SphinxTrain

SphinxBase und *SphinxTrain* stellen die grundlegenden Module für den Aufbau eines Sprachmodells bereit. SphinxBase beinhaltet APIs, die von SphinxTrain und den Recognizern PocketSphinx, Sphinx3 und Sphinx4 genutzt werden. [Sphc] SphinxTrain besteht aus einigen Perl-Skripten, mit denen ein Acoustic Model trainiert werden kann. Als *Training* wird der Prozess bezeichnet, womit ein Modell aus verschiedenen Merkmalssequenzen *lernt*. In Vorbereitung auf das Training müssen vorliegende Sprachaufnahmen in *wav*-Dateien in *mfc*-Dateien mithilfe des Skripts *make_feats.pl* gewandelt werden. Sind alle Dateien vollständig, kann der Trainingslauf mit *RunAll.pl* gestartet werden. Während des Trainings werden werden Hidden-Markov-Modelle erstellt und anschließend durch den *Baum-Welch-Algorithmus* trainiert. [Sphd] [Ger06, 4.2.2]

4.1.3 Die Recognizer

Recognizer bzw. *Erkenner* sind nach dem Training für die *eigentliche Spracherkennung* zuständig, evaluieren ein Modell und geben die Wortfehlerrate an. Von CMU Sphinx werden verschiedene *Recognizer* bereitgestellt: *Sphinx4, Sphinx3* und *PocketSphinx*. Sphinx3 ist in C geschrieben und auf eine hohe Erkennungsrate ausgerichtet. Dabei wurde jedoch die Laufzeit vernachlässigt und Sphinx3 kann für komplizierte Spracherkennungsaufgaben in Interaktionen nicht verwendet werden. Mit der Java-Entwicklung Sphinx4 wurde versucht, diese Probleme zu beheben und es ist nun möglich, Modelle während des Laufzeit zu erweitertn. PocketSphinx ist für den mobilen Einsatzbereich und eingebettete Systeme entwickelt.

4.1.4 CMU Sphinx und Sprachmodelle verschiedener Sprachen

CMU Sphinx bietet *fertige* Sprachmodelle einiger Sprachen an. Dazu gehören auch verschiedene englisch- und US-englischsprachige Modelle, da CMU Sphinx ursprünglich in den USA konzipiert war. Es gibt u. a. Modelle der Sprachen Russisch, Französisch, Spanisch und mexikanisches Spanisch sowie Mandarin und Deutsch. Der Umfang der angegebenen Sprachmodelle ist sehr unterschiedlich. Ein Sprachmodell für die russische Sprache hat eine Größe von ca. 100 MB, ein chinesisches Modell 85 MB, das mexikanische 50

MB, ein US-englisches 108 MB, ein französisches Modell 197 MB und ein deutsches nur rund 8 MB. Die *Größe* eines Sprachmodells sagt prinziell nichts über dessen Qualität aus, zeigt aber einen gewissen Aufwand und so ist das französischsprachige Modell durch ein Projekt an der *Universite du Maine* entstanden. [spha] [DEMM05] Einige Modelle, auch das relativ kleine deutsche, entstammen dem *VoxForge-Projekt*. Dahinter verbirgt sich ein Open-Source-Projekt, das Sprachdaten verschiedener Nationalitäten sammelt und *frei* zur Verfügung stellt. [vox12] Der Wortschatz des deutschen Sprachmodells mit derzeit rund 3000 Einträgen im phonetischen Wörterbuch und 4000 eingesprochenen Sätzen ist sehr eingeschränkt und beinhaltet viele Wörter aus der Mathematik, Informationstechnolgie und Rechtswissenschaft. In einer vorherigen Projektarbeit wurde an der TU Chemnitz ein kleines Sprachmodell zur Erkennung von Beiträgen aus Lokalnachrichten unter Berücksichtigung des sächsischen Dialektes entwickelt. Viele Wörter wurden dabei mit unterschiedlichen Aussprachevarianten angegeben. Bei der Entwicklung wurde das deutschsprachige Modell von VoxForge eingebunden. Es wurde festgestellt, dass das phonetische Wörterbuch zum Teil maschinell erstellt ist und gravierende Fehler enthält. Diese Fehler wurden korrigiert und die Erkennungsrate lag bei ca. 12 Prozent. [FTRL11]

4.2 Vom Text zum Language Model

4.2.1 Auswahl des Wortschatzes

Ausgangpunkt für das Sprachmodell bilden 70 Operationsberichte, die von der Klinik für Allgemein- und Viszeralchirurgie vom Klinikum Chemnitz durch den Chefarzt Prof. Dr. Boese-Landgraf mit freundlicher Genehmigung zur Verfügung gestellt wurden. Die Protokolle wurden von drei verschiedenen Operationsbereich aus drei verschiedenen (und häufig erkrankten) Fachbereichen aus der Allgemein- und Viszeralchirurgie gewählt. Die Struktur von Operationsberichten am Klinikum Chemnitz ist durch eine Vorlage des elektronischen Verwaltungsystems von *SAP* vorgegeben. Ein Teil der Daten wird über ein Eingabeformular des Systems vor und nach einer Operation durch das Personal eingegeben. Zu diesen Daten gehören *Allgemeine Patientendaten*, *Zeitliche Angaben zur Operation*, *Risikofaktoren und Infektionsrisiko*, *OP-Team* sowie *Operationsdiagnosen*. Ein weiterer Teil des Dokuments besteht aus einem *Verbalen OP-Bericht*, woraus der Textkorpus für die Sprachmodelle zusammengestellt ist. Ein Überblick über die Größe des zusammengestellten Korpus ist in Tabelle 1 dargestellt.

Bereich	# Dokumente	Σ Wörter	Min	Max	Ø
1	30	7167	63	626	238,9
2	20	3755	110	298	187,7
3	20	5956	89	615	297,8
1-3	70	16878	63	626	241,1

Tabelle 1: Überblick Textkorpus

4.2.2 Bearbreitung des Eingabetextes und Entstehung des Language Models

Da mithilfe des CMUlmtk Worthäufigkeiten im Zusammenhang mit Grammatikstrukturen erstellt werden können, wurde ein Python-Skript entworfen, mit dem von einem gegebenen Text Satzende durch den Punkt (nach Ausschluss von Abkürzungen) und Absätze erkannt werden können. Werden OP-Berichte diktiert, so werden Satzzeichen wie Punkt, Komma sowie runde und geschweifte Klammern in der Regel mit diktiert. Diese Zeichen wurden deshalb wie eigenständige Wörter betrachtet und von umstehenden Zeichen getrennt. Der bearbeitete Eingabetext bildet nun die Grundlage für das Language Model und wurde mit dem CMUlmtk und 3-Grammen (s. 4.1.1) erstellt.

4.3 Entwicklung des Acoustic Models

Der Textkorpus des Acoustic Models bezieht sich nur auf einen Teil des Gesamtkorpus des Language Models und ist in der Regel auch viel kleiner als das Language Model, da ein sehr großes Language Model feinere Wahrscheinlichkeiten und Wortabhängigkeiten bestmmen kann. Aus den insgesamt 70 Protokollen wurden für das Acoustic Model insgesamt 20 mit einem Umfang von 4130 Wörtern und 332 Sätzen ausgewählt. Aus dem ersten Bereich wurden zehn, aus dem zweiten und dritten jeweils fünf Dokumente für das Training des Acoustic Models zufällig gewählt.

4.3.1 Aufnahme und Bearbeitung des Audiomaterials

Von vier Ärzten der Klinik wurde mit einem speziellen Headset für Sprachaufnahme und *Audacity* die OP-Berichte in das *.wav*-Format mit (16 kHz, 16 Bit und mono) eingesprochen. Da Aussagen nur in einer Länge zwischen fünf und 30 Sekunden für ein optimales Training des Sprachmodells günstig sind, wurden die Diktate satzweise geschnitten. [sphb] Insgesamt sind viermal 332 Audiodateien entstanden. Da die Aufnahme sehr aufwendig war, kam die Idee auf, die Aussagen durch eine Sprachsynthese erzeugen zu lassen. Da das Programm *eSpeak* sich leicht durch Skripte ansprechen lässt, wurde ein Python-Skript geschrieben, das den Text *vorliest* und in *.wav*-Dateien speichert. Die entstandenen Dateien wurden später für das Training verwendet. In Tabelle 2 ist eine Übersicht der Einsprechzeiten der vier verschiedenen Sprecher (A, B, C, D) und der Sprachsynthese dargestellt.

4.4 Erstellung des phonetischen Wörterbuches

Das phonetische Wörterbuch gibt in einer *dic*-Datei für alle Wörter die Aussprache in Form der phonetischen Darstellung an. Pro Zeile ist ein Wort mit der dazugehörigen Aussprache beschrieben. Ein phonetisches Wörterbuch mit 4678 Einträgen aus einem vorherigen Projekt bildet hier die Grundlage. [FTRL11] Der Texkorpus des Acoustic Models besteht aus 1273 Wörtern. eSpeak ermöglicht auch eine maschinelle phonetische Tran-

Sprecher	Teil 1	Teil 2	Teil 3	Σ
A	24:46	06:51	09:43	41:20
B	20:36	06:10	08:34	35:20
C	19:22	04:58	08:35	32:55
D	28:50	08:25	12:17	49:32
Ø	23:24	06:36	09:47	39:47
Σ(A,B,C,D)	1:33:34	0:26:24	0:39:09	2:39:07
eSpeak	19:22	05:54	08:16	33:32
Ø(A,B,C,D,eSpeak)	22:35	06:28	09:29	38:32
Σ(A,B,C,D,eSpeak)	1:52:56	32:18	47:25	3:12:39

Tabelle 2: Übersicht der Einsprechzeit

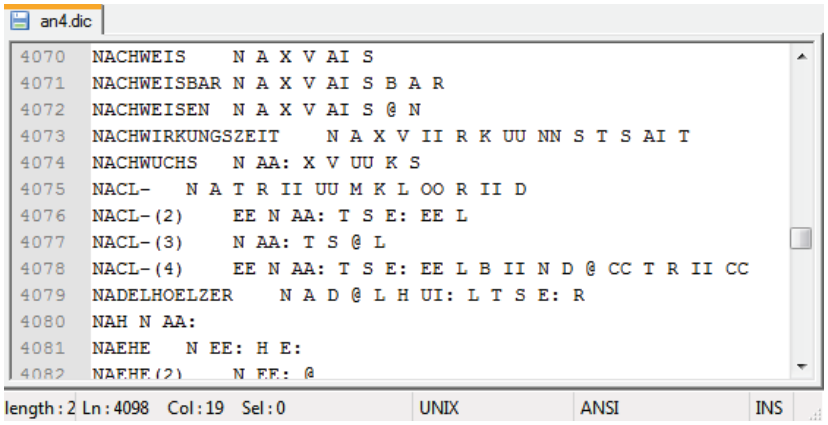


Abbildung 2: Ausschnitt aus dem phonetischen Wörterbuch

skription von Wörtern der deutschen Sprache. Jedoch kann so nur eine kleine Datenmenge bearbeitet werden. Auf automatische Transkription wurde verzichtet, u. a. auch aus dem Grund, da medizinische Vokabulare mit vielen Eigennamen, Fachbegriffen, Abkürzungen sowie Wörtern der lateinischen und griechischen Sprache sehr sensibel sind. Des Weiteren werden einige Wörter von unterschiedlichen Personen leicht variiert ausgesprochen, was auch zum Teil durch regionale Dialekte bedingt ist. Sehr viele Wörter wurden deshalb mit unterschiedlicher Aussprache gelistet. Der entstandene Gesamtkorpus besteht aus 6791 Einträgen (Ausschnitt, s. Abb. 2).

4.5 Zusammenstellung aller Module zum Sprachmodell

Bevor alle Bestandteile des Sprachmodells miteinander verbunden werden können, müssen die Werkzeuge aus 4.1.2 und 4.1.3 in einem Verzeichnis und aus Einfachheitsgründen unter Linux installiert werden. Als Recognizer wird Sphinx3 verwendet, da es für eine hohe

Erkennungsrate entwickelt wurde. Im Anschluss werden alle notwendigen Module des Sprachmodells wie im Tutorial [Sin08] beschrieben in einem gesonderten Ordner gesammelt. Um Fehler auszuschließen wurde sich während der Implementierung am Beispiel *an4* des Tutorials orientiert. Der Ordner *an4* besteht anfangs aus zwei Unterordnern: *etc* und *wav*. *wav* enthält alle Audiodateien. Aufnahmen, die von einem Sprecher sind, werden in jeweils einzelnen Unterverzeichnissen abgelegt. In *etc* werden alle weiteren Dateien gesammelt: Das phonetische Wörterbuch wird als *.dic* abgespeichert. Die Phonemliste (*.phone*) des VoxForge-Projektes wird mit den 40 Phonemen der deutschen Sprache übernommen und nicht abgeändert. Eine *.filler*-Datei enthält alle Filler-Phoneme bzw. stille Ereignisse wie Pausen. Mit der *.fileids*-Datei werden alle Pfade der Audiodateien angegeben, die das Modell verwenden soll. Die *.transcription*-Datei enthält zeilenweise alle Sätze des Acoustic Models und einen Verweis zur jeweiligen Audiodatei. Zu guter Letzt wird das erstellte Language Model als *.lm*-Datei ebenfalls in dem Ordner abgespeichert. Sind *.wav*-Dateien die Grundlage des Sprachmodells, so werden diese im nächsten Schritt in *.mfc*-Dateien bzw. 13-dimensionale Merkmalsvektoren gewandelt. Nun sind alle Dateien vollständig und das Training kann gestartet werden. Anschließend kann das Trainingsset decodiert und gleichzeitig evaluiert werden. So wie hier beschrieben, wurden verschiedene Sprachmodelle aufgebaut, die in den nächsten Abschnitten weiter betrachtet werden.

5 Auswertung / Evaluation

Aus den verschiedenen drei OP-Bereichen und jeweligen Sprachaufnahmen wurden verschiedene Sprachmodelle zusammengestellt und anschließend evaluiert. Zu Beginn wurden lediglich verschiedene Language Models betrachtet, danach vollständig zusammengesetzte Sprachmodelle. Im Anschluss erfolgt ein Test mit der Spracherkennungssoftware *Dragon Medical* mit einem Sprecher sowie ein Vergleich.

5.1 Evaluation der N-Gramm-Statistiken

Die N-Gramm-Statistiken eines Language Models können mit den Größen *Perplexität* und *Entropie* auf ihre *Qualität* untersucht werden. Über die Schnittstelle *evallm* aus dem CMUlmtk können diese beiden Größen ausgegeben werden. Da durch das Toolkit N-Gramm-Statistiken unterschiedlicher Größe erstellt werden können, wurden verschiedene N-Gramm-Statistiken bis zu $N = 7$ untersucht. Es wurde der Gesamtkorpus und die drei verschiedenen einzelnen Teilbereiche betrachtet. Die Entropie und die Perplexität werden immer in Bezug zu einem Stichprobentext betrachtet. Hier wurde dafür der jeweilige Eingabetext verwendet. Die Perplexität beschreibt, wie groß die Anzahl der Wörter ist, die im Mittel zur Fortsetzung einer Wortfolge in Betracht gezogen bzw. zur Vervollständigung der Historie eingesetzt werden kann. Der Wert der Perplexität sollte möglichst klein gehalten werden, damit die Anzahl der Wörter, die im Mittel zur Fortsetzung einer Wortfolge betrachtet werden, minimal ist. [Wen04, 3.2] [Fin03, 6.4] Die Entropie wird als der mittlere Informationszuwachs pro Wort definiert. Ein kleiner Wert der Entropie im Ver-

gleich zu einem größeren bedeutet, dass das Modell *sicher* ist, welches Wort als nächstes kommt. Je mehr Abhängigkeiten im Modell sind, desto kleiner wird die Entropie. [PK08, 13.2.8.2] Es hat sich gezeigt, dass Statistiken mit einer Größe von $N = 3$ die kleinsten Perplexitäts- und Entropiewerte im Vergleich zu großer werdenden N aufweisen. Die Statistiken der Teile 2 und 3 zeigten minimal bessere Werte im Vergleich zu der Statistik des Korpus 1 und der Statistik des Gesamtkorpus. Daraus folgt, dass die 3-Gramm-Statistiken der verschiedenen Bereiche und des Gesamtkorpus annähernd gleich aussagekräftig sind.

5.2 Auswertung Wortfehlerraten verschiedener Sprachmodelle

Die *Wortfehlerrate* bzw. die *Word-Error-Rate (WER)* berechnet sich aus der Summe aller ersetzten, ausgelassenen und eingefügten Wörter geteilt durch die Anzahl aller Wörter, die ein System erkennen soll:

$$WER = \frac{\#\ \textit{Ersetzungen} + \#\ \textit{Auslassungen} + \#\ \textit{Einfügungen}}{\#\ \textit{zu erkennende Wörter}} \cdot 100\% \qquad (1)$$

Es wurden verschiedene Sprachmodelle erzeugt und anschließend auf ihre Fehlerraten untersucht. - Für jeden der vier Sprecher und die durch Synthese erzeugten Sprachsignale wurde ein Modell für jeden Teilbereich des Korpus und für jeden Sprecher (einschließlich eSpeak) wurde auch ein Modell mit dem Gesamtkorpus trainiert. (Für die Trainingsläufe aller Modelle wurde als Language Model eine 3-Gramm-Statistik über den Gesamtkorpus verwendet.) Es wurden die Teilbereiche des Korpus für alle Sprecher trainiert, einmal mit und einmal ohne den Daten, die eSpeak erzeugt hat. Im Anschluss wurde ein Modell trainiert, das alle (einmal mit und einmal ohne eSpeak) Trainingsdaten beinhaltet. Es zeigte sich, dass Modelle, die sprecherabhängig trainiert wurden, kleinere und somit

Sprecher	Teilkorpus (TK)			
	1	2	3	Σ(1,2,3)
A	20,3 %	43,1 %	34,0 %	14,3 %
B	20,9 %	54,9 %	24,9 %	23,0 %
C	65,1 %	74,7 %	66,4 %	55,6 %
D	19,2 %	41,2 %	63,2 %	15,6 %
eSpeak	89,3 %	91,9 %	93,2 %	72,9 %
A, B, C, D	53,3 %	78,2 %	36,0 %	52,9 %
A, B, C, D, eSpeak	59,2 %	54,4 %	46,5 %	63,6 %

Tabelle 3: Übersicht unterschiedlicher Wortfehlerraten

bessere Wortfehlerraten aufwiesen. Die jeweiligen Gesamtmodelle der sprecherabhängigen Modelle zeigten eine geringere Fehlerrate als die einzeln trainierten Teilbereiche. Der Teilkorpus mit den wenigsten Trainingsdaten, TK 2, hatte im Vergleich zu den beiden anderen Teilen und dem Gesamtset, die schlechtesten Quoten bei allen sprecherabhängi-

gen Modellen. Die Modelle mit mehr Trainingsdaten erzielten bessere Quoten im Vergleich zu den Modellen mit weniger Daten. Die synthetisch erzeugten Daten enthielten in den sprecherabhängigen Modellen die schlechtesten Fehlerraten. Eine Sprachsynthese, die nicht an medizinische Wortschätze angepasst ist, ist somit ungeeignet, Sprachmodelle mit medizinischen Inhalten zu trainieren. Sollten Sprachmodelle mit medizinischen Wörtern synthetisch trainiert werden, dann muss die Sprachsynthese vorher daran angepasst werden. Die Modelle, die mit den Daten aller Sprecher (ohne eSpeak-Daten) trainiert wurden, zeigten erhöhte Fehlerquoten gegenüber den sprecherabhängigen Modellen. Die sprecherunabhängigen Modelle zeigten keine Verbesserungen im Vergleich zu den einzelnen sprecherabhängigen Modellen. Es wird angenommen, dass die Sprachmerkmale der verschiedenen Sprecher sehr unterschiedlich sind und sich gegenseitig zu sehr negativ beeinflusst haben. Die sprecherabhängigen Modelle erwiesen sich somit als die besseren Modelle. Die Sprachmodelle, die aus allen verfügbaren Sprachsignalen einschließlich der synthetisch erzeugten Daten zusammengestellt wurden, zeigten im Vergleich zu den Daten ohne der Sprachsynthese schlechtere Quoten - außer im TK 2. Auch hier hat sich gezeigt, dass die synthetisch erzeugten Daten einen schlechten Einfluss auf die Fehlerraten hatten. Ein Blick in die Datei, die während des Decodiervorgangs alle erkannten Wörter sammelt, zeigte, dass viele medizinische Fachbegriffe richtig erkannt wurden. Gleich klingende Wörter verursachten jedoch einen Großteil der Fehler. So wurde zum Beispiel statt *„hier“ „die“*, statt *„man“ „dann“* und statt *„Oberschenkel“ „Total Schenkel“* erkannt. Durch ein spezielles Einzelworttraining mit Fachbegriffen und mit einer Vergrößerung des Trainingssets könnte ein großer Teil dieser Fehler beseitigt werden.

5.3 Ein Vergleich mit Dragon Medical

Für den medizinischen Anwendungsbereich der Spracherkennung wurde *Dragon Medical* entwickelt. Vor der ersten Anwendung muss ein Benutzerprofil erstellt und ein relativ kurzer Trainingstext vorgelesen werden. Sprecher A hat das Training der Version 11 (3/2011) der Software absolviert. Durch die *Autoumsetzung* der Software wurden *.wav*-Dateien mit den Aufnahmen von Sprecher A in Text gewandelt. Dragon Medical sollte einen Korpus von 4 625 Wörtern (einschließlich Punkte, Kommata, Klammern und Bindestriche) erkennen. In Tabelle 4 ist dargestellt, wie hoch die Fehlerrate in den einzelnen Bereichen und im Gesamtkorpus ist. In Tabelle 5 ist der direkte Vergleich der Fehlerraten zwischen den eigenen entwickelten Sprachmodellen mithilfe CMU Sphinx und denen aus dem Testlauf von Dragon Medical dargestellt. Erstaunlicherweise ist die WER in beiden Modellen des ersten Bereiches fast gleich, dafür war im zweiten Teil die Quote von Dragon Medical deutlich besser, was hier auf die kleine Größe des Modells zurückzuführen ist. Ebenso trifft dies auf den dritten Teil zu. Das gesamte Trainingsset von Sprecher A hat allerdings im Vergleich zur Fehlerrate von Dragon einen minimal besseren Wert. Daraus zeigt sich, dass viele Wörter des eigenen Sprachmodells u. a. durch unterschiedliche Aussprachemöglichkeiten im phonetischen Wörterbuch an den speziellen Anwendungsfall angepasst sind und somit deutlich weniger Fehler auftreten als im Vergleich zu Dragon Medical. Der Testlauf zeigte auch, dass Dragon Medical einige medizinische Fachbegriffe und Eigennamen nicht

Text	Fehler	#Wörter	Wortfehlerrate
1	513	2692	19,1 %
2	133	774	17,2 %
3	208	1159	17,9 %
Σ(1,2,3)	854	4625	18,5 %

Tabelle 4: Dragon Medical - Test - Übersicht unterschiedlicher Fehlerraten

Teilkorpus	Wortfehlerrate	
1	20,3%	(CMU Sphinx)
	19,1%	(Dragon)
2	43,1%	(CMU Sphinx)
	17,2%	(Dragon)
3	34,0%	(CMU Sphinx)
	17,9%	(Dragon)
Σ(1,2,3)	14,3%	(CMU Sphinx)
	18,5%	(Dragon)

Tabelle 5: Der direkte Vergleich - WER von Dragon Medical und CMU Sphinx

richtig erkannte. Wörter, die keine medizinischen Begriffe sind, wurden wiederum gut erkannt. Bei häufig auftretenden Fachwörtern, die falsch ausgegeben wurden, fiel auf, dass diese immer in der gleichen falschen Schreibweise ausgegeben wurden. Statt *„Taurolin“* wurde zum Beispiel sehr oft *„Heroin“* ausgegeben, statt *„Vena“* *„Wien“* und statt *„Redondrainage“* *„Rettungsdrainage“*. Durch ein spezielles und aufwendiges Einzelworttraining aus dem Umfang der Software können solche Wörter durch den Benutzer in das System aufgenommen werden und die Fehlerrate würde sich erheblich verringern.

6 Zusammenfassung und Ausblick

Die entstandenen Sprachmodelle wurden mit einem Gesamtumfang von rund 160 Minuten Aufnahmen und rund 34 Minuten synthetisch erzeugter Sprache trainiert. Die Wortschatzgröße des (gesamten) Acoustic Models bestand aus rund 4 700 Wörtern, das dazugehörige Vokabular aus 1 300 und das Language Modell bestand aus ca. 17 000. Die entstandenen Fehlerraten der sprecherabhängigen Sprachmodelle des Gesamtkorpus lagen zwischen 14,3% und 55,6%. Das Modell, das alle Trainingsdaten (ohne Sprachsynthese) enthielt, zeigte eine Wortfehlerrate von rund 53%. Eine Spracherkennung, die praktisch eingesetzt werden soll, muss eine Erkennungsrate von mindestens 95% aufweisen bzw. eine Fehlerrate von nicht mehr als 5%. Daher können die Modelle in einem praktischen Einsatz - schon allein aufgrund der stark begrenzten Wortschatzgröße - nicht verwendet

werden. Durch einen direkten Vergleich der Trainingsdaten mit Dragon Medical hat sich jedoch gezeigt, dass hier entstandene Sprachmodelle für den Anwendungsbereich sehr angepasst sind und somit eine Grundlage für eine Weiterentwicklung bilden. Der Korpus des Sprachmodells müsste mit vielen zusätzlichen Trainingsdaten erneut gefüllt werden. Des Weiteren hat sich entgegen der Erwartung gezeigt, dass die Wortfehlerraten der einzelnen Teilbereiche (1, 2, 3) im Vergleich zur WER des gesamten Korpus ($\Sigma(1, 2, 3)$) nicht besser waren. Ein Grund dafür war jedoch die zu kleine Größe der Trainingssets der einzelnen Teilbereiche. Werden OP-Protokolle im Klinikum Chemnitz in das elektronische Datenverwaltungssystem einfügt, so immer eine Diagnoseschlüsselnummer mit angegeben. In Anbetracht vieler verschiedener Operationstypen, schon allein in der Allgemein- und Viszeralchirurgie, könnte man in einer Weiterführung unterschiedliche Sprachmodelle für den jeweiligen OP-Bereich entwickeln. Wird dann zum Beispiel ein OP-Bericht von einer Gallenoperation diktiert, so könnte man aufgrund einer Voreinstellung, die durch das Klinikpersonal bereits vor der Operation in das System eingegeben wird, ein bestimmtes Sprachmodell, dem OP-Berichten von Gallenoperationen zu Grunde liegen, angesprochen werden. Die Zusammenstellung der Sprachmodelle ist jedoch sehr aufwendig. Schon allein die Sammlung der Sprachaufnahmen verschiedener Personen war mit einem sehr hohen zeitlichen Aufwand verbunden, da die Trainingsdaten für extra eingesprochen wurden. Daraufhin wurden mithilfe der Sprachsynthese eSpeak Sprachsignale erzeugt. Es zeigte sich jedoch, dass diese Daten die Wortfehlerraten im Vergleich zu den eingesprochen Daten nicht verbesserten oder (bis auf eine Ausnahme) weitaus schlechtere Fehlerquoten entstanden. eSpeak kann demzufolge für Trainingsläufe von Sprachmodellen mit medizinischen Anwendungswortschätzen nicht verwendet werden oder müsste vor einem erneuten Versuch angepasst werden. Alternativ zu eSpeak könnte aber auch eine andere Sprachsynthese, zum Beispiel die Microsoft-Speech-API, für einen Trainingslauf zum Einsatz kommen. Allerdings sollten daraus resultierende Ergebnisse aufgrund des sehr speziellen Anwendungswortschatzes sehr genau untersucht werden.

Literatur

[CR97] Philip R. Clarkson und Ronald Rosenfeld. Statistical Language Modeling Using the CMU-Cambridge Toolkit. In *ESCA Eurospeech*, 1997. http://www.cs.cmu.edu/ŕoni/papers/SLMTKV2eurospeech97.pdf - Letzter Zugriff: 03.06.2012.

[DEMM05] Paul Deleglise, Yannick Esteve, Sylvain Meignier und Teva Merlin. The LIUM speech transcription system: a CMU Sphinx III-based system for french broadcast news. *Proc. of Interspeech'05, Lisbon (Portugal)*, 09/2005. http://lium3.univ-lemans.fr/lium_d5/sites/default/files/LIUM_Interspeech05.pdf - Letzter Zugriff: 03.06.2012.

[Fin03] Gernot A. Fink. *Mustererkennung mit Markov-Modellen*. B.G. Teubner Verlag, 1. Auflage, 2003.

[FTRL11] Maria Friess, Marina Trinks, Claudia Rohde und Christina Lohr. Sprachadaption für den Dialekt Sächsisch. Teamprojekt, Technische Universität Chemnitz, 04 2011.

[Ger06] Sebastian Germesin. Spracherkennung mit dynamisch geladenen, spezifischen Akustikmodellen. Bachelorarbeit, Saarbrücken, Universitätdes Saarlandes, 01 2006. http://www.dfki.de/k̃ipp/seminar/germesin-bachelor.pdf - Letzter Zugriff: 21.03.2012.

[mus] (Muster-)Berufsordnung für die in Deutschland tätigen Ärztinnen und Ärzte. http://www.bundesaerztekammer.de/downloads/MBO_08_20111.pdf - Letzter Zugriff: 03.06.2012.

[PK08] Beat Pfister und Tobias Kaufmann. *Sprachverarbeitung: Grundlagen und Methoden der Sprachsynthese und Spracherkennung*. Springer, 2008.

[Sin08] Evandro Gouvêaand Rita Singh. Robust group's Open Source Tutorial, 09 2008. http://www.speech.cs.cmu.edu/sphinx/tutorial.html - Letzter Zugriff: 03.06.2012.

[spha] Acoustic and Language Models for CMU Sphinx. http://sourceforge.net/projects/cmusphinx/files/Acoustic%20and%20Language%

[sphb] Training Acoustic Model For CMUSphinx. http://cmusphinx.sourceforge.net/wiki/ tutorialam - Letzter Zugriff: 03.06.2012.

[Sphc] CMU Sphinx. SphinxBase. http://sourceforge.net/projects/cmusphinx/files/sphinxbase/0.7/sphinxbase-0.7.tar.gz/download - Letzter Zugriff: 03.06.2012.

[Sphd] CMU Sphinx. SphinxTrain. http://sourceforge.net/projects/cmusphinx/files/sphinxtrain/1.0.7/sphinxtrain-1.0.7.tar.gz/download - Letzter Zugriff: 03.06.2012.

[Ste12] Dr. Johannes Steinmüller. Sprachverstehen (Vorlesungsskript) (WS 2011/12), 2012. http://www.tu-chemnitz.de/informatik/KI/edu/spraver/ - Letzter Zugriff: 03.06.2012.

[Uni12] Carnegie Mellon University. CMUSphinx - Open Source Toolkit For Speech Recognition, 2012. http://cmusphinx.sourceforge.net/wiki/download - Letzter Zugriff: 03.06.2012.

[vox12] VoxForge, 2012. http://www.voxforge.org/ - Letzter Zugriff: 03.06.2012.

[Wen04] Prof. Dr. Andreas Wendemuth. *Grundlagen der statistischen Sprachverarbeitung*. Oldenbourg Verlag, 2004.

Performance Loss on Virtual Machines

Yu Zhang, René Oertel and Wolfgang Rehm

zhayu@hrz.tu-chemnitz.de
{rene.oertel,rehm}@cs.tu-chemnitz.de

Abstract: Performance loss is unfavorable for virtual machine to be widely applied to high performance computing. Although much efforts were put to seek the potential bottlenecks on virtual machines and attempted to reduce them, and research showed that among all aspects of the virtualization, operations involving the virtual machine monitor and I/O devices tend to be inefficient, with the emergence of the ever more dedicated applications and sophisticated virtual softwares, bottlenecks for performance loss are getting more and more difficult to be tracked. As Parsec benchmark suite is rich in workload type, we examined the performance loss of its included applications on a set of virtual machines. Key aspects, including system call, CPU and memory utilization, I/O as well as Cache Miss Rate were taken into consideration. Through analysis of the abnormal trends and the statistics during program execution, we gained a deeper insight into the root causes of performance loss. At least system calls handling process synchronization and I/O data exchanges were identified as the main causes of performance loss for the majority of virtual machines. Besides, we believe CPU utilization to be an indicator to the system performance, and that frequent context switch triggered off by frequent I/O request may reduce the CPU utilization seriously. However, significant impact of *cache miss rate* to performance loss introduced by virtual machine was not found in our practice. Way to improve the performance of such system calls on virtual machines should be considered.

1 Introduction

Virtual machine enables hardware resources to be exploited more efficiently and costs of hardware purchasing and maintaining to be reduced by a wide margin. However, due to a performance overhead suffered by the majority of production applications, it has not gained a widespread utilization on the HPC cluster yet. Efforts to seek the performance loss has never stopped since the invention of the virtual machine. With the ever growing complexities of virtualization softwares and the diversity of the production workloads, it is increasingly difficult to understand the behavior of an application on a virtual machine. Although most of the previous studies pointed out that I/O was particularly troublesome for a virtual machine, bottlenecks in an ordinary case is never a simple thing to be identified, considering the complicated interaction between guest and host systems with involvement of virtual machine monitor. As performance of a virtual machine is compared against the physical one, one may wondering that difference in configuration between the two has caused the performance gap. In the optimal case, if configurations of the virtualized hardwares are identical to that of their physical counterpart, identical performance may also

be yielded. Unfortunately, being implemented on the basis of hardware, it's impossible to eliminate the difference in configurations thoroughly, hence the performance gap between the two. To the majority of production workloads, controlling it under an acceptable level will be sufficient.

To analyze the performance loss of the virtual machine, sufficient statistics during execution of the benchmark application is necessary, which may reveal the characteristics of the program workload from a dynamic perspective. We traced the behaviors of different kind of workloads by gathering statistics when programs were running. Comparisons between the ill-behaved ones and normal ones, virtual ones and native ones were performed. Configuration issues key to performance of most programs were examined.

For Parsec benchmark suite includes a variety of workloads from real world, KVM, VirtualBox, Xen(Para) and VMware ESXi Server are the main stream of modern virtualization technologies, performance overheads encountered by them may be typical ones met also in production practice, identifying and solving of them will be contributions to both virtual machines research and application.

The rest of the paper is organized as follows: In Section 2, we survey the related work. In Section 3 we review a case of performance loss with concrete benchmark suite on a number of typical virtualization platforms. We discuss the possible reasons for this in Section 4. Experiments and results were presented in Section 5, finally we concluded and anticipated in Section 6.

2 Related Work

Much work focusing on performance overhead [1] [2] pointed out that I/O virtualization, scheduling, system calls and page fault handing [3] are costly, even on the most developed virtualization platform Xen and VMware ESXi Server. F. Benevenuto etc. [2] and H. Mousa etc. [4] suggested to characterize the workloads by breaking them down to the cycle per instruction (CPI) performance metric for a fine granularity of sampling. Having measured the basic Linux operations such as the kernel calls, context switch, page allocation and file reading, M. Guevara and C. Gregg [5] concluded that reading from a file and inter-process communication exhibit a significant overhead. Also with system call, K. Najafi [6] attributes the slowdown of `mmap()` system call to the process scheduler and the repeated pagefault handling of the memory subsystem. According to Q. Ali etc. [7] the HPC workloads on VMware's virtualization platform achieve close to native performance, in some cases even 20% better than native. In the paper [8], I. Ahmad proposed two novel techniques, one uses the number of commands in flight to dynamically adjust the coalescing rate in fine-grained steps and to use future I/O events to avoid the need of high resolution, the other reduces the number of inter-process interrupts while keeping the latency bounded, and with these they improved the CPU efficiency up to 17%. With the Parsec benchmark suite, A. Navarro etc [9] identified the I/O bottleneck and load imbalance as the main causes of some of the performance degradations with pipeline parallelism, and proposed the parallel I/O as a remedy for them. Performance Impact of buffer cache

replacement algorithm were taken into consideration by A. R. Butt etc. [10], they showed that the kernel pre-fetching can have a significant impact on the relative performance in terms of the number of actual disk I/Os of many well-known replacement algorithms.

There are other works on the performance profiling, monitoring and tracing as [11], [12] and [13] etc., which may all be helpful for obtaining better understandings of the performance overhead by tracing the important statistics during the interaction between the guest and host systems.

3 Benchmark

3.1 Configuration of the Host and Guest Machines

The benchmark platform incorporates four identical host machines, each with one of the following virtualization software installed: VirtualBox 4.0.1, KVM, VMware ESXi 5 Server and Xen-4.0. Guest systems were configured individually on each one of them, depending on the features of the virtualization software underlying. Detailed information about the configuration is listed in Table 1 .

Table 1: Host machine configuration

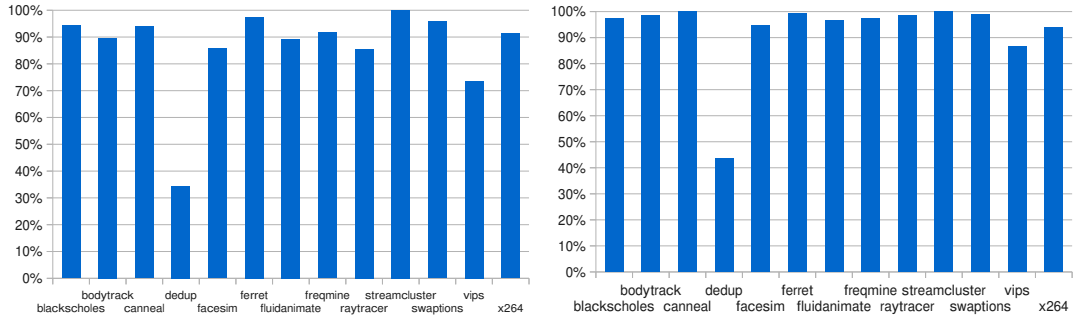
Host	Configuration
Processor	Dual-Core AMD Opteron(tm) Processor 2218
Cpu MHz	2600.000
Cache size	1024 KB
Cpu cores	2
Bogomips	5226.48
TLB size	1024 4K pages
Memory	DDR2 8192 MB
Hard disk	250 GB
Operating System	Debian Squeeze 6.0.2 Linux

Configurations among guest machines are due to specific virtualization software features not strictly identical, while the number of processors, memory size and storage volume are kept the same. Furthermore, in order to make the performances more comparable between virtual and native cases, four processors were assigned to each guest. Based on this we

launched a series of benchmarks. Performance comparisons were made between the guest machine against the host. For simplicity multiple threads was not taken into consideration in our practice, which we left for further discussion.

3.2 Benchmark Results

As showed in Figure 1 , it turned out that most of the applications perform nearly at a native speed on the virtual machines, with only a few exceptions suffering a significant performance loss on each virtual machine. With a huge benchmark suite such as Parsec, it's really a challenge to see the actual reasons behind the scenes. One practical approach is to capture the detailed information during the program execution, to make sure which stage in the processing is not so efficient, that reduced the performance of the entire system.



(a) VirtualBox

(b) KVM

(c) VMware ESXi 5

(d) Xen(para)

Figure 1: Performance overview on different virtualization platforms

Obviously performance loss occurred on virtualization platforms varies from one to the other. However, once an application suffers a heavy loss on almost every virtual machine, it could not be taken as by accident. One example - `dedup`, lost over half of its speed on VirtualBox and on KVM, and more than 30% on Xen(para) and VMware ESXi 5 Server. Others like `vips` and `x264` also experienced common penalties of more or less than 10%. It may suggest us something similar happened in spite of the differences among the virtualization softwares in dealing with some problematic operations. We resort to the principle

of virtual machine implementation as a guideline to our discussion.

4 Performance Loss on Virtual Machines

J. E. Smith and R. Nair [14] summarized several key reasons for performance degradation:

- Setup: The initialization process before a guest machine come to life, initializing the timing facilities, program counter and other registers.
- Emulation: A guest system is expected to execute most of its instructions in a native way. However, emulation by a virtual machine monitor becomes necessary if sensitive instructions are encountered during the execution, which is well known as inefficient.
- Interrupt handling: The virtual machine monitor catches the once a interrupt is generated by the guest system, then may hand it over to the guest operating system.
- State saving: Saving the states of the guest machine introduces extra overhead when control transfer to the virtual machine monitor occurs.
- Bookkeeping: As an intrinsic requirement of the virtual machine, the virtual machine monitor need to perform some special operation to keep the behavior equivalent to that of the real machine.
- Time elongation: Memory access for instance is far from trivial a thing than in the native case, page tables on the guest machine, page tables on the host machine, shadow tables are all accessed mapping to the real memory address from the guest machine atop.

Reasons listed above deal with overhead on processor and memory access, of which we focused on the impacts of interrupt handling and state saving. Setup is beyond our discussion for the benchmark programs were only launched when the virtual machine come to its life. Bookkeeping and Time elongation occur all the way when virtual machine is in operation.

As the performance of a virtual machine is evaluated, we made the assumption that the native performance is always 100%. It actually set a standard for the virtual machine not only in performance, but also in system configuration. Efforts seeking the cause of performance gap is roughly equal to that of seeking the difference in system configuration between the two. With the statistics gathered from the execution, we attempted to figure out the exact behaviors of benchmark applications on the virtualized hardwares, e.g. how much CPU time and memory size were taken as their resources, how many times of system call, interrupt as well as context switch occurred meanwhile, does the cache miss rates really matters, and how does the total performance get influenced by all these factors.

5 Execution Analysis

5.1 System Calls and Interrupts

Once a system call is issued by the guest, the virtual machine monitor is responsible to intercept it and let the host to handle it as it should, control is returned back to guest operating system immediately upon completion. Even the virtual machine monitor is by-passed in dealing with system calls, a diminished overhead is still there. That made us believe that system calls is an important factor in our performance study. After many times of benchmark on virtual machines, we came to realize that on virtual machines the ill-behaved applications tend to spend quite a lot of time in kernel mode. Figure 2 depicted how much time each benchmark program spends in kernel mode on virtual machines as well as on the host as native, regarding the total execution time as a whole.

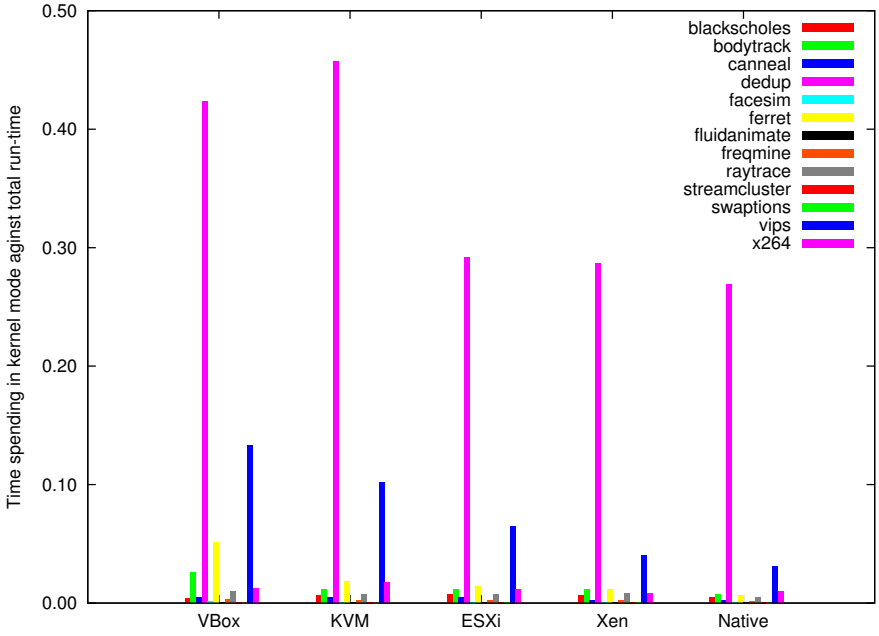


Figure 2: Time spending in kernel mode against total execution-time

The figure exhibits proportions of time spending in kernel mode. `dedup` spends nearly 30% of its total execution time on the host, guest machines of VMware ESXi 5 Server and Xen, and even 10% or more higher on that of VirtualBox and KVM.

To see which kind of system calls may dominate in kernel mode, system call reports were generated with the help of a system monitoring tool - `strace`. As depicted in Figure 3, we identified immediately `futex()` as the bulk of system calls `dedup` and `vips` issued. In the worst case the time cost of a system call may increased by a magnitude of 10 on the guest of VirtualBox against that on the host. In contrast, such cost on guest of VMware

ESXi 5 Server is diminished by a large margin, and that on Xen guest is even lower than that on the host. If the virtual machine monitor is unable to cope with it properly, even the non-expensive system calls in native would be proved much more expensive on a virtual machine, as the figure suggested.

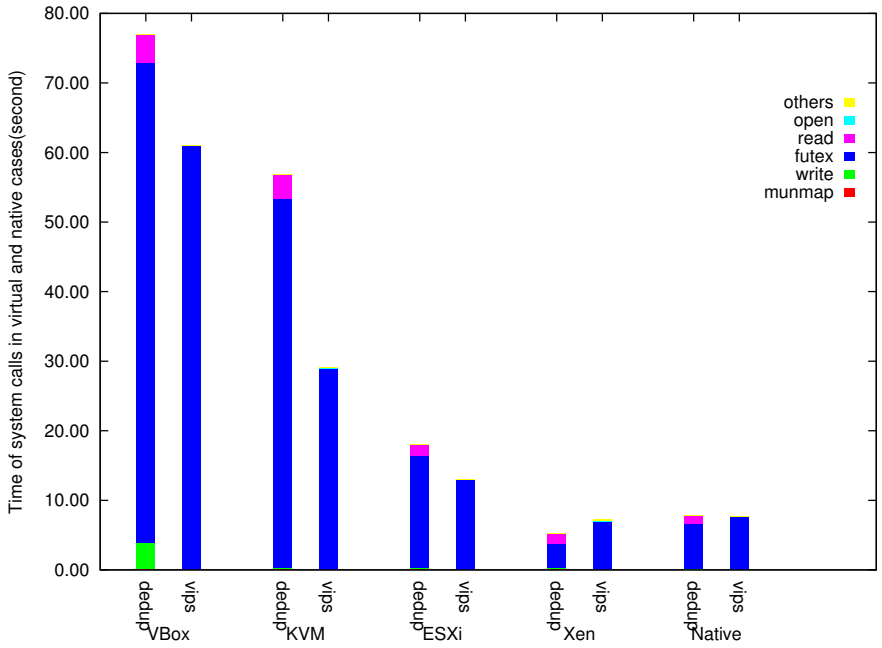


Figure 3: Time of system calls during execution in virtual and native cases

Let $f_{syscall}$ be the *number of system call per second*, $t_{syscall}$ be the *average time cost per call*, P be *Physical (Host)*, and V be *Virtual Machine*. The *total time cost of system calls* may be estimated with the equation $T_{syscall} = \sum f_{syscall} \cdot t_{syscall}$. From both theory [1] and our practice, we know that for each system call the equation $f_{syscall}(V) = f_{syscall}(P)$ holds true. As a certain kind of system calls on a virtual machine are particularly troublesome to be coped with, $t_{syscall}(V)$ may be multiple greater than $t_{syscall}(P)$. When it happens that such system call takes a large proportion of time in the kernel mode and that the program control need to enter into the kernel mode frequently, the cost of system calls on a virtual machine increases sharply, finally adding to the overall cost. In other words, a virtual machine needs much more time in dealing with such system calls in this situation.

As `futex()` is implemented as a means of process or thread locking and synchronization in kernel space, it operates based on two central behaviors: WAIT and WAKE, a reasonable explanation of its frequent concurrence is that multiple of synchronization taken place during the program execution. In addition, `read()` and `write()` also amount much, the virtual I/O devices should be responsible.

[1] equality nature of virtual machine - program should behave almost the same if not exactly identical as they do on the host machine

5.2 Cache Miss Rates

For a modern processor with out-of-order execution and pipeline features, effective use of multiple levels of cache is a key to a smooth execution. *cache miss rate* is an index for it, the lower it is, the smoother the execution will be. An L1 miss will typically cost around 10 cycles, and an L2 miss can cost as much as 200 cycles. Cache access therefore need to be careful profiled for the sake of system performance. After cache miss rate for all Parsec applications on guest and host machines were examined with `cachegrind` [15], we saw no noticeable increase brought by virtual machines, hence no clear evidence shows a necessary connection between the *cache miss rate* and the heavy performance loss brought by virtual machines. Instead, the *cache miss rate* seems to be more of a nature predetermined by the workload itself rather than the fault of virtual machines. At least it has been verified by the Parsec benchmarks so far. Table 2 [2] presents the *cache miss rate* for all Parsec programs on the virtual and host machines.

5.3 Hardware Resource Utilization

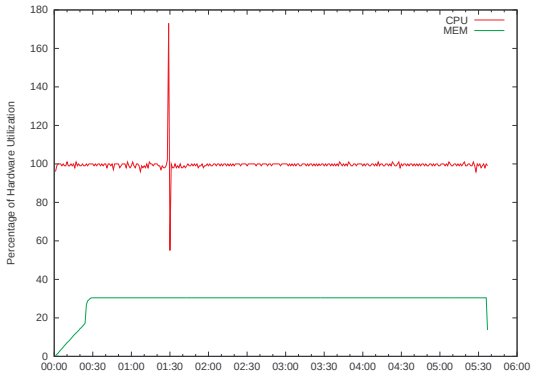
No matter on host or on guest machines, system resources like the CPU and memory are always shared by multiple of kernel and user threads. System resource utilization reveals the nature of a specific kind of workload in a dynamic way. Once CPU or memory get hogged by a certain process, others may be starved for lack of available resources temporarily, the overall system performance may be effected. We monitored the variation of the CPU and memory utilization the whole duration of the process, and made a comparison of that among guest and host machines (see Figure 4). To be noticed is the scale of the Y-axis, since the virtual machines were configured in accordance with the host machine, in this case, four virtual processors, a CPU utilization over 100% may suggest that one processor is overloaded, with the rest load being shifted to other processors.

Difference is immediately seen between the two types of behaviors. Keeping well balanced burdens on the hardware, `blackscholes` and `fluidanimate` yield almost native performance on virtual machines. `dedup` and `vips`, however, behave abnormally. `dedup` overloads the CPU with a peak of computation task, beyond the capacity of a single core immediately upon its arrival. Moreover, it swallows much memory than any other workloads. `vips` is not able to utilize the CPU efficiently, constantly interrupted from execution. Although the memory of virtual machines were configured to 2048MB - a quarter of that on the host, we found that a workload on the virtual machine consumes nearly the same amount of memory as that on the host, as long as within the maximum available size. Applications may run at nearly native speeds keeping a processor utilization close to 100% as the above two. However, if for a certain reason they blow the CPU with a task constantly changing in intensity up to 50%, one could not expect the virtual machine turns out a matching performance as that on the host any more.

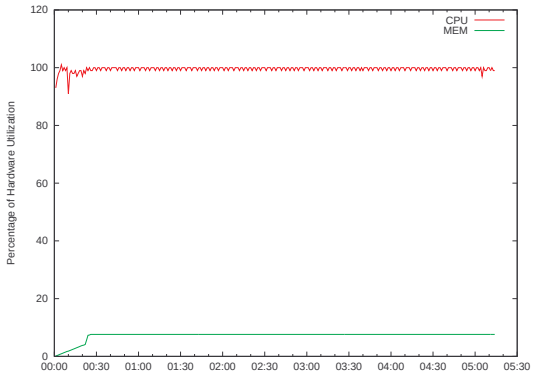
[2] `cachegrind` simulates and annotates a program line-by-line with the cache misses, records the results as: I1 - L1 instruction cache reads and misses; D1 - L1 data cache reads and read misses, writes and write misses; L2i - L2 instruction cache reads and read misses; L2d - L2 data cache reads and read misses, writes and writes misses

Table 2: Cache miss rates

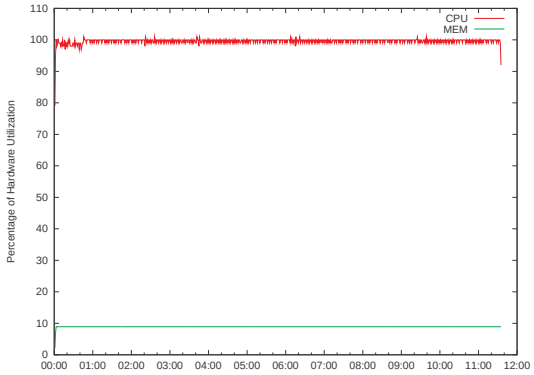
Application	Cache Miss Rate(%)	VBox	KVM	ESXi	Xen	Native
blackscholes	I1	0.00	0.00	0.00	0.00	0.00
	D1	0.10	0.10	0.10	0.10	0.10
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	0.10	0.10	0.10	0.10	0.10
bodytrack	I1	0.01	0.01	0.01	0.01	0.01
	D1	2.50	2.50	2.50	2.50	2.50
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	0.00	0.00	0.00	0.00	0.00
canneal	I1	0.10	0.10	0.10	0.10	0.10
	D1	6.90	6.90	6.90	6.90	6.90
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	6.00	6.20	6.00	6.00	6.00
dedup	I1	0.00	0.00	0.00	0.00	0.00
	D1	1.30	1.40	1.40	1.40	1.40
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	0.40	0.40	0.40	0.40	0.40
facesim	I1	0.08	0.08	0.08	0.08	0.08
	D1	1.60	1.60	1.50	1.60	1.50
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	1.30	1.30	1.30	1.30	1.30
ferret	I1	0.00	0.00	0.00	0.00	0.00
	D1	1.60	2.30	1.60	1.60	1.70
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	0.30	1.20	0.30	0.30	0.30
fluidanimate	I1	0.00	0.00	0.00	0.00	0.00
	D1	0.40	0.40	0.40	0.40	0.40
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	0.30	0.30	0.30	0.30	0.30
freqmine	I1	0.00	0.00	0.00	0.00	0.00
	D1	0.40	0.40	0.40	0.40	0.40
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	0.10	0.10	0.10	0.10	0.10
raytrace	I1	0.04	0.04	0.04	0.04	0.04
	D1	0.30	0.30	0.30	0.30	0.30
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	0.10	0.10	0.10	0.10	0.10
streamcluster	I1	0.00	0.00	0.00	0.00	0.00
	D1	3.30	3.30	3.30	3.30	3.30
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	3.20	3.20	3.20	3.20	3.20
swaptions	I1	0.00	0.00	0.00	0.00	0.00
	D1	0.00	0.00	0.00	0.00	0.00
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	0.00	0.00	0.00	0.00	0.00
vips	I1	0.00	0.00	0.00	0.00	0.00
	D1	1.70	1.70	1.70	1.70	1.70
	L2i	0.00	0.00	0.00	0.00	0.00
	L2d	0.30	0.60	0.30	0.30	0.30
x264	I1	0.39	0.45	0.45	0.45	0.45
	D1	1.80	2.30	2.30	2.30	2.30
	L2i	0.00	0.01	0.00	0.00	0.00
	L2d	0.50	0.60	0.60	0.60	0.60



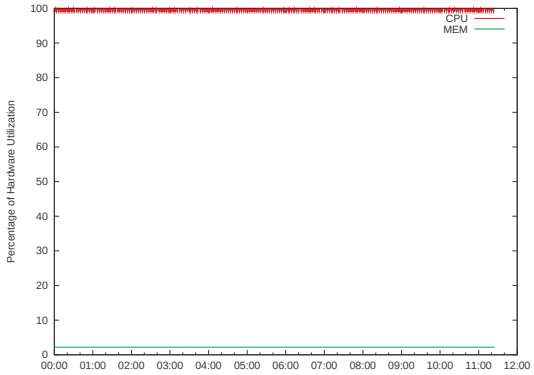
(a) blackscholes on KVM guest



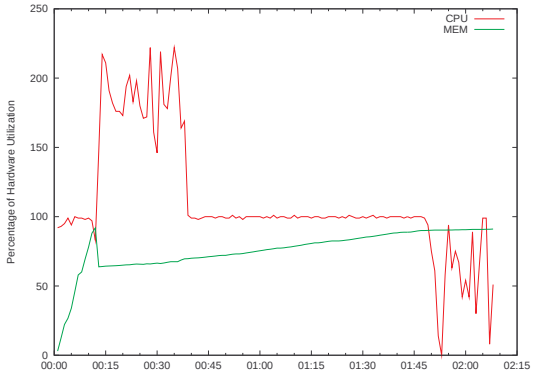
(b) blackscholes on host



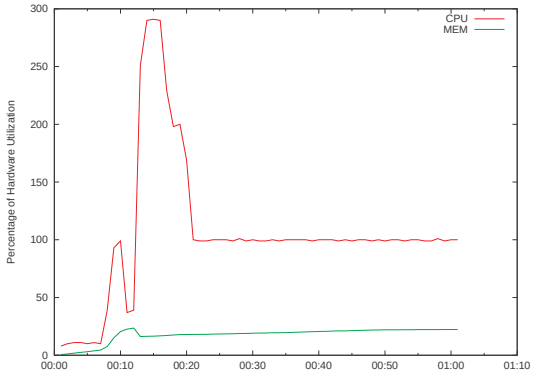
(c) fluidanimate on KVM guest



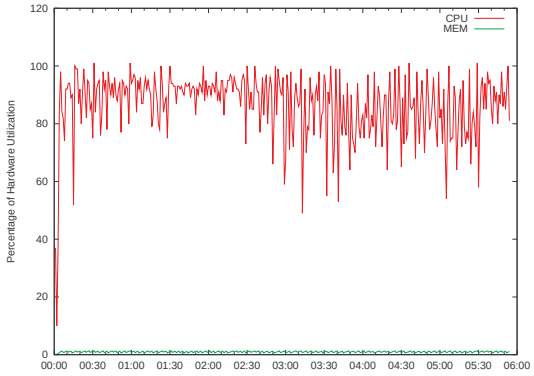
(d) fluidanimate on host



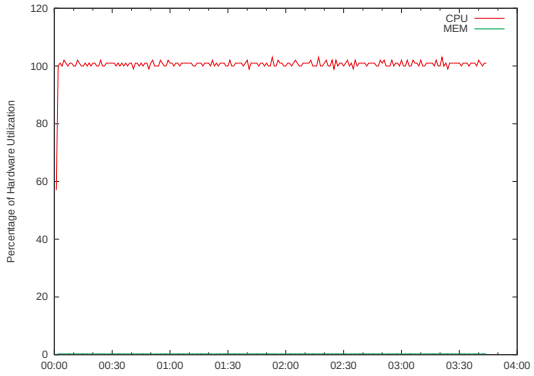
(e) dedup on KVM guest



(f) dedup on host

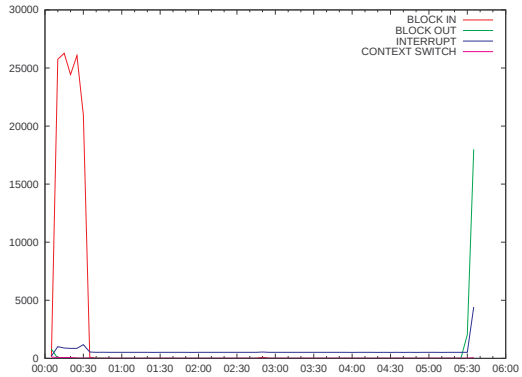


(g) vips on KVM guest

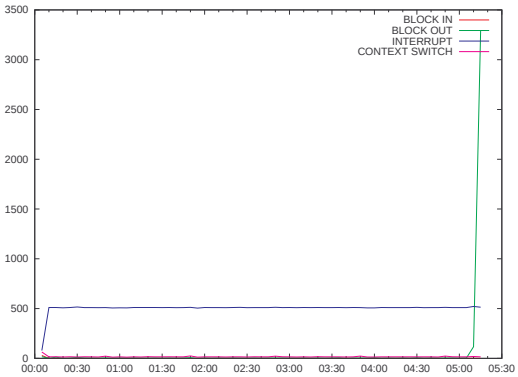


(h) vips on host

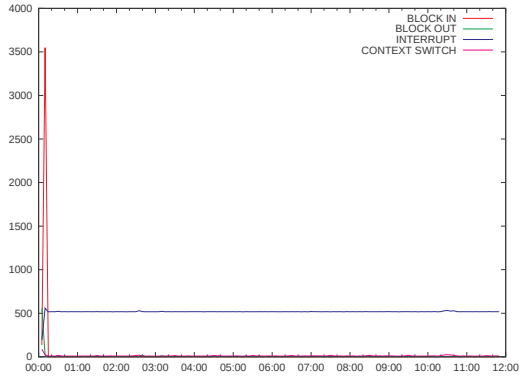
Figure 4: Contrast between system resource utilizations in virtual and native cases



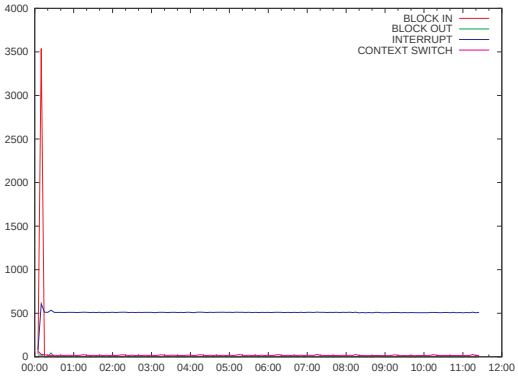
(a) blackscholes on KVM guest



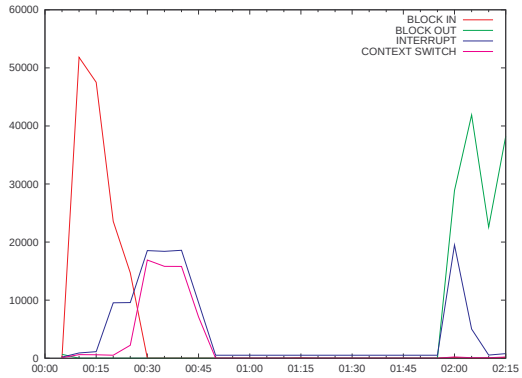
(b) blackscholes on host



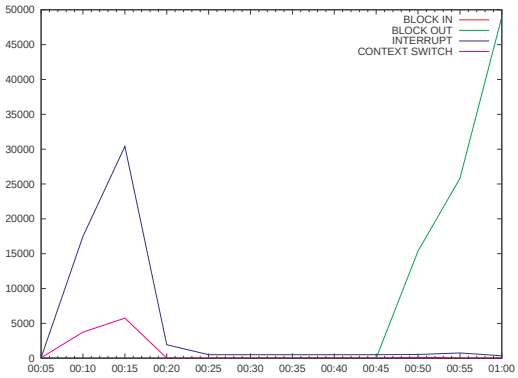
(c) fluidanimate on KVM guest



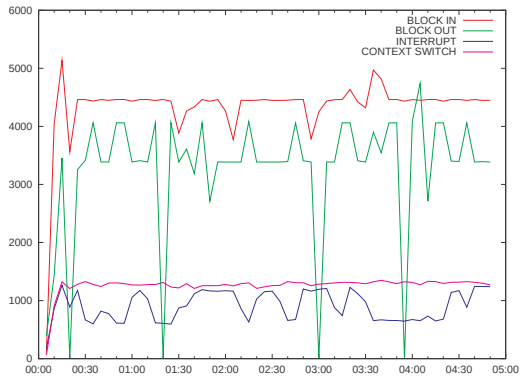
(d) fluidanimate on host



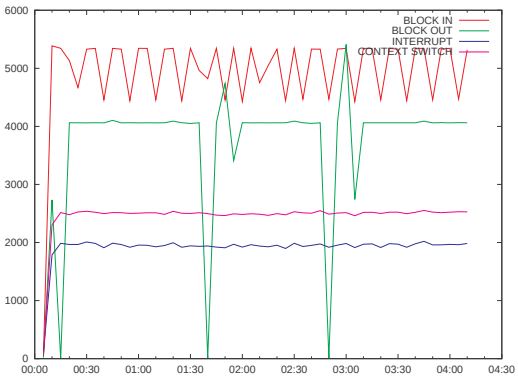
(e) dedup on KVM guest



(f) dedup on host



(g) vips on KVM guest



(h) vips on host

Figure 5: Contrast between I/O, interrupt and context switch in virtual and native cases

5.4 I/O, Interrupt and Context Switch

I/O request diverts the control flow from a running program to an interrupt handling, and interrupt handling triggers off context switch. The context of an executing process consists of the CPU state, including contents of all registers and all necessary informations of memory accessing for the switched-out process. When a context switch caused by an I/O request occurs, control is transfered from the running program to an I/O interrupt handler, or from a handler for a low-priority interrupt to that of a higher priority interrupt [16], It is a computing process of storing and restoring the state (context) of a CPU so that execution can be resumed from the same point at a later time. This enables multiple processes to share a single CPU. It is normally computationally-intensive and may deliver a strong impact to the performance.

Let f_{cs} be the *number of context switch occurrences per second*, t_{cs} be the *average time cost per context switch*, t_{io} be the *average time cost per I/O operation*, p_{io} be the *percentage of context switch caused by I/O Interrupt*, and t_o be the *average time spending on other operations per second*, the CPU utilization may roughly be estimated by $u_{cpu} = 1 - (f_{cs} \cdot t_{cs} + p_{io} \cdot f_{cs} \cdot t_{io} + t_o)$. As stated by [17], t_{cs} ranges from several microseconds to more than a thousand microseconds, depending on whether a data size fits the cache well. t_{io} is due to notoriously inefficient I/O on virtual machines greater, a large part of CPU time would also be taken away in case of heavy I/O traffic. In general, for a given virtual machine, the values of t_{cs} and t_{io} are most likely to be greater than that on the host. For a given application, the f_{cs}, p_{io} keep constant. The greater the values of t_{cs} and t_{io} are, the lower the CPU utilization will be. The tendency was reflected by the scenes shown in Figure 4 and Figure 5 . When the I/O requests on a virtual machine reached and kept at a rate of over 4000 blocks per second, interrupts and context switches also raised to a level of 1000-2000 times per second, which bring down the CPU utilization by about 10%. As the I/O request comes at a rate of about 50000 blocks per second, the interrupts and context switches may happen 20000 times per second, the CPU will be totally kept busy with them, with the total performance being reduced by about 40% or more. Applications with such characteristics will not be handled well on virtual machines.

6 Conclusion and Further Work

With the performance study of a set of virtual machines, insight into performance loss was summarized as follows: 1. Not all system calls contribute to the performance loss, we currently know that system calls performing kernel space process locking or synchronization, system calls performing I/O data exchange tend to slow down a virtual machine if the virtual machine monitor is not very capable of dealing with them. 2. CPU utilization is a critical index to the whole system performance. Frequent I/O request results in frequent context switch, which may reduce the CPU utilization by a large margin. 3. Cache miss rate has no significant impact to performance loss introduced by virtual machine. In future, works to diminish the cost of such system calls on virtual machines should be considered.

Reference

[1] N. Huber, M. von Quast, *Evaluating and Modeling Virtualization Performance Overhead for Cloud Environments*. Fasanengarten 5, Karlsruhe, Germany, Feb. 2011.

[2] F. Benevenuto, C. Fernandes, M. Santos, V. Almeida, J. Almeida, *A Quantitative Analysis of the Xen Virtualization Overhead*. Computer Science Department, Federal University of Minas Gerais, Brazil, Jul. 2010.

[3] U. F. Minhas, J. Yadav, A. Aboulnaga, K. Salem, *Database Systems on Virtual Machines: How Much do You Lose?*. University of Waterloo, Jan. 2008.

[4] H. Mousa, K. Doshi, E. Ould-Ahmed-Vall, *Characterizing Performance in Virtualized Execution*. Department of Computer Science, University of California, Santa Barbara, Santa Barbara, CA 93106, Jan. 2008.

[5] M. Guevara, C. Gregg, *Measuring Basic Linux Operations*. CS 656 Operating Systems - Springer 2009, Feb. 2009.

[6] K. Najafi, Professor Eddie Kohler, Steve VanDeBogart, Jun. 2008.

[7] Q. Ali, V. Kiriansky, J. Simons and P. Zaroo, *Performance Evaluation of HPC Benchmarks on VMware's ESXi Server*. VMware Inc., Sep. 2011.

[8] I. Ahmad, A. Gulati, A. Mashtizadeh, M. Austruy, *Improving Performance with Interrupt Coalescing for Virtual Machine Disk IO in VMware ESX Server*. VMware Inc., Palo Alto, CA 94304, Apr. 2009.

[9] A. Navarro, R. Asenjo, S. Tabik, *Load Balancing Using Work-stealing for Pipeline Parallelism in Emerging Applications*. IBM Research Report, RC24732 (W0901-066) January 20, 2009, Computer Science.

[10] A. R. Butt, C. Gniady, Y.C. Hu, *The Performance Impact of Kernel Prefetching on Buffer Cache Replacement Algorithms*. Purdue Univeristy, West Lafayette, IN 47907.

[11] R. Nikolaev, G. Back *Perfctr-Xen: A Framework for Performance Counter Virtualization*. Virginia Polytechnic Institute, Blacksburg, Mar. 2011.

[12] B. Li, J. Li, T. Wo, C. Hu, L. Zhong, *A VMM-based System Call Interposition Framework for Program Monitoring*. School of Computer Science and Engineering, Beihang University, Beijing, China, Dec. 2011.

[13] J. Pfoh, C. Schneider, C. Eckert, *Nitro: Hardware-based System Call Tracing for Virtual Machines*. Technische Universität München, Munich, Germany, Nov. 2011.

[14] J. E. Smith, R. Nair, *Virtual Machines - Versatile Platforms for Systems and Processes*. Morgan Kaufmann, Elsevier, San Francisco, 2005, 416-417.

[15] Cachegrind: a Cache-miss Profiler.
`http://wwwcdf.pd.infn.it/valgrind/cg_main.html`

[16] Behrooz Parhami, *Computer Architecture - From Microprocessors to Supercomputers*. Oxford University Press, New York, Oxford, 2005.

[17] C. Li, C. Ding and K. Shen, *Quantifying The Cost of Context Switch*. Dept. of Computer Science, University of Rochester, 2007.

Triplestore Evaluation unter Verwendung des DBpedia SPARQL Benchmark

Oliver Grund
Technische Universität Chemnitz, Fakultät für Informatik, D-09107 Chemnitz

oliver.grund@informatik.tu-chemnitz.de

Zusammenfassung

In absehbarer Zukunft wird das effiziente Speichern und Abfragen von Daten im RDF Format immer mehr an Bedeutung gewinnen. Dafür werden sogenannte Triplestores verwendet. Doch zwischen den jeweiligen Implementierungen verschiedener Anbieter gibt es teilweise deutliche Performanceunterschiede. Der Artikel stellt die Aufgabe des Benchmarking von aktuellen Triplestores im Rahmen eines Forschungsseminars vor. Dazu werden kurz die Techniken und Methoden erläutert, die hierfür zum Einsatz kommen. Besonderes Augenmerk liegt auf dem DBpedia SPARQL Benchmark und der Entstehung der darin verwendeten Query-Templates. Schließlich werden die zu testenden Triplestore Implementierungen knapp vorgestellt, gefolgt von einem kurzen Ausblick auf die Testumgebung.

1 Einleitung

Das Semantic Web ist eine Weiterentwicklung des WWW und soll es mittelfristig möglich machen, dass die Informationen im Web besser zugänglich werden. Dazu wird eine formale Beschreibung von strukturierten Aussagen über Entitäten und Ressourcen bereitgestellt. Diese erlaubt den durch das Web verbundenen Maschinen, die Informationen nach ihrem semantischen Gehalt zu durchsuchen und entsprechend zu verarbeiten. Die grundlegende Technologiekomponente des Semantic Web ist das Resource Description Framework (RDF)[1], ein verbreiteter und offizieller Standard, den das W3C 2004 veröffentlicht hat. Demnach besteht jede elementare Aussage aus Subjekt, Prädikat und Objekt, dem sogenannten RDF-Tripel.

Für eine zukünftige Verbreitung und Umsetzung vieler Semantic Web und Linked Data Technologien, spielt die Speicherung von RDF-Daten, das vorherige Serialisieren dieser Daten sowie deren effiziente und performante Abfrage eine zentrale Rolle. Diese Aufgaben werden von Triplestores übernommen, die somit zum Rückgrat der Infrastruktur im Bereich der Semantic Web Anwendungen werden. Die verschiedenen Triplestore Implementierungen verfolgen oft unterschiedliche Ansätze diesen Herausforderungen gerecht

[1] http://www.w3.org/TR/rdf-concepts/

zu werden und zusätzlich eine hohe Performance und Stabilität zu gewährleistern. Daher ist es unerlässlich, verlässliche Analysemethoden zur Evaluation von Triplestores anzuwenden, um eine fundierte Grundlage für die Entscheidung über die konkret eingesetzte Implementierung zur Verfügung zu haben.

Für diesen Zweck wurden verschiedene RDF Datenbank Benchmarks entwickelt, um die Vergleichbarkeit der Performance von verschiedenen Triplestore-Implementierungen zu möglichen. Beispiele hierfür sind der Berlin SPARQL Benchmark (BSBM)[2] oder der Lehigh University Benchmark (LUBM)[3]. Sie testen die RDF-Ladegeschwindigkeit, bestimmte SPARQL-Funktionalitäten, die Geschwindigkeit der SPARQL Anfragen sowie die zur Bearbeitung der Anfragen implizit benötigten Inferenzen. Die Funktionsweise vieler solcher Benchmarks ist jedoch nicht auf die spezifischen Besonderheiten des RDF Datenmodells ausgelegt, sondern ähnelt von den Methoden und Datenstrukturen her den für relationale Datenbanken entworfenen Benchmarks. Doch die große Heterogenität der Klassen und Properties, besonders in großen RDF knowledge bases, verlangt eine sehr spezifische Arbeitsweise, da diese sich nicht ohne Probleme auf das relationale Datenmodell zurückführen lassen. Daher sind die typischen Charakteristiken für das Laden, Ändern und Abfragen von RDF-Daten teilweise ganz Andere als bei herkömmlichen Datenbanken mit relationalem Datenbestand. [MUA10]

2 DBpedia SPARQL Benchmark

Um die besonderen Anforderungen, welche an RDF Triplestores gestellt werden, genauer untersuchen zu können, entwickelten mehrere Beteiligte vom internationalen DBpedia Projekt[4] einen Benchmark namens DBpedia SPARQL Benchmark (DBPSB)[5]. Dieser ist in Java implementiert und als Open Source Projekt frei verfügbar. Es werden auch die Daten der DBpedia knowledge base für Interessierte bereitgestellt, sodass eine vergleichbare Grundlage zur Evaluierung von anderen Triplestore-Implementationen möglich ist. Durch ein spezialisiertes Verfahren von Analyse und Verarbeitung der an DBpedia gestellten Anfragen wurden 25 Query-Templates extrahiert, die verwendet werden können, um eigene Triplestore-Installationen zu testen. Statt synthetische Testdaten zu verwenden, werden praktisch eingesetzte RDF-Daten genutzt. Das ist von großem Vorteil, da so die Evaluation den Bedingungen im realen Einsatz sehr nahe kommt. Der spezifisch für Triplestores entworfene native SPARQL Benchmark lässt somit genaue Schlüsse auf die Leistungsfähigkeit der jeweiligen Implementation zu und unterstützt schließlich die Entscheidungsfindung bei der Auswahl des geeigneten Triplestores.

Die Ausführung des Benchmarks läuft in 3 Phasen ab. Nachdem die zu Grunde liegenden RDF-Daten geladen wurden, wird das System neu hochgefahren, um eventuell vorhandene Inhalte im Zwischenspeicher zurückzusetzen. Anschließend folgt eine Warm-up Phase, in welcher dem System zufällige Anfragen gestellt werden, die von den später im

[2] http://www4.wiwiss.fu-berlin.de/bizer/BerlinSPARQLBenchmark/spec/
[3] http://swat.cse.lehigh.edu/projects/lubm/
[4] http://dbpedia.org/
[5] http://aksw.org/Projects/DBPSB

echten Testlauf gestellten Anfragen verschieden sind. Schließlich beginnt die tatsächliche Testphase, in der über einen festgelegten Zeitraum (z.B. 60 Minuten) die vorher speziell ausgewählten Anfragen gestellt werden. Die resultierenden Antwortzeiten werden in zwei unterschiedlichen Metriken zusammengefasst, in die Gesamtperformance des Systems und die Performance der einzelnen Query-Templates. Die Gesamtperformance wird in Query Mixes per Hour (QMpH) angegeben, also der Anzahl an Anfragen die während einer Stunde vom System beantwortet werden können. Das Ergebnis für die Messungen der einzelnen Anfragen wird in Queries per Second (QpS) bestimmt. Diese Einheit beinhaltet also die Anzahl der bearbeiteten Anfragen pro Sekunde, die das System für einen bestimmten Anfragetyp erfolgreich beantworten kann. Dies erlaubt eine fundierte Auskunft, welche Triplestore Implementierung eine bestimmte Kombination von SPARQL Features besser unterstützt als die Konkurrenz.

Im Juni 2011 veröffentlichten die Entwickler des DBPSB die Ergebnisse ihrer Leistungsmessungen nach der Durchführung des Benchmarks auf einigen der beliebtesten und meistgenutzten Triplestores: Virtuoso, Sesame, Jena-TDB und BigOWLIM. Die genauen Resultate sind auf der Projektseite des DBPSB einzusehen. Sie machen deutlich, dass die Performanceunterschiede zwischen den jeweiligen Implementierungen größer ausfallen als es zu erwarten war. Bei einzelnen Anfragetypen schwanken die Antwortzeiten teils drastisch. In vielen Bereichen konnte sich Virtuoso als performantester Triplestore durchsetzen. [MLAN11]

3 SPARQL

Um über die Endpunktschnittstellen auf die RDF Tripel des jeweiligen Stores zuzugreifen, wird als Anfragesprache die SPARQL Protocol And RDF Query Language verwendet, kurz SPARQL[6], welche 2008 als offizielle Recommendation des W3C freigegeben wurde. SPARQL verwendet eine SQL ähnliche Syntax, die das Extrahieren und Modifizieren von Daten aus bestehenden RDF Graphen ermöglicht. Weiterhin lassen sich Sortierkriterien festlegen und Eingrenzungen durch Filterung der Ergebnismenge anhand von konkreten Regeln vornehmen. Auch die Struktur der Ergebnisse, die beispielsweise wieder als Tripel oder auch in tabellarischer Form zurückgeliefert werden können, lässt sich festlegen.[7]

Das genaue Referenzieren der gewünschten Daten wird ermöglicht durch die Angabe verschiedener Operatoren. Mit dem Schlüsselwort OPTIONAL lassen sich zum Beispiel Anfragemuster definieren, welche nicht zwingend sondern nur optional vorhanden sein müssen. Das Schlüsselwort UNION erlaubt hingegen die Angabe alternativer Teile eines Musters. Das zurückgelieferte Ergebnis entspricht der Vereinigung der jeweiligen Teilergebnisse, also allen Daten die mindestens eine der Bedingungen erfüllen. Weiterhin können Optionen und Alternativen kombiniert werden, sodass sich mehrere Graph-Muster durch Konjunktion und Disjunktion zu einer einzigen Anfrage zusammenfügen.

[6] `http://www.w3.org/TR/rdf-sparql-query/`
[7] Eine hilfreiche SPARQL Einführung: `http://jena.apache.org/tutorials/sparql.html`

```
PREFIX  rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
PREFIX  ex:  <http://www.example.org/>.
SELECT    ?x ?firmensitz ?name
WHERE   { ?x rdf:type ex:Autohersteller .
          { ?x ex:hatFirmensitzIn ?firmensitz . } UNION
          { ?x ex:hatHauptstandortIn ?firmensitz . } OPTIONAL
          { ?x ex:hatNamen ?name . } }
```

Listing 1: Beispiel zur Kombination von Optionen und Alternativen

Im hier dargestellten Beispiel ist eine solche Kombination enthalten. Die Anfrage liefert eine Tabelle mit je einer Spalte für die URI, den Firmensitz und den Namen des Autoherstellers. Es werden demnach alle Autohersteller ausgegeben, die entweder einen Firmensitz oder einen Hauptstandort haben, sowie der Name des Autoherstellers, falls es einen im RDF Graphen gibt.

Für einen praxisnahen Triplestore-Benchmark ist eine gezielte Auswahl von generischen, realistischen Anfragetypen von großer Bedeutung, um für die Praxis verwertbare Ergebnisse zu erzielen. Dafür verwenden die Entwickler des DBPSB ein gesondertes Verfahren. In diesem werden durch Anfrageanalyse und –gruppierung einige spezielle Anfragen ausgewählt. Die Basis dafür bieten die über einen Zeitraum von mehreren Monaten an den SPARQL Endpunkt von DBpedia gestellten Anfragen. Dieser Prozess beginnt mit der Selektion der am häufigsten ausgeführten Anfragen. Diese werden danach durch verschiedene Vereinfachungen auf eine möglichst einheitliche Form gebracht, indem von Besonderheiten abstrahiert wird und sehr ähnliche Varianten des gleichen Anfragetypus zusammengefasst werden. So werden u.a. die Präfixe und SPARQL Schlüsselwörter entfernt sowie die Namen der Variablen angeglichen, um eine bessere Vergleichbarkeit der einzelnen Anfragen zu gewährleisten. Anschließend werden diese mit speziellen Algorithmen auf ihre Ähnlichkeit überprüft und es wird eine Gruppierung von gleichartigen Anfragen vorgenommen. Daraus lassen sich dann einzelne Anfrageprototypen erzeugen, die einen Großteil der gesamten Anfragenmenge abdecken.

Um die Leistungsfähigkeit der jeweiligen Triplestore-Implementierungen genauer zu erfassen, ist es nötig, die Bearbeitung der von SPARQL unterstützten Features sowohl einzeln als auch in Kombination methodisch abzuprüfen, indem die generierte Menge der Anfrageprototypen nach diesen Merkmalen durchsucht und die passenden Anfragen ausgewählt werden. Dabei spielt die Anwendung der UNION- und OPTIONAL-Funktion eine Rolle, aber auch Funktionen zur Filterung und Eingrenzung wie FILTER und DISTINCT. Schließlich sind noch die Anzahl der insgesamt enthaltenen Graph-Muster und die der JOIN-Operationen ausschlaggebend.

Die nun ausgewählten Anfrageprototypen werden anschließend in Templates konvertiert, indem einzelne Teile der Anfrage durch Platzhalter ersetzt und später mit Hilfe einer vorbereiteten Liste von Werten variiert werden.

```
PREFIX  foaf: <http://xmlns.com/foaf/0.1/>.
PREFIX  rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
SELECT    ?var2
WHERE   { ?var3   foaf:homepage ?var2 .
          ?var3   rdf:type      %%var%% .   }
```

Listing 2: angepasster Auszug aus den DBPSB Query-Templates inkl. Platzhalter

Der hier im Beispiel verwendete Platzhalter (%%var%%) variiert den RDF-Typ der Variable var3 mit vorher aus dem RDF-Graphen extrahierten Daten. Dies ermöglicht, dass trotz der schematisch generierten Anfrage immer Ergebnisse zurückgeliefert werden und sich die ausgeführten Anfragen trotzdem unterscheiden. Letzteres dient vor allem dazu, Cache-Effekte zu vermeiden. Denn die mehrfache Ausführung derselben Anfrage würden die Triplestores sehr schnell bearbeiten können, was das Messergebnis stark verfälschen würde, da in der Realität nur höchst selten eine solch große Anzahl identischer Anfragen hinter einander gestellt wird. [SAN+11]

4 Evalutaion

Im Zuge des aktuellen Forschungsseminars Modern Software Engineering and Semantic Web des Sommersemesters 2012 unter Leitung von Dr. Sören Auer, übernehme ich die Aufgabe, mit Hilfe des DBPSB die Performance von drei weiteren Triplestore Implementierungen zu testen: 4Store[8], BigData[9] und AllegroGraph[10].

4Store ist ein in C geschriebener Triplestore, der von Garlik entwickelt wurde und frei verfügbar ist, da er unter der GNU General Public Licence Version 3 steht. Das Projekt stellt eine performante, sichere, stabile und skalierbare RDF Datenbank mit Query Engine zur Verfügung und läuft unter unix-ähnlichen Betriebssystemen. Es wurde optimiert um auf Shared-Nothing Clustersystemen mit bis zu 32 Knoten zu arbeiten, läuft aber auch auf Einzelrechnern. Schon einige knowledge bases mit mehreren Millarden Tripeln nutzen heute 4Store als Plattform. Es enthält einen SPARQL HTTP Server der Anfragen per SPARQL Protocol beantwortet und bietet Client Bibliotheken unter anderem für PHP, Ruby und Python.

Bei der BigData RDF Database handelt es sich um einen Triple- bzw. Quadstore, der eine schnelle, verlässliche und stark skalierbare RDF Datenbank bietet. Das unter der GNU General Public Licence Version 2 stehende Projekt wird dank einer aktiven Community ständig weiterentwickelt. BigData unterstützt eine Volltextsuche, ebenso RDFS- als auch teilweise OWL-Inferenzen. Es ist außerdem in der Lage, verteilte Operationen auf einem Cluster mit dynamischen Indizes Bereich durchzuführen, sodass der Cluster mit der zunehmenden Größe der Daten skalieren kann, ohne dass diese beim Hinzufügen von zusätzlichen Knoten jedes Mal neu geladen werden müssen. Das Java-Projekt unterstützt

[8] http://4store.org/
[9] http://www.bigdata.com/bigdata/blog/
[10] http://www.franz.com/agraph/allegrograph/

eine Reihe von SPARQL Servern, wie beispielsweise den OpenRDF Sesame HTTP Server oder den integrierten Nano SPARQL Server, der eine einfache REST API bietet.

Das kommerzielle AllegroGraph wurde von Franz Inc. entwickelt und in Common Lisp implementiert. Es bietet eine moderne, leistungsfähige Datenbank für RDF-Graphen und verspricht eine effiziente Hauptspeichernutzung in Kombination mit Disk-basierter Speicherung, sodass es mit Milliarden von Tripeln skaliert und gleichzeitig eine performante Interaktion ermöglicht. AllegroGraph unterstützt auch RDFS++ Reasoning, eine nichtstandardisierte Eigenentwicklung und Kombination aus RDFS- und OWL-Inferenzen, die auf Wunsch eine sehr schnelle, dafür aber unvollständige Antwort zurückliefert. Auch zahlreiche Backup-Funktionen sind enthalten. Die neuste Version stellt eine SPARQL 1.1 Query Engine bereit, verbessert nochmals die Performance und bietet zur Gewährleistung der Sicherheit eine Client Authentifizierung mit X.509 Zertifikaten an. Der Zugriff auf den Server erfolgt über die REST-Schnittstelle und unterstützt Adapter für viele verschiedene Sprachen, u.a. Sesame Java, Sesame Jena, Python und Lisp. Es existieren aber dank mehrerer Community Projekte auch Open-Source-Adapter für Ruby, Scala und Perl. Der Hersteller bietet optional eine kostenlose AllegroGraph Free Server Edition an, die allerdings auf 5 Millionen Tripel limitiert ist.

Die innerhalb des Forschungsseminars gestellte Aufgabe beinhaltet also die Installation und Konfiguration der Triplestore Implementationen, das Laden der RDF-Daten sowie die Durchführung der Testdurchläufe des DBPSB auf allen drei Installationen. Um eine möglichst gute Vergleichbarkeit der Resultate zu erreichen, werden die Messungen auf einer sehr ähnlichen Hardware durchgeführt, wie bei den vorherigen Tests. Der Server besitzt zwei AMD Opteron 4184 Prozessoren (je 6 Kerne á 2.80GHz), 32GB RAM, ein 6TB RAID-5 HDD Verbund und läuft unter Ubuntu Server 10.10.

Aktuell dauert die Vorbereitung des Servers noch an, daher liegen momentan noch keine konkreten Ergebnisse vor. Sobald die ersten Resultate bereit stehen, werden sie wohl auf der DBPSB Projektseite veröffentlicht.

Literatur

[MLAN11] Mohamed Morsey, Jens Lehmann, Sören Auer und Axel Cyrille Ngonga Ngomo. DBpedia SPARQL Benchmark – Performance Assessment with Real Queries on Real Data, 2011.

[MUA10] Michael Martin, Jörg Unbehauen und Sören Auer. Improving the Performance of Semantic Web Applications with SPARQL Query Caching, 2010.

[SAN+11] Saeedeh Shekarpour, Sören Auer, Axel Cyrille Ngonga Ngomo, Daniel Gerber, Sebastian Hellmann und Claus Stadler. Keyword-driven SPARQL Query Generation Leveraging Background Knowledge, 2011.

Supporting Semantic Interoperability in Inter-Widget-Communication-enabled User Interface Mashups

Sebastian Heil, Olexiy Chudnovskyy * and Martin Gaedke *

Chemnitz Univeristy of Technology
sebastian.heil@informatik.tu-chemnitz.de
olexiy.chudnovskyy@informatik.tu-chemnitz.de
martin.gaedke@informatik.tu-chemnitz.de

Abstract: Recent developments in the field of user interface mashups have acknowledged the necessity of Inter-Widget Communication. However, supporting IWC in UI mashups which are not pre-composed is not trivial. This paper illustrates the necessity of IWC for UI mashups, details disadvantages of current approaches and identifies requirements which we believe will help creating better solutions. We also present a first draft of the IWC framework we propose to address the shortcomings identified.

1 Introduction

Within the last years the development of web applications in the form of *user interface mashups (UI mashups)* has become increasingly important. [WDJS11] Their asynchronous and event-driven nature [SM09] and their ability to spontaneously create custom views on specific issues by composing independently developed, re-usable components, called *widgets*, have spawned a wide variety of user interface mashup applications and frameworks. [Wil12]

One of the major challenges faced by them is to enable communication between widgets, which is called *Inter-Widget Communication (IWC)*. Without IWC, each widget in a mashup works in isolation inducing redundancy as widgets are not enabled to co-operate and share common functionality thus causing them to become more extensive and also less reusable.

What is more, IWC has a strong impact on usability. [Wil12] Imagine, for instance, a mashup application concerned with travel which could feature a social travel advisory widget listing hotels highly recommended by the customer's friends, a map widget for displaying possible destinations, a translator widget to translate foreign-language hotel descriptions, a calendar widget displaying the personal calendars of the customer and a booking widget to book accommodation and flights.

*academic supervisor

Without IWC, the usage of this mashup requires several manual actions: The customer browses the list of recommended hotels and checks their locations by manually copying the address from the hotel description into the location prompt of the map widget. He then reviews the hotel description in the advisory widget. In order to translate it, he copies its text into the translator widget. After checking the room availability in the booking widget, the customer wants to cross-check the dates against his calendars. In order to do so, he has to manually leaf through the calendar widget to get to the corresponding dates. Ultimately, he books his stay and manually adds his planned vacations to his calendar by creating an event in the calendar widget.

As illustrated by this example, without interactivity between widgets, users have to perform multiple manual actions in order to synchronise widgets and simulate communication mitigating the usability of the mashup application.

2 Related work

Recent UI mashup projects like Geppeto[1] or Scrapplet[2] are aware of the necessity of IWC and have developed a wide range of different approaches. [SvS09] In 2011, Zuzak et al. conducted a thorough analysis of cross-context communication methods covering more than 30 systems many of which can and have been used for IWC. [ZIB11]

As of today, in 2012, there are two major competitors in the field of widget specifications: OpenSocial Gadgets and W3C Widgets. These specifications detail packaging, configuration, metadata, communication and security aspects. While specified for OpenSocial Gadgets, the W3C Widgets specifications lack a description of IWC. As more and more widget engines supporting W3C Widgets come into existence, the lack of a normative description of IWC has created a wide range of different approaches. [Goo11][CÍ1b][CÍ1a]

3 Current research challenges

Though many of these systems may work well in *Orchestrated UI mashups* [WDJS11] (pre-composed mashups) where experts, called mashup designers or mashup creators [SM09], build mashup applications by composing widgets with knowledge of the various data formats used and ensuring interoperability by thoroughly selecting widgets or manually defining the required transformations [SM09], there are not many solutions supporting interoperability in non-pre-composed, i.e. *Choreographed UI mashups*. [WDJS11]

However, the necessity of Inter-Widget Communication in these mashups, where end-users compose custom user interfaces by adding widgets, is evident [WDJS11], especially if the resulting solutions shall overcome being mere portal sites without any inter-activity between their components. [DST+11] The travel booking scenario previously described

[1] http://www.geppeto.fer.hr/
[2] http://www.scrapplet.com/

could refer to an orchestrated mashup as well. In order to enable IWC between arbitrary widgets employing different data formats in the messages exchanged between them, an IWC framework has to provide means of non-explicit communication while, by strictly separating semantic concepts from syntax, supporting the required transformation of data formats. Non-explicit communication refers to communication patterns which do not require explicit addressing of messages such as, for instance, broadcast messaging.

4 Requirements

In order to provide basic and easy-to-use communication abilities and alleviate the shortcomings in independent development of Widgets previously described, an appropriate Inter-Widget Communication framework should comply with the following criteria:

1. Independent widget development
2. Ad-hoc widget communication
3. Syntax-independent communication about semantically relevant subjects
4. Extensibility and re-use

1 Independent widget development A good solution shall enable widget developers to create their widgets without any a-priori knowledge of other widgets which may or may not be present in the parent context of their own widget at runtime. This suggests that widgets shall not communicate by exchanging messages explicitly directed to one or several specific recipients nor by reading from and writing to any shared memory dedicated to but one widget or a specific group of widgets. Instead, widgets shall rather communicate in a way which allows for the participation of any interested party. The idea behind is to obey to the principle of loose coupling proven successful in Web Engineering. [PWB09]

2 Ad-hoc widget communication A solution complying with this criterion shall provide possibilities to inform and receive notifications from other widgets about relevant events if the developer considers this necessary. The ad-hoc feature of Choreographed UI mashups necessitates the delivery of messages to all interested widgets in a continuously changing environment: As the mashup composition is not predefined, end-users may add or remove widgets at runtime. Removing a widget shall not cause interruptions in the communication flows between other widgets, any widget newly added to the running mashup shall be able to participate in the communication immediately.

3 Syntax-independent, transparent communication about semantically relevant subjects In order to comply with this requirement, widgets shall be able to communicate about relevant subjects regardless of specific syntax. This implies that a good solution shall provide tools to grant a strict separation of syntax and semantics by performing data format transformations if necessary.

4 Extensibility and re-use If the IWC-framework does not provide a transparent data format transformation for a particular widget as described above, it shall enable the widget developer to extend its functionality by supplying an implementation of the missing transformation in a standardized way. Moreover, considering criterion 3, the solution shall enable data transformations implemented for one widget to be re-used by other widgets requiring them.

5 Proposed IWC framework concept

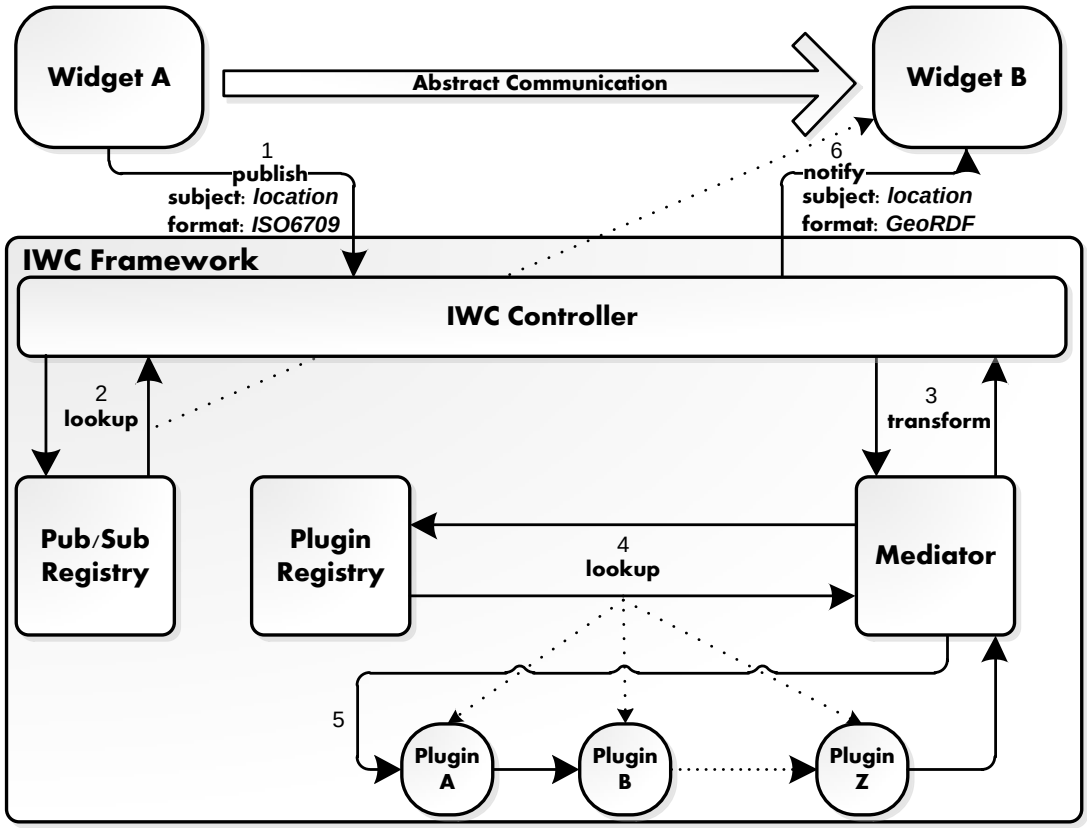


Figure 1: Proposed IWC framework architecture and publication flow

In order to solve the problems described in section 1, we propose the IWC framework architecture depicted in Figure 1. Being a "common model for choreographed communication" we join Wilson in advocating "a *publish-subscribe messaging* approach". [Wil12] It shows the publication flow triggered by Widget A publishing a message with plugins A to Z performing the transformations and subscribed Widget B receiving the message.

The framework provides a basic publish-subscribe interface. This interface is extended by additional format parameters inspired by *HTTP content negotiation*. [FGM+99] All publish-messages declare the format they are employing. Likewise, widgets subscribing to a certain subject state the formats they are able to understand, corresponding to the

"Accept header field" in an HTTP request.

Using the Mediator component, the framework transforms data from source to target format. This is done by querying the Plugin Registry for a list of plugins supporting transformation from source to target format. By performing their transformations forwarding the output of a plugin to the subsequent one in the order given by the Plugin Registry, the Mediator finally yields the data in the desired target format.

The idea of chaining plugins adds support for transformations where there is no plugin available which explicitly performs the transformation required, but which can be achieved by a sequence of transformations with the corresponding plugins available.

Unfortunately, the use of several plugins introduces a certain degree of complexity into the transformation process. To address this issue, we advocate the use of a central data format, a lingua franca as it were, for each topic so plugins merely have to transform a specific data format from and to this intermediate format consequently limiting complexity by restricting actual chain lengths to two plugins.

Our approach aims at identifying communication subjects commonly encountered in user interface mashups. This is inspired by the default intents collated by the Web Intents Task Force. [Web12] In addition to that, the *Web Intents* specification enables to register and request any service by defining the "action" parameter. Moreover, the "type" parameter which is a "string indicating the type of the data payload" [BHK12] ensures the description of the syntax employed in the intent data. We resemble this with subject and format.

So far, we have identified the following default IWC subjects:

1. Geo Location
2. Date
3. Search
4. Message (Email / SMS / Chat) Received / Sent
5. Content Created / Uploaded
6. Selection

Additionally, we aim at supporting any subject apart from those listed above in the same way the Web Intents specification does by defining a "Typed Datum Multicast" which allows to publish data on any subject and of any type.

6 Conclusion and future work

To conclude, we underlined the necessity of IWC for UI mashups and outlined the problems encountered in current approaches, in particular, for choreographed UI mashups. We consider the lack of independency in widget development which is introduced by recent solutions the most crucial issue. In order to alleviate this, this paper proposed several requirements for UI mashup IWC approaches and presented a first draft to increase widget development independency and semantic interoperability by applying well-known patterns such as plugin architecture, content negotiation and Web Intents to the IWC domain.

As our research on this topic still is at an early stage, we will elaborate on these requirements and adapt the architectural draft accordingly. Moreover, we intend to identify example scenarios for IWC-enabled choreographed UI mashups and derive further default subjects from them. This will also affect the decision on how to perform the transformations. The next step will be to implement the IWC framework, test the implementation in these scenarios and validate the framework against our requirements. Moreover, the process of implementing will likely yield interesting insights which we would like to report on by composing them into a set of best practices.

References

[BHK12] Greg Billock, James Hawkins, and Paul Kinlan. Web Intents. *W3C Editor's Draft*, 2012.

[CÍ1a] Marcos Cáceres. Widget Interface. *W3C Candidate Recommendation*, 2011.

[CÍ1b] Marcos Cáceres. Widget Packaging and XML Configuration. *W3C Recommendation*, 2011.

[DST+11] Florian Daniel, Stefano Soi, Stefano Tranquillini, Fabio Casati, Chang Heng, and Li Yan. Distributed Orchestration of User Interfaces. *Information Systems*, 37(6):539–556, 2011.

[FGM+99] Roy Fielding, J Getty, J Mogul, H Frystyk, L Masinter, P Leach, and Tim Berners-Lee. RFC 2616: Hypertext Transfer Protocol – HTTP/1.1, 1999.

[Goo11] Google Inc. OpenSocial Core Gadget Specification 2.0.1, 2011.

[PWB09] Cesare Pautasso, Erik Wilde, and U C Berkeley. Why is the Web Loosely Coupled ? A Multi-Faceted Metric for Service Design. In *Proceedings of the 18th International World Wide Web Conference (WWW2009)*, Madrid, Spain, 2009.

[SM09] Michele Stecca and Massimo Maresca. An execution platform for event driven mashups. *Proceedings of the 11th International Conference on Information Integration and Web-based Applications & Services - iiWAS '09*, page 33, 2009.

[SvS09] Srbljic Siniša, Dejan Škvorc, and Daniel Skrobo. Widget-Oriented Consumer Programming. *AUTOMATIKA: Journal for Control, Measurement, Electronics, Computing and Communications*, 50:252–264, 2009.

[WDJS11] Scott Wilson, Florian Daniel, Uwe Jugel, and Stefano Soi. Orchestrated user interface mashups using w3c widgets. In *Proceedings of ComposableWeb*. Springer, 2011.

[Web12] Web Intents Task Force. webintents.org, 2012.

[Wil12] Scott Wilson. Design challenges for user-interface mashups: user control and usability in inter-widget communications, 2012.

[ZIB11] Ivan Zuzak, Marko Ivankovic, and Ivan Budiselic. A Classification Framework for Web Browser Cross-Context Communication. *Arxiv preprint arXiv:1108.4770*, 2011.

Methodische Analyse von Eigenschaften einer vertrauten Struktur für eine explorative Visualisierung im Kontext des Semantischen Web

Stefanie Oertel

TU Chemnitz, Fakultät für Informatik
mail@stefanieoertel.com

Abstract: Leider sind Endnutzer selten die Zielgruppe aktuell entwickelter Wissensrepräsentationen im Kontext des Semantischen Web. Um sie beim Erfassen komplexer Sachverhalte zu unterstützen und den kognitiven Aufwand zu reduzieren, müssen sie stets zu Beginn des Designprozesses einbezogen werden. Grundlage für die vorliegende Arbeit bilden zwei vorangegangene Studien, die sich mit der Analogie physischer Substanzen zum Einsatz für die Gestaltung von Interaktionen befassen. Mein Beitrag ist ein methodischer Ansatz für die Analyse von Eigenschaften einer vertrauten Struktur, der potentielle Nutzer vor der Konzeptentwicklung in den Designprozess einbezieht. Untersuchungsgrundlage bildet die Substanz Schaum.

1 Einleitung

Das Semantische Web ist die Erweiterung des derzeitigen Web [GC06] und dient der Konzeptbeschreibung und der Beschreibung von Relationen in Wissensdomänen [PB06]. Ziel ist die eindeutige Beschreibung von Aussagen [Luc11] und eine damit verbundene Maschinen-lesbare Repräsentation der Bedeutungen. Informationen sollen automatisch interpretiert, verarbeitet, verknüpft und in neuen Zusammenhang gesetzt werden. Wissen muss man demnach strukturieren und formalisieren. Laut Tim Berners-Lee [PB06] soll das Semantische Web die gemeinsame Kooperation von Mensch und Maschine befähigen, um die Informationsbedürfnisse von Nutzern zu befriedigen. Dies verdeutlicht, dass vorrangig der Endnutzer von den entwickelten Technologien profitieren sollte. Den Nutzen des Semantischen Web belegt man demnach durch für den Nutzer optimierte Anwendungen. Aktuelle Nutzer verfügen über weniger technisches Know-how, als die damaligen „Techies“ und leider sind Navigationselemente oft in einer solchen Fülle vorhanden, dass sie den Nutzer eher überlasten, als ihm zu nutzen [PB06]. An diesen Stand müssen sich auch Anwendungen anpassen, indem sie leicht verständlich und minimalistisch gestaltet sind. Ist im Folgenden vom Endnutzer die Rede, ist damit kein Fachexperte und keine Maschine gemeint, sondern Nutzer, die wenig technisches Verständnis aufweisen.

Ontologien sind zentrale Bausteine des Semantischen Web [PB06]. Bei Ontologien handelt sich um Netze von Hierarchien, in welchen eine Menge von Begriffen einer bestimmten Domäne in logischen Relationen zueinander stehen. Die Sammlung beinhaltet Worte, die

Begriffe und semantische Relationen bezeichnen [ES06], zerlegt in SPO-Tripel (Subjekt, Prädikat, Objekt) [AQTF08]. Ontologien nutzt man zum Annotieren von Daten mit Metadaten zum Vermitteln der Bedeutung [GC06] und somit für den Austausch von Wissen.

Wissen ist die Expertise und Fähigkeiten einer Person über ein theoretisches oder praktisches Verständnis zu einem Gegenstand in einer bestimmten Zeit [Bha08]. Das Konzept der „Wasseranalogie des Wissens“ erläutert Wissen mit Bezug auf Ontologien. Es unterscheidet Wissen in fest, flüssig und gasförmig. Gasförmiges Wissen ist unstrukturiert und schlecht greifbar, wie es bei E-Mails, Blogs und Konversationen der Fall ist. Dagegen ist flüssiges Wissen teilstrukturiert und basiert auf einem gemeinsam kontrollierten Vokabular, so zum Beispiel bei Folksonomies [RR01, PB06]. Folksonomies („folk“ und „taxonomy“) sind Taxonomien, generiert durch die Zuweisung von Schlagworten durch die Community einer Webseite. Der Vorteil liegt in denen für Maschinen interpretierbaren semantischen Inhalten, die der Nutzer dieser Informationen selbst klassifiziert [ST10]. Festes Wissen ist strukturiert und greifbar und dient demnach als tragfähige Ontologie einer Wissensdomäne [RR01, PB06].

Eine Ontologie-visualisierende Killer-Applikation[1] existiert bislang nicht [LSR09]. Endnutzer, welche kein bis wenig Grundwissen aus dem technischen Bereich besitzen, können mit denen hierfür gewünschten vielfältigen Anforderungen[2] wenig anfangen. Steve Krugs erstes Gesetz der Usability „Don’t make me think!“ sollte auch in diesem Fall Fuß fassen. Effektive Darstellungen müssen demnach klar, naheliegend und selbsterklärend sein, da Nutzer sich nicht damit befassen wollen, wie etwas funktioniert [Kru06]. Da Wissen des Weiteren in direkter Relation zum Lernprozess steht [Bha08], sind Nutzer mit einer großen Bereitstellung von Funktionen überlastet. Infolgedessen ist mit Fokus auf den Endnutzer ein nicht-technischer und mehr auf den Endnutzer bezogener Ansatz für die Visualisierung der Semantik im Web vorzuziehen.

2 Verwandte Arbeiten

2.1 Natural User Interface

Es sind nicht Daten selbst, die einen Endnutzer interessieren, sondern durch Daten übermittelte Informationen. Da Daten das reale Leben repräsentieren [Yau11], ist es naheliegend, sie auf Basis von für den Endnutzer Wohlbekanntes abzubilden. Diesen Ansatz findet man auch im Bereich des Natural User Interface (NUI), dessen Ziel es ist, eine natürliche, intuitive Interaktion mit dem Computer zu gestalten. Hierbei interagieren virtuelle Dinge wie reale Objekte. Der Nutzer setzt erlernte, von unterschiedlichen Aspekten abhängige Handlungsmuster ein, welche dessen Wissen repräsentieren. Die Visualisierung spielt im Falle des NUI eine untergeordnete Rolle [Hen10].

[1] Eine Killerapplikation wäre eine Anwendung zum Begünstigen von Wissensaustausch.

[2] Liste für Visualisierungsanforderungen: `http://wordnet.princeton.edu/man/wngloss.7WN.html#toc3` (Zuletzt besucht: 30.05.2012)

2.2 Ontologie-Visualisierungsmethoden

Es existieren diverse Methoden [KHL+07] für die Visualisierung des Semantischen Web. Möglich ist auch bereits bestehende Konzepte für diese Zwecke zu adaptieren. [KHL+07] gibt einen guten Überblick über Ontologie-Visualisierungsmethoden. Sie lassen sich grob in die Kategorien „Indented list“, „Node-link and tree“, „Zoomable“, „Space-filling“, „Focus + context or distortion“ und „3D Information Landscapes“ einordnen. Der Großteil von Ontologie-Visualisierungstools verwendet Graphen oder auch Maps [LSR09].

2.3 Visualisierung durch die Assoziation mit physischen Substanzen

Brade et al. untersuchen in der Studie „Nutzung inhärenter Interaktionsangebote von Substanzen des Alltags“ physische Substanzen auf Interaktionsmöglichkeiten und Übertragbarkeiten auf die Mensch-Computer-Interaktion. Die Eigenschaften klassifizieren sie in Größe, Verformbarkeit, Interaktionsform, Verhalten, Reversibilität und Zustandsübergang. Ziel dieses Ansatzes sei die Reduktion der Bedienkomplexität interaktiver Systeme. Bei den gewählten Substanzen handelt es sich um farbiges Öl, Seifenblasen, Eier, Reis und Magnete[3] [BKK+11].

Bei der Wahl einer Substanz muss man abhängig der Zielgruppen die Vorkenntnisse und Verhaltensweisen der Nutzer berücksichtigen. Bei dieser Arbeit sei demnach zu prüfen, wie vielen Nutzern die Verwendung von beispielsweise farbigen Öl nicht geläufig ist. Gibt es Farbrückstände bei dem Interagieren mit farbigen Öl? Kann man farbige Öle vermischen? Ohne eine vorangehende Eigenschaftsanalyse entkoppelter interner (mentaler) Repräsentationen[4] von potentiellen Endnutzern über die jeweilige Substanz ist nicht davon auszugehen, dass Endnutzer Eigenschaften einer übertragenden Substanz in das Digitale mit deren realen Eigenschaften assoziieren. Ohne die Assoziationsmöglichkeit müssen potenzielle Endnutzer die Interaktionsform zunächst erlernen, wodurch sich die Bedienung vorerst erschwert. Ein Vorteil der Nutzung von Substanzen des Alltags für die Reduktion der Bedienungskomplexität ist somit nicht gegeben. Die Klassifizierung unterschiedlicher Substanzen dient der Auswahl einer geeigneten Visualisierungsgrundlage, birgt jedoch die Gefahr, vorhandenes Potenzial nicht auszuschöpfen, indem man weitere adaptierbare Eigenschaften nicht erkennt.

Eine weitere Studie von Brade et al. [BHG11] befasst sich mit der Idee der Assoziation einer Zellstruktur und Flüssigkeits-Metapher, um somit durch externe Repräsentationen den Nutzer bei dessen internen (mentalen) Repräsentationen zu unterstützen. Gerade zu Beginn dieses Prozesses erwarte der Nutzer Assoziationen und potenzielle Relationen. Daran anschließend ändere sich dessen Gedankenfluss über die gesammelten Informationen. Die Studie basiert auf den drei Metaphern Flüssigkeit, Seifenblasencluster und sogenannte „Realtime Blobs“ [Iva12], welche man sich als eine Art digitales Öl vorstellen kann. In

[3] Hierzu fand ein Workshop an der TU Dresden statt (2010): `http://www.youtube.com/watch?v=ygAq8JHGtvM` (Zuletzt besucht: 20.06.2012)

[4] Entkoppelte interne (mentale) Repräs. sind im Vorfeld vorhandene Vorstellungen. Gekoppelte interne (mentale) Repräs. sind wahrnehmungsgebunden (z. B. Prototypenevaluierung, Experiment mit der Substanz).

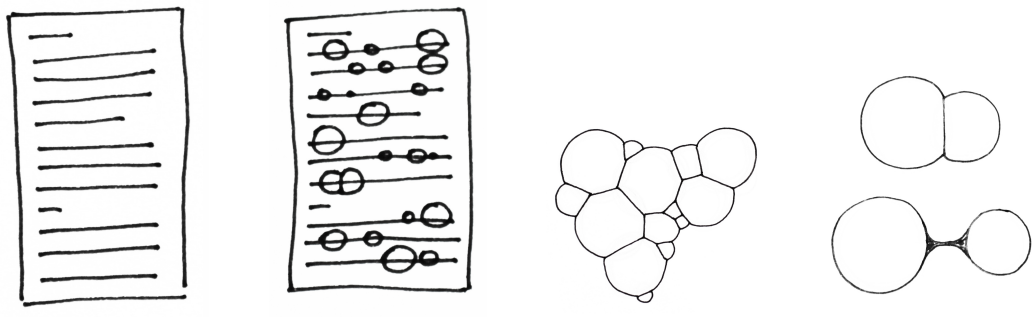
diesem Fall fand eine Evaluierung der Prototypen statt.

Zwar zog man im Rahmen dieser Studie den Endnutzer in den Designprozess mit ein, jedoch zu spät um entkoppelte interne (mentale) Repräsentationen zu analysieren. Des Weiteren weisen die Visualisierungen keinen nachvollziehbaren Bezug zu einem konkreten Kontext auf. Es fehlen exemplarische Szenarien, die die Verwendung physischer Substanzen nachvollziehbar veranschaulichen. Bezieht man sich beispielweise auf das Semantische Web und ersetzt die Zellstruktur und Flüssigkeits-Metapher durch die Metapher Schaum, lässt sich diese durch die Wasseranalogie des Wissens nachvollziehbar darstellen. Darauf wird im Punkt 3.1 näher eingegangen. Punkt 3.4 beschäftigt sich damit, wie sich Schaum und Seifenblasen aus Sicht der Endnutzer unterscheiden.

3 Eigenschaftenanalyse von Schaum

3.1 Entwurf des methodischen Ansatzes

Bei der Visualisierung soll es sich um etwas dem Endnutzer Vertrauten handeln, worüber dieser Vorkenntnisse besitzt. Schaum greift als Untersuchungsgrundlage das Konzept der „Wasseranalogie des Wissens“ auf. Gasförmiges Wissen ist unstrukturiert und schlecht greifbar wie im Beispiel von Blogs (Abb. 1a), wo man Informationen erst suchen muss. Markiert man Subjekte, Prädikate und Objekte in dem Text, bildet sich eine Struktur über den Informationen, wodurch das Wissen zum Teil greifbar wird (Abb. 1b). Zusammengefasst zu Schaum werden Relationen sichtbar (Abb. 1c). Die resultierende, greifbare Struktur unterstützt den Endnutzer die Informationen gesamtheitlich zu erfassen. Abbildung 1d zeigt ein SPO-Tripel. Das Prädikat ist der Schnittpunkt zwischen Subjekt und Objekt. Es sind unterschiedliche Darstellungen möglich.



(a) gasförmig (b) flüssig (c) fest (d) SPO-Tripel

Abbildung 1: Skizzenhafte Veranschaulichung der Einordnung von Schaum in das Konzept der Wasseranalogie des Wissens

Um die explorative Visualisierung für den Endnutzer klar, naheliegend und einfach zu gestalten, muss der Endnutzer bereits zu Beginn in den Designprozess einbezogen wer-

den. Die Eigenschaften sollen mit Hilfe von „Mixed Methods“ analysiert werden. Dies ist die Kombination aus einer qualitativen und quantitativen Methode, die sich auf dieselbe Fragestellung beziehen [JOT07]. Die qualitative Analyse verdeutlicht die Vielfältigkeit von Eigenschaften durch das Zusammentragen durch Endnutzer. Auf diese Weise werden individuelle, entkoppelte interne (mentale) Repräsentationen erfasst. Diese individuellen Sichtweisen werden infolgedessen durch eine quantitative Online-Umfrage gefiltert, um Eigenschaften zu identifizieren, die nicht auf einzelne Annahmen beruhen.

3.2 Qualitative Analyse

Um zunächst viele Eigenschaften zu identifizieren, begann ich mit einem qualitativen Ansatz mit Hilfe des Brainstormings[5] „Welche Eigenschaften besitzt Schaum?“ an dem mindestens sechs Personen im Alter von 24 bis 29 teilnahmen. Dazu zählen ein Fachexperte und fünf Endnutzer. Da es sich um ein öffentliches Online-Brainstorming handelt, können weitere Personen teilgenommen haben. Die resultierenden Antworten bilden die Basis für die folgende quantitative Analyse[6].

3.3 Quantitative Analyse

Die quantitative Analyse fand in Form eines Fragebogens statt. An der Befragung nahmen 14 Personen im Alter von 21 bis 30 teil. Dazu zählen ein Fachexperte und 13 Endnutzer. Bei der qualitativen, als auch quantitativen Analyse handelt es sich um Medienkommunikations-, Informatik- und Designstudenten, wissenschaftlichen Mitarbeitern und Nutzer mit einer technischen Ausbildung. Die Fragen sollten schnell beantwortet werden, um die entkoppelten internen (mentalen) Repräsentationen der befragten Personen zu identifizieren. Daher verzichtete ich unter anderem auf Anschauungsmaterial. Die Verwendung von abstrakten Begriffen dient dazu, ausreichend Raum für Interpretationsfreiheit zu gewähren und das variierende Vokabular der Teilnehmer zu berücksichtigen. Die Fragen wurden in einer Ratingskala (Abstufung: Ja, Tendenz zu Ja, Tendenz zu Nein, Nein, Freitext) dargeboten.

Die Klassifikation der Antworten erfolgt grob in zustimmend (Ja + Tendenz zu Ja) und in ablehnend (Nein + Tendenz zu Nein). Beispielsweise weist die Eigenschaft „Färbung“ zu 79 Prozent zustimmende Aussagen auf: 7 Personen antworteten mit „Ja“, vier mit „Tendenz zu Ja“, zwei mit „Tendenz zu Nein“ und eine mit „Nein“. Dies ergibt ein Ergebnis von 11 von 14 möglichen zustimmenden Aussagen, also 79 Prozent. Der Übersicht halber sind die Antworten des Weiteren in die Stufen „$<50\%$“, „$\geq 50\%$ und $<75\%$“, „$\geq 75\%$ und $<100\%$“, sowie „100%“ unterteilt. Die quantitative Auswertung des Fragebogens ist in Tabelle 1 dargestellt.

[5] Das Brainstorming findet man hier: `http://www.brainr.de/brainstorming/show/24921-welche-eigenschaften-besitzt-schaum` (Zuletzt besucht: 19.06.2012)

[6] Der Fragebogen: `https://docs.google.com/spreadsheet/viewform?fromEmail=true&formkey=dHNQdTFqUmt1WW5hWU12Z0c5cjRSUHc6MQ` (Zuletzt besucht: 19.06.2012)

Tabelle 1: Auswertung des Fragebogens

Eigenschaft	**Ja**	**Tendenz zu Ja**	**Tendenz zu Nein**	**Nein**
Stufe 1: Zustimmendes Ergebnis von $<$50%				
Antibläschen	0	0	1	13
2-Dimensional	2	0	5	7
Einfrieren	1	2	4	6
Elemente können eckig sein	1	2	5	6
Erster Gedanke: diverse Schaumarten	3	0	3	8
Spiegelung	5	1	5	2
Stufe 2: Zustimmendes Ergebnis von $\geq$50% und $<$75%				
Verformbarkeit einzelner Elemente	4	4	6	0
Reversibel	3	6	3	1
Tragfähigkeit leichter Dinge	5	4	3	2
Erzeugung von Geräuschen	4	5	3	2
Stufe 3: Zustimmendes Ergebnis von $\geq$75% und $<$100%				
Gemeinsame Schnittpunkte von Blasen	6	5	3	0
An Form anpassbar	3	8	2	1
Zusammensetzen von Blasen	7	4	2	1
Färbung	7	4	2	1
Schaum kann Dinge einschließen	7	4	2	1
Selbstauflösend	4	7	1	2
Teilbar	5	7	1	1
Variable Größe einzelner Elemente	10	2	1	1
Variable Größe von Schaum	9	3	0	2
Verformbarkeit von Schaum	9	4	0	1
Irreversibel	7	6	1	0
Erster Gedanke: Badeschaum	10	3	0	1
3-Dimensional	12	1	0	1
Stufe 4: Zustimmendes Ergebnis von 100%				
Verbund	7	7	0	0
Indirekt veränderbar	6	8	0	0
Direkt veränderbar	11	3	0	0
Veränderbar durch Werkzeuge	10	4	0	0
Transparenz	8	6	0	0

Ob und in welcher Form die genannten Eigenschaften in das Konzept einfließen, wird zu einem späteren Zeitpunkt festgelegt. Hierzu sind weitere Untersuchungen und Festlegungen notwendig. Es ist jedoch bereits erkennbar, dass Schaum eine eigenschaftsreiche Substanz darstellt.

3.4 Abgrenzung Schaum von Seifenblasen

Da bereits in den genannten Studien [BKK+11, BHG11] Schaum-ähnliche Substanzen, die Seifenblase, beziehungsweise die Zellstruktur und Flüssigkeits-Metapher, verwendet werden, kommt die Frage auf, worin sich Schaum und Seifenblasen unterscheiden. Im Rahmen der quantitativen Analyse integrierte ich aufgrund dessen die qualitative Frage „Wohin besteht der Vorteil Schaum anstelle von Seifenblasen als Grundlage für ein Interface zu verwenden?“. Neben sich enthaltenden Stimmen, kristallisiert sich die allgemeine Meinung heraus, Seifenblasen seien losgelöst voneinander (zwei Personen), wohingegen Schaum einen Verbund darstellt (vier Personen). Seifenblasen seien ein Teil von Schaum (zwei Personen). Sie würde man jedoch eher mit der Assoziation des Platzens (eine Person) in Verbindung bringen, ganz im Gegensatz zu Schaum (eine Person). In Bezug auf die explorative Visualisierung stellt jede Blase im Schaum einen Begriff dar. Relationen sind durch den Verbund von Begriffen gekennzeichnet. Zwei angrenzende Blasen bilden ein SPO-Tripel.

4 Fazit und Ausblick

Schaum bildet als vertrautes Medium eine eigenschaftsreiche Grundlage für die Visualisierung des Semantischen Web. Folgend sollen im Rahmen meiner Masterarbeit zunächst Personas und Use Cases generiert werden, um exemplarisch anhand von Anwendungsszenarien eine konkrete Aussage darüber treffen zu können, um welche Informationen es sich bei der explorativen Visualisierung im Kontext des Semantischen Web handeln wird. Es muss überprüft werden, welche Anforderungen Endnutzer an das Semantische Web stellen und welche Eigenschaft-abhängige Assoziationen auf Basis dieser Anforderungen möglich sind. Erst dann entscheidet sich, wie und ob die genannten Eigenschaften in das Konzept einfließen. Daraus entstehende Low-Fidelity-Prototypen sind zu evaluieren. Die resultierenden Erkenntnisse fließen in verbesserter Form in die abschließenden Designvarianten ein. Die zentralen Fragestellungen sind, welche Vor- und Nachteile sich aus dieser Visualisierungsmöglichkeit auf Seiten des Endnutzers ergeben.

Literatur

[AQTF08] Richard Allen, Kai Qian, Lixin Tao und Xiang Fu. *Web Development with JavaScript and AJAX Illuminated*. Jones & Bartlett Publishers, 2008.

[Bha08] Nadeem Bhatti. Web Based Semantic Visualization to Explore Knowledge Spaces – An Approach for Learning by Exploring. In Joseph Luca und Edgar R. Weippl, Hrsg., *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2008*, Seiten 312–322, Vienna, Austria, Juni 2008. AACE.

[BHG11] M. Brade, J. Heseler und R. Groh. An Interface for Visual Information-Gathering during Web Browsing Sessions: BrainDump-A Versatile Visual Workspace for Memorizing

and Organizing Information. In *ACHI 2011, The Fourth International Conference on Advances in Computer-Human Interactions*, Seiten 112–119, 2011.

[BKK+11] M. Brade, M. Keck, D. Kammer, A. Salmen und R. Groh. Nutzung inhärenter Interaktionsangebote von Substanzen des Alltags. *Mensch & Computer 2011*, Seiten 37–41, 2011.

[ES06] Marc Ehrig und Rudi Studer. Wissensvernetzung durch Ontologien. In Tassilo Pellegrini und Andreas Blumauer, Hrsg., *Semantic Web*, X.media.press, Seiten 469–484. Springer Berlin Heidelberg, 2006.

[GC06] Vladimir Geroimenko und Chaomei Chen. *Visualizing the Semantic Web: XML-based Internet and Information Visualization*. Springer London, 2006.

[Hen10] Wolfgang Henseler. Von GUI zu NUI. Die nächste Generation des Interfacedesigns., 2010. `http://createordie.de/cod/artikel/Von-GUI-zu-NUI-2818.html` (Zuletzt besucht: 26.04.2012).

[Iva12] Den Ivanov. Realtime Blobs, 2012. Den Ivanov blog. `http://www.cleoag.ru/labs/flash4/009/blobs.html` (Zuletzt besucht: 20.06.2012).

[JOT07] R. Burke Johnson, Anthony J. Onwuegbuzie und Lisa A. Turner. Toward a Definition of Mixed Methods Research. *Journal of Mixed Methods Research*, 1(2):112–133, 2007.

[KHL+07] Akrivi Katifori, Constantin Halatsis, George Lepouras, Costas Vassilakis und Eugenia Giannopoulou. Ontology visualization methods – a survey. *ACM Comput. Surv.*, 39(4), November 2007.

[Kru06] Steve Krug. *Don't make me think! Web Usability. Das intuitive Web*. mitp-Verlag, 2006.

[LSR09] Monika Lanzenberger, Jennifer Sampson und Markus Rester. Visualization in Ontology Tools. In Leonard Barolli, Fatos Xhafa und Hui-Huang Hsu, Hrsg., *CISIS*, Seiten 705–711. IEEE Computer Society, 2009.

[Luc11] Ulrike Lucke. *Netzbasierte Systeme in Lehre und Forschung: innovative IT-Infrastrukturen für die Hochschule der Zukunft*. Logos-Verlag Berlin, 2011.

[PB06] Tassilo Pellegrini und Andreas Blumauer. *Semantic Web: Wege zur vernetzten Wissensgesellschaft*. Springer, 2006.

[RR01] Gabi Reinmann-Rothmeier. Wissen managen: Das Münchener Modell, 2001.

[ST10] Alexander Stocker und Klaus Tochtermann. *Wissenstransfer mit Wikis und Weblogs: Fallstudien zum erfolgreichen Einsatz von Web 2.0 in Unternehmen*. Gabler Verlag, 2010.

[Yau11] Nathan Yau. *Visualize This: The FlowingData Guide to Design, Visualization, and Statistics*. Wiley, 2011.

Semantische Anreicherung bei der Suche nach Kulturgütern auf multilingualen Daten

Daniel Richter

Technische Universität Chemnitz, Fakultät für Informatik
Daniel.Richter@informatik.tu-chemnitz.de

Zusammenfassung: Diese Arbeit stellt den Plan für die noch anzufertigende Diplomarbeit des Autors mit dem Titel „Information Retrieval mit dem Xtrieval Framework für *cultural heritage* Datenbestände“ vor. Dabei soll das Information Retrieval Framework Xtrieval an die Datenbestände der Europeana angepasst und den damit verbundenen speziellen Bedarf von internationalen und mehrsprachigen Datenbeständen des Kulturerbes umgesetzt werden. Insbesondere wird in dieser Arbeit auf das Vorhaben eingegangen, die Suchanfragen und -ergebnisse mit Linked Open Data semantisch anzureichern.

1 Einleitung

Suchmaschinen helfen uns seit Jahren, unseren Bedarf an Informationen zu befriedigen. Ihre Hauptaufgabe besteht darin, für eine Suchanfrage zu entscheiden, welche Dokumente aus den ihnen zur Verfügung stehenden Korpora relevant sind und welche nicht. In der Wissenschaft beschäftigt sich das Information Retrieval [Ri79] schon seit längerer Zeit mit der Fragestellung, wie der Prozess der Informationsrückgewinnung noch weiter verbessert werden könnte, um letztendlich dem Nutzer genau das zu präsentieren, was er sucht. Ein Problem dabei ist, dass die Suchmaschinen oft nicht wirklich verstehen, was der Nutzer mit der Suchanfrage meint und welches Informationsbedürfnis sich entsprechend dahinter verbirgt. Für einfaches Textretrieval, bei welchem nur betrachtet wird, ob die angefragten Worte auf die eine oder andere Art in den Dokumenten des Korpus vorhanden sind [FB92], mag das ausreichen, aber spätestens, wenn der Nutzer selbst nicht genau weiß, was er tatsächlich sucht, werden neue Ansätze nötig.

Im Bereich der Computerlinguistik existieren verschiedene Ansätze, die natürliche Sprache den Maschinen verständlich zu machen. Während sich grundlegende Schritte der Textvorverarbeitung heutzutage schon automatisch lösen lassen, bleibt das wirkliche Verstehen in Form einer semantischen und pragmatischen Analyse noch immer eine Herausforderung. [Ca07]

2 Problemformulierung

Im Rahmen der CLEF Initiative[1] behandelt die Aufgabe CHiC[2] speziell das Problem der Suche in Beschreibungsdaten von Kulturgütern. CLEF bietet eine Plattform zur Evaluation multilingualer Information Retrieval Systeme, wobei der Schwerpunkt auf europäischen Sprachen liegt. Der Korpus von CHiC wird 2012 durch die Europeana[3] bereitgestellt, welche eine zentrale Anlaufstelle für digitale Quellen aus Museen, Büchereien, Archiven und Sammlungen in ganz Europa bietet. Dieses Projekt ermöglicht damit, an einem Ort gezielt Recherchen zu einem großen Teil der europäischen Kultur anstellen zu können. Durch die vielfältigen Möglichkeiten, die Datensätze zu durchforsten, kann der Suchende leicht für ihn unbekannte Kulturgüter finden. Die zentrale Vernetzung gibt auch den kleineren Ausstellungen die Chance, ihre Exponate einer breiteren Zielgruppe zu präsentieren, genauso wie die schon bekannteren Institutionen noch mehr Menschen erreichen können.

Da es sich bei der Europeana unter anderem um eine Art Information Retrieval System handelt, liegt es nahe, dieses zu verbessern, um den Menschen ein noch ausgereifteres Erlebnis beim Durchsuchen der europäischen Kulturgüter zu bieten. Dazu bietet es sich an, im Rahmen eines Evaluation Forums neue wissenschaftliche Ideen und Systeme zur Lösung spezieller Teilaufgaben zu testen. Bei CHiC besteht die Aufgabenstellung daher in diesem Jahr aus drei Unteraufgaben:

- Um die prinzipielle Leistung des Information Retrieval Systems beurteilen zu können, sieht die erste Problemstellung die Standardaufgabe vor. Dabei müssen für eine bestimmte Anzahl von vorgegebenen Suchanfragen vom System relevante Dokumente gefunden werden.
- Die zweite Aufgabe bildet ein schon spezielleres Nutzerbedürfnis ab. Wenn der Nutzer nicht genau weiß, was er sucht und was es zu einem Thema alles gibt, ist es hilfreich, wenn die Suchmaschine einen Überblick über vorhandene Objekte zu einem relativ allgemeinen Thema bieten kann. Das Information Retrieval System soll daher eine möglichst große Vielfalt an Dokumenten zu verschiedenen Anfragen liefern.
- Die dritte Aufgabe soll ebenfalls helfen, Nutzern, welche nicht genau wissen, wonach sie tatsächlich suchen, eine Hilfestellung zu geben. Die Suchanfragen sollen dabei semantisch angereichert werden, um das eigentliche Informationsbedürfnis, welches sich hinter einer Suchanfrage verbirgt, besser verstehen zu können und damit den Nutzer schneller auf die Spur zu dem zu bringen, was er wirklich sucht. Dabei soll das System ein oder mehrere Konzepte liefern, wie die ursprüngliche Suchanfrage erweitert oder verändert werden könnte, um dieses Ziel zu erreichen.

1 Conference and Labs of the Evaluation Forum (http://www.clef-initiative.eu/)
2 Cultural Heritage in CLEF (http://www.promise-noe.eu/chic-2012)
3 http://www.europeana.eu/

3 Plan zur semantischen Anreicherung von Suchanfragen

Im Folgenden wird grob skizziert, wie die dritte Teilaufgabe angegangen werden soll. Abbildung 1 liefert dabei eine schematische Darstellung des Prozesses.

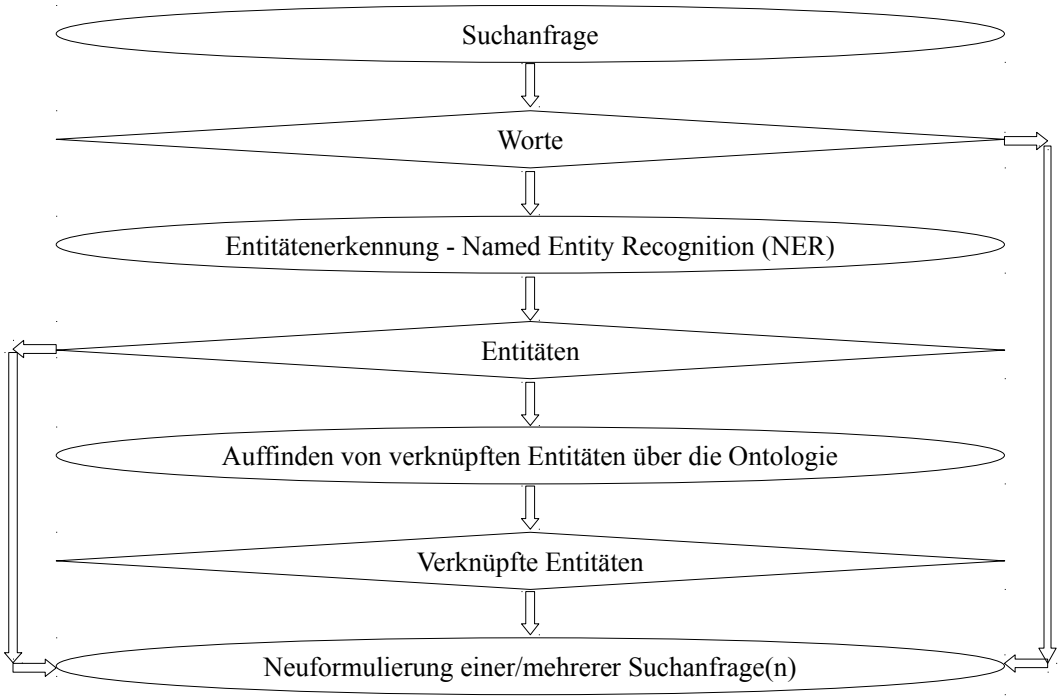


Abbildung 1: Schematische Darstellung der semantischen Anreicherung von Suchanfragen

Zunächst wird die Suchanfrage in Worte zerlegt. Mit diesen wird anschließend eine Entitätenerkennung durchgeführt. Entitäten sind benannte Dinge oder Konzepte. Um möglichst spezielle Entitäten zu erkennen, werden anfangs noch mehrere nebeneinander stehende Worte zusammengefasst. Praktisch bedeutet das, dass der Prozess mit der gesamten Anzahl an Worten, welche in der Suchanfrage vorhanden sind, gestartet wird und dann die Wortanzahl immer weiter reduziert wird und auf jeder Verarbeitungsstufe alle Kombinationen von nebeneinander stehenden Worten auf Entitäten überprüft werden, bis am Ende nur noch auf Einzelwortebene geprüft wird. Falls zum Schluss einzelne Worte mehreren Entitäten zugeordnet sind, werden dabei die entfernt, welche eine geringere Wortanzahl haben, da es sich dabei um allgemeinere Entitäten handelt, welche folglich einen geringeren Informationsgehalt besitzen. Falls keine Entitäten gefunden werden, müssen alle Worte nochmals eine weniger scharfe Suche nach Entitäten durchlaufen, wobei dann nur auf Teilübereinstimmung mit bekannten Entitäten überprüft wird. Falls dann immer noch keine Entitäten erkannt wurden, kann die semantische Anreicherung nicht durchgeführt werden.

Andernfalls wird für jede Entität nach verknüpften Entitäten gesucht. Dabei hilfreich ist eine Ontologie [Gr09], da diese das Weltwissen[4] repräsentiert und angibt, welche Eigenschaften eine Entität besitzt und welchen Klassen[5] sie zugeordnet werden kann. Außerdem werden über die Ontologie auch die Relationen zwischen den verschiedenen Klassen dargestellt, so dass von sehr speziellen Klassen auf allgemeine Klassen geschlossen werden kann. Dadurch ist es möglich, andere Entitäten zu finden, welche den gleichen Klassen entsprechen oder wenigstens einer gewissen Teilmenge.

Mit Hilfe der Worte aus der initialen Suchanfrage, den darin erkannten Entitäten und den damit verknüpften Entitäten kann anschließend eine oder mehrere neue Suchanfragen formuliert werden. Neue Entitäten auf gleicher Ebene, d. h. der gleichen speziellen Klassen, können dem Nutzer als weitere Vorschläge zum gleichen Thema zur Suche angeboten werden. Entitäten von Oberklassen führen den Nutzer von seiner eventuell sehr speziellen Anfrage zu einem allgemeineren Überblick über das Thema. Dabei muss allerdings darauf geachtet werden, dass keine Vertreter zu allgemeiner Klassen angeboten werden, da die Anzahl an gefundenen Dokumenten mit der neuen Suchanfrage dann schnell unüberschaubar und weniger nützlich sein kann.

Um das System auszubalancieren, bietet es sich an, mit der ursprünglichen Suchanfrage eine tatsächliche Suche auf der Datenbasis auszuführen und anschließend auch mit den neu erstellten Suchanfragen wirklich zu suchen. Damit kann man die Anzahl der gefundenen Ergebnisse vergleichen und daraus ableiten, welche neuen Suchanfragen potentiell besser geeignet sind und dadurch eine Reihenfolge erstellen, in welcher sie dem Nutzer präsentiert werden sollten. Mit Hilfe von Konzepten des maschinellen Lernens kann dieser Algorithmus noch dahingehend angepasst werden, zu entscheiden, welche Anzahl an Dokumenten gut ist oder wie das Verhältnis von Anzahl Dokumente mit der ursprünglichen Suchanfrage zu Anzahl Dokumente mit der neuen Suchanfrage idealerweise sein sollte.

4 Praktische Umsetzung

Für die Suche wird das Information Retrieval Framework Xtrieval [KWE08] zum Einsatz kommen. Dieses in Java[6] programmierte Framework bietet vielfältige Möglichkeiten ein Information Retrieval System aufzubauen und damit zu suchen. Die Hauptkomponenten ermöglichen dabei den Indexaufbau, die Suche und die Evaluation. Außerdem gibt es eine grafische Benutzerschnittstelle, welche aber im vorliegenden Anwendungsfall nicht gebraucht wird, da das Framework durch den Javacode der Anwendung verwendet wird. Sowohl für die Vorverarbeitung der zu indexierenden Dokumente als auch der Suchanfragen stehen mehrere Möglichkeiten zur Verfügung und

4 Gemeint ist hierbei nicht das gesamte Wissen der Welt, sondern eine Sammlung von Wissen über die Welt im Kontext, wie sie bei der Forschung zur Künstlichen Intelligenz zum Einsatz kommt (vgl. [Se91]).

5 Klassen bilden in diesem Zusammenhang Mengen, Begriffe, Typen oder Ansammlungen, deren Entitäten bestimmte gleiche Eigenschaften aufweisen.

6 http://www.java.com/

es können auch neue implementiert werden. Für den Indexaufbau kann aus verschiedenen Suchmaschinen ausgewählt werden. Dabei gibt es mehrere Suchmodelle, welche dann bei der Suche zum Einsatz kommen können. Ebenfalls implementiert sind mehrere Metriken zur Evaluation der Ergebnisse, sowie diverse Wege ein gewisses Feedback der Relevanz automatisch in den Retrievalprozess einfließen zu lassen.

Bei der Entitätenerkennung und den Daten zur semantischen Anreicherung wird primär DBpedia[7] verwendet werden. Dabei handelt es sich um eine Linked Open Data Resource, welche auf den Daten von Wikipedia[8] basiert und diese automatisch extrahiert. Linked Data ist ein Konzept, welches Daten aus verschiedenen Quellen mit verschiedenem Hintergrund versucht in eine Beziehung zu bringen und durch typisierte Links miteinander zu verbinden. [BHB09] Diese Daten müssen nicht zwingend frei verfügbar sein, aber Linked Open Data erweitert das Konzept genau um diese Komponente. Neben DBpedia existieren daher noch eine große Vielfalt an Datenbanken, welche mit dieser verknüpft sind und Informationen zu vielen Themen liefern.[9]

DBpedia stellt verschiedene Möglichkeiten bereit, die vorhandenen Daten abzufragen. Für traditionelle Suchanfragen nach Entitäten bietet sich DBpedia Lookup[10] an. Damit kann die Entitätenerkennung realisiert werden. Für die Datenabfrage zur semantischen Anreicherung gibt es einen SPARQL-Endpoint. [HBF09] Für die Einbindung der Dienste in das Javaprogramm wird auf den Beispielprogrammen von Mark Watsons Openbook „Practical Semantic Web and Linked Data Applications“ [Wa11] aufgebaut.

Quellen

[BHB09] Bizer, Christian; Heath, Tom; Berners-Lee, Tim: Linked Data - The Story So Far; In: IJSWIS, Vol. 5, Issue 3, Seiten 1-22, 2009. [http://tomheath.com/papers/bizer-heath-berners-lee-ijswis-linked-data.pdf]

[Ca07] Callan, Jamie et al.: Meeting of the MINDS: An Information Retrieval Research Agenda. In: SIGIR Forum, Vol. 41, No. 2. (December 2007), Seiten 25-34. [http://www.itl.nist.gov/iaui/894.02/MINDS/FINAL/IR.web.pdf]

[FB92] Frakes, William B.; Baeza-Yates, Ricardo A. (Hrsg.): Information Retrieval: Data Structures & Algorithms. Prentice-Hall, 1992.

[Gr09] Gruber, Tom: Ontology. In: Encyclopedia of Database Systems, Ling Liu, M. Tamer Özsu (Hrsg.), Springer-Verlag, 2009. [http://tomgruber.org/writing/ontology-definition-2007.htm]

[HBF09] Hartig, Olaf; Bizer, Christian; Freytag, Johann-Christoph: Executing SPARQL Queries over the Web of Linked Data; In Proceedings of the 8th International Semantic Web Conference (ISWC'09), Washington, DC, USA, 2009. [http://www2.informatik.hu-berlin.de/%7Ehartig/files/HartigEtAl_QueryTheWeb_ISWC09_Preprint.pdf]

7 http://dbpedia.org/
8 http://wikipedia.org/
9 http://richard.cyganiak.de/2007/10/lod/
10 http://dbpedia.org/lookup

[KWE08] Kürsten, Jens; Wilhelm, Thomas; Eibl, Maximilian (2008): Extensible Retrieval and Evaluation Framework: Xtrieval. In: Baumeister, Joachim; Atzmüller, Martin (Hrsg.) LWA 2008, 6.-8. Oktober 2008, Würzburg. - Technical Report Nr. 448, Department of Computer Science, Universität Würzburg, 4 Seiten. [http://archiv.tu-chemnitz.de/pub/2009/0012]

[Ri79] van Rijsbergen, C. J.: Information Retrieval. Butterworth, 1979. [http://www.dcs.gla.ac.uk/Keith/Preface.html]

[Se91] Seel, Norbert M.: Weltwissen und mentale Modelle. Göttingen: Verlag für Psychologie Hogrefe, 1991.

[Wa11] Watson, Mark: Practical Semantic Web and Linked Data Applications, Java, Scala, Clojure, and JRuby Edition;. Lulu Com, 2011. [http://www.markwatson.com/opencontent/book_java.pdf]

Alle im Internet verlinkten Quellen und Webseiten wurden zuletzt am 2012-06-03 abgerufen.

Design von Objekterkennungssystemen basierend auf dem visuellen System des Menschen

Michael Teichmann

michael.teichmann@informatik.tu-chemnitz.de

Abstract: Ein Weg um Objekterkennungssysteme noch leistungsfähiger zu gestalten, ist die Orientierung an der Natur. In diesem Artikel wird ein am menschlichen visuellen System orientiertes Modelldesign für die Erkennung beliebiger Objekte vorgestellt. Die Objekterkennungsleistung des Modells wird sukzessive über mehrere Hierarchiestufen aufgebaut. Mit zunehmender Hierarchiestufe werden Merkmale zunehmender Komplexität transformations-invarianter repräsentiert. Das Modell erlaubt die Rückprojektion von höheren auf niedrigere Verarbeitungsstufen, was eine dynamische Rauschminderung während der Verarbeitung ermöglicht. Darüber hinaus ist die Verarbeitungsleistung der einzelnen Neuronen im Modell nicht fest vorgegeben, sondern wird mittels eines biologisch plausiblen neuronalen Lernalgorithmus erreicht.

Einleitung

Trotz allem technologischen Fortschritts bei Objekterkennungssystemen unterliegen diese immer noch starken Einschränkungen in ihrer Anwendbarkeit. Häufig sind sie nur für spezielle Aufgabenstellungen geeignet, wie zum Beispiel die Gesichtserkennung oder die Erkennung von Verkehrszeichen. Selten jedoch sind die Systeme in der Lage, beliebige Objekte in beliebiger Zahl zu identifizieren, und dies unabhängig von der jeweiligen Betrachterposition. Dem Menschen fallen hingegen diese verschiedenartigsten Aufgaben äußerst leicht. Somit ist es folgerichtig das menschliche visuelle System und seine Mechanismen und Verarbeitungsprozesse zu untersuchen, um geeignete Mechanismen zu identifizieren und diese für das maschinelle Verstehen von Bildinhalten nutzbar zu machen. Die wesentlichsten Fragestellungen hierbei sind: Wie werden Objekte und ihre Merkmale im Gehirn verarbeitet bzw. repräsentiert? und Wie hat das Gehirn diese Art der Verarbeitung bzw. Repräsentation hervorgebracht? Im Folgenden wird ein Überblick über die wesentlichen Strukturen des Gehirns, welche in die visuelle Wahrnehmung involviert sind, gegeben. Darüber hinaus wird betrachtet, wie die einzelnen Neuronen im visuellen Kortex Bildmerkmale verarbeiten sowie repräsentieren und wie sie diese Fähigkeiten erlernen können. Aus diesem Wissen heraus wird ein mögliches, durch den Menschen inspiriertes, Modell eines Objekterkennungssystems vorgestellt und die Daten zu bereits implementierten und evaluierten Teilen dieses Systems werden betrachtet.

Neurophysiologie des visuellen Systems

Retina und Lateral Geniculus Nucleus

Bevor die Strukturen des visuellen Kortex betrachtet werden, ist es zunächst wichtig zu wissen, welcher Verarbeitung die Lichtinformationen auf dem Weg vom Auge hin zum Kortex unterworfen sind. Ohne zu sehr auf die Details eingehen zu wollen, sind für die Verarbeitung in der Retina die Ganglienzellen von besonderer Bedeutung. Ganglienzellen erhalten von mehreren Photorezeptoren Signale und besitzen eine sogenannte Zentrum-Umfeld-Organisation, das heißt ihr rezeptives Feld ist so organisiert, dass sie durch helles Licht im Zentrum mit dunklerer Umgebung, oder umgekehrt, besonders angeregt werden. Die Axone der Ganglienzellen bilden den Sehnerv, welcher die Signale der Retina weiter zum Lateral Geniculus Nucleus (LGN) transportiert. Der LGN ist Teil des Thalamus und wird häufig als eine Art Relais für die visuellen Informationen betrachtet. Die rezeptiven Felder im LGN besitzen ebenfalls eine Zentrum-Umfeld-Organisation. An ihnen können verschiedene Effekte gemessen werden, wie zum Beispiel Kontrastanpassungen, auf diese soll hier aber nicht weiter eingegangen werden [CDM$^+$05].

Der primäre visuelle Kortex

Die so vorverarbeiteten Bildinformationen erreichen nun den primären visuellen Kortex (V1). Ein kortikales Areal besteht typischerweise aus sechs Schichten. In V1 sind die Schichten vier und drei von besonderer Bedeutung. Die Axonen des LGN enden in der Schicht vier. In dieser Schicht wurden die sogenannten einfachen Zellen, „simple cells“, gefunden. Diese Bezeichnung geht zurück auf die grundlegenden Arbeiten der Nobelpreisträger Hubel und Wiesel. Sie untersuchten das Antwortverhalten von Zellen in V1 und entdeckten dabei, dass sich die rezeptiven Felder, der von ihnen einfache Zellen genannten Neuronen, relativ einfach durch Karten von anregenden und hemmenden Regionen beschreiben ließen [HW62]. Diese Regionen waren deutlich elongiert und hatten kantenförmigen Charakter. Dieses Aussehen erklärten sie sich damit, dass einfache Zellen ihre Informationen von mehreren koaktiven LGN Zellen erhalten, so dass die Summation der LGN Felder zu einem kantenförmigen Feld führt. Jones und Palmer (1987) stellten fest, dass diese rezeptive Feldstruktur sehr gut durch Gaborfunktionen beschreibbar ist [JP87]. Dies ermöglichte es später, Modelle für die Verarbeitung im primären visuellen Kortex zu entwickeln. Darüber hinaus konnten nun die Resultate von Lernalgorithmen für das Erlernen dieser rezeptiven Felder auf einfache Weise durch einen Datenfit getestet werden. Außer den einfachen Zellen fanden Hubel und Wiesel auch noch sogenannte komplexe Zellen, „complex cells“. Wie der Name schon vermuten lässt, ließen sich diese Zellen eben nicht so einfach beschreiben. Im Wesentlichen wurde festgestellt, dass diese Zellen auf die selben Stimuli reagieren wie auch einfache Zellen, allerdings ohne dass die präsentierten Lichtbalken exakt an einer bestimmten Position sein mussten. Dies konnte dadurch erklärt werden, dass die komplexen Zellen vorwiegend in der Schicht drei von V1 gefunden wurden, in welche die Axonen von Schicht vier wachsen. Demzufolge reagierten komplexe Zellen auf die Aktivitäten verschiedener einfacher Zellen, welche ähnliche rezeptive Felder an räumlich leicht unterschiedlichen Orten aufweisen.

Die Teilung in zwei Verarbeitungspfade
Nach der Verarbeitung in V1 trennen sich die Verarbeitungswege in zwei Pfade, den dorsalen Pfad und den ventralen Pfad. Der dorsale Pfad beinhaltet eine Reihe von Arealen, denen eine wichtige Rolle in der Analyse von Bewegungen, räumlichem Bewusstsein, und der Lenkung von Körperbewegungen zugesprochen wird. Wohingegen der ventrale Pfad eine Reihe von Arealen beinhaltet, welche eine wichtige Rolle in der Formrepräsentation und der Objekterkennung spielen. Da sich dieser Artikel mit der Objekterkennung beschäftigt, wird sich im Folgenden auf den ventralen Pfad konzentriert.

Wichtige Areale des ventralen Pfades
Die für diesen Artikel wichtigen Areale des ventralen Pfades sind die Areale V2, V4 und der inferortemporale Kortex (IT). Leider ist eine so genaue Charakterisierung der Repräsentation durch diese Areale nicht mehr so leicht möglich. Schon bei V2 ist unklar, welches Aussehen die rezeptiven Felder der Neuronen haben. Was aber klar ist, ist, dass V2 Zellen auf komplexere Stimuli als V1 Zellen reagieren. Es wird angenommen, dass die Neuronen auf Stimuli aus zusammengesetzten Kanten reagieren oder auf Texturen [HV03]. Neuronen in V4 reagieren auf noch komplexere Stimuli [KT94], z.B. einzelne Ansichten von einfachen Objekten. Die Neuronen in IT, welches eigentlich aus mehreren Arealen besteht, reagieren bereits auf sehr komplexe Stimuli und sind teilweise selektiv für einzelne Klassen von Objekten [Tan96]. Was mit Sicherheit über die Repräsentation von Merkmalen in diesen Arealen gesagt werden kann, ist, dass die Komplexität mit jeder weiteren Verarbeitungsstufe zunimmt.

Vom Gehirn zum Modell

Im Folgenden Abschnitt wird ein Modell vorgestellt, welches, ausgehend vom Wissen über die Struktur des Gehirns und seinen Lernmechanismen, eine Basis für ein für beliebige Aufgaben geeignetes Objekterkennungssystem sein kann.

Struktur
Das Modell besteht aus mehreren Arealen, welche in ihrer Funktion auf den wichtigsten Arealen des ventralen Pfades basieren (siehe Abbildung 1). Die Netzwerkstruktur des Modells kann wie folgt beschrieben werden. Die Eingabeschicht des Netzwerkes ist die LGN Schicht. Sie wird gefolgt von den drei visuellen Arealen V1, V2 und V4. Jedes dieser Areale besteht aus jeweils zwei Schichten, einer sogenannten Eigenschaftsextraktionsschicht und einer Eigenschaftsbündelungsschicht. Die Eigenschaftsextraktionsschicht erhält ihre Eingaben von der jeweils letzten Schicht des vorhergehenden Areals, welches entweder im Falle von V1 die LGN Schicht oder im Falle der weiteren Areale die jeweilige Eigenschaftsbündelungsschicht des Vorgängerareals ist. Das oberste Areal stellt das IT Areal dar. Es besteht nur aus einer einzigen Schicht, welche ihre Eingaben von der Eigenschaftsbündelungsschicht von V4 erhält. Als Eingabe für das Netzwerk dient monochromes Bildmaterial, welches insofern aufbereitet wird, als dass eine Normalisierung im

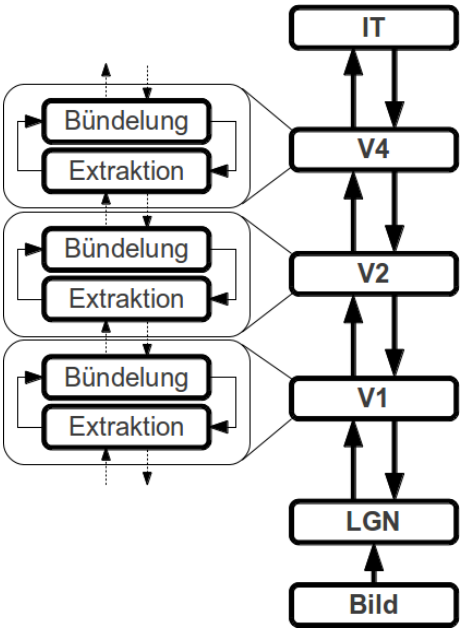


Abbildung 1: Modellskizze. Die Bildeingaben werden nach der Vorverarbeitung auf die LGN Schicht gesetzt. Von dort werden sie durch jeweils zwei Schichten der Areale V1, V2 und V4 verarbeitet. Das oberste Areal IT besteht aus einer Schicht und fungiert als eine Art Klassifikator. Die Rückwärtsverbindungen projizieren Informationen höherer Areale zurück zu niedrigeren Arealen und verbessern so die Verarbeitung dieser.

Frequenzspektrum und eine Zerlegung in Zentrum-Umfeld Antworten mit entweder exzitatorischem oder inhibitorischem Zentrum stattfindet [OF96, WH09]. Diese Art der Vorverarbeitung ist eine Standardmethode zur Generierung geeigneter LGN ähnlicher Neuronenantworten. Die Antworten werden als Aktivitäten auf die LGN Schicht des Netzwerkes gesetzt.

Rezeptive Felder und Lernen

Die V1 Neuronen der Eigenschaftsextraktionsschicht erhalten aus diesen Bildinformationen jeweils einen kleinen Ausschnitt als Eingabe, welches dem relativ kleinen rezeptivem Feld von kortikalen V1 Zellen Rechnung trägt. Dies geschieht in der Form, dass immer eine Gruppe von Neuronen denselben Bildausschnitt erhält und die jeweiligen Nachbargruppen einen um jeweils einen Bildpunkt verschobenen Bildausschnitt erhalten, so dass alle Neuronen gemeinsam das gesamte Bild verarbeiten. Um eine Dekorrelation im Netzwerk zu erreichen und damit Redundanzen zwischen der Repräsentation einzelner Neuronen zu reduzieren, sind alle Neuronen, welche teilweise gleiche Vorwärtseingaben erhalten, lateral inhibitorisch miteinander verbunden [Föl90, WH09, TWH12]. Die V1 Neuronen der Eigenschaftsbündelungsschicht formen ebenfalls Gruppen, deren Neuronen ihre Eingaben von jeweils einer Gruppe der Vorgängerschicht erhalten. Dieses Prinzip wird in den Arealen V2 und V4 fortgeschrieben, allerdings einhergehend mit einer Verdopplung der rezeptiven Feldgröße im Vergleich zum Vorgängerareal, bezogen auf den Bildraum. Das IT Areal nimmt in diesem Modell eine Sonderrolle ein. Seine Neuronen haben rezeptive Felder, die den gesamten Bildraum abdecken. Sie fungieren als eine Art Klassifikatoren, d.h. ihre Aktivitäten repräsentieren die im Bild jeweilig vorhanden Objekte. Diese Art der Struktur führt dazu, dass die Komplexität der repräsentierten Bildelemente mit jeder Verarbeitungsstufe zunimmt und gleichzeitig, durch die Bündelung ähnlicher Repräsentationen in jeder Stufe, die Robustheit, d.h. die Invarianz, gegenüber Transformationen zunimmt.

Zu jeder Vorwärtsverbindung im Netzwerk existiert ebenfalls eine Rückwärtsverbindung, welche modulierend auf das jeweilige Zielneuron wirkt [WH09]. Diese Rückwärtsverbindungen nutzen die Informationen über bereits erkannte Merkmale oder Objekte und projizieren sie zurück in die niedrigeren Verarbeitungsstufen. Dies führt zu einer dynamischen Rauschreduktion während der Verarbeitung und kann ebenso genutzt werden, um Aufmerksamkeitseffekte zu modellieren [Ham05].

Die Vorwärtsverbindungen wie auch die Rückwärtsverbindungen sind lernbar. Für die genaue Implementation eines geeigneten Lernalgorithmus sei auf Teichmann et al. (2012) bzw. Wiltschut und Hamker (2009) verwiesen [TWH12, WH09]. Hier nur soviel: der vorgeschlagene Algorithmus arbeitet unüberwacht und bildet die von kortikalen Synapsen bekannten Lernmechanismen der Langzeit-Potenzierung und Langzeit-Depression ab. Zusätzlich wird für die Eigenschaftsbündelungsschichten die zeitliche Trägheit synaptischer Prozesse ausgenutzt [TWH12]. Ausgehend von zufällig initialisierten Gewichten erlernen so alle Neuronen gleichzeitig ihr jeweiliges rezeptives Feld. Das Lernen findet kontinuierlich statt und die Präsentationsdauer eines einzelnen Bildes unterliegt keinerlei Beschränkungen. Allerdings ist für das generelle Lernergebnis entscheidend, dass die Eingabesequenzen eine ähnliche inhaltliche wie zeitliche Struktur aufweisen wie Realwelt-Szenen. Dies ist insofern wichtig, da die Neuronen Ähnlichkeiten zwischen Merkmalen aufgrund ihrer Transformationen in der Zeit erlernen. Das heißt, Objekte sollten nicht in schneller Folge als Schnappschuss präsentiert werden, sondern längere Zeit im Bild verbleiben und dabei den in der Realwelt üblichen Transformationen unterliegen.

Eine Besonderheit bezüglich des Lernens im Modellareal IT ist, dass hier die Neuronenaktivität (teil-)überwacht auf Basis der jeweils im Bild enthaltenen Objekte modifiziert werden kann. Dies forciert die Neuronen in dieser Schicht dazu, selektiv für einzelne Objekte zu werden, ohne dabei die Fähigkeit der Population einzuschränken, auf alle in einer Szene vorhanden Objekte zu reagieren. Die Teilüberwachtheit (semi-supervised) ist insofern von Bedeutung, da nicht jedes Objekt in einer Szene vor dem Training bekannt und auch nicht jede Szene annotiert sein muss. Die Anzahl der Objekte, welche durch die IT Schicht repräsentiert werden können, ist nicht durch die Größe dieser beschränkt, da die Anzahl der Neuronen in IT frei skalierbar ist. Die Zahl der möglichen trennbaren Muster ist nur von der Qualität der Verarbeitung in den Vorgängerarealen abhängig. Denn diese sukzessive Zerlegung des Bildraumes in einen neuronalen Merkmalsraum ist entscheidend für die lineare Trennbarkeit der einzelnen Objekte [DC07].

Diskussion

Das hier gewählte Netzwerkdesign ist nicht von Grund auf neu. Es basiert auf dem seit vielen Jahren vorhanden Konzept der hierarchischen Verarbeitung visueller Inhalte [Fuk80, WR97, RP99, Ser06]. Bereits Fukushima mit dem „Neocognitron“ wie auch Riesenhuber und Poggio mit „HMAX“ nutzten sukzessive, aufeinander folgende Eigenschaftsextrationsschichten und Eigenschaftsbündelungsschichten. Im Gegensatz zum hier vorgestellten Ansatz sind in ihren Ansätzen die Eigenschaftsbündelungsschichten durch einfache Maximumsfunktionen realisiert und nicht aufgrund der im Bildmaterial vorkommen-

den Ähnlichkeiten erlernt. Dies funktioniert in beiden Ansätzen nur aufgrund einer fest vorgegebenen Struktur der jeweiligen Vorgängerschichten. Während bei HMAX die Eigenschaften, welche die jeweiligen Neuronen repräsentieren, fest vorprogrammiert sind, erlaubt das Neocognitron zwar das Erlernen der Eigenschaften, kopiert diese aber auf andere vordefinierte Neuronen mit leicht verschobenen rezeptiven Feldern, so dass die Ähnlichkeitsrelation per Design fest steht. Ein ähnliches Prinzip wird auch bei dem neueren Modell von Serre genutzt. In der Lernphase werden einzelne Bildausschnitte verwendet, um sie in einem sogenannten „imprint“ Verfahren als rezeptives Feld der jeweiligen Neuronen zu nutzen, gleichzeitig werden die Ausschnitte noch in ihrer Größe und Lage variiert und bilden so das rezeptive Feld weiterer Neuronen. Aufgrund dieses Mechanismus ist auch hier die Ähnlichkeitsrelation bekannt. Im Gegensatz dazu verwenden Wallis und Rolls gar keine spezielle Schicht, welche Ähnlichkeitsrelationen extrahiert, sondern setzen darauf, dass dies implizit in den Verarbeitungsschichten erfolgt. Trotzdem hat ihr Modelldesign starken Bezug zu unserem Ansatz, da sie eine vergleichbare Struktur von sich sukzessive vergrößernden rezeptiven Felder nutzen. Hierzu verwenden sie eine Gauß definierte Verknüpfungsstruktur, die kreisförmige rezeptive Felder erzeugt, welche sich bei benachbarten Zellen überlappen. Auch orientieren sie sich beim Anstieg der rezeptiven Feldgröße am biologischen Vorbild.

Im Gegensatz zu den oben genannten Ansätzen wird bei dem hier vorgestellten Modell besonderer Wert darauf gelegt, dass die Verarbeitung, d.h. die erlernten rezeptiven Felder, sich möglichst nah am Vorbild der Natur befinden. Dies konnte auch bereits in mehreren Teilstudien gezeigt werden. Das Modell von Wiltschut und Hamker (2009) zum Erlernen von rezeptiven Feldern, ähnlich derer von einfachen Zellen, hat hervorragende Ergebnisse im direkten Vergleich zu gefundenen Feldern im Gehirn von Makaken Affen gezeigt. In Teichmann et al. (2012) wurde anhand von komplexen Zellen gezeigt, wie die damit verbundene invariante Repräsentation natürlicher Stimuli erlernt werden kann. Ebenfalls, aber noch unveröffentlicht, wurden erste erfolgreiche Versuche unternommen, die Erlernbarkeit und Vergleichbarkeit von rezeptiven Feldern des V2 zu untersuchen.

Literatur

[CDM+05] M. Carandini, J. B. Demb, V. Mante, D. J. Tolhurst, Y. Dan, B. A. Olshausen, J. L. Gallant und N. C. Rust. Do we know what the early visual system does? *The Journal of neuroscience*, 25(46):10577–97, 2005.

[DC07] J. J. DiCarlo und D. D. Cox. Untangling invariant object recognition. *Trends in cognitive sciences*, 11(8):333–41, August 2007.

[Föl90] P. Földiák. Forming sparse representations by local anti-Hebbian learning. *Biological cybernetics*, 237(5349):55–56, Mai 1990.

[Fuk80] K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 202, 1980.

[Ham05] F. H. Hamker. The reentry hypothesis: the putative interaction of the frontal eye field, ventrolateral prefrontal cortex, and areas V4, IT for attention and eye movement. *Cerebral cortex (New York, N.Y. : 1991)*, 15(4):431–47, April 2005.

[HV03] J. Hegdé und D. C. Van Essen. Strategies of shape representation in macaque visual area V2. *Visual neuroscience*, 20(3):313–28, 2003.

[HW62] D. H. Hubel und T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106–154, Januar 1962.

[JP87] J. P. Jones und L. A. Palmer. An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex. *Journal of neurophysiology*, 58(6):1233–58, Dezember 1987.

[KT94] E. Kobatake und K. Tanaka. Neuronal selectivities to complex object features in the ventral visual pathway of the macaque cerebral cortex. *Journal of neurophysiology*, 71(3):856–67, Marz 1994.

[OF96] B. A. Olshausen und D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–9, 1996.

[RP99] M. Riesenhuber und T. Poggio. Hierarchical models of object recognition in cortex. *Nature neuroscience*, 2(11):1019–25, November 1999.

[Ser06] T. Serre. *Learning a Dictionary of Shape-Components in Visual Cortex: Comparison with Neurons, Humans and Machines*. Doctor thesis, Massachusetts Institute of Technology, 2006.

[Tan96] K. Tanaka. Inferotemporal cortex and object vision. *Annual review of neuroscience*, 19:109–39, Januar 1996.

[TWH12] M. Teichmann, J. Wiltschut und F. H. Hamker. Learning invariance from natural images inspired by observations in the primary visual cortex. *Neural computation*, 24(5):1271–96, Mai 2012.

[WH09] J. Wiltschut und F. H. Hamker. Efficient coding correlates with spatial frequency tuning in a model of V1 receptive field organization. *Visual neuroscience*, 26(1):21–34, 2009.

[WR97] G. Wallis und E. T. Rolls. Invariant face and object recognition in the visual system. *Progress in neurobiology*, 51(2):167–94, Februar 1997.

Performance in der Microsoft Access Datenbank

Babak Bastan

babak.bastan@s2010.tu-chemnitz.de

Aufteilung der Datenbank, Ja oder Nein?

Eine Access Datenbank ist aufgeteilt wenn wir sie in Front-end und Back-end verteilen. Der Back- end besteht aus Daten (Tabellen) und Front end besteht aus Formulare, Codes, Berichte und etc.

Der Front-end kann auch die lokale Tabelle, die nur die lokalen Einstellungen für den Front-end haben, enthalten.

Die verknüpfte Tabelle soll nicht nur immer in der Mehrbenutzerumgebung verwendet werden, sondern auch in dem Fall, in dem wir nur einen Benutzer haben. Das bedeutet, dass die Aufteilung der Datenbank ein Software Entwurf ist.

Warum soll die aufgeteilte Datenbank in der Mehrbenutzerumgebung verwendet werden?

Access Datenbank ist vorbreitet um mehrere Benutzer gleichzeitig durch die Aufteilung der Datenbank in Front-end und Back-end zu dienen. Jeder Benutzer hat eine Kopie von Applikation, die nicht geteilt ist (Front-end) .Die Daten (Back-end) sind geteilt durch die Verknüpfung des Frontends zu der Tabelle, die auf dem Back-end liegen. Eigentlich wird nur das Front-end direkt durch Access geöffnet. Z.B ist eine Webseite ein Backend und unser Browser ist ein Frontend.[1]

Wann sollen wir unsere Datenbank aufteilen?

Gründe zur Aufteilung einer Datenbank:[2]

- Die Aufteilung der Tabellen ermöglicht, jeder Front-end und Back-end 2 GIG zu sein (zusammen 4 GIG), deswegen können wir die Kapazitätbeschränkung einer einzelnen Datenbank überwinden.

- Anderer Punkt ist Sicherheit wenn eine Datenbank aufgeteilt wird, kann der Benutzer nicht zufällig oder absichtlich die Tabellen löschen, weil er keinen direkten Zugriff zu den Tabellen hat.

- Wenn die Accessdatenbank voraussichtlich in der Zukunft so groß würde, wäre es besser die Accessdatenbank in Front-end und Back-end aufgeteilt zu haben, da sehr einfach der Back-end auf andere Datenbank Z.B MSSQL Server übertragen werden kann und die Front-ends sehr einfach mit dem neuen Back-end verknüpft werden können.

- Die Performanz wird erhöht, weil der Benutzer eine Kopie von der Applikation besitzt und aus diesem Grund muss die Applikation nicht auf dem Server ausgeführt werden. Die Formulare werden von dem lokalen Speicherplatz geladen, dass ganz deutlich schneller ist, als das Laden vom Server durch LAN.

- Der Netzwerkstau wird verringert, weil nur die Anfragen durch das Netz übertragen warden (nicht die Formulare oder ...)

- Ohne aufgeteilte Datenbank wird die Wahrscheinlichkeit der Datenbankskorruption erhöht, weil meistens die Korruptionen in der Accessdatenbank sich in der Formularen, Reports oder Modulen ereignet, deshalb wenn ein Front-end Kaput geht, beeinflusst nur den Benutzer, der diese Kopie vom Front-end verwendet und nicht alle Benutzer.

Verwenden der 8.3-Bennenungskonventionen für Dateien

Wenn der Name der Datenbankdatei länger als acht Zeichen ist oder die Datenbank liegt innerhalb eines Ordnens, dessen Namen aus mehr als acht Zeichen besteht oder die Dateinnamenerweiterung umfasst mehr als drei Zeichen, ruft Access die **GetShortPathNameW** funktion für jede Abfrage auf. Das bedeutet, wenn wir dieser Regel nicht folgen, müssen zusätzliche Schritte (Ausführung der **GetShortPathNameW** Funktion) durchgeführt werden und das kostet Zeit.[3]

Komprimieren und Reparieren

Um eine hohe Performance in MS- Access zu haben soll die Datenbank regelmäßig komprimiert werden.

Die Daten werden in einer Datenbank unzusammenhängend gespeichert.Aus diesem Grund verringert sich die Performance der Datenbank, weil das unsere Datenbank irgendwie groß wird .Komprimieren fragmentiert die Datenbank und verwendet den Speicherplatz effizient. [4]

In Frontend/Backend Modell soll nur das Frontend komprimiert werden.

Unterdatenblatt Eigenschaft

Microsoft Access hat eine besondere Eigenschaft, dass das Sehen der „One to Many“ Beziehungen zwischen die Tabellen ermöglicht.

Im Default der Name vom Unterdatenblatt auf [Auto] stellten und wenn eine Tabelle geöffnet ist,sucht das Datenblatt automatisch das zusammenhängende Unterdatenblatt, um die Daten anzuzeigen.Brauchen wir diese Möglichkeit nicht, können wir den Namen von Unterdatenblatt auf [None] einsetzen und dann erhalten wir mehr Performance besonders für die Access-Projekte, die im Netz ausgeführt und aufgeteilt sind.[5]

Recordset [6]

Recordset ist eigentlich ein in Memory-container für Daten. Es gibt drei verschiedenen Typen in MS Access: dynasets, snapshot und Table:

Dynasets (Dynamic set)

In dieser Methode werden zuerst alle Primärschlüssel aufgerufen. Dannwird fürjeden Datensatz eine zusätzliche Anfrage gestellt um den Rest der Spalte, die zu dem Primärschlüssel gehört, abzurufen .Das bedeutet wenn eine Anfrage 100 Datensätze zurückgeben sollte, muss es 101 Anfrage an den Server gestellt werden. Dynasets sind erreichbar wenn erste 20 Datensätze zurückgebracht werden und das ist geeignet wenn die Delete, Update und Add Befehle durchgeführt werden sollen.Dynaset ist für jeden Benutzer lokal und unsere locale Hinzufügungen und Löschungen werden in dem Dynaset wiedergegeben .Diese Änderungen werden nicht den anderen Benutzern angezeigt.

Snapshot

Snapshot ist eine statistische, read-only Repräsentation von Daten, Wenn die Daten durch Snapshot zurückgebracht werden, werden alle von den Daten auf einmal übertragen und das Ergebnis wird auf dem Arbeitsplatz des Benutzersrechners gespeichert .In diesem Fall, wenn wir eine große Menge von Daten haben, verringert sich die Performance, weil hier alle gesamten Datensätze heruntergeladen werden sollen und dann kommen sie zum Vorschein .Wenn wir kleine Mengen von Daten haben oder keine Änderungen ausgeführt werden, erhöht das Snapshot die Performance z.B. Für Comboboxs, Listboxes, in der es keine Delete, Update,Add Befehle gibt, kann der Snapshot genutzt werden.

Table:

Tableobjekt ist eine logische Repräsentation der Physikalischen Tabelle. Diese Repräsentation ist für alle Benutzer sichtbar (in dem Fall, in dem es mehrere Benutzer gibt) und sie sind in der Lage, alle Datensätze in der original Tabelle durch *Tableobject* variable zu sehen.

Diese Methoden haben Vor- und Nachteile und die Verwendung von ihnen kommt auf dem Scenario an .Z.B haben wir in einem Formular keine große Menge von Daten aber in diesem Formular ist der Benutzer in der Lage, einen neuen Satz einzufügen. Aus diesem Grund können wir behaupten, dass Snapshot für uns nicht geeignet ist. Ein Zweifel gibt es zwischen dem *Dynaset* und dem Table, das sieht so aus, dass *Tableobject* für die Mehrbenutzerumgebung geeignet ist, aber gemäß der msdn Webseite soll der *Dynaset* in der Mehrbenutzerumgebung, in der wir verknüpfte Tabellen haben, verwendet werden.

“Only dynaset-type or snapshot-type Recordset objects can be created with linked tables or tables in Microsoft Access database engine-connected ODBC databases.” [7]

Einschalten des JET Executionplans[8]

In Oracle oder MS SQL Server kann man sehr einfach und grafisch den Explainplan verwenden aber hier soll MS Access gezwungen werden, aber wie?
Das hat etwas mit Registry zu tun. Folgen Sie bitte diesen Weg in der Registry: (bitte beachten Sie, dass die Registry ein gefährliches Gebiet ist, zuerst sichern Sie Ihr Registry dann setzten Sie die Änderungen ein)

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\JET\4.0\Engines

legen Sie ein Schlüssel, der Debug heisst, wie folgt:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\JET\4.0\Engines\Debug

dann fügen sie bitte in Debug eine Zeichenfolge ein, die JETSHOWPLAN heißt und stellen deren Werte mit ON ein.

Die Showplandatei wird Irgendwo auf Ihrem Speicherplatz gespeichert .Sie können sie einfach suchen, um deren Platz klar zu stellen.Im nächsten Schritt soll eine Abfrage ausgeführt werden.Öffnen Sie die ShowPlandatei.OUT und Sie können das Ergebnis sehen.[9]

Literaturverzeichnis:

[1] http://www.fmsinc.com/microsoftaccess/databasesplitter/index.html [accessed 24.01.2012]Wendell Bell & Associates Inc.Why Split a Database?

[2] http://www.hitechcoach.com/index.php?option=com_content&view=article&id=35:split-your-access-database-into-application-anddata&catid=24:design [accessed 20.01.2012]

[3] http://support.microsoft.com/kb/889588/de [accessed 18,01,2012].Optimieren der Netzwerkleistung von Office Access und der Jet-Datenbank-Engine mit Windows 2000- und Windows XP-Clients

[4] Microsoft. Komprimieren und Reparieren einer Access-Datei

http://office.microsoft.com/de-ch/access-help/komprimieren-und-reparieren-einer-access-datei-HP005187449.aspx[access 24.02.2012]

[5] Haught,D and Chung,L. Microsoft Access Performance Tips to Speed up Your Access Databases http://www.fmsinc.com/free/newtips/access/subdatasheetname.asp [accessed 11.04.2012]

[6] Microsoft.Verwenden von Tabelle-Objekten und Dynaset-Snapshot-Objekte in VB http://support.microsoft.com/kb/109218/de [accessed 30.01.2012]

[7] Microsoft. Recordset Object (DAO) http://msdn.microsoft.com/en us/library/ff197799.aspx [accessed 13.02.2012]

[8] Fritchey,G.Execution Plan Basics http://www.simple-talk.com/sql/performance/execution-plan-basics/ [accessed 1.03.2012]

[9] TechRepublic.Use Microsoft Jet's ShowPlan to write more efficient queries http://www.techrepublic.com/article/use-microsoft-jets-showplan-to-write-more-efficient-queries/5064388 [accessed 11.04.2012]

FPGA-basierte Hardwarebeschleunigung für Echtzeitbildverarbeitung und Fusion

Stephan Blokzyl und Wolfram Hardt
Fakultät für Informatik
Technische Informatik
Technische Universität Chemnitz
D-09107 Chemnitz

{stephan.blokzyl;wolfram.hardt}@informatik.tu-chemnitz.de

1 Motivation

Für Anwendungen, in denen Strukturen und Objekte erkannt werden sollen, eignen sich insbesondere elektrooptische (EO) Sensoren (Kameras). Sie sind kostengünstig und stellen qualitativ gute, hochaufgelöste Bilddaten zur Verfügung. Diese sehr großen Datenmengen in endlicher, definierter Zeit zu verwalten und zu verarbeiten ist eine zentrale Herausforderung, vor allem in ressourcenlimitierten, eingebetteten Systemen. So umfasst bspw. die Verarbeitung von FullHD-Farbbildern ca. 156 Millionen Operationen pro Sekunde[1], wenn nur eine Berechnung pro Pixel nötig ist (vgl. [GJB03]).
General- oder Special-Purpose-Prozessoren sind bei wachsender Systemkomplexität nicht mehr geeignet, diesen Herausforderungen zu begegnen (vgl. [PMF+11][RR09]). Eine Lösung bieten rekonfigurierbare, applikationsspezifische, eingebettete Systeme, wie z.B. Field-Programmable Gate Arrays (FPGAs). Sie verbinden die Flexibilität von Prozessoren mit den Vorteilen anwendungsspezifischer, integrierter Schaltkreise. Sie erlauben eine Optimierung der Verarbeitungsarchitektur auf Basis der funktionalen und zeitlichen Anforderung der zugrundeliegenden Anwendung. Zusammen mit der unbegrenzten Vielfalt konfigurierbarer On-Chip-Schnittstellen für seriellen oder parallelen Datenaustausch, qualifizieren sich FPGAs im besonderen Maße für High-Performance-Computing-Anwendungen.
Im Rahmen der Forschungsarbeiten werden Konzepte der hardwarebeschleunigten Bildverarbeitung untersucht und entwickelt. Am Beispiel der Detektion von Objekten soll gezeigt werden, welchen Beitrag Hardwarebeschleunigung liefern kann, komplexe Verfahren wie die Bildverarbeitungskette zu beschleunigen. Der Fokus liegt auf der Anpassung der Verarbeitungskette, um eine Abbildbarkeit der Algorithmen innerhalb der durch die Hardware gegebenen Randbindungen zu ermöglichen. Ziel dabei ist die Abbildung aller Schritte der Bildverarbeitungskette: Von der Sensordatenakquise über Vorverarbeitung, Segmentierung, Fusion, Klassifikation, Interpretation/Ableitung von System- und Umweltzuständen, bis zur Verteilung der Ergebnisse.

[1] Auflösung 1920x1080 Pixel, bei einer Framerate von 30Hz

2 Stand der Forschung

Der praktische Einsatz hardwarebeschleunigter Bildverarbeitung wird von verschiedenen Forschergruppen weltweit untersucht. Durch Parallelisierung können z.B. unabhängige Pixeloperationen für große Bildbereiche synchron mit Hilfe von Logik berechnet werden.
So stellte Gribbon bereits 2003 ein FPGA-basiertes Verfahren zur Korrektur radialer Verzeichnung vor [GJB03]. Saldanha präsentierte 2010 eine hardwareimplementierte Methode zur Detektion von Kratzern bei der industriellen Bildentwicklung [SHB10]. Andere Schritte der Bildverarbeitungskette, wie z.B. die Klassifikation von Objekten und Gesichtern werden in [PB10], [MAHD11] und [CMOK09] beschrieben. Ferner wird Hardwarebeschleunigung z.B. für Bilddemosaicing, Filterung, Transformation, Kompression, Verschlüsselungsverfahren und vielen weiteren Methoden verwendet.
Es zeigt sich, dass Hardwarebeschleunigung bereits bei einzelnen Schritten der Bildverarbeitungskette erfolgreich eingesetzt wird. Die vollständige Integration der Verarbeitungskette in Hardware ist Ziel der Forschungen. Dabei liegt der Fokus auf dem Einsatz vieler paralleler, unterschiedlicher Low-Level-Algorithmen und der Fusion ihrer heterogenen Ergebnisse zu einer Gesamtdetektion. Im Ergebnis entsteht ein robuster Detektor mit deutlich verbesserter Detektionsgüte im Vergleich zu den Low-Level-Detektoren.

3 Echtzeitbildverarbeitung

Echtzeitbildverarbeitung bedeutet, dass die Gesamtverarbeitungszeit kleiner gleich der reziproken Framerate der EO Sensorik ist. Die Gesamtverarbeitungszeit ist die Summe der für die Sensordatenakquisition, -verarbeitung und Ergebnisverteilung benötigten Zeiten. Diese Betrachtung separiert die wichtigen Phasen der Echtzeitbildverarbeitung: Sensordatenakquisition, Sensordatenverarbeitung/-auswertung sowie Ergebnisverteilung. Diese drei Phasen sollen im Folgenden kurz beschrieben werden.

3.1 Sensordatenakquisition und Ergebnisverteilung

EO Sensoren verfügen über vielfältige Protokoll- und Schnittstellenstandards für die Sensordatenübergabe und -steuerung. FPGAs unterstützen diese Standards durch die Einbindung von Schnittstellenmodulen in Form von unabhängigen IP[2]-Cores. Die Entkopplung der Datenakquise von der Verarbeitung unterstreicht die hohe Flexibilität von rekonfigurierbaren FPGAs. Bei Änderung der Schnittstelle (z.B. bei Sensorwechsel) und gleichbleibender Detektionsaufgabe wird ausschließlich das Interfacemodul adaptiert.
Die für einen Anwender (Subscriber) nötigen Resultate werden auf Basis eines beliebigen Protokolls verpackt und anschließend entsprechend einer spezifizierten Transporttechnologie übermittelt. Das Protokoll und die Übertragungstechnologie werden von der Anwendung festgelegt.

[2] Intellectual Property

3.2 Sensordatenverarbeitung

Die Sensordatenverarbeitung auf einem FPGA erlaubt ein hohes Maß an Parallelisierung. Damit große Mengen EO Sensordaten in Echtzeit verarbeitet werden können, werden zwei Strategien untersucht: *Datenparallelisierung* und *Funktionsparallelisierung*.
Bei der Datenparallelisierung werden die Sensordaten in Subsets fragmentiert und an parallel angeordnete, homogene Verarbeitungseinheiten weitergereicht. Die Teilergebnisse aus den parallelen Verarbeitungseinheiten werden anschließend wieder zu einem Gesamtergebnis zusammengeführt (Synchronisation). Diese Strategie der Lastverteilung beschleunigt die Ausführung einzelner, einander unabhängiger Verarbeitungsschritte.
Die Funktionsparallelisierung prozessiert die Sensordaten zeitgleich mit unterschiedlichen Bildverarbeitungsmethoden (z.B. Kanten-, Eckendetektoren, Regionensegmentierung o.a.). Dabei werden die Bilddaten an parallele, heterogene Verarbeitungseinheiten weitergereicht, die unterschiedliche Zwischenergebnisse generieren. Diese sind im nächsten Schritt Grundlage für eine Hypothesenfusion, welche die Zwischenergebnisse zu einer Gesamtdetektion fusioniert. Alle Detektionen werden anschließend klassifiziert und mit einem Vertrauensmaß bewertet. Die Fusion kann z.B. mit einem Fuzzy-basierten Ansatz, einem Bayes-Filtermodell [LS97][MAHD11] oder einem Künstlichen Neuralen Netz [ZS03] erfolgen. Diese, vor allem bei softwarebasierten Anwendungen etablierten Verfahren, werden auf ihre Implementierbarkeit in Hardware untersucht und adaptiert. Alternative Fusionsmethoden sollen entwickelt, und im Kontext der hardwarebeschleunigten Bildverarbeitung bewertet werden. Die Fusion von Low-Level-Detektionen (z.B. Kanten und Ecken) zu einem Segment höheren Abstraktionsniveaus (z.B. Viereckssegment) kann zu einer signifikanten Steigerung der Detektionsgüte und der Robustheit gegen Störungen führen.
Die Strategien der Daten- und Funktionsparallelisierung fließen in komplexes System aus einfachen Low-Level-Bildverarbeitungsalgorithmen ein, das eine robuste Detektion von Objekten und Strukturen in Abhängigkeit einer spezifizierten Wissensbasis erlaubt.

4 Projekteinordnung und Konzeptumsetzung

Bildverarbeitungsverfahren sind immer an eine bestimmte Applikation (Detektionsziel) adaptiert, bei der nur eine gut modellierte Wissensbasis und die richtige Algorithmenwahl robuste Detektionsergebnisse erzielen. Die Modularität der Daten- und Funktionsparallelisierung erlaubt eine flexible, baukastenartige Anpassung des Bildverarbeitungssystems. Es kann für eine spezielle Anwendung vorkonfiguriert (statische Rekonfiguration) sowie im laufenden Betrieb modifiziert (Änderung des Detektionsziels) und reparametrisiert (Justierung der Detektionsparameter) werden (dynamische Rekonfiguration).
Diese Eigenschaften sind, in Verbindung mit der Energieeffizienz und dem geringen Gewicht von eingebetteten Systemen, im Kontext der unbemannten Flugaufklärung von besonderem Interesse. So soll die hardwarebasierte Bildverarbeitung auf einem UAV[3] in verschiedenen Operationsphasen die Auswertung sehr großer Mengen Bilddaten beschleunigen und verbessern.

[3] UAV steht für *Unmanned Aerial Vehicle*, dt. Unbemanntes Luftfahrzeug

Die Arbeiten finden dabei innerhalb des Cassidian Open Innovation Projektes statt, einem kooperativen Forschungsprojekt mit acht Forschungsinstituten und Hochschulen sowie dem Industriepartner EADS Cassidian. Ziel der Initiative ist die Entwicklung und Reifung von für die Luftfahrt relevanter Technologien.
Im Rahmen der hier am Lehrstuhl bearbeiteten Forschungsthemen entstehen dazu geeignete Datenverteilungs-, Kommunikations- und Speicherkonzepte, die einen wesentlichen Beitrag zur Beschleunigung der Bildverarbeitung leisten. Eine effiziente Modellierung und Aufteilung der Low-Level-Bildverarbeitungsalgorithmen und der Schritte der Bildverarbeitungskette in Hard- und Software (Hardware-Software-Codesign) stellt einen weiteren Forschungsschwerpunkt dar. Zentraler Punkt der aktuellen Arbeiten ist die Bewertung der Hardwareumsetzbarkeit und die Hardwareimplementierung von verschiedenen Low-Level-Bildverarbeitungsansätzen, sowie die Entwicklung geeigneter Verfahren zur Fusion der heterogenen Ergebnisse auf der Meta-Ebene.

Literatur

[CMOK09] Cho, Mirzaei, Oberg und Kastner. FPGA-Based Face Detection System Using Haar Classifiers. In *Proceedings of the ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, Seiten 103–112, 2009.

[GJB03] Gribbon, Johnston und Bailey. *A Real-time FPGA Implementation of a Lens Distortion Correction Algorithm with Bilinear Interpolation*. Institute of Information Sciences and Technology, Massey University, Palmerston North, New Zealand, 2003.

[LS97] Lee und Salcic. High-performance FPGA-based implementation of Kalman filter. *Microprocessors and Microsystems*, 21(4):257–265, 1997.

[MAHD11] Meng, Appiah, Hunter und Dickinson. FPGA Implementation of Naive Bayes Classifier for Visual Object Recognition. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Seiten 123–128, 2011.

[PB10] Papadonikolakis und Bouganis. A Novel FPGA-based SVM Classifier. In *Proceedings of International Conference on Field-Programmable Technology (FPT)*, Seiten 283–286, 2010.

[PMF+11] Pacholik, Muller, Fengler, Machleidt und Franke. GPU vs FPGA: Example Application on White Light Interferometry. In *Proceedings of International Conference on Reconfigurable Computing and FPGAs*, Seiten 481–486, 2011.

[RR09] Rosenband und Rosenband. A design case study: CPU vs. GPGPU vs. FPGA. In *Proceedings of International Conference on Formal Methods and Models for Co-Design*, Seiten 69–72, 2009.

[SHB10] Saldanha, Hartmann und Bobda. Scratch Detector - A FPGA Based System for Scratch Detection in Industrial Picture Development. In *Proceedings of International Symposium on Industrial Embedded Systems (SIES)*, Seiten 57–62, 2010.

[ZS03] Zhu und Sutton. FPGA Implementations of Neural Networks - a Survey of a Decade of Progress. In *Proceedings of 13th International Conference on Field Programmable Logic and Applications*, Seite Poster presentation, 2003.

Robustheit eingebetteter Systeme durch entscheidungstheoretische Betrachtungen

Ariane Heller
Technische Universität Chemnitz
Fakultät für Informatik
Professur Technische Informatik
ariane.heller@informatik.tu-chemnitz.de

Abstract: Umweltbedingungen und damit einhergehend Betriebsbedingungen für IT-Systeme können sich plötzlich und unerwartet ändern. Es wirken verschiedenste Störungen der Umwelt auf das System. Um einen drohenden Systemausfall und damit verbundene wirtschaftliche Verluste bis hin zur Gefährdung von Menschenleben abzuwenden, müssen Systeme robust auf veränderte Betriebsbedingungen reagieren.

1 Einleitung

Lange Zeit standen die funktionalen Aspekte bei der Entwicklung eingebetteter Systeme im Vordergrund. Die zunehmende Verbreitung von solchen Systemen in zentralen Bereichen unserer Gesellschaft wie Verkehrs-, Finanz- oder Gesundheitssystem führte zu einer Verlagerung des Schwerpunktes auf nichtfunktionale Eigenschaften. In den letzten Jahren haben sich nichtfunktionale Eigenschaften wie Sicherheit, Performanz, Echtzeitfähigkeit und Robustheit vom auserlesenen Merkmal hochspezialisierter Systeme zu unverzichtbaren Schlüsseleigenschaften in einer Vielzahl von Anwendungsfeldern entwickelt. Im entscheidungstheoretischen Ansatz wird die Analyse des Systemzustands mittels einer definierten Robustheitsfunktion mit der statistischen Entscheidungstheorie zur Modellierung von verschiedenen Entscheidungsmodellen bei einwirkenden Störereignissen verbunden. In Abhängigkeit der Einordnung des aktuellen Systemzustands erfolgt die Zuordnung zur Robustheitsfunktion und ein entsprechendes Entscheidungsmodell wird gewählt [HH12].

2 Stand der Technik

Der Begriff Robustheit ist im Allgemeinen nicht fest definiert. Es bestehen vielfältige Definitionen, die einsatz- und kontextabhängig sind. In [ea10] ist ein zusammenfassender Überblick zur begrifflichen Definition von Robustheit mit der Unterscheidung in qualitative und quantitative Robustheitserklärungen zu finden. Die statistische Entscheidungstheorie befasst sich mit der Herleitung optimaler Entscheidungen auf Basis statistischer Daten und Bewertungen der Entscheidung. Dabei werden Randbedingungen formuliert, die die

Auswirkung bzw. Folge einer Entscheidung betreffen: Nutzen, Verlust, Kosten oder Risiken [Men63]. Der Ansatz der statistischen Entscheidungstheorie wird in der Informatik bereits erfolgreich in der maschinellen Sprachverarbeitung eingesetzt [Ney03].

3 Entscheidungstheoretischer Ansatz

Ein eingebettetes System F wird durch Systemeigenschaften und diese wiederum durch Systemparameter beschrieben. Das sich in einer Umgebung U befindliche System kann durch Störereignisse $E_i \in \mathcal{E}$ mit $\mathcal{E} = \{E_1, \ldots, E_k\}$ der Umgebung beeinflusst werden. Jedem Parameter des Systems wird ein Störereignis der Umgebung zu geordnet und als Einfluss definiert. Grundlage für die Analyse der Systemrobustheit und die Formulierung eines Modells ist der Zusammenhang zwischen Störereignissen und die Auswirkung dieser auf die Systemparameter. Die Einflussfunktion $L_i\,(x)$ gibt den Grad der Störung des Systems in Abhängigkeit vom aktuellen Wert x des i-ten Systemparameters an mit $\mathcal{X}_i$ als Wertebereich. Der Wertebereich umfasst die drei Bereiche $\mathcal{X}_i^*$ Sollbereich, $\mathcal{X}_i^{**}$ Störbereich und $\mathcal{X}_i^{***}$ Toleranzbereich mit $\mathcal{X}_i = \mathcal{X}_i^* \cup \mathcal{X}_i^{**} \cup \mathcal{X}_i^{***}$. Solange der Wert des Systemparameters sich im Sollbereich $\mathcal{X}^*$ befindet, liegt keine Störung des Systems hinsichtlich des i-ten Systemparameter vor mit $L_i\,(x) = 0$ für $x \in \mathcal{X}^*$. Falls $x \in \mathcal{X}^{**}$ (Störbereich) ist der i-te Systemparameter nicht mehr kontrollier- und beeinflussbar und das System gilt als gestört mit einer Einflussfunktion $L_i\,(x) = \propto$ für $x \in \mathcal{X}^{**}$. Der Toleranzbereich $\mathcal{X}_i^{***}$ repräsentiert eingeschränkte Funktionsfähigkeit, welcher leichte bzw. kleinere Störungen erlaubt. Das System ist eingeschränkt funktionsfähig mit einer Einflussfunktion $0 < L_i\,(x) < \propto$ für $x \in \mathcal{X}^{***}$. Dabei sind $\mathcal{X}_i^*$ und $\mathcal{X}_i^{**}$ obligatorische Bereiche und somit immer vorhanden. Der Bereich $\mathcal{X}_i^{***}$ kann auch 0 sein. Das bedeutet es ist kein Toleranzbereich vorhanden. Toleranzbereiche können einseitig oder zweiseitig vorliegen.

3.1 Analyse des Systemzustand

Der Zustand des Systems wird mittels der Einflussfunktionen und der daraus folgenden Robustheitsfunktion analysiert. Hierfür werden zunächst die notwendigen Eigenschaften einer Robustheitsfunktion $R_{F,L}(L_1, ...L_k)$ definiert: normalisierte bzw. normierte Funktion, nichtnegative Funktion und absolute Robustheit bei $R_{F,L}(L_1, ...L_k) = 1$. Gilt für wenigstens ein $L_i = \infty$, $i = 1, ..., k$, dann liegt auch keine Robustheit vor. Es gilt dann $R_{F,L}(L_1, ...L_k) = 0$. Ebenfalls sollte Monotonie in den einzelnen Komponenten L_i, $i = 1, ..., k$, vorliegen. Falls $L_i \leq L_i'$ gilt, dann sollte $R_{F,L}(L_1, ..., L_i, ..., L_k) \geq R_{F,L}(L_1, ..., L_i', ..., L_k)$ gelten. Das System F wird durch k Systemparameter $x_i \in \mathcal{X}_i$, $i = 1, ..., k$, beschrieben. Es sind $L_i(x_i)$, $i = 1, ..., k$, die zugehörigen Einflussfunktionen und die Funktion $R_{F,L}$ mit den oben angegebenen Eigenschaften heißt Robustheitsfunktion für das System F bezüglich des Einflussfunktionensystems $L = \{L_1, ..., L_k\}$. Für jeden Parametervektor $\vec{x}_k \in \vec{\mathcal{X}} = \times_{i=1}^{k} \mathcal{X}_i$ ist $R_{F,L}(L_1(x_1), ..., L_k(x_k))$ der aktuelle Ro-

bustheitswert des Systems bei Vorliegen des Systemparametervektors $\vec{x}_k$. Die Robustheit stellt somit eine Kennzahl dar, die den Einfluss von Störereignissen auf das System anzeigt. Sinkt die Robustheit des Systems merklich sollten andere Entscheidungen hinsichtlich der Reaktion auf Störereignisse getroffen werden. Ziel ist es, dass System vor Eintritt in den Störbereich zu bewahren und eine größtmögliche Robustheit sicherzustellen.

3.2 Statistische Entscheidungsansatz

Die statistische Entscheidungstheorie ist eine Erweiterung der Spieltheorie in dem beobachtungsabhängige Entscheidungen zugelassen werden. In der Regel werden 2-Personen-Spiele des Statistikers gegen die Natur betrachtet. Aufgabe des Statistikers ist es, auf einen von der Natur bereits festgelegten Zustand zu reagieren. Der Statistiker hat dabei die Möglichkeit, die Natur vor seiner Entscheidung zu beobachten. Er erhält in Abhängigkeit vom Zustand der Natur mehr oder weniger genaue Hinweise über den Zustand der Natur. Diese Informationen können vom Statistiker in die Entscheidungsfindung eingebracht werden. Es beschreibt Θ die Zustandsmenge der Natur mit der Menge der möglichen Störereignissen. Auf jedes Störereignis kann das System mit einer Aktion $\mathcal{A}$ reagieren. In Abhängigkeit vom vorliegenden Zustand der Natur und der gewählten Reaktion des System entsteht ein möglicher Verlust L. Der Verlust kann dabei als monetäre Einheit oder auch Zeiteinheit verstanden werden. Der Verlust wird durch eine Verlustfunktion $L : \Theta \times \mathcal{A}$ beschrieben. Indem das System mittels eingesetzter Sensoren beobachtungsabhängig entscheidet und die Beobachtungsgröße eine Zufallsgröße X ist, wird auch der entstehende Verlust zu einer Zufallsgröße. Aus diesem Grund ist der mittlere Verlust zu betrachten. Zu gegebenen $\theta_i \in \Theta$ und $d_j \in \mathcal{D}$ heißt R Risiko der Entscheidungsregel d_j:

$$R\left(\theta_i, d_j\right) = E_{\theta_i} L\left(\theta_i, d_j\left(X\right)\right) = \sum_{k=1}^{m} L\left(\theta_i, d_j\left(x_k\right)\right) P_{\theta_i}\left(X = x_k\right) \tag{1}$$

Jedem $x \in \mathcal{X}$ wird nun über eine Entscheidungsregel d eine Aktion $a \in \mathcal{A}$ zugeordnet. Weiterhin ist die Wahrscheinlichkeit anzugeben, dass bei Vorliegen des Zustands θ_i die Beobachtungsgröße X den Wert $x_j \in \mathcal{X}$ besitzt. Hierfür sei $\theta_i \in \Theta$ und $x_j \in \mathcal{X}$ gegeben.

Mit der Linearen Ordnung kann eine optimale Entscheidung ausgewählt werden. Die wichtigsten Optimalitätskriterien der Linearen Ordnung sind das Minimax-Kriterium und das Bayes-Kriterium.
Das Minimax-Kriterium hat das Min-Max-Prinzip zur Grundlage und besteht darin, das maximale Risiko zu minimieren. Für jede Entscheidungsregel wird der maximal mögliche mittlere Verlust bzw. maximale Risiko des Parameter θ betrachtet. Das Minimax-Kriterium verfügt über ein geringes Maß an Risikobereitschaft. Es wird sich sehr vorsichtig verhalten, um für den schlechtesten aller möglichen Fälle noch das bestmögliche Ergebnis zu erzielen.
Das Bayes-Kriterium ist ein weiteres Kriterium mit dem eine Lineare Ordnung der Entscheidungsregeln erreicht werden kann. Dabei wird angenommen, dass für die in der Natur auftretenden Störereignisse eine Wahrscheinlichkeitsverteilung bekannt ist. Diese Wahr-

scheinlichkeitsverteilungen werden als a-priori-Wahrscheinlichkeiten für den Parameter θ bezeichnet. Das Bayes-Risiko einer Entscheidungsregel ergibt sich als Mittelwert über die Risikofunktion bzgl. der a-priori-Verteilung Q. Bezeichnet dazu q_i die gemäss a-priori-Verteilung vorliegende Wahrscheinlichkeit dafür, dass die Natur den Parameter θ_i wählt, dann ergibt sich das Bayes-Risiko einer Entscheidungsregel d bzgl. der a-priori-Verteilung Q zu

$$R(Q,d) = E_Q R(\theta,d) = \sum_{i=1}^{s} R\left(\theta_i,d\right) q_i \tag{2}$$

Die Entscheidungsregel $d_B \in \mathcal{D}$, für die gilt $R(d_B,Q) = \min_{d\in\mathcal{D}} R(d,Q)$ heißt Bayes-Regel.
In Abhängigkeit vom Robustheitswert erfolgt der Einsatz der Entscheidungsmodelle. Dabei unterscheiden sich die Entscheidungsmodelle in verschiedenen Optimalitätskriterien und somit angepassten Entscheidungsstrategien. Zeigt der Systemzustand einen niedrigen Robustheitswert an, so wird auf eine Entscheidung nach dem Minimax-Kriterium zurückgegriffen, da in diesem Fall zurückhaltender entschieden wird. Befindet sich der Robustheitswert in einem unkritischen Bereich wird ein Entscheidungsmodell mit dem Bayes-Kriterium eingesetzt. Zwischen beiden Entscheidungsmodellen können beliebige Abstufungen erfolgen, indem die optimale Entscheidungsstrategie sich der des Minimax- bzw. Bayes-Kriterium annähert.

4 Zusammenfassung

Der entscheidungstheoretische Ansatz setzt sich aus der Analyse des Systemzustands und der Wahl des Entscheidungsmodells zusammen. Zur Bewertung des aktuellen Systemzustands des eingebetteten Systems wurde eine Robustheitsfunktion mit definierten Eigenschaften vorgestellt. Für das Entscheidungsmodell wurde die statistische Entscheidungstheorie herangezogen und Optimalitätskriterien erläutert, die die Grundlage für unterschiedliche Entscheidungsmodelle darstellen.

Literatur

[ea10] Martin Radetzki et al. Robustheit nanoelektronischer Schaltungen und Systeme. Bericht, VDE Verlag GmbH, 2010.

[HH12] Ariane Heller und Wolfram Hardt. Entscheidungstheoretischer Ansatz für robuste eingebettete Systeme. Dresdner Arbeitstagung Schaltungs- und Systementwurf (DASS), Seiten 112–117. Fraunhofer-Institut für Integrierte Schaltungen, May 2012. 978-3-8396-0404-5.

[Men63] Günter Menges. Kriterien optimaler Entscheidungen unter Ungewißheit. *Statistical Papers*, 4(1), 1963. DOI: 10.1007/BF02923046.

[Ney03] H. Ney. Maschinelle Sprachverarbeitung- Der statistische Ansatz in der Spracherkennung und Sprachübersetzung. *Informatik Spektrum*, 2, Mai 2003.

Rendering und Verarbeitung massiver Punktwolken

Thomas Kanzok, Paul Rosenthal
Fakultät für Informatik, Juniorprofessur Visual Computing
Technische Universität Chemnitz
thomas.kanzok@informatik.tu-chemnitz.de
paul.rosenthal@informatik.tu-chemnitz.de

Abstract: Das Paper beschreibt ein Forschungsvorhaben im Bereich der Visualisierung und Vearbeitung von Punktwolken, die durch Laserscanning von Brücken gewonnen wurden. Es werden die beiden Teilproblembereiche Visualisierung und Verarbeitung vorgestellt, exisitierende Ansätze besprochen und die geplante Herangehensweise erklärt.

1 Einleitung

Öffentliche Infrastrukturbauwerke, insbesondere Brücken, müssen in regelmäßigen Abständen begutachtet werden um den gegenwärtigen Zustand zu dokumentieren, frühzeitig Schäden zu erkennen und Handlungsempfehlungen für Reparaturen oder eventuelle Sperrungen ableiten zu können. Das Gutachten umfasst dabei detaillierte Messungen der Strukturen und Fotos von problematischen Bereichen. Allerdings wird der traditionell manuelle Arbeitsablauf zunehmend durch vollautomatische Vermessung der Bauwerke mittels terrestrischer Laserscanner-Systeme ersetzt. Diese generieren eine dichte Wolke aus Abtastpunkten, welche direkt vor Ort oder später am PC verarbeitet werden können. Die erzeugten Scans liefern neben millimetergenauen geometrischen Informationen auch Farbwerte für jeden Messpunkt. Bei ausreichend hoher Auflösung der Scans werden damit neben der direkten Vermessung auch viele Fotos überflüssig.

Die Verwendung automatischer Laserscanner erzeugt allerdings auch neue Probleme. Damit die Bauingenieure die gewonnenen Daten verarbeiten können, müssen Sie zunächst geeignet visualisiert werden. Dabei erweist sich die enorme Menge der erzeugten Daten als Fluch und Segen zugleich. Einerseits ermöglicht sie es, die Punkte direkt – also ohne sie vorher in ein Dreiecksnetz umzuwandeln – zu rendern, ohne dass dabei große Löcher im Bild auftreten, andererseits können schnell Datenmengen in Größenordnungen von mehreren Gigabyte entstehen. Solche Mengen an Daten können nicht ständig im Arbeitsspeicher, geschweige denn im Grafikspeicher gehalten werden. Also müssen Wege gefunden werden, immer genau die Daten in Haupt- oder Arbeitsspeicher vorzuhalten, die auch tatsächlich gesehen werden können, während der Rest auf die Festplatte ausgelagert werden muss.

Dabei ist auch zu klären, ob eine schnellere Hauptspeicheranbindung der Grafikkarte mittels PCI Express quasi das ”Live-Streaming”der Daten zumindest aus dem RAM erlaubt.

Zum Beispiel liefern die 16 PCIe-Lanes der Radeon HD7970 mit 15,38 GB/s genügend Bandbreite, um circa eine Milliarde farbige Punkte (12 Byte Position, 4 Byte Farbe) pro Sekunde zu verarbeiten – oder, um die gerade noch angenehme Bildwiederholrate von 20 Bildern pro Sekunde einzuhalten, etwa 51 Millionen Punkte pro Bild. Andererseits sieht die Spezifikation dieser Karte 264 GB/s und damit etwa die 17-fache PCIe-Bandbreite für den Grafikspeicher vor, was entweder für mehr Details oder höhere Frameraten genutzt werden könnte.

Zur Verarbeitung der digitalen Modelle müssen dem Nutzer die gleichen Möglichkeiten an die Hand gegeben werden, die auch vor Ort an den Bauwerken existieren. Dazu gehört das Messen von Strecken und Flächen und die genaue Inspektion von gefährdeten Bereichen. Zusätzlich ermöglicht die computergestützte Verarbeitung aber auch das Hinzufügen von Beschriftungen, Links und Detailfotos direkt in das Modell, was am realen Objekt nicht möglich ist.

2 Problemfeld Rendering

Die Verwendung von Punkten als Renderingprimitive ist ein schon seit langem erforschter Bereich [KB04]. Als Quasi-Standard hat sich dabei das sogenannte Splatting [BHZK05] etabliert, welches die Punkte durch elliptische Flächenelemente ersetzt, auf denen lokale Basisfunktionen für die Oberflächenrekonstruktion im Bildraum definiert werden können. Allerdings erzeugen heutige Laserscanner so dichte Punktwolken, dass üblicherweise allein mit den Punkten beinahe geschlossenen Flächen gerendert werden können. Verbleibende kleine Lücken können dann effizient im Bildraum geschlossen werden [RL08].

Das Hauptproblem stellt die Datenverwaltung dar. Während aktuelle Grafikkarten ohne Probleme in der Lage sind über 30 Millionen Punkte mit mindestens 20 Bildern pro Sekunde zu verarbeiten, falls sie im schnellen Grafikspeicher vorliegen, ist das Streaming der benötigten Punkte von der Festplatte zu langsam, um noch interagieren zu können. Um die Daten, die auf der Grafikkarte gehalten werden müssen, zu beschränken, gibt es im Wesentlichen zwei große Gruppen von Verfahren: zum einen das Aussortieren von Punkten, die vom gegenwärtigen Betrachterstandpunkt nicht gesehen werden können (*Culling*), zum anderen die Verwendung unterschiedlicher Auflösungsstufen für weiter entfernte Teile des Modells (*Level Of Detail, LOD*).

Für effizientes Culling werden die Punkte in eine den Raum aufteilende Baumstruktur sortiert, die es ermöglicht, schnell große nicht sichtbare Teile zu verwerfen [GZPG10, RD10]. Die Traversierung dieser Strukturen erfolgt dann üblicherweise auf CPU-Seite, da das nötige dynamische Speichermanagement auf GPUs problematisch ist.

Die Idee hinter den Level-Of-Detail Methoden ist, dass ein Objekt auf dem Bildschirm eine kleinere Fläche einnimmt, je weiter es vom Betrachter entfernt ist. Diese Abschätzung kann genutzt werden, um eine Detailstufe des Modells auszuwählen, bei der dem Betrachter kein qualitativer Unterschied auffällt. Besonders Punktdaten eignen sich gut für derartige Repräsentationen, da bei geeigneter Sortierung der Punkte [DVS03] einfach die ersten n Punkte des sortierten Punkt-Feldes gerendert werden müssen.

Das geplante System soll das CPU-lastige Culling mit einem GPU-basierten LOD-Ansatz verbinden, um die beiden leistungsfähigsten Recheneinheiten aktueller PCs optimal auszulasten. Dafür ist eine geeignete Datenstruktur zu finden, die diese Verbindung ermöglicht und die dabei insbesondere auf GPU-Seite platzsparend ist, damit die sichtbaren Punkte auch tatsächlich im Grafikspeicher gehalten werden können. Für eventuelle Nachbearbeitung, Beleuchtung oder Illustratives Rendering setzen wir auf bildraumbasierte Verfahren, die sich an den Methoden des Deferred Shading [DWS+88, Shi05] orientieren.

3 Problemfeld Verarbeitung

Die gescannten Daten sind allerdings nicht frei von Störungen. Zunächst dauert ein kompletter Rundum-Scan für eine Position mit der momentan verwendeten Ausrüstung etwa eine Minute, in der zuerst die Positionen der Punkte und erst in einem zweiten Schritt ihre Farben aufgenommen werden. Hat sich die Umgebung in der Zeit zwischen den Scans verändert, etwa weil sich Personen oder Fahrzeuge bewegt haben, werden den Positionen eventuell falsche Farben zugeordnet. Außerdem müssen oft mehrere Scans von unterschiedlichen Standpunkten aufgenommen werden, um ein Objekt vollständig zu erfassen. Zwischen zwei Scans können sich die Umgebungsbedingungen ebenfalls drastisch ändern, etwa durch verschiedene Lichtverhältnisse aufgrund von Bewölkung oder unterschiedlichen Tageszeiten.

Bevor man sinnvoll mit den Daten arbeiten kann, müssen diese Artefakte entfernt werden. Die Lichtverhältnisse sind dabei ein Problem, welches wahrscheinlich nur schwer global zu lösen ist. Allerdings ist es möglich, zumindest unauffällige Übergänge zwischen den einzelnen Scans zu erzeugen. Aber auch eine Bereinigung der geometrischen Artefakte sollte sich über die Konsistenz geometrischer Merkmale in mehreren überlappenden Scans bewerkstelligen lassen.

Die wichtigste Verarbeitungsfunktion stellt das Messen von Strecken und Flächen dar. Während euklidische Messungen für Strecken und Streckenzüge, sowie planare Flächenberechnung kein Problem darstellen, gestaltet sich die Berechnung der Längen und Flächen von Pfaden, die auf der Objektoberfläche liegen sollen schwieriger. Für diese *geodätischen* Messungen, die für Dreiecksnetze exakt lösbar sind [SSK+05], muss eine lokale Oberflächenrekonstruktion durchgeführt werden. Auch dafür sind unzählige Algorithmen bekannt [ABK98, KBH06, KSO04], eine vollständige Rekonstruktion der Bauwerke ist allerdings nicht gewünscht, da die Vorverarbeitungszeit minimal gehalten werden soll. Die Herausforderung besteht also hier vor allem darin, nur den Bereich der Punktwolken zu isolieren, der für einen geodätischen Pfad relevant ist.

4 Zusammenfassung

Im Rahmen des vorgestellten Projektes soll ein Verarbeitungssystem für große Punktwolken entstehen, das es Bauingenieuren ermöglicht, weite Teile des momentanen Arbeitsab-

laufes an einem Arbeitsplatzrechner erledigen zu können. Nach der einmal erfolgten und vor Ort grob überprüften Erfassung der Daten soll eine Begutachtung dann ausschließlich am digitalen Modell möglich sein. Damit wird auch eine effiziente Dokumentation des Bauzustandes und der einfache Vergleich desselben über mehrere Jahre möglich.

Literatur

[ABK98] Nina Amenta, Marshall Bern und Manolis Kamvysselis. A new Voronoi-based surface reconstruction algorithm. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '98, Seiten 415–421, New York, NY, USA, 1998. ACM.

[BHZK05] Mario Botsch, A. Hornung, M. Zwicker und L. Kobbelt. High-quality surface splatting on today's GPUs. In *Point-Based Graphics, 2005. Eurographics/IEEE VGTC Symposium Proceedings*, Seiten 17 – 141, june 2005.

[DVS03] C. Dachsbacher, C. Vogelgsang und M. Stamminger. Sequential point trees. In *ACM Transactions on Graphics (TOG)*, Jgg. 22, Seite 657–662, 2003.

[DWS+88] Michael Deering, Stephanie Winner, Bic Schediwy, Chris Duffy und Neil Hunt. The triangle processor and normal vector shader: a VLSI system for high performance graphics. *SIGGRAPH Comput. Graph.*, 22(4):21–30, Juni 1988.

[GZPG10] P. Goswami, Y. Zhang, R. Pajarola und E. Gobbetti. High quality interactive rendering of massive point models using multi-way kd-trees. In *Computer Graphics and Applications (PG), 2010 18th Pacific Conference on*, Seiten 93 —- 100, 2010.

[KB04] Leif Kobbelt und Mario Botsch. A survey of point-based techniques in computer graphics. *Comput. Graph.*, 28:801 – 814, December 2004.

[KBH06] Michael Kazhdan, Matthew Bolitho und Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, SGP '06, Seiten 61–70, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.

[KSO04] Ravikrishna Kolluri, Jonathan Richard Shewchuk und James F. O'Brien. Spectral surface reconstruction from noisy point clouds. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, SGP '04, Seiten 11–21, New York, NY, USA, 2004. ACM.

[RD10] Rico Richter und Jürgen Döllner. Out-of-core real-time visualization of massive 3D point clouds. In *Proceedings of the 7th International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa*, AFRIGRAPH '10, Seiten 121–128, New York, NY, USA, 2010. ACM.

[RL08] Paul Rosenthal und Lars Linsen. Image-space point cloud rendering. In *Proceedings of Computer Graphics International*, Seiten 136–143, 2008.

[Shi05] O. Shishkovtsov. Deferred shading in stalker. *GPU Gems*, 2:143–166, 2005.

[SSK+05] V. Surazhsky, T. Surazhsky, D. Kirsanov, S. J Gortler und H. Hoppe. Fast exact and approximate geodesics on meshes. *ACM Transactions on Graphics (TOG)*, 24(3):560, 2005.

Multimedia-Verarbeitung und -Adaptierung in IP-Netzen

Albrecht Kurze

Technische Universität Chemnitz
Albrecht.Kurze@informatik.tu-chemnitz.de

Bereits jetzt erzeugt der Transport von Multimedia-Inhalten, vorrangig Videodaten, die Hauptlast in (mobilen) IP-Netzen. Prognosen gehen davon aus, dass sich dieser Trend in den nächsten Jahren noch verstärken wird [VNI12]. Die Schaffung von Interoperabilität beim Zugriff auf Multimedia-Ressourcen wie Videos/Video-Streams und Fragen in den Bereichen Quality-of-Service (QoS) bzw. Quality-of-Experience (QoE) erscheinen trotz verschiedener Forschungsarbeiten der letzten Jahre noch immer nicht befriedigend gelöst bzw. umgesetzt.

Zwar gibt es vom MPEG-Konsortium entsprechende Konzepte, z.B. innerhalb von MPEG-21 wie Universal Multimedia Access (UMA) und Digital Item Adaptation (DIA) [VET05], doch sind diese Modelle nicht unmittelbar auf Realwelt-Szenarien anwendbar. Die klassischen Ansätze zur Verbesserung von QoS/QoE zielen auf ein Traffic Management (Bandbreitenverwaltung) und optimierte Zwischenspeicher. Zusätzlich sind in neuerer Zeit im Rahmen verschiedener Bestrebungen rund um „Future Internet“ (Content Centric/Aware Networking) Themen von Interesse, die auch eine Medienverarbeitung in IP-Netzen selbst einbeziehen, z.B. [BAU11].

Die aktive Unterstützung von Bereitstellung, Verarbeitung und Transport der AV-Medien durch zusätzliche Netzelemente bzw. Funktionseinheiten, z.B. in Form von Gateways und Proxies, sollte sich als hilfreich erweisen und neue Szenarien für die Nutzung von Multimedia-Angebote ermöglichen. Dies sind z.B. personalisierte Mediendienste oder Optimierungen, die darauf abzielen, AV-Medien überhaupt nutzbar zu machen, Netzbandbreite einzusparen und QoS/QoE durch Anpassungen zu verbessern.

Besonders die Nutzung von mobilem Video ist durch die Heterogenität der genutzten Geräte, verfügbaren Netze und deren Charakteristiken sowie technischen Parameter der angebotenen Medien stark fragmentiert. Die Vielfalt an genutzten Formaten, Codecs und Protokollen führt leicht zu Konstellationen, in denen Inkompatibilitäten und andere Einschränkungen eine Nutzung nur suboptimal erlauben, erschweren oder gar unmöglich machen. Um trotz solcher Probleme dem Nutzer den Zugang und eine „optimale“ Nutzung zu ermöglichen, könnten Multimedia-Adaptierungen helfen. Diese Adaptierungen können, je nachdem wie groß die Interoperabilitätslücke ist, trivial ausfallen, z.B. durch automatische Selektion eines inhaltsgleichen, aber kompatiblen Formats, oder hochkomplex sein und mehrstufige, genau aufeinander abgestimmte, rechen- und ressourcenintensive Verarbeitungsschritte auf verschiedenen Ebenen erfordern.

Das in der Arbeit entworfene Konzept soll sich möglichst auch für Realwelt-Szenarien eignen, dabei sind Anpassungen notwendig für:

- Access (Referenzierung, Einbettung)
- Transport/Delivery (Protokolle)
- Content (Formate, Codecs, Inhalte)

Es ist denkbar, Teile der dafür notwendigen Funktionalitäten in vorhandenden, etablierten Netzelemente z.B. Routern oder Komponenten in Form von Network Support Functions (NSF) wie Traffic Management Systemen und Deep Packet Inspection (DPI), zu nutzen bzw. in diese zu integrieren. Andere Funktionalitäten müssen neue, spezielle Elemente in Form von Adaptation Functions (AF) bereitstellen. Diese sind zunächst funktional abstrakt konzipiert.

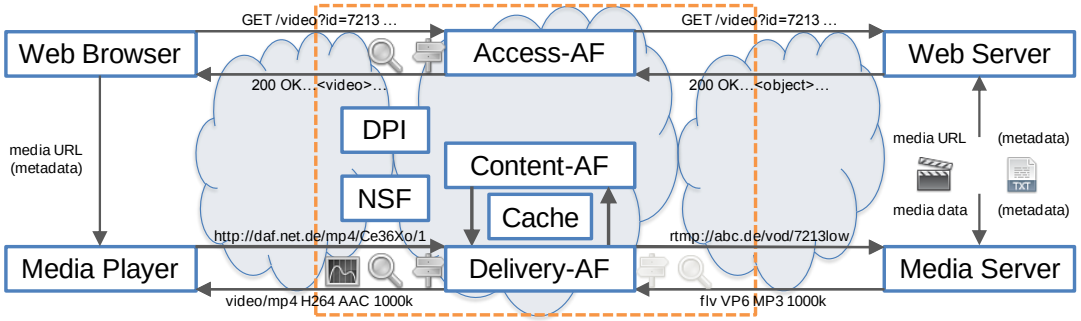


Abbildung 1: Konzept der Funktionsblöcke zur Verarbeitung und Anpassung (innerhalb des orangenen Rahmens)

Die involvierten Komponenten sollen beim Zugriff potentielle Probleme der Interoperabilität bzw. fehlender Bandbreiten erkennen und automatisch alle notwendigen Anpassungen durchführen. Dabei ist zu klären, ob diese Art der Verarbeitung und Anpassung von Multimedia-Ressourcen im Netzwerk für Nutzer und Anbieter auch unbemerkt erfolgen kann, so dass auch ein Einsatz in nichtkooperativen Umgebungen möglich ist. Im besten Fall sind damit weder auf Anbieterseite, also da wo AV-Medien erzeugt und bereitgestellt werden, z.B. Web-/Medien-/Streaming-Server, noch auf Nutzungsseite (Medienplayer etc.) besondere Maßnahmen bzw. eine aktive Kooperation notwendig (transparentes System).

In einer Access-AF werden Referenzen bzw. Einbettungen zu relevanten AV-Medien bzw. zugeordneten technischen Metadaten, z.B. in per HTTP transportierten HTML-Seiten, erkannt und ggf. angepasst, so dass die Videoinhalte prinzipiell auf dem abrufenden Endgerät nutzbar werden. Dabei sollen die Fähigkeiten und Einschränkungen (Capabilities/Constrains) der Endgeräte bei der Medienwiedergabe und ggf. auch die Netzwerkcharakteristiken berücksichtigt werden. Falls dazu Anpassungen des Medientransports bzw. der Mediendaten selbst notwendig sein sollten, werden die Funktionalitäten der Delivery-AF und Content-AF bemüht.

In der Delivery-AF werden die angepassten Medien-URLs auf die Original-URLs abgebildet. Dabei soll zwischen den Protokollen für Session-Verwaltung, Steuerung und Medientransport vermittelt werden. Für die angepassten Medien-URLs ist eine Erweiterung der W3C-Media Fragments[1] eine mögliche Variante, die zusätzlich zu den bereits im Standard aufgeführten Anpassungsmöglichkeiten auch Parameter, z.B. zur Anpassung der Protokolle oder Codecs, beinhaltet. Es soll untersucht werden, ob sich mit solchen „Media Processing URLs“ die gewünschte Funktionalität herstellen lässt. Alternativen dazu wären ad-hoc erzeugte Meta-Beschreibungen der notwendigen Anpassungen, z.B. in einem XML-Format, ggf. unter Anwendung etablierter Multimedia-Beschreibungen wie SMIL.

Falls durch Access-AF oder Delivery-AF erkannt wird, dass auch Anpassungen der eigentlichen AV-Mediendaten notwendig sind, übernimmt die Content-AF diese Aufgabe. Dies kann von einfachen und effizienten Anpassungen auf Ebene des ursprünglich AV-Bitstreams, über die Ebene der kodierten AV-Elementarströme bei Wechsel des Container-Formats (Transmuxing) bis hin zu aufwendigen Decode und Encode (Transcode) der eigentlichen Medienessenzen reichen, wenn z.B. Wechsel der Codecs oder Bildgrößenänderungen notwendig sein sollten.

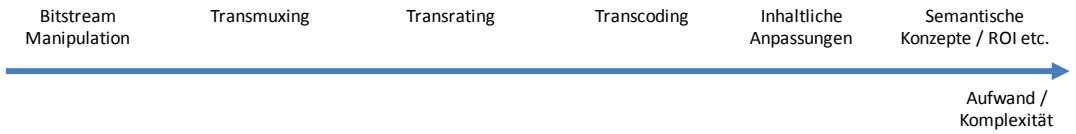


Abbildung 2: Aufwandabschätzung verschiedener Verarbeitungs- und Anpassungsstufen

Im Gegensatz zu den Modellen und Annahmen zu a priori Bitstream-adaptierbaren AV-Inhalten wie z.B. in [TIM06] und den dort gezeigten generischen Anpassungswerkzeugen, liegen AV-Medien heutzutage typischerweise in Ausprägungen vor, die meist keine einfache AV-Bitstream-Manipulation zur Erzielung der gewünschten Anpassungen zulassen. Entsprechend sind komplexe Verarbeitungs- und Anpassungsschritte in der Content-AF zu meistern.

Bei der Verarbeitung und Adaptierung von AV-Medien in IP-Netzen liegt eine der Herausforderungen in der effizienten und flexiblen Verknüpfung und Steuerung der verfügbaren Ressourcen:

- Rechenleistung in den verschiedenen Komponenten
- Speicher in den verschiedenen Komponenten
- verfügbare Bandbreite zwischen den Komponenten

Aus den Überlegungen zum geschilderten Konzept ergeben sich einige Problempunkte, für die bislang noch keine befriedigenden Antworten gefunden werden konnten.

[1] http://www.w3.org/2008/WebVideo/Fragments/ (zuletzt aufgerufen im Juni 2012)

Die Vielzahl möglicher Varianten, wie AV-Medien bzw. evtl. vorhandene zugehörige Metadaten „In-the-Wild“ referenziert und genutzt werden, z.B. in typischen HTML-Webdokumenten, ist quasi unüberschaubar.

Entsprechend bleibt abzuschätzen, welche Einschränkungen für die Access-AF nicht umgangen werden können, bzw. was die Mindestanforderungen für die Beschreibung und Nutzung der AV-Medien sind (technische Metadaten). Daraus wird sich auch die Grenze zwischen notwendiger „Kooperation“ und dem Ansatz eines unbemerkt agierenden Systems ergeben.

Eine große Zahl gleichzeitig durchzuführender Anpassungen führt zwangsläufig zu einem Skalierungsproblem hinsichtlich der o.g. Ressourcen. Deshalb ist zu prüfen, inwieweit sich ein solches Konzept in etablierte Strukturen der digitalen Mediendistribution wie z.B. CDNs (Content Distribution Networks) integrieren lässt.

Neben der Anpassung vorhandener AV-Medien ist auch interessant, wie eine IP-basierte Medienproduktion [KUR11] von derartigen Konzepten profitieren kann, um z.B. durch die direkte Kopplung von Produktion und Distribution personalisierte Mediendienste zu realisieren. Unter Nutzung der gleichen Funktionsblöcke werden erweiterte Anwendungen möglich. Quasi beliebige, per IP nutzbare AV-Medienquellen lassen sich flexibel in jeweils passenden „Media-Flows“ für die angestrebten Mediensenken (Endgeräte/Zielplattform) verarbeiten. Besonders im Bereich der Content-AF ergeben sich z.B. durch Bildmischung aus verschiedenen AV-Quellen, zusätzlichen Overlays etc. neue, jeweils einzigartige Repräsentationen der ursprünglichen Quellen. Konkrete Anwendungsszenarien könnten z.B. individuell ausgewählte und skalierte Bildausschnitte aus (fixen) Panoramaaufnahmen oder ein wahlfrei aus verschiedenen Blickwinkeln zusammengestellter Videomix („Personal View“) sein.

Literaturverzeichnis

[BAU11] Bauer, M; Braun, S.; Domschmitz, P.: Media Processing in the Future Internet. In Proc. Euroview 2011, Würzburg, 2011

[KUR11] Kurze, A.; Knauf, R.: A Scalable Open Source Framework for Live Media Production and Distribution. In Proceedings of 14th ITG Conference on Electronic Media Technology, Dortmund, Germany, March 23-24, 2011, ISBN 978-3-00-033964-6.

[TIM08] Timmerer, C.: Generic Adpatation of Scalable Multimedia Resources, VDM Verlag, Saarbrücken, 2008, ISBN 3639003969

[VET05] Vetro, A.; Timmerer, C.: Digital Item Adaptation: Overview of Standardization and Research Activities. In IEEE Transactions on Multimedia, vol. 7, no. 3, June 2005, pp. 418-426

[VNI12] Cisco VNI – Visual Networking Index, 2012, online verfügbar unter http://www.cisco.com/en/US/netsol/ns827/networking_solutions_sub_solution.html (zuletzt abgerufen im Juni 2012)

Dynamische Ressourcenverwaltung in Hierarchischen Heterogenen Verteilten Eingebetteten Systemen

Sven Schneider

sven.schneider@cs.tu-chemnitz.de

Der immer fortschreitende Einsatz eingebetteter Systeme hat Computer zu einem wesentlichen Bestandteil moderner Technologie gemacht. Eingebettete Systeme werden unter anderem in Kommunikationsinfrastrukturen und Kommunikationsgeräten, industriellen Steuerungen und im Verkehrswesen eingesetzt. Das Verkehrswesen, welches Erdgebundene Fahrzeuge, Luftfahrzeuge sowie Verkehrsleiteinrichtungen umfasst, stellt besonders hohe Anforderungen an Eingebettete Systeme. So muß ein solches System eine hohe Betriebssicherheit aufweisen und in Echtzeit auf seine Umgebung reagieren, wobei das Gewicht, die Größe und der Energieverbrauch des Systems stark begrenzt sind. Diese Anforderungen haben im Bereich der Unbemannten Autonomen Luftfahrzeugen (UAVs) eine besondere Relevanz. Da diese Systeme sich mit einer hohen Geschwindigkeit fortbewegen müssen, ist der Zeitrahmen, in dem ein solches System auf die Umwelt reagieren muß, sehr beschränkt und fest. Somit ist der Betrieb als Echtzeitsystem unabdingbar. Zusätzlich wird der Operationsradius eines UAV wesentlich durch die Größe, das Gewicht und den Energieverbrauch bestimmt.

Eine weitere Herausforderung für UAVs ist die Komplexität des Systems. Da ein UAV Entscheidungen ohne den Eingriff durch Menschen trifft, muß das System selbständig auf der Umwelt angemessene Reaktionen schließen. Diese Aufgabe kann in mehrere Unteraufgaben, wie z.B. das Aufnehmen und Interpretieren von Sensoreingaben, mehrere Sensoreingaben verknüpfen, besondere Situationen erkennen, Entscheidungsfindung oder die Umsetzung einer Entscheidung, unterteilt werden. Die Unteraufgaben können dabei hierarchisch weiter unterteilt werden. Da diese Aufgaben von weitgehend voneinander unabhängigen Eingebetteten Systemen [GP07] verarbeitet werden, die miteinander durch ein Kommunikationsnetzwerk [Inc06] verbunden sind, handelt es sich dabei um heterogene Cluster. Dieser Trend führte dazu, daß in Eingebetteten Systemen Paradigmen, die Ursprünglich aus dem Bereich des High Performance Computing stammen, wie zum Beispiel das Message Passing [Mes09] aus dem Cluster- und Grid-Computing, eingesetzt werden.

Sowohl die Plattform als auch die Anwendungen eingebetteter Systeme haben eine hierarchische Struktur. So werden zum BeispielpDaten von einem Videosensor meist lokal vorverarbeitet und auf einem oder mehreren eingebetteten Systemen weiterverarbeitet.

Missionen, die durch ein UAV durchgeführt werden, werden in verschiedene Phasen, wie starten, in das eigentliche Missionsgebiet fliegen, die Mission ausführen, zurückfliegen und landen, aufgeteilt. Dabei ist es sinnvoll anzunehmen, daß die Flugsteuerung bei der Landung erheblich genauer sein muß, als beim Überfliegen einer bestimmten Zone im

Missionsgebiet. Das bedeutet aber auch, daß sich der Ressourcenbedarf zur Bewältigung einzelner Teilaufgaben während der Mission stark ändern kann. Durch eine günstige Mehrfachverwendung der Ressourcen ist es daher möglich Gewicht und Größe, und damit auch indirekt den Energieverbrauch, solcher Systeme zu senken. Standardtechnologien zur Mehrfachverwendung von Ressourcen in Software sind unter anderem dynamische Speicherverwaltung, Multi-Threading oder das dynamische Laden von Softwaremodulen und -bibliotheken. In Hardwaresystemen ermöglicht es die Technologie der Dynamischen Partiellen Rekonfigurierung die Hardwareschaltung während des Betriebs zu verändern. Damit können Grundlegende Hardwareressourcen wie Chipfläche von mehreren Tasks nacheinander genutzt werden.

Aktuelle Forschungsprojekte beschäftigen sich damit Methoden der künstlichen Intelligenz in Komponenten zu integrieren, die die Durchführung der Mission überwachen und steuern [HWK+11]. Die Verwendung von künstlicher Intelligenz erhöht die Wahrscheinlichkeit eine Mission erfolgreich abzuschließen, vor allem wenn die Umwelt nur unscharf erkannt wird oder unvorhergesehene Situationen auftreten. Ein großer Nachteil dieser Methoden ist es aber, daß es schwieriger wird vorherzusagen, welche Aufgaben gleichzeitig durchgeführt werden oder welche Ressourcenanforderungen sie haben werden. Daher müssen sich diese Systeme dynamisch an ein sich änderndes Aufgabenprofil anpassen. Das führt dazu, daß die Ressourcen ebenfalls zur Laufzeit verwaltet werden müssen.

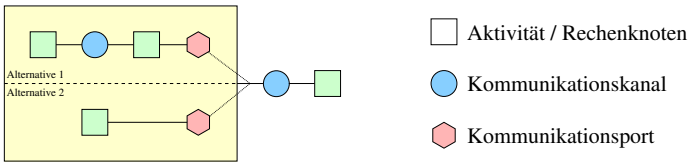


Abbildung 1: Repräsentation einer Anwendung

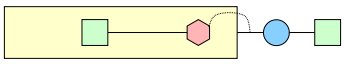


Abbildung 2: Repräsentation der Infrastruktur

In der Dissertation wird die dynamische Verwaltung der Ressourcen und die Durchsetzung von Ressourcenzuteilungen zur Laufzeit betrachtet. Zu den wesentlichen Punkten wird Aufstellung eines Systemmodells gehören, welches es ermöglichen soll, Anwendungen (Abbildung 1) und Infrastruktur (Abbildung 2) so zu beschreiben, daß eine Laufzeitumgebung (Abbildung 3) eine sinnvolle Ressourcenverwaltung durchführen kann. Das Systemmodell muß dabei die besonderen Eigenschaften der Heterogenität und Hierarchie der Anwendungen sowie der Infrastruktur berücksichtigen und durch eine geeignete Datenstruktur darstellbar sein.

Im Gegensatz zu den existierenden Ansätzen von Teich et al. [HTRE02, SKHT06] soll das Modell zur Laufzeit verarbeitet werden und mehrere Ressourcenarten gleichzeitig unterstützten. Das Systemmodell lehnt sich im Bereich der Anwendungsdarstellung an das MAGELLAN-Projekt [CV01] an, wobei im Kontext dieser Arbeit die Darstellung des Kontrollflusses zugunsten eines höheren Abstraktionsgrades aufgegeben wird. Anders als

in den auf Taskgraphen basierenden Modellen, stellen die Verbindungen potentiell asynchrone Kommunikation dar.

Die Datenstruktur bildet hierarchisch organisierte Graphen ab. Annotationen an die Graphen werden die Ressourcenanforderungen bei Anwendungen und die Ressourcenangebote bei der Infrastruktur darstellen. In Anwendungen darf ein Unterblock zudem Umsetzungsalternativen enthalten.

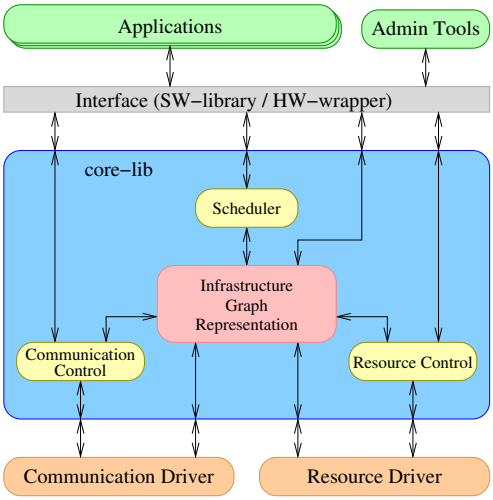


Abbildung 3: Architektur der Laufzeitumgebung

Ein weiterer wichtiger Aspekt wird der Algorithmus [SMH09] zum Finden einer gültigen Abbildung (Abbildung 4) zwischen der Anwendungen und der Infrastruktur sein. Hierbei wird eine Abbildung als gültig angesehen, wenn alle geforderten Anwendungen auf der Infrastruktur so platziert werden können, daß alle Ressourcenanforderungen einer Umsetzungsalternative der Anwendungen erfüllt sind. Der Algorithmus wird heuristisch konstruktiv arbeiten, wobei falsche Entscheidungen durch Backtracking rückgängig gemacht werden können. Um potentiell günstige Entscheidungen treffen zu können, müssen die komplexen Ressourcenbeschreibungen so reduziert werden, daß verschiedene Möglichkeiten, bezüglich ihrer Erfolgsaussichten auf eine gültige Abbildung, miteinander verglichen werden können.

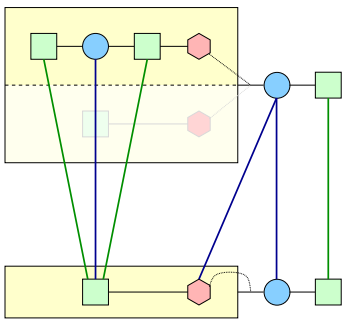


Abbildung 4: Platzierung einer Anwendung auf einer Infrastruktur

Desweiteren entsteht eine Prototypische Ausführungsplattform (Abbildung 5), auf der das Systemmodell und der Abbildungsalgorithmus auf deren Umsetzbarkeit geprüft werden sollen.

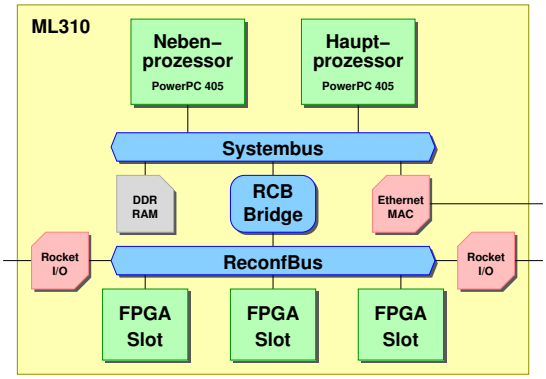


Abbildung 5: Architektur eines Knotens der Ausführungsplattform

Literatur

[CV01] K.S. Chatha und R. Vemurl. MAGELLAN: multiway hardware-software partitioning and scheduling for latency minimization of hierarchical control-dataflow task graphs. In *Hardware/Software Codesign, 2001. CODES 2001. Proceedings of the Ninth International Symposium on*, Seiten 42 –47, 2001.

[GP07] R. Garside und F.J. Pighetti. Integrating Modular Avionics: A new role emerges. In *Digital Avionics Systems Conference, 2007. DASC '07. IEEE/AIAA 26th*, Seiten 2.A.2–1 –2.A.2–5, oct. 2007.

[HTRE02] Christian Haubelt, Jürgen Teich, Kai Richter und Rolf Ernst. System Design for Flexibility. In *In Proceedings of Design, Automation and Test in Europe (DATE'02)*, Seiten 854–861. IEEE, 3 2002.

[HWK+11] J. Halbig, A. Windisch, P. Kingsbury, N. Oswald und W. Hardt. Integration of realtime decision-making in time-triggered software architectures for certifiable autonomous unmanned systems. In *Instrumentation Control and Automation (ICA), 2011 2nd International Conference on*, Seiten 284 –289, nov. 2011.

[Inc06] Aeronautical Radio Inc. ARINC Specification 429-17 Part 1., 2006. Prepared by the Airlines Electronical Engeneering Commitee.

[Mes09] Message Passing Interface Forum. *MPI: A Message-Passing Interface Standard, Version 2.2*. High Performance Computing Center Stuttgart (HLRS), September 2009.

[SKHT06] Thilo Streichert, Dirk Koch, Christian Haubelt und Jürgen Teich. Modeling and design of fault-tolerant and self-adaptive reconfigurable networked embedded systems. *EURASIP J. Embedded Syst.*, 2006(1):9–9, Januar 2006.

[SMH09] S. Schneider, A. Meisel und W. Hardt. Communication-Aware Hierarchical Online-Placement in Heterogeneous Reconfigurable Systems. In *Rapid System Prototyping, 2009. RSP '09. IEEE/IFIP International Symposium on*, Seiten 61 –67, june 2009.

Integration von OntoWiki in SharePoint Foundation

Martin Wegner

mweg@hrz.tu-chemnitz.de

Abstract: Die Einbeziehung von modernen Wissensbasen in vorhandene Unternehmensstrukturen ist eine der aktuellen Herausforderungen in der Informatik. Aus diesem Grund wird in dem folgenden Paper die mögliche Integration von OntoWiki in Microsoft SharePoint Foundation untersucht.

1 Überblick

1.1 Software

OntoWiki ist eine webbasierte Umgebung zur Verwaltung strukturierter Informationen. Es stellt Werkzeuge zur Navigation, Visualisierung und Erstellung von RDF basierten Wissensbasen bereit. [TFH10] Entwickelt wird die Software hauptsächlich von der Forschergruppe AKSW.

SharePoint Foundation (fortan SharePoint genannt) ist eine von Microsoft hergestellte Software, die eine Weboberfläche bereitstellt. Diese soll Mitarbeitern eines Unternehmens die Zusammenarbeit erleichtern. Wichtige Funktionen sind unter anderem Dokumentverwaltungen und gemeinsame Kalender. SharePoint steht auch für kommerzielle Zwecke kostenlos zur Verfügung. Zur Zielgruppe gehören „Organisationen und Unternehmenseinheiten aller Größen“ [ST10].

1.2 Szenario und Testumgebung

Als Szenario wird ein mittelständiges Unternehmen angenommen, dass einen Server mit dem kommerziellen, jedoch kostenfreien Microsoft SharePoint Foundation betreibt. Geplant ist die Einbindung von Ontowiki als Wissensbasis. Daraus leitet sich die Testumgebung ab, welche aus einem Netzwerk mit zwei Rechnern besteht. Als Betriebssystem wird auf beiden Rechnern ein Windows Server 2008 R2 mit Service Pack 1 eingesetzt. Der Rechner 1 wird für OntoWiki genutzt. Dafür war zusätzlich die Installation von XAMPP inklusive Apache und MySql Server nötig. SharePoint ist mit allen Softwarevoraussetzung auf Rechner 2 installiert. Wie die Installation im Einzelnen abläuft, wird im folgenden Kapitel genauer erläutert.

2 Installation

OntoWiki ist eine Open Source Software und kann kostenlos heruntergeladen[1] und verwendet werden. Nach erfolgreicher XAMPP Installation ist die Einrichtung von OntoWiki nach der Anleitung der verlinkten Homepage[2] kein Problem.

Der Download[3] von **SharePoint** ist ebenfalls kostenlos. Der Setupmanager lädt alle Softwarevoraussetzungen nach und konfiguriert die nötigen Serverrollen. Dadurch müssen nach der Installation keine weiteren Einstellungen vorgenommen werden.

3 Mögliche Verknüpfungspunkte

3.1 Verlinkung

Die einfachste Verbindung zwischen zwei Webseiten ist eine Verlinkung. Diese lassen sich bei SharePoint im Webinterface unter *Einfügen –> Verknüpfung* einfügen. Zusätzlich besteht die Möglichkeit, Links in Listen zu organisieren. Dazu muss einfach unter *Einfügen –> Neue Liste –> Hyperlinks* eine neue Liste erstellt werden, die dann auf mehreren Seiten verwendet und zentral administriert werden kann.

3.2 Einbettung

Für die Einbettung einer anderen Webseite stellt SharePoint ein Steuerelement zur Verfügung. Dieses kann wie folgt eingefügt werden.

- Einfügen –> Webpart
- Medien und Inhalt –> Seitenviewer
- Menüpfeil (rechts oben im Steuerelement) –> Webpart bearbeiten
- Name, URL und Layout anpassen

Das Ergebnis ist in Abbildung 1 zu sehen. Die Einbindung einer Website mithilfe eines *<iframe>* ist nicht möglich, da SharePointwebseiten auf *XHTML 1.0 strict* beschränkt sind und der *Tag* deshalb nicht zulässig ist.

[1] Download OntoWiki `https://github.com/AKSW/OntoWiki/downloads/`

[2] Installationsanleitung OntoWiki `https://github.com/AKSW/OntoWiki/wiki/Setup/`

[3] Download SharePoint Foundation `http://www.microsoft.com/de-de/download/details.aspx?id=5970`

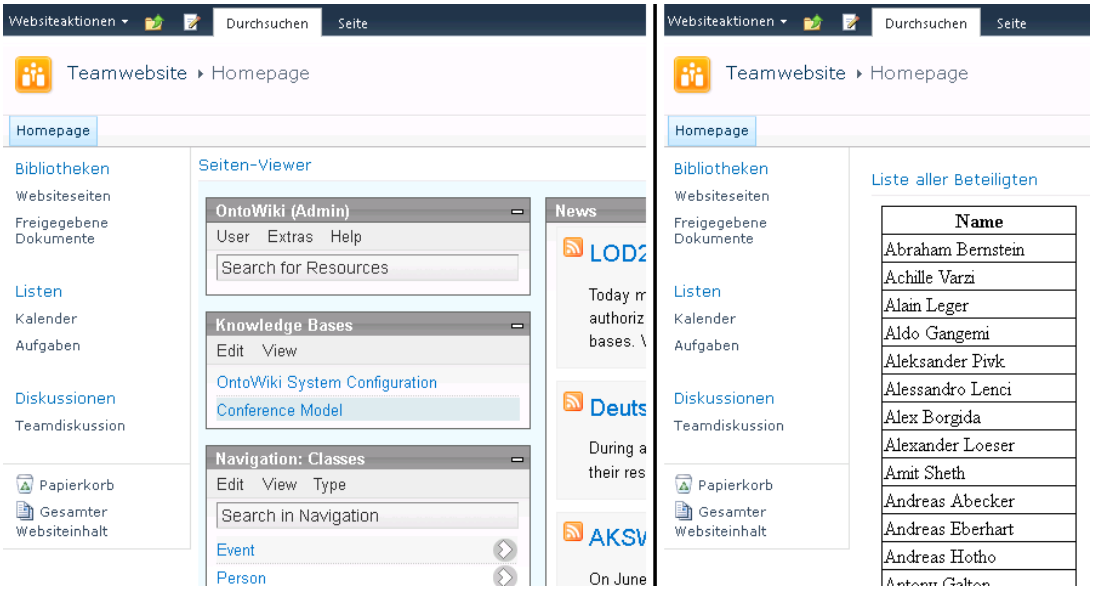


Abbildung 1: Ergebnis der Untersuchung (links: Einbettung / rechts: MySql Abfrage)

3.3 Zugriff auf die MySql Tabellen

Der MySQL ODBC Connector[4] erlaubt es, MySql Tabellen in SharePoint als Datenquellen zu nutzen. Dies geschieht allerdings mit Einschränkungen, denn es sind nur statische SQL-Befehle ohne Variablen möglich. Trotzdem können nützliche Informationen aus der Datenbank angezeigt werden.

Vorausgesetzt die Wissenbasis aus dem verlinkten Einführungs-Tutorial[5] wurde importiert, ist es möglich, mit dem folgenden SQL Befehl die Liste aller Teilnehmer einer bestimmten Konferenz anzuzeigen.

```
1 Select 'o' as 'Name' from 'ef_stmt' where 'p' = 'http://www
    .w3.org/2000/01/rdf-schema#label' AND 's' = ANY (Select
    'o' from 'ef_stmt' where 'p' = 'http://3ba.se/
    conferences/pcMember')
```

Werden die Schritte zur Erstellung einer MySql Abfrage entsprechend der verlinkten Anleitung[6] durchgeführt, wird eine Tabelle mit allen Teilnehmern angezeigt. Das Ergebnis ist in Abbildung 1 zu sehen. Die MySql Abfrage wird jedes mal ausgeführt, wenn die Seite aufgerufen wird.

[4] Download MySQL ODBC Connector http://www.mysql.de/downloads/connector/odbc/
[5] OntoWiki - First Steps http://ontowiki.net/Projects/OntoWiki/FirstSteps/
[6] Tutorial - MySql Daten in SharePoint http://www.wind-soft.de/blog/ecm/entry/mysql-daten-in-SharePoint-anzeigen/

3.4 Plugin

Um ein Webpart Plugin für SharePoint zu entwickeln, ist ein SharePoint Server notwendig. Es ist nicht geplant, diesen im Rahmen des Seminars zu installieren. Der Vollständigkeit halber soll diese Möglichkeit aber erwähnt werden.

4 Fazit

Die Integration von OntoWiki in SharePoint Foundation ist nur eingeschränkt möglich. Zwar können mit SQL Befehlen Daten aus OntoWiki anzeigt werden, aber ein Datum im richtigen Format hinzuzufügen, ist vermutlich nur mit erheblichen Aufwand machbar.

Würde OntoWiki einen Microsoft SQL Server zur Datenspeicherung nutzen, könnte dieser ohne Plugin direkt als Datenquelle angebunden und die Tabellen direkt bearbeitet werden. Der Nutzen davon ist aber fraglich, da Endanwender die Daten wahrscheinlich nur schwer korrekt bearbeiten können.

Die beste Möglichkeit der Integration von Ontowiki in Microsoft SharePoint Foundation scheint die Einbettung zu sein, weil nur so alle Daten von Ontowiki betrachtet und bearbeitet werden können.

5 Ausblick

Es ist möglich mit Hilfe von Microsoft Visual Studio und dem kostenpflichtigen SharePoint Server, Plugins für SharePoint zu entwickeln. Mit einem solchen Plugin wäre es wahrscheinlich möglich, die von Ontowiki bereit gestellte SPARQL Schnittstelle zu nutzen. Diese Möglichkeit bedarf weiterer Untersuchung und könnte Gegenstand einer weiterführenden Arbeit sein.

Literatur

[ST10] Microsoft Office System und Servers Team. *Erste Schritte mit Microsoft SharePoint Foundation 2010*. Microsoft Corporation, http://www.microsoft.com/de-de/download/details.aspx?id=22229, November 2010.

[TFH10] Sebastian Tramp, Philipp Frischmuth und Norman Heino. OntoWiki – a Semantic Data Wiki Enabling the Collaborative Creation and (Linked Data) Publication of RDF Knowledge Bases. In Oscar Corcho und Johanna Voelker, Hrsg., *Demo Proceedings of the EKAW 2010*, October 2010.

Autorenverzeichnis

Chemnitzer Informatik-Berichte

In der Reihe der Chemnitzer Informatik-Berichte sind folgende Berichte erschienen:

CSR-06-07 Karsten Hilbert, Guido Brunnett, A Texture-Based Appearance Preserving Level of Detail Algorithm for Real-time Rendering of High Quality Images, August 2006, Chemnitz

CSR-06-08 David Brunner, Guido Brunnett, Robin Strand, A High-Perpormance Parallel Thinning Approach Using a Non-Cubic Grid Structure, September 2006, Chemnitz

CSR-06-09 El-Ashry, Peter Köchel, Sebastian Schüler, On Models and Solutions for the Allocation of Transportation Resources in Hub-and-Spoke Systems, September 2006, Chemnitz

CSR-06-10 Raphael Kunis, Gudula Rünger, Michael Schwind, Dokumentenmanagement für Verwaltungsvorgänge im E-Government, Oktober 2006, Chemnitz

CSR-06-11 Daniel Beer, Jörg Dümmler, Gudula Rünger, Transformation ereignisgesteuerter Prozeßketten in Workflowbeschreibungen im XPDL-Format, Oktober 2006, Chemnitz

CSR-07-01 David Brunner, Guido Brunnett, High Quality Force Field Approximation in Linear Time and its Application to Skeletonization, April 2007, Chemnitz

CSR-07-02 Torsten Hoefler, Torsten Mehlan, Wolfgang Rehm (Eds.), Kommunikation in Clusterrechnern und Clusterverbundsystemen, Tagungsband zum 2. Workshop, Februar 2007, Chemnitz

CSR-07-03 Matthias Vodel, Mirko Caspar, Wolfram Hardt, Energy-Balanced Cooperative Routing Approach for Radio Standard Spanning Mobile Ad Hoc Networks, Oktober 2007, Chemnitz

CSR-07-04 Matthias Vodel, Mirko Caspar, Wolfram Hardt, A Concept for Radio Standard Spanning Communication in Mobile Ad Hoc Networks, Oktober 2007, Chemnitz

CSR-07-05 Raphael Kunis, Gudula Rünger, RAfEG: Referenz-Systemarchitektur und prototypische Umsetzung - Ausschnitt aus dem Abschlussbericht zum Projekt "Referenzarchitektur für E-Government" (RAfEG) -, Dezember 2007, Chemnitz

CSR-08-01 Johannes Steinmüller, Holger Langner, Marc Ritter, Jens Zeidler (Hrsg.), 15 Jahre Künstliche Intelligenz an der TU Chemnitz, April 2008, Chemnitz

CSR-08-02 Petr Kroha, José Emilio Labra Gayo, Using Semantic Web Technology in Requirements Specifications, November 2008, Chemnitz

CSR-09-01 Amin Coja-Oghlan, Andreas Goerdt, André Lanka, Spectral Partitioning of Random Graphs with Given Expected Degrees - Detailed Version, Januar 2009, Chemnitz

Chemnitzer Informatik-Berichte

CSR-09-02 Enrico Kienel, Guido Brunnett, GPU-Accelerated Contour Extraction on Large Images Using Snakes, Februar 2009, Chemnitz

CSR-09-03 Peter Köchel, Simulation Optimisation: Approaches, Examples, and Experiences, März 2009, Chemnitz

CSR-09-04 Maximilian Eibl, Jens Kürsten, Marc Ritter (Hrsg.), Workshop Audiovisuelle Medien: WAM 2009, Juni 2009, Chemnitz

CSR-09-05 Christian Hörr, Elisabeth Lindinger, Guido Brunnett, Considerations on Technical Sketch Generation from 3D Scanned Cultural Heritage, September 2009, Chemnitz

CSR-09-06 Christian Hörr, Elisabeth Lindinger, Guido Brunnett, New Paradigms for Automated Classification of Pottery, September 2009, Chemnitz

CSR-10-01 Maximilian Eibl, Jens Kürsten, Robert Knauf, Marc Ritter , Workshop Audiovisuelle Medien, Mai 2010, Chemnitz

CSR-10-02 Thomas Reichel, Gudula Rünger, Daniel Steger, Haibin Xu, IT-Unterstützung zur energiesensitiven Produktentwicklung,
Juli 2010, Chemnitz

CSR-10-03 Björn Krellner, Thomas Reichel, Gudula Rünger, Marvin Ferber, Sascha Hunold, Thomas Rauber, Jürgen Berndt, Ingo Nobbers, Transformation monolithischer Business-Softwaresysteme in verteilte, workflowbasierte Client-Server-Architekturen, Juli 2010, Chemnitz

CSR-10-04 Björn Krellner, Gudula Rünger, Daniel Steger, Anforderungen an ein Datenmodell für energiesensitive Prozessketten von Powertrain-Komponenten, Juli 2010, Chemnitz

CSR-11-01 David Brunner, Guido Brunnett, Closing feature regions, März 2011, Chemnitz

CSR-11-02 Tom Kühnert, David Brunner, Guido Brunnett, Betrachtungen zur Skelettextraktion umformtechnischer Bauteile, März 2011, Chemnitz

CSR-11-03 Uranchimeg Tudevdagva, Wolfram Hardt, A new evaluation model for eLearning programs, Dezember 2011, Chemnitz

CSR-12-01 Studentensymposium Informatik Chemnitz 2012, Tagungsband zum 1. Studentensymposium Chemnitz vom 4. Juli 2012, Juni 2012, Chemnitz