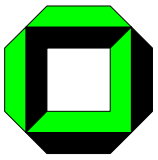

Universität Karlsruhe
Fakultät für Informatik
Institut für Telematik
76128 Karlsruhe

Netzwerk-Management und Hochleistungs-Kommunikation

Teil XXI

Seminar WS 1999/2000

Herausgeber:
Meng Gan
Hartmut Ritter
Dr. Jochen Schiller

Universität Karlsruhe (TH)
Institut für Telematik
http://www.telematik.informatik.uni-karlsruhe.de/

Interner Bericht 2000-3
ISSN 1432–7864

Der vorliegende Interne Bericht enthält die Beiträge zum Seminar „Netzwerk-Management und Hochleistungs-Kommunikation“, das im Wintersemester 1999/2000 zum einundzwanzigsten (!) Mal stattgefunden hat.

Die Themenauswahl kann grob in folgende vier Blöcke gegliedert werden:

Ein großer Block ist dieses Mal der Mobilkommunikation gewidmet, die auf fast allen Schichten des ISO-OSI-Referenzmodelles Änderungen erfordert. Betrachtet werden die Grundlagen für drahtlose lokale Netzwerke, Erweiterungen von TCP für die Mobilkommunikation und mobile Ad-hoc-Netzwerke. Die sogenannte dritte Generation des Mobilfunks ist ein weiteres Thema.

Die Grundlagen für Mobilität und hohe Datenraten auf Seiten des Endsytems werden im zweiten Block betrachtet, in dem Betriebssysteme für kleine, insbesondere mobile Geräte beschrieben werden und neuartige Architekturen des Netzwerk-Subsytems klassischer Endsysteme oder Server vorgestellt werden.

Der dritte Block ist dem Bereich Netzwerk-Management gewidmet, allerdings dieses Mal mit nur einem Beitrag über Netzwerküberwachung mit RMON.

Die Frage der Dienstintegration und Konvergenz der Netze wird in den beiden Beiträgen des letzten Blocks beschrieben: Computer-Telefon-Integration vereint bisher noch sehr stark voneinander getrennte Netze, „Konvergenz oder Divergenz?“ ist die Frage angesichts der Trends im Netzwerkbereich, die der letzte Beitrag stellt.

Inhaltsverzeichnis

Vorwort

Das Seminar „Netzwerk-Management und Hochleistungs-Kommunikation“ am Institut für Telematik erfreut sich weiterhin großer Beliebtheit. Die Telematik als Verbindung von Telekommunikation und Informatik entfaltet immer mehr von ihrer Dynamik. Dies zeigt sich an der breiten öffentlichen Diskussion über die zukünftige Bedeutung des Internet, das ja schon lange dem akademischen Bereich entwachsen ist, ebenso wie an der wachsenden Bedeutung der Mobilkommunikation.

Die Verbindung von Mobilkommunikation und Internet, die auch unter dem Schlagwort *fixed-mobile-convergence* diskutiert wird, strahlt in viele Bereiche aus, die zunächst nicht betroffen zu sein scheinen. Zukünftige mobile Systeme erfordern Änderungen beispielsweise ebenso auf der Ebene der Transportprotokolle wie im Bereich der Betriebssystemarchitektur. Betriebssysteme für mobile Kommunikation stehen vor der nicht unbedingt neuen, aber jetzt viel stärker geforderten Aufgabe, geringen Ressourcenbedarf (z.B. Speicher, Energie) mit Anpassungsfähigkeit (Adaptivität) zu verbinden.

Auf der anderen Seite darf das Kernnetz, der zweite Teil der Verbindung von Mobilkommunikation und Internet, nicht vergessen werden. Die Last, die durch mobile Clients generiert wird, konzentriert sich beim typischen Web-Verkehr auf Server im Festnetz, die leicht zum Engpass bei der von den Endkunden wahrgenommenen Antwortzeit auf Anfragen werden können. Dazu kommen Probleme der Verwaltbarkeit dieser Netze, die nur noch durch Dezentralisierung zu lösen ist. Die Integration der klassischen Sprach- und Datennetze und die nicht zuletzt daraus resultierende Konvergenz verschiedener Netze wird die Telematik in Zukunft stark beschäftigen und voranbringen.

Jetzt liegt auch der nunmehr 21. Seminarband als Interner Bericht vor. Durch die engagierte Mitarbeit der beteiligten Studenten konnte so zumindest ein Ausschnitt aus dem komplexen und umfassenden Themengebiet klar und übersichtlich präsentiert werden. Für den Fleiß und das Engagement der Seminaristinnen und Seminaristen sei daher an dieser Stelle recht herzlich gedankt.

Die weiterhin gute Resonanz bei den Studenten bestätigt uns darin, auch im kommenden Sommersemester 2000 ein derartiges Seminar – natürlich mit geändertem aktuellem Inhalt – durchzuführen, so dass bald ein weiterer Interner Bericht mit neuen Forschungsergebnissen aus innovativen Seminarbeiträgen erscheinen wird. Doch vorerst sollen im vorliegenden Band folgende Themengebiete vorgestellt werden:

Wireless LANs

Drahtlose lokale Netze (Wireless LANs) kommen dem Wunsch nach drahtloser, mobiler Datenkommunikation nach. Der Beitrag beschreibt den in diesem Bereich relevanten Standard IEEE 802.11 und diskutiert Probleme auf der Netzwerkschicht, die durch den nun möglichen schnellen Wechsel eines Rechners zwischen verschiedenen Subnetzen auftreten.

Mobile TCP - Erweiterung und Verbesserung von TCP in mobilen Umgebungen

Das Transportprotokoll TCP wurde für Festnetze konzipiert und optimiert. In Mobilfunkverbindungen treten neue Probleme wie vermehrt auftretende, stark korrelierte Bitfehler auf der phyisikalischen Schicht und kurzzeitige vollständige Aussetzer der Netzwerkverbindung auf, mit denen die Mechanismen von TCP nicht umgehen können. Dadurch wird die Leistungsfähigkeit von TCP stark eingeschränkt. Der Beitrag stellt verschiedene Erweiterungen und Verbesserungen von TCP vor, um es tauglich für die mobile Kommunikation zu machen.

Mobile Ad-hoc Netzwerke

Mobile Ad-hoc Netzwerke sind Netzwerke, in denen spontan, d.h. kurzfristig, verschiedene mobile Geräte miteinander kommunizieren. Wesentlich dabei ist das Fehlen einer festen Infrastruktur, auf die beispielsweise ein Mobilfunksystem wie GSM zugreift. Der Beitrag diskutiert verschiedene Realisierungsmöglichkeiten (Bluetooth, Manet) und auftretende Problem (Routing, Sicherheit) bei Mobilen Ad-hoc Netzwerken.

3GPP - Mobilfunk der 3. Generation

Der Mobilfunk, der z.B. in Europa mit GSM extreme Wachstumsraten zu verzeichnen hat, steht vor dem Übergang zur Integration von Sprach- und Datendiensten in ein universelles Mobilfunksystem. Unter dem Namen IMT-2000 (International Mobile Telecommunication) sollen Systeme der sogenannten dritten Generation eingeführt werden, die diese Integration weltweit ermöglichen und vereinheitlichen. Der Beitrag stellt verschiedene Aspekte dieser neuen Generation vor.

Betriebssysteme mit kleinem footprint

Die Bedeutung von mobiler Kommunikation einerseits und eingebetteten Systemen andererseits rückt den Aspekt des Betriebssystems wieder in den Vordergrund. Neue Betriebssysteme mit geringem Speicherbedarf, dem sogenannten *footprint*, werden entwickelt, von denen eine hohe Konfigurierbarkeit gefordert wird, die aber auch selber anpassungsfähig an Ressourcenschwankungen sein müssen. Der Beitrag vergleicht verschiedene unter diesen Aspekten neu entwickelte Betriebssysteme, die den Anspruch erheben, besonders für mobile Geräte geeignet zu sein.

Weg mit dem Flaschenhals im Rechner

Die treibende Kraft der Entwicklung des Internet ist derzeit der WWW-Verkehr, der auf dem klassischen Client-Server-Prinzip basiert. Um dem Endnutzer mit seinem festen oder mobilen Client eine gewisse Qualität des WWW-Dienstes bieten zu können, sind also neben schnellen Zugangs- und Kernnetzen auch effiziente Architekturen auf der Seite der Server notwendig. Hier wird vor allem das Netzwerk-Subsystem oft zum Flaschenhals, der die vom Server gesendete Datenrate drosselt. In diesem Beitrag wird daher auf verbesserte Architekturen eingegangen, die diesen Engpass öffnen.

RMON und RMON2 - effizientes “remote network monitoring”

Netzwerkmanagement gewinnt angesichts weiter wachsender Kernnetze und den steigenden Anforderungen an die zeitliche Verfügbarkeit von Internet-Verbindungen weiter an Bedeutung. Dabei besteht die Gefahr, dass mit steigender Ausdehnung und Vermaschung des Netzes die Verwaltbarkeit sinkt. Dieser Komplexität begegnen die in diesem Beitrag vorgestellten Ansätze, die dezentralisiertes Netzwerkmanagement vorschlagen, um Probleme nah an dem Ort des Entstehens überwachen und vermeiden zu können.

Computer-Telefon-Integration (CTI)

Die Computer-Telefon-Integration ist ein wichtiges Beispiel für die beginnende Verschmelzung von ehemals reinen Sprach- und Datendiensten. Die Computer-Telefon-Integration, wie sie typischerweise in Call-Centern eingesetzt wird, beschränkt sich dabei nicht mehr nur auf das lokale Netz, sondern zielt auch darauf ab, über das WWW Dienste wie *Voice-over-IP* (Sprache über IP-Netze) anzubieten. Der Beitrag stellt verschiedene Architekturen der CTI vor.

Konvergenz der Netze

Die Konvergenz der Netze ist ein übergreifendes Thema aus dem Bereich der Telematik, die die alte Frage aufgreift, ob ein einheitliches, diensteintegrierendes Netzwerk denkbar und umsetzbar ist. Mit dem Scheitern von ATM als allumfassendes Netzwerk scheinen jetzt IP-Netzwerke diesen Anspruch aufzugreifen, allerdings sind dort noch einige Probleme auf dem Weg zum umfassenden, dienstgüteunterstützenden Netzwerk zu lösen. Es muss daher offen bleiben, ob IP-Netzwerke tatsächlich die Netzwerke sind, auf die die Entwicklung konvergiert. Der Beitrag diskutiert einige Gründe und Voraussetzungen für eine Konvergenz der Netze.

Wireless LANs

Diana Kaufmann

Kurzfassung

Wireless LANs gewinnen heutzutage immer mehr an Bedeutung und Nutzen. Nach einer kurzen Einführung in die drahtlosen lokalen Netzwerke sollen grundlegende Themen wie die Netzwerkarchitektur und der IEEE-Standard 802.11 der Wireless LANs vorgestellt sowie anhand eines Beispiels der praxisnahe Einsatz gezeigt werden. Auch spezielle Mechanismen werden beschrieben wie Handover, der die Mobilität besonders gut zum Ausdruck bringt, das DHCP, durch welches sich einiges an Konfigurationen ersparen läßt, und Mobile IP, die die jederzeitige Erreichbarkeit, welche heutzutage unbedingt notwendig ist, ermöglicht. Man wird sehen, daß die drahtlosen Netze in naher Zukunft immer mehr die herkömmlichen Festnetze ersetzen werden.

1 Einleitung

Die Funktechnik findet in der heutigen Zeit immer mehr Verbreitung: Nicht nur moderne Hochleistungsnetzwerke werden drahtlos betrieben, sondern auch immer häufiger lokale Netze. Das ist verständlich, denn diese kabellose Technik bringt viele Vorteile mit sich:

Man ist sehr flexibel, wenn es darum geht, kurzfristig ein Netzwerk zu erstellen, das eventuell hinterher wieder abgebaut werden soll. Ebenso muß man zum Beispiel bei der Planung eines Hauses nicht berücksichtigen, welche Räume miteinander vernetzt werden sollen, sondern kann dies auch noch nachträglich festlegen. Eine Erleichterung sind drahtlose Netze vor allem in historischen Gebäuden, in denen am Mauerwerk nichts verändert werden darf: Bei der Verkabelung von Endgeräten hätte man die Qual der Unterbringung der Kabel in Kabelschächten, was man sich durch ein Funknetz ersparen kann. Auch für den Einsatz bei Naturkatastrophen sind Funkverbindungen sehr hilfreich: Wenn beispielsweise bei einem Erdbeben verschiedene Stationen voneinander getrennt werden, die aber unbedingt miteinander kommunizieren müssen, könnte ein drahtloses Netz dem Ganzen standhalten, während eine verkabelte Infrastruktur zusammenbrechen würde.

Jedoch sollte man auch die Nachteile von drahtlosen Netzwerken nicht übersehen: Die Funkstrecken sind bei Weitem nicht so belastbar wie gegenwärtige Kabelverbindungen. Das bedeutet, daß die Übertragungsrate geringer ist und mehr Übertragungsfehler zustandekommen, wodurch die Dienstgüte eingeschränkt wird. Die Hardware für drahtlose Netze ist zur Zeit noch um einiges teurer als die bisherige Hardware für Kabelnetze; es sind internationale Regelungen notwendig, die weltweit bestimmte Frequenzbereiche zur Nutzung von lokalen Netzwerken freigeben; die Funkstrahlen rufen oft Störungen von anderen elektrischen Geräten hervor. Dadurch, daß die Datenpakete das Medium „Luft“ benutzen, die „ jedermann frei zugänglich“ ist, wird das Risiko gegenüber Lauschangriffen höher. Deswegen müssen zusätzliche Sicherheitsvorkehrungen getroffen werden. Außerdem ist bis heute umstritten, inwiefern sich die elektromagnetischen Wellen auf die Gesundheit des Menschen auswirken.

2 Grundlagen von Wireless LANs

Um aber - die Nachteile einmal außer Betracht lassend - Nutzen aus den Vorteilen ziehen zu können, wird der Standard der „Wireless LANs“, der drahtlosen lokalen Netzwerke also, ständig weiterentwickelt.

Dabei gilt es zuerst einmal, die Verwendung von „drahtlos“ und „mobil“ zu unterscheiden: Eine drahtlose Verbindung besteht aus einer Luftschnittstelle, über die per Funk oder Infrarot kommuniziert wird. Der Begriff „mobil“ besagt, daß das entsprechende Endgerät leicht zu transportieren ist. Zum Beispiel kann ein Desktop-Computer durchaus drahtlos vernetzt sein, wenn er mit der notwendigen Hardware ausgestattet ist. Ein durch eine Steckverbindung an ein Ethernet angeschlossener Laptop ist dagegen mobil, aber drahtgebunden.

Weiterhin gibt es Funk- und Infrarot-Schnittstellen, von denen beide ihre Vor- und Nachteile haben: Infrarot-Schnittstellen sind einfach und billig herzustellen und mittlerweile schon sehr oft in mobile Geräte eingebaut. Zur Übertragung zwischen Infrarot-Sender und -Empfänger muß jedoch Sichtkontakt herrschen. Die Funkübertragung hat auch durch Wände hindurch noch eine Reichweite von einigen Metern, und es steht mehr Bandbreite zur Verfügung als bei der Infrarot-Übertragung. Wie schon anfangs erwähnt, können allerdings durch die elektromagnetischen Wellen andere elektrische Geräte gestört werden. Deshalb muß man abwägen, welche Übertragungsart jeweils besser geeignet ist; hier soll aber die Funkübertragung in den Vordergrund gerückt werden, weil es unmöglich ist, in drahtlosen lokalen Netzen immer Sichtkontakt aufrecht zu erhalten.

Grundsätzlich unterscheidet man bei den drahtlosen Netzen zwischen einem Infrastruktur- und einem Ad-hoc-Netzwerk.

Das Infrastruktur-Netzwerk zeichnet sich dadurch aus, daß die Kommunikation zweier mobiler Endsysteme mittels einer Basisstation erfolgt, die den Access Point (=Zugangspunkt, AP) für die mobilen Endgeräte darstellt. Sie ist über Kabel mit dem Festnetz verbunden und leitet die Datenpakete von einem mobilen Endgerät sowohl zu anderen drahtlosen Teilnehmern als auch in das Festnetz weiter und umgekehrt.

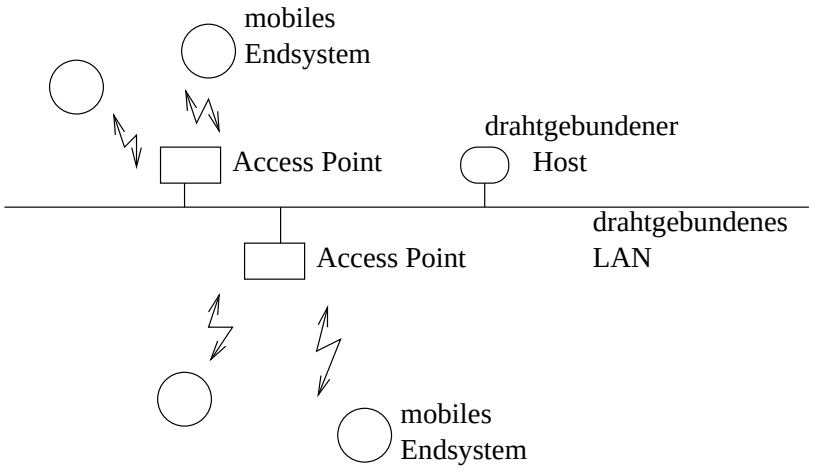

Abbildung 1: Infrastruktur-Netzwerk

Als Ad-hoc-Netzwerk bezeichnet man zwei oder mehrere mobile Endsysteme, die direkt, also ohne Basisstation, miteinander kommunizieren. Diese müssen dabei in gegenseitiger Reichweite liegen oder ihre Daten von anderen mobilen Teilnehmern weiterleiten lassen. Diese Endsysteme, die alle die gleiche Frequenz benutzen, werden als eine Zelle bezeichnet.

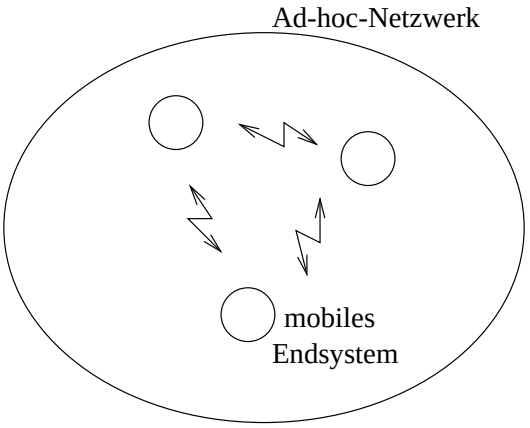

Abbildung 2: Ad-hoc-Netzwerk

2.1 Netzarchitektur

Der Anschluß an das Festnetz erfolgt, wie bereits in Abschnitt 2 erwähnt, durch Access Points, die zunächst die Datenpakete in ein Distribution System (DS) weiterleiten. Ein solches Distribution System kann mehrere Access Points miteinander verbinden und kommuniziert schließlich über ein Portal mit dem Festnetz, wobei Access Points, Distribution System und Portal als Brücke fungieren. Die mobilen Endsysteme, auch Stationen genannt, die Zugriff auf einen gemeinsamen Access Point haben, bezeichnet man als ein Basic Service Set (BSS). Zusammen mit allen anderen Basic Service Sets und dem Distribution System bilden sie ein Extended Service Set (ESS).

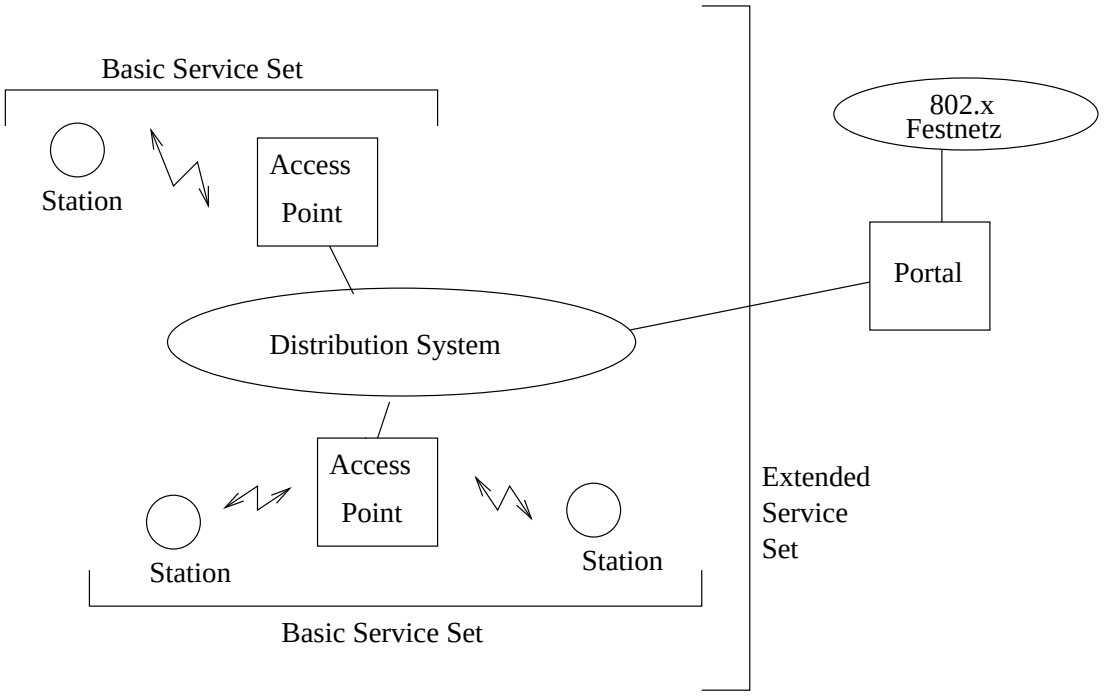

Abbildung 3: Netzarchitektur

2.2 Der Standard IEEE 802.11

Wie die anderen IEEE-Standards 802.x legt auch der Standard IEEE 802.11 für Wireless LANs den Zugriff auf Schicht 1 und 2 des ISO/OSI-Schichtenmodells fest. Bild 4 zeigt, wie ein drahtloses Netz mittels einer Brücke an ein Festnetz angebunden wird.

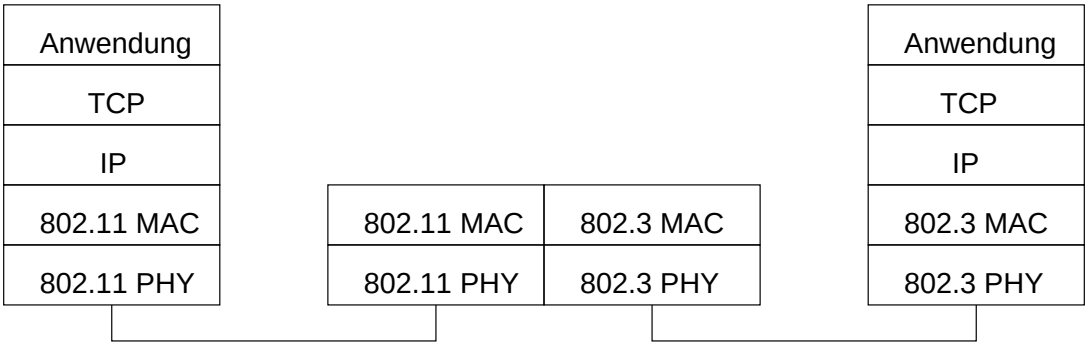

Abbildung 4: Anbindung eines drahtlosen Netzes an ein Festnetz

Die MAC-Schicht, also Schicht 2a und somit ein Teil der Sicherungsschicht, hat in IEEE 802.11 die zusätzliche Funktion der Verschlüsselung von Daten, was bei der Luftschnittstelle sehr wichtig ist. Weiterhin stellt sie die Option einer Fragmentierung bereit. Diese Aufteilung eines großen in mehrere kleine Pakete verringert bei der Übertragung die Fehlerwahrscheinlichkeit, die umso kleiner wird, je kleiner auch die Datenpakete sind. Außerdem sorgt die MAC-Schicht für die Synchronisation der mobilen Endgeräte, indem sie regelmäßig sogenannte „Beacon"-Signale versendet. Synchronisation bedeutet hierbei sowohl das Abgleichen der Uhren als auch das Auffinden eines Netzes.

Der IEEE-Standard 802.11 stellt verschiedene Varianten der physikalischen Schicht zur Verfügung: 2 basieren auf Funkübertragung im lizenzfreien 2,4 GHz-Band und eine auf Infrarotübertragung im 850 - 950 nm-Bereich. Übertragen werden wahlweise 1 oder 2 MBit/s. Das wird realisiert, indem der Header eines Datenpaketes mit 1 MBit/s gesendet wird und, falls die empfangende Station eine höhere Übertragungsrate zuläßt, auch mit 2 MBit/s fortgefahren werden kann. Nach neueren Standards, die gerade in Entwicklung sind, sollen bei IEEE 802.11a mit 5 GHz allerdings auch 20 MBit/s und bei IEEE 802.11b, wiederum mit 2,4 GHz, 10 MBit/s zulässig sein.

2.3 Bandspreizverfahren

Da Funkübertragungen generell sehr störanfällig sind und vor allem Signale, die nur in einem schmalen Frequenzbereich übertragen werden, schnell ausgelöscht werden können, wurden sogenannte Bandspreizverfahren eingeführt, die das Signal über einen größeren Signalbereich „spreizen": Das ursprüngliche Signal, das nur eine geringe Bandbreite in Anspruch nimmt, wird zum Beispiel mit einer Zufallszahlenfolge, welche „chipping sequence" genannt wird, verknüpft. Dadurch reicht das Signal in dem Beispiel aus Abbildung 5 über eine siebenfache Bandbreite des Ausgangssignals und ist somit weniger störanfällig.

Hierbei entsteht jedoch das Problem, daß Signale, die vorher ihre zugeteilte Bandbreite hatten, sich durch das Spreizen jetzt überlappen. Um dies zu verhindern, gibt es zwei Möglichkeiten:

- Beim „Frequency Hopping Spread Spectrum" (FHSS) wechseln die Sender nach einer bestimmten Zeit jeweils ihre Frequenzen, wobei keine zwei Sender die gleiche Folge von Frequenzwechseln haben dürfen.

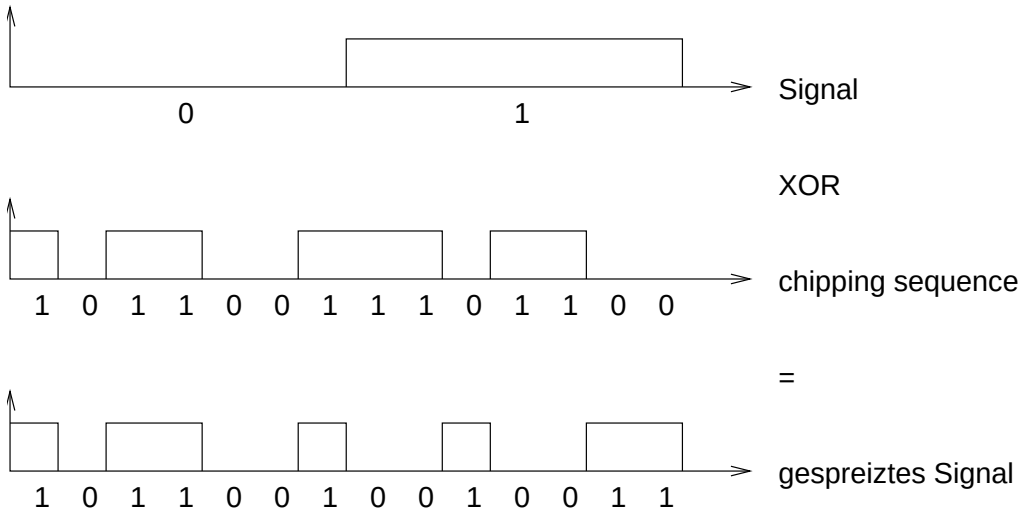

Abbildung 5: Bandspreizen durch eine Zufallszahlenfolge

- Beim „Direct Sequence Spread Spectrum“ (DSSS) verknüpfen die Sender das Signal mit einer gegebenen Zufallszahlenfolge, wobei ebenfalls sichergestellt werden muß, daß keine zwei Sender die gleich Zahlenfolge besitzen.

2.4 Beispiel: WaveLAN-II

Die Firma Lucent Technology, die sich seit längerer Zeit mit der Herstellung von Hardware für drahtlose Netze beschäftigt, brachte 1997 ihr neues Produkt „WaveLAN-II“ auf den Markt. Dieses arbeitet nach dem IEEE 802.11-Standard mit 2 MBit/s auf dem 2,4 GHz-Band; durch geschickte Codierung kann aber auch eine Übertragungsleistung von 10 MBit/s erreicht werden. WaveLAN-II benutzt in den USA 11 und in Europa 13 unterschiedliche Kanäle. Ein entscheidender Vorteil dabei ist, daß die für WaveLAN-II bereitgestellten Access Points auf mehreren Kanälen arbeiten können, so daß mehrere Funkzellen installiert werden können, die sich überlappen. Somit kann auch die Bandbreite besser ausgenutzt werden.

WaveLAN-II wurde am Institut für Telematik erfolgreich installiert und getestet [DoGR99]. Allerdings weisen die Messungen mit der etwas neueren WaveLAN-Turbo-Karte, die explizit auf 10 MBit/s ausgelegt ist, eine Übertragungsrate von nur ca. 3,5 MBit/s auf, während ein normales 10 MBit/s-Ethernet mit der 10 MByte-Testdatei immerhin 7 MBit/s zustande brachte. Daran kann man sehen, daß die Funktechnik zur Zeit doch noch einiger Verbesserung bedarf.

3 Handover in mobilen Netzen

Bisher wurde davon ausgegangen, daß ein mobiles Endsystem immer in Reichweite eines Access Points ist, über den es kommunizieren kann. Gerade aber bei tragbaren Notebooks etc. ist es wünschenswert, sich von einem Ort zum anderen bewegen zu können (=„Roaming“) und somit auch während der Datenübertragung den Access Point zu wechseln, was als „Handover“ bezeichnet wird. Das Handover läuft folgendermaßen ab:

Wenn eine Endstation bemerkt, daß die Übertragungsqualität einen bestimmten Schwellwert unterschreitet, beginnt sie, den Luftraum nach anderen Access Points abzusuchen. Dabei versendet sie entweder selbst Proben auf verschiedenen Kanälen (=„aktives Scanning“) oder wartet auf Beacon-Signale (siehe Abschnitt 2.2) der Access Points (=„passives Scanning“).

Aus den gefundenen Access Points wählt sich die mobile Station denjenigen mit dem stärksten Signal aus und sendet ihm ein „Reassociation Request“-Paket, woraufhin der Access Point mit einem „Reassociation Response“ antwortet. In diesem Falle war das Handover, also das Wechseln von einem Access Point zum anderen, erfolgreich; andernfalls muß das mobile Endgerät mit dem Scanning fortfahren.

Nach einem erfolgreichen Handover teilt der neue Access Point dem Distribution System die Anwesenheit des Endsystems in seinem Funkbereich mit, damit das DS seine Datenbank erneuern kann. Dieses Update ist notwendig, damit das DS die für das mobile Endsystem bestimmten Pakete korrekt an den richtigen Access Point weiterleiten kann und dem alten Access Point mitteilen kann, daß die mobile Station sich nun nicht mehr in dessen Funkbereich aufhält.

4 DHCP

Es wurde nun beschrieben, wie mobile Endsysteme innerhalb eines Distribution Systems miteinander kommunizieren und ihren Access Point wechseln. Jedoch wurde nicht besprochen, wie sie von außerhalb erreichbar sind, d.h. woher beliebige Endsysteme wissen, an welchem Access Point sie sich gerade befinden.

Normalerweise sind Endgeräte eindeutig durch ihre IP-Adresse zu identifizieren. Datenpakete werden an diese Endgeräte weitergeleitet, indem sie in das entsprechende Subnetzwerk geschickt werden, welches sie an die ihm bekannten Hosts weiterleitet.

Im Fall der mobilen Endgeräte aber ist das Problem, daß sie sich theoretisch über jeden beliebigen Access Point an ein Netzwerk anbinden können - gleich, welche IP-Adresse dieses Netzwerk hat oder welcher Klasse es angehört. Wenn das mobile Gerät also eine feste IP-Adresse hätte, müßte es irgendeinem Netzwerk angehören, welches immer über den Aufenthalt dieses Geräts informiert wäre und dessen Pakete weiterleiten würde.

Da dies schwer realisierbar ist, wird hier das DHCP (=Dynamic Host Configuration Protocol) verwendet: Auf eine DHCP-Anfrage beim DHCP-Server bekommt ein neu in ein Netzwerk eingegliederter Computer Daten wie DNS-Server, Router-Tabelle, Subnetzmaske, Domainname und IP-Adresse mitgeteilt, wobei die IP-Adresse ein Schwerpunkt des DHCP ist und auch in diesem Fall besonders betrachtet werden soll. Funktional läuft das Ganze folgendermaßen ab:

Ein Client (also in diesem Fall das mobile Endsystem) stellt eine DHCPDISCOVER-Anfrage mittels Broadcast im Subnetz an die erreichbaren Server, auf das diese mit einem DHCPOFFER antworten. Der Client vergleicht die von den Servern „angebotenen“ Konfigurationen und wählt sich eine aus. Der Server, von dem diese ausgewählte Konfiguration stammt, bekommt vom Client ein DHCPREQUEST mit dem Parameter „options“. Alle anderen Server erhalten den gleichen Request, allerdings mit dem Parameter „reject“, was ihnen anzeigt, daß diese Konfiguration wieder frei ist und für andere Clients verwendet werden kann. Der ausgewählte Server bestätigt die Anfrage des Clients mit einem DHCPACK bzw. lehnt sie mit einem DHCPNACK ab. Im Falle einer Bestätigung ist die Initialisierung erfolgreich abgeschlossen und der Client hat nun eine IP-Adresse zugewiesen bekommen. Wenn ein Client fertig ist und das Netzwerk verläßt, sollte er seine Konfiguration mit einem DHCPRELEASE wieder für andere Clients freigeben.

Da die IP-Adresse dynamisch vergeben wird und irgendwann den Client wechseln wird, erhält das mobile Endsystem sie auch nur für eine gewisse Zeit, welche „lease time“ genannt wird. Jedes Mal nach Ablauf dieser lease time muß der Client seine Konfiguration inklusive der IP-Adresse mittels eines erneuten DHCPREQUEST verlängern oder sie abgeben. Dies

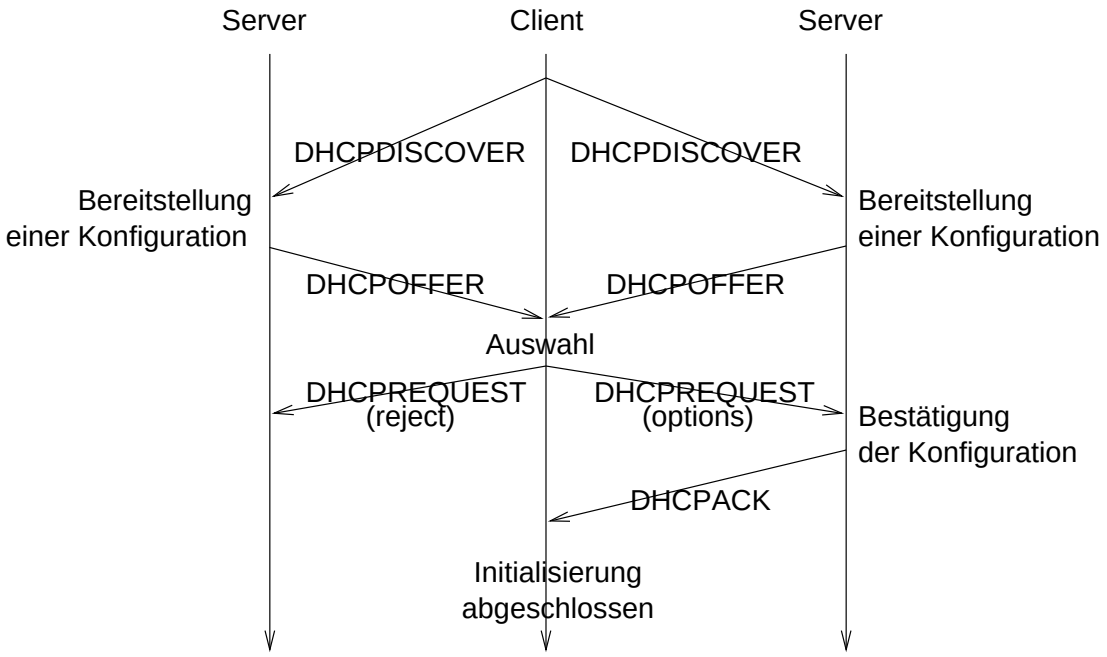

Abbildung 6: Initialisierung mittels DHCP

dient dazu, daß nicht mehr benutzte IP-Adressen - entstanden dadurch etwa, daß der Funkkontakt zu einer Station zusammengebrochen ist oder diese sich nicht mit DHCPRELEASE abgemeldet hat - wieder an andere Clients vergeben werden können.

Es sei darauf hingewiesen, daß das DHCP bezüglich der Sicherheit den Nachteil besitzt, daß jeder beliebige Client sich von einem DHCP-Server eine IP-Adresse anfordern kann und sowohl der Client dem Server als auch umgekehrt der Server dem Client blindes Vertrauen entgegenbringen muß.

Mit Hilfe des DHCP-Protokolls weiß also der DHCP-Server immer, an welchem Access Point die mobile Endstation sich aufhält und welche IP-Adresse sie besitzt, so daß die ankommenden Pakete korrekt weitergeleitet werden können.

5 Mobile IP

DHCP funktioniert aber nur, solange sich die mobile Station im eigenen Subnetz aufhält: Die Pakete in diesem Netzwerk gelangen zu dem jeweiligen DHCP-Server, der über die momentane IP-Adresse Bescheid weiß und die Pakete weiterleiten kann.

Nun kann es aber vorkommen, daß man beispielsweise unterwegs ist und sich mit seinem mobilen Endgerät über irgendein Fremdnetz Zugang zum Internet verschaffen möchte. In diesem Fall reicht ein DHCP-Server nicht mehr aus, denn angenommen, das mobile Endgerät bekommt dadurch eine kurzfristige IP-Adresse. Dann ist das eine Adresse aus dem jeweiligen Fremdnetz, in dem sich der DHCP-Server befindet. Diese neue IP-Adresse ist aber niemandem bekannt, man kann also auch keine Daten an die mobile Station schicken, weil man nicht weiß, wie man sie adressieren muß. Und falls es möglich sein sollte, dem Endgerät eine Adresse aus dessen eigenem Subnetz zuzuweisen, stellt sich das Problem, daß die Router die Datenpakete an dieses Subnetz weiterleiten, es dann aber verwerfen, weil es dort zu dieser Zeit keinen Host mit der entsprechenden IP-Adresse gibt.

Eine Lösung dafür bietet Mobile IP, eine Erweiterung des Internet Protokolls IP der ISO/OSI-Schicht 3. Bei diesem Protokoll behält das mobile Endsystem stets eine fest zugewiesene

IP-Adresse bei, sein momentaner Aufenthaltsort ist aber immer registriert, wie in 5.1 gleich beschrieben werden wird, damit keine Pakete verworfen werden, weil der Host nicht erreichbar ist. Betrachtet werden nun in den folgenden Abschnitten die Eingliederung eines Hosts aus seinem Heimatnetz in das Fremdnetz und der danach mögliche Datentransfer.

5.1 Netzintegration

Jedes mobile Endsystem, das den Punkt seines Netzanschlusses unter Beibehaltung seiner IP-Adresse wechselt, wird ein „Mobile Node“ genannt. Ein Mobile Node besitzt ein Heimatnetz, zu dem es laut seiner IP-Adresse gehört. Im Heimatnetz eines jeden Mobile Nodes gibt es einen „Home Agent“, eine Einheit, die im Normalfall auf einem Router dieses Subnetzes untergebracht ist. Der Home Agent ist stets über den Aufenthaltsort des Mobile Nodes informiert und nimmt auch die Datenpakete an, die an ihn geschickt werden. Ebenso befinden sich in den Fremdnetzen, also den Netzen, in denen sich der Mobile Node zur Zeit der Kommunikation aufhält, die sogenannten „Foreign Agents“, die ebenfalls auf einem Router untergebracht sind und über alle Mobile Nodes Bescheid wissen, die sich zur Zeit in diesem Fremdnetz befinden.

Wechselt ein Mobile Node beispielsweise von seinem Heimatnetz in ein Fremdnetz, so wartet er entweder auf ein Broadcast-Paket eines Foreign Agents - ein sogenanntes „Agent Advertisement“, das sowohl Foreign Agents als auch Home Agents in regelmäßigen Abständen versenden, um ihr Dasein anzuzeigen -, oder er schickt selbst mittels Broadcasting eine Anfrage nach einem Foreign Agent ab, um einen solchen zu finden. Zur Registrierung übergibt der Mobile Node dem gefundenen Foreign Agent seine IP-Adresse und weitere verschlüsselte Informationen. Der Foreign Agent nimmt Kontakt mit dem Home Agent des Mobile Nodes auf und übermittelt ihm die IP-Adresse zur Identifizierung des Mobile Nodes und die verschlüsselten Informationen zur Verifizierung der Identität. Der Home Agent bestätigt dem Foreign Agent eine erfolgreiche Verifizierung; daraufhin kann der Home Agent, der den Aufenthaltsort des Mobile Nodes verwaltet, seine Daten aktualisieren und der Foreign Agent den neuen Mobile Node in seiner Tabelle registrieren. Die IP-Adresse des Foreign Agent wird zur „Care-of Address“ des Mobile Nodes, unter der er zusätzlich erreichbar ist. Somit ist er in das Fremdnetz eingegliedert und wieder von überall her auch unter seiner eigenen IP-Adresse ansprechbar. (Siehe auch Abbildung 7.)

5.2 Datentransfer

Schickt nun ein Sender ein Datenpaket an einen Mobile Node, so wird dieses Paket der IP-Adresse zufolge erst einmal an dessen Heimatnetz geroutet(1, Abbildung 8). Dort wird es von dem Home Agent entgegengenommen, der das Paket direkt an den Mobile Node weiterleitet, falls sich dieser gerade im Heimatnetz aufhält. Andernfalls weiß der Home Agent, wo sich der Mobile Node zur Zeit befindet, und schickt das Datenpaket folgendermaßen weiter:

Das ganze Paket wird gekapselt, d.h. es wird ein zweiter IP-Header vorne angehängt, der als Absender-Adresse die IP-Adresse des Home Agent und als Empfänger-Adresse die Care-of Address des Mobile Nodes enthält. Das ursprüngliche Datenpaket bleibt dabei unverändert und wird als Nutzlast übertragen.

Mit diesem neuen Header wird das Paket nun durch das Internet „getunnelt“ bis zum entsprechenden Subnetz, zu dem die Care-of Address gehört(2). Der Foreign Agent, der mit der Care-of Address den Tunnelendpunkt darstellt, entkapselt das Paket wieder und schickt das ursprüngliche Datenpaket direkt weiter zum Mobile Node(3). Dadurch, daß dieser das Originalpaket erhält, kann der Mobile Node daraus leicht ersehen, wer der Absender war, und Antworten auf direktem Wege zu diesem zurückschicken(4).

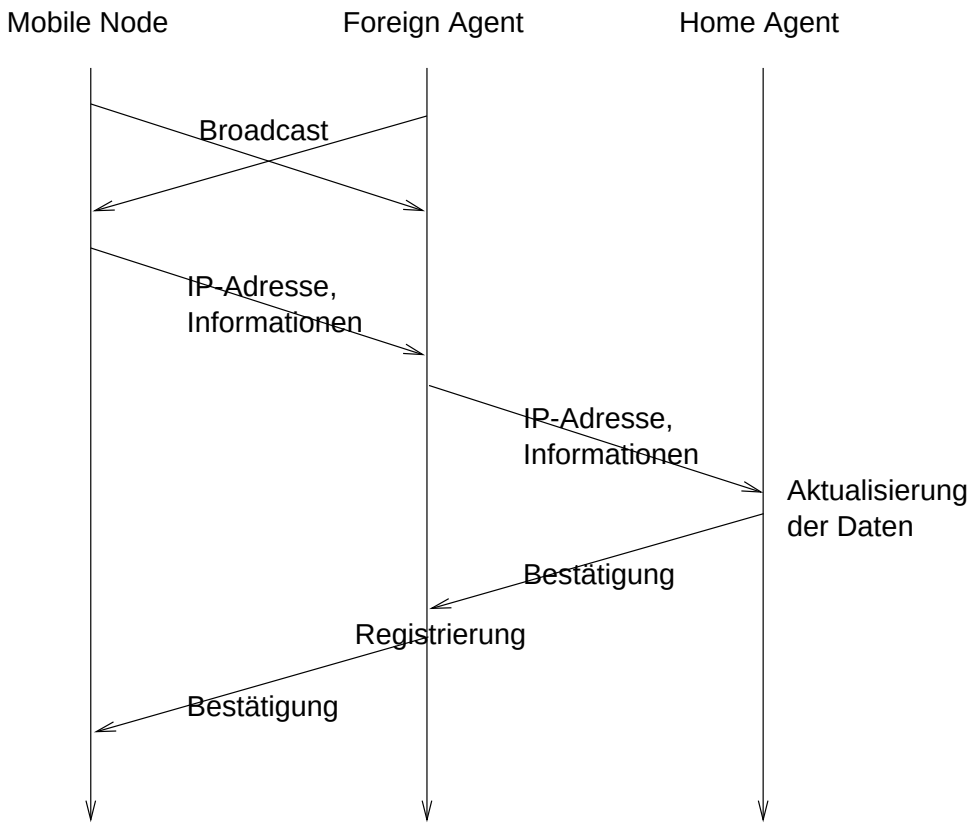

Abbildung 7: Netzintegration eines Mobile Nodes

Der Home Agent teilt dem Absender unterdessen die Care-of Address des Mobile Nodes mit, so daß der Absender von nun an seine Pakete direkt an das Fremdnetz schicken kann, in dem sich der Mobile Node befindet. Dies steigert die Effizienz und vermeidet ein unnötiges Überlasten des Internets.

Mit diesem Mobile IP-Protokoll wurde die Kompatibilität zum herkömmlichen IP-Protokoll bewahrt. Gleichzeitig bleibt den höheren Schichten verborgen, daß es sich um ein mobiles Endgerät handelt. Dieses kann seinen Zugangspunkt zum Netz auch während der Aufrechterhaltung einer Verbindung wechseln, und die Sicherheit bleibt gewährt, indem sich der Host immer erst authentifizieren muß, bevor er in ein neues Fremdnetz integriert wird.

6 Aussichten

Wie man sieht, wurde auf dem Gebiet der Wireless LANs in jüngster Zeit sehr viel entwickelt; und diese Entwicklung hat noch lange kein Ende gefunden. Im Gegenteil - mit den neuen Standards IEEE 802.11a und 802.11b fangen die drahtlosen Netze erst an, in Bezug auf die Übertragungsgeschwindigkeit mit den Festnetzen zu konkurrieren. Auch bezüglich der Fehlerwahrscheinlichkeit bei der Übertragung wird sich in der nächsten Zeit noch einiges ändern. Bleibt also abzuwarten, ob und wann die drahtlosen Netze die Festnetze vollständig ersetzen werden.

Für die interessierten Leser, die sich noch eingehender mit Wireless LANs beschäftigen möchten, finden sich weitere Informationen in den Büchern „Mobile Communications“ [Schi00], „Computer Networks“ [Tane96] und „Lehr- und Übungsbuch Telematik“ [KrRe00].

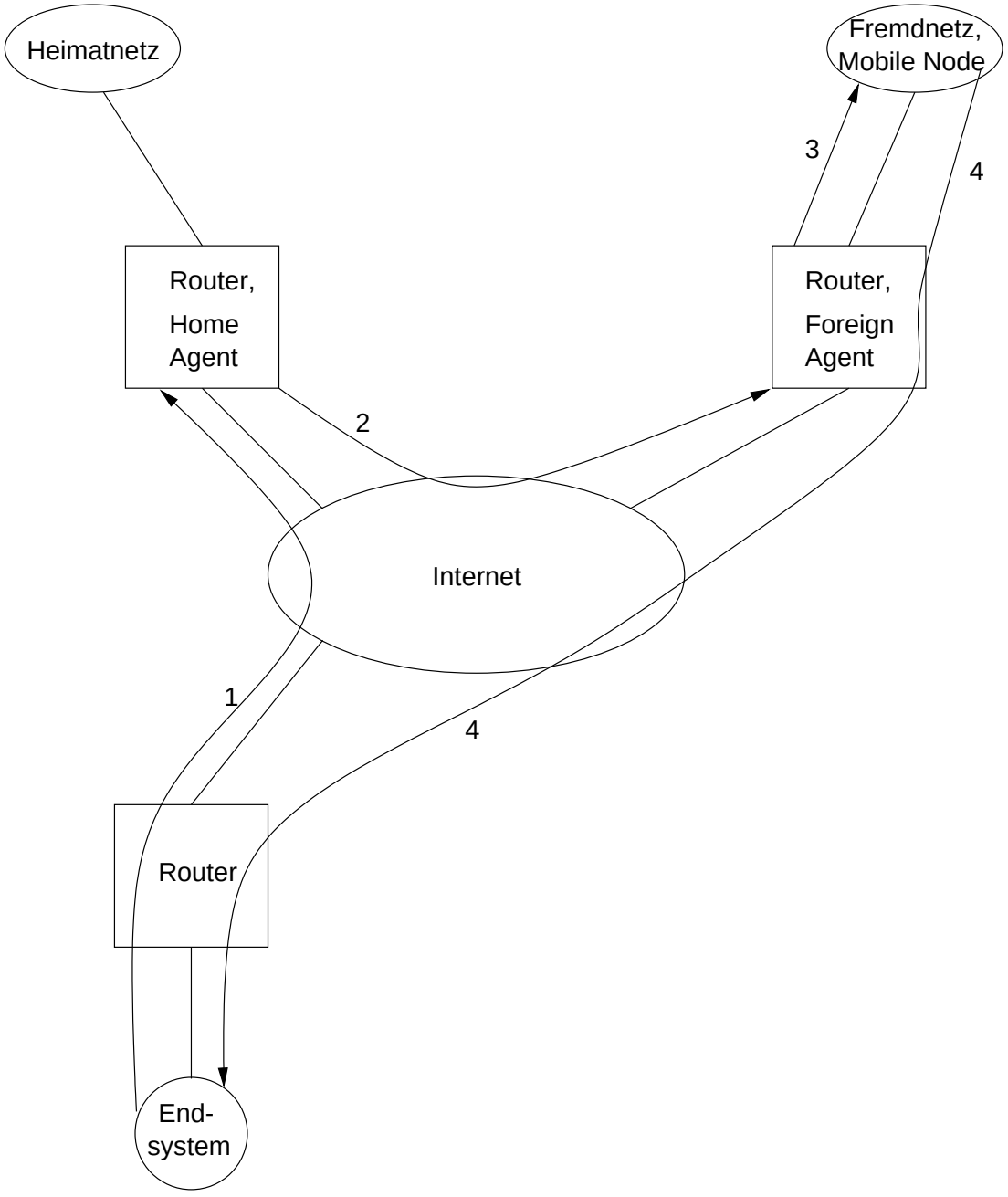

Abbildung 8: Datentransfer mittels Mobile IP

Literatur

[DoGR99] Elmar Dorner, Meng Gan und Verena Rose. Planung, Installation und Betrieb eines lokalen Funknetzes. Technischer Bericht, Institut für Telematik, Universität Karlsruhe (TH), August 1999.

[KrRe00] Gerhard Krüger und Dietrich Reschke. *Lehr- und Übungsbuch Telematik*. Fachbuchverlag Leipzig. 2000.

[Schi00] Dr. Jochen Schiller. *Mobile Communications*, Kapitel 7 und 9, S. 161–214, 255–289. Addison-Wesley. 2000.

[Tane96] Andrew S. Tanenbaum. *Computer Networks*. Prentice-Hall. 3rd Edition, 1996.

Mobile TCP - Erweiterungen und Verbesserung von TCP in mobilen Umgebungen

Milena Neumann

Kurzfassung

TCP wurde für Festnetze mit hoher Bandbreite und niedriger Bitfehlerrate entwickelt, in solchen Netzen funktioniert es auch gut. Drahtlose Transportverbindungen leiden oft unter Problemen wie hohe Bitfehlerrate, Verbindungsabbrüche durch Abschattungen oder Handovers sowie niedrige Bandbreite, die sich außerdem dynamisch verändern kann. Traditionelles TCP geht bei Paketverlusten immer von einer Netzüberlastung aus und reagiert mit speziellen Maßnahmen, die eine schnelle Auflösung des Staus ermöglichen. Diese Annahme ist jedoch meist falsch in drahtlosen Netzen, wo Paketverluste durch Übertragungsfehler verursacht werden und führt dazu, dass die Leistung des TCP drastisch sinkt. In dieser Ausarbeitung wird zuerst das Verhalten von TCP in mobiler Umgebung beschrieben. Dann werden verschiedene TCP-Modifikationen vorgestellt, die entwickelt wurden um die Effizienz des TCP in der mobilen Umgebung zu verbessern.

1 Einleitung

Unter verschiedenen Trends in Netzwerktechnologie und Kommunikation in den 90er Jahren fallen zwei deutlich auf: die rapide globale Expansion des Internet zum Zugriff auf Informationsressourcen sowie das massive Wachstum der Nutzung von mobilen Geräten zur Kommunikation. Die Hauptursache des ersten Trends ist das schnelle Wachstum vom World Wide Web und die riesige Menge an Informationen, die dort für jeden zugänglich sind. Der zweite Trend, der der drahtlosen Kommunikationen, ist durch die Fortschritte in der Hardwareherstellung und in der Technologie zellularer Telefonnetze zustandegekommen. Abbildung 1 zeigt diese zwei rasch wachsende Trends.

Man könnte erwarten, dass das Zusammenspiel der beiden oben genannten Trends - Internet auf der einen Seite und Mobilität auf der anderen - in einer wirkungsvollen Kombination resultieren und drahtlose Datennetzwerke und -dienste den Markt überfluten würden. Die Beobachtung des Marktes zeigt jedoch, dass drahtlose Datennetzwerke und -dienste noch weit davon entfernt sind ausgereift zu sein. Es mangelt immer noch an einer breitgefächerter Profilierung solcher Systeme.

Warum ist das der Fall? Kann es sein, dass das angeborene Bedürfnis des Benutzers an drahtlosen Datensystemen nicht existiert? Verschiedene Forschungen widersprechen dem jedoch und zeigen, dass Benutzer den bequemen drahtlosen Internetzugang, den diese Systeme bereitstellen, bevorzugen. Das folgende Zitat von Georg H. Heilmeier erscheint dabei als eine Zukunftsvision: „People and their mashines should be able to access information and communicate with each other easily and securely, in any medium or combination of media - voice, data, image, video, or multimedia - any time, anywhere, in a timely, cost-effective way.“ Leider wird die Realisierung dieser Vision durch die Besonderheiten der Kommunikation in der mobilen Umgebung erschwert. Nicht nur Hardwaretechnologien, sondern auch Kommunikationsprotokolle müssen weiterentwickelt werden, um diese Vision zu verwirklichen.

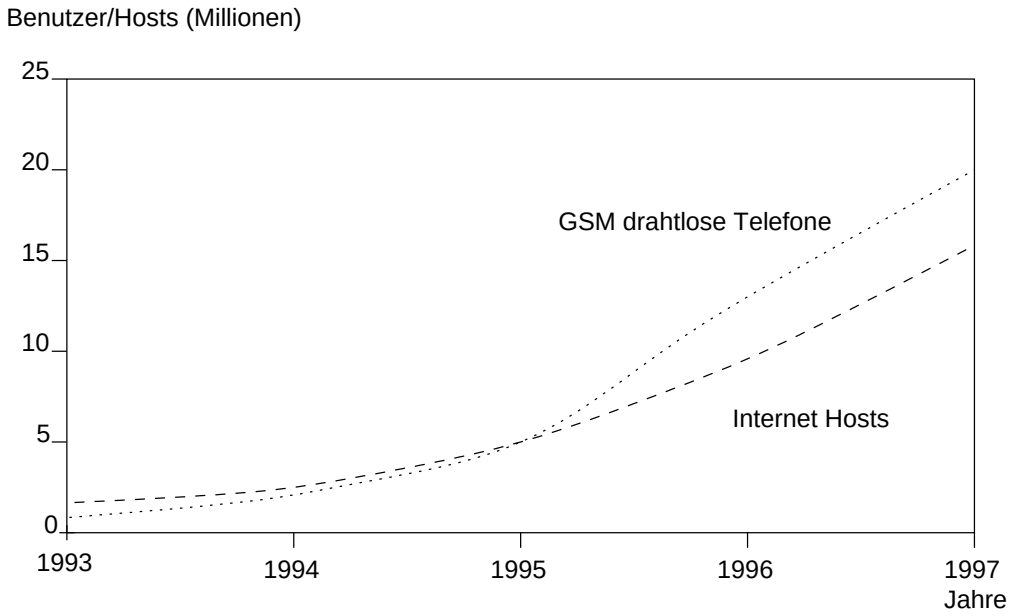

Abbildung 1: Wachstum der Anzahl von GMS Benutzer und Internet Hosts. (*Quelle: Ericsson, Inc. and Matthew Gray, MIT*)

2 TCP in der mobilen Umgebung

2.1 Effekte der Mobilität

Drahtlose Transportverbindungen zeichnen sich durch ihre Unzuverlässigkeit aus. Pakete, die über eine Funkstrecke übertragen werden, gehen oft verloren oder werden verfälscht. Außerdem entstehen in solchen Netzen oft lange Verzögerungen, was zu einem Timeout führen kann. Diese Effekte werden verursacht durch

- hohe Übertragungsfehlerrate auf der drahtlosen Strecke
- Verbindungsunterbrechungen während Handovers
- (wiederholte) Verbindungsabbrüche aufgrund Abschattungen durch Hindernisse
- Entfernung des mobilen Gerätes aus der Reichweite der Basisstation
- schmale Bandbreite, niedrige Übertragungsrate
- begrenzte Energieressourcen mobiler Endgeräte
- Geräteheterogenität.

Auch wenn sich das mobile Endgerät nicht bewegt, drahtlose Kommunikationen leiden unter großer Paketverlustrate aufgrund physikalischer Übertragungsfehler durch Abschattungen, Interferenzen mit Wassermolekülen in der Luft, falsche Antennenausrichtung, Dämpfung, Mehrwegeausbreitung, etc. Im zellularen Mobilfunknetz, wenn sich der Benutzer zwischen den Zellen bewegt, kann es zu kurzen Übertragungsunterbrechungen kommen, während das mobile Endgerät das Handover zu der nächsten Basisstation ausführt. Unterbrechungen können außerdem durch physikalische Hindernisse verursacht werden, die Funksignale blockieren, z.B. Gebäude. Wenn sich im Bereich einer Zelle zu viele Benutzer aufhalten, kann es

passieren, dass die vorhandene Bandbreite für neu ankommende Benutzer nicht ausreicht - Zellenblockierung, die über eine längere Zeitperiode dauern kann - das kann auch zu den Unterbrechungszuständen gezählt werden. Die Unterbrechungen können mehrere Sekunden dauern, verlorene Datensegmente und Quittungen sind die Folgen.

Das Problem verschlimmert sich noch im Falle sehr kleiner Zellen oder wenn sich der Benutzer sehr schnell bewegt. Dabei treten Unterbrechungen viel öfter auf, wiederholte Unterbrechungszustände führen zu den seriellen Timeouts auf der Seite des TCP-Senders. Ein serielles Timeout tritt ein, wenn mehrere aufeinanderfolgende Versuche dasselbe Segment zu Übertragen gescheitert sind, weil die Funkstrecke gerade in dieser Zeit blockiert war. Der Timer auf der Senderseite wird nach jedem unglücklichen Versuch verdoppelt, das kann zu einer Untätigkeit von bis zu 1 Minute führen!

Mobile Endgeräte zeichnen sich durch ihre immense Heterogenität aus. Die große Vielfalt der Technologien drahtloser Kommunikation reicht von den drahtlosen Local Area Netzwerken bis zu Wide Area Satellitensystemen. Diese Technologien haben verschiedene Charakteristika (z.B. Bandbreite, Latenzzeiten, Fehlerrate, Ausdehnung, etc.) die die Identifizierung der Ursachen für schlechte Performance erschweren. Die existierenden Ansätze des zuverlässigen Datentransfers haben Schwierigkeiten im Umgang mit der Heterogenität drahtloser Netzwerke.

2.2 Traditionelles TCP

Ein Transportschichtprotokol wie TCP wurde entwickelt für feste Netze mit „festen“ Endgeräten. Sowohl die Hardware, als auch die Software für solche Netzwerke ist relativ gut ausgereift und funktioniert im Vergleich mit drahtlosen Netzwerken stabil und zuverlässig. Deshalb entstehen Übertragungsfehler in festen Netzwerken eher selten. Dennoch können auch in solchen Netzwerken Pakete verlorengehen, allerdings aus einem anderen Grund. Ein hohes Verkehrsaufkommen in irgendeinem Router der Kommunikationsstrecke kann dazu führen, dass die Pufferkapazität des Routers nicht mehr ausreicht. Die Strecke ist überlastet, der Router kann die Pakete nicht mehr schnell genug weiterleiten. In dieser Situation bleibt dem Router nichts anderes übrig, als die ankommenden Pakete zu verwerfen.

Aufgrund fehlender Quittung stellt der Sender fest, dass ein Paket verlorengegangen ist. Dabei geht er davon aus, dass im Netz eine Überlastung aufgetreten ist. Übertragungswiederholungen wären jetzt zwecklos und könnten den Stau noch vergrößern. Statt dessen reagiert der TCP-Sender mit speziellen Maßnahmen, die notwendig sind, damit sich die Überlastung auflösen kann: er verlangsamt seine Übertragung, um so das Verkehrsaufkommen im überlasteten Netzknoten zu vermindern. Das Gleiche tun natürlich auch alle anderen TCP-Sender, die am denselben Stau beteiligt sind, nur auf diese Weise kann sich die Überlastung bald auflösen. Dieses faire Zusammenwirken der TCP-Verbindungen im globalen Internet ist für das Netz der Netze lebenswichtig.

Der Zustand in den die TCP-Instanz nach einer festgestellten Überlastung übergeht, heißt *slow start*, siehe Abbildung 2. TCP verwendet zur Überlastungsüberwachung eine spezielle Fenstertechnik, wobei der Sender für jede Übertragung ein sogenanntes Überlastungsfenster für einen Empfänger festlegt. Die Anfangsgröße des Überlastungsfensters ist immer 1 (ein TCP-Paket). Wenn alles gut geht, d.h. Quittungen kommen rechtzeitig und in richtiger Reihenfolge, verdoppelt sich das Überlastungsfenster nach jeder erfolgreichen Übertragung. Das exponentielle Wachstum des Fensters dauert, bis die Fenstergröße einen gewissen Schwellwert erreicht hat. Danach wird das Wachstum linear, nun erhöht sich die Fenstergröße nach Erhalt einer Quittung jedes Mal um 1.

Kommt es auf der Senderseite zu einem Timeout aufgrund fehlender Quittung oder zu wiederholten Quittungen für dasselbe Paket - deutet das auf einen Paketverlust hin. Sofort setzt

der Sender die Größe des Überlastungsfensters wieder auf 1, die neue Überlastungschwelle auf die Hälfte des aktuellen Wertes und der *slow start* Algorithmus beginnt von vorn.

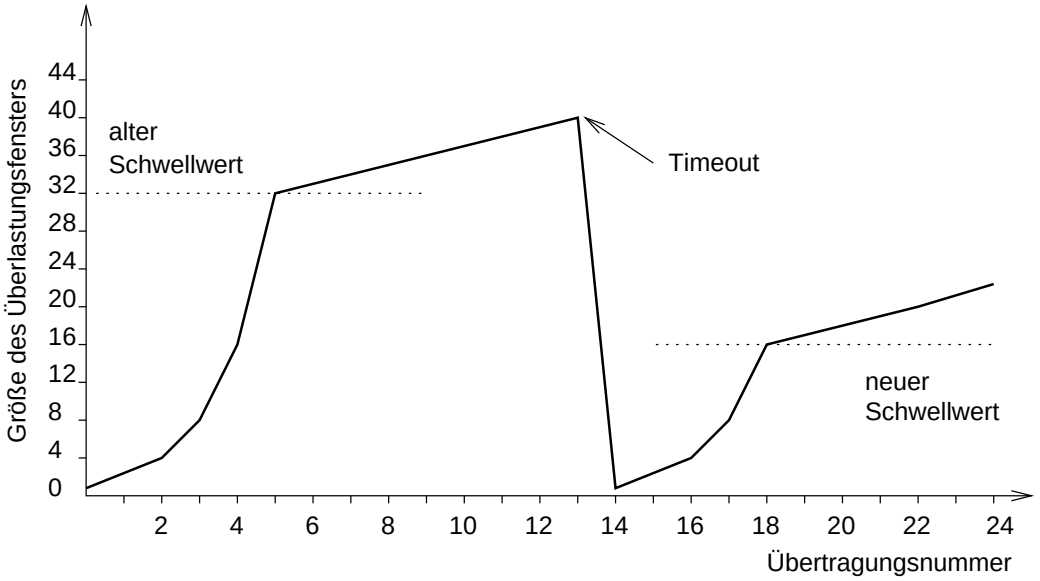

Abbildung 2: Der *slow start* Algorithmus von TCP.

Wenn traditionelles TCP-Protokoll bei mobilen Sendern oder Empfängern eingesetzt wird, verschlechtert sich seine Leistung drastisch. Das Problem ist, dass die vorhandenen TCP-Mechanismen zwischen verschiedenen Ursachen für Paketverluste nicht unterscheiden können. TCP kennt einfach keine anderen Ursachen als Überlastung und übergeht deshalb immer in den *slow start*. Leider nutzt dieses Verhalten in der mobilen Umgebung nichts, hier entstehen Paketverluste wegen Übertragungsfehler, siehe Abschnitt 2.1. Die bewährten TCP-Mechanismen, die im Festnetzbereich das Internet zusammenhalten, führen dazu, dass der Einsatz vom unveränderten TCP zusammen mit drahtlosen oder mobilen Endgeräten in einem katastrophalen Leistungsabfall des Transportprotokolls resultiert.

Um eine akzeptable Effizienz des Transportprotokolls beim Einsatz mit mobilen Endgeräten zu erreichen muß TCP den Gegebenheiten der mobilen Umgebung angepasst werden. Auf der anderen Seite kann man TCP nicht komplett verändern weil es bereits auf Millionen Hosts läuft. Man kann auch nicht den *slow start* Algorithmus einfach abschalten: ohne diesen Mechanismus würde das globale Internet womöglich in einem Chaos zusammenbrechen. Aus diesen Überlegungen folgt, dass alle mobilitätunterstützenden TCP-Modifikationen kompatibel zu Standard-TCP sein müssen und möglichst keine Änderungen der bereits installierten Basis erfordern.

3 Indirektes TCP

Im Jahr 1995 wurde Indirektes TCP vorgestellt. Diese TCP-Modifikation sieht eine Auftrennung der Transportverbindung in zwei Verbindungen vor, siehe Abbildung 3. Das Ziel war die Effizienz des TCP in der mobilen Umgebung zu verbessern, ohne Änderungen im Festnetz-Bereich vornehmen zu müssen. Als eine gut geeignete Stelle für die Trennung sind z.B. der *foreign agent* im Mobile IP oder die Basisstation im zellularen Mobilfunknetz denkbar, die bei diesem Verfahren als Proxy funktionieren.

Die Aufteilung in zwei Verbindungen hat folgende Vorteile [BaB.95] :

- sie trennt die Flußkontrolle und die Überlastungskontrolle auf der drahtlosen Strecke von denselben im Festnetz. Diese Trennung ist erwünscht, weil die entsprechende Charakteristika der beiden Netzarten enorm unterschiedlich sind: die Festnetze (Ethernet oder zukünftig ATM) werden Tag für Tag schneller und zuverlässiger, während die drahtlosen Verbindungen immer noch sehr langsam und extrem anfällig für Rauschen und Signaldämpfung sind, was zu einer höher Bitfehlerrate führt
- ein spezielles Transportprotokoll für die drahtlose Strecke kann solche „mobilspezifische“ Ereignisse unterstützen wie Verbindungsunterbrechung, Bewegung des Benutzers sowie andere Eigenschaften berücksichtigen, wie z.B. verfügbare Bandbreite etc.
- die Basisstationen können dem mobilen Host einen Großteil der Verbindungsverwaltung abnehmen. So kann ein mobiles Endgerät, auf dem ein sehr einfaches Protokoll zur Kommunikation mit der Basisstation läuft, Dienste des Festnetzes nutzen, z.B. WWW. Ohne Trennung müßte der ganze TCP/IP-Turm auf dem mobilen Gerät laufen.

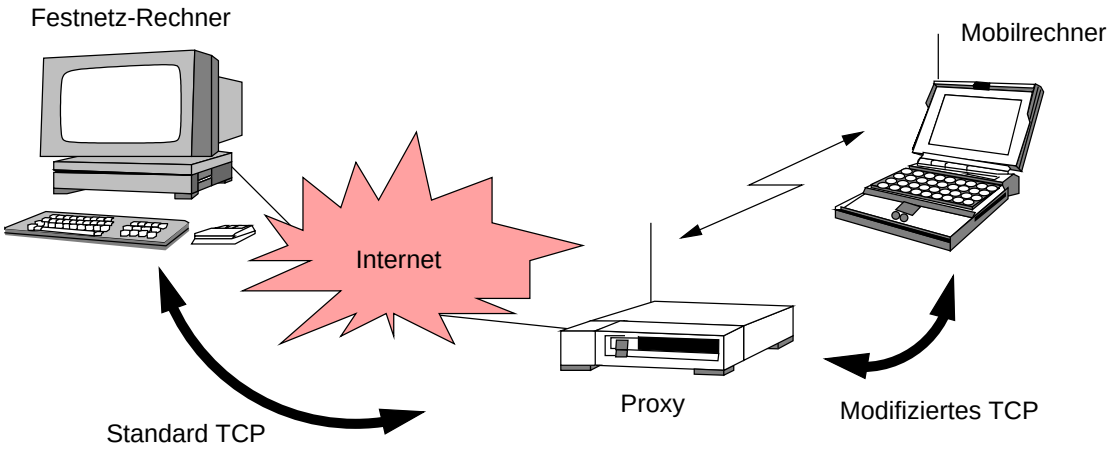

Abbildung 3: Indirektes TCP.

Zwischen dem Festnetz-Host und dem Proxy läuft das Standard-TCP. Die Rechner im Festnetz merken von der Trennung nichts. Zwischen dem Proxy und dem mobilen Endgerät wird ein modifiziertes TCP-Protokoll eingesetzt, das für die mobile Umgebung optimiert ist.

Wenn der Festnetz-Rechner ein Paket abschickt, wird das Paket vom Proxy bestätigt. Dann versucht er das Paket an das mobile Endgerät weiterzuleiten. Wenn das mobile Endgerät das Paket erhält, sendet es eine Bestätigung. Die ist jedoch nur für den Proxy von Bedeutung. Wenn ein Paket auf der Funkstrecke wegen eines Übertragungsfehlers verlorengeht, bekommt der Festnetz-Rechner das nicht mit. Es ist jetzt die Aufgabe des Proxy, das Paket erneut zu übertragen und so den zuverlässigen Datentransfer zu gewährleisten.

Wenn das mobile Endgerät ein Paket sendet, bestätigt der Proxy dieses und versucht es zum Empfänger im Festnetz weiterzuleiten. Das mobile Gerät merkt Paketverluste auf der Funkstrecke viel schneller und kann die Übertragung sofort wiederholen. Paketverluste im Festnetz werden nun vom Proxy behandelt.

Der Einsatz vom I-TCP in der mobilen Umgebung bringt folgende Vorteile:

- es sind keine Änderungen im Festnetzbereich erforderlich
- dank strikter Trennung in zwei Verbindungen können sich die Fehler auf der drahtlosen Strecke nicht ins Festnetz fortpflanzen. Ohne Trennung würden die Übertragungswiederholungen verlorener Pakete zwischen dem mobilen Endgerät und dem Festnetz-Rechner stattfinden und sich so über das ganze Netz erstrecken

- bei I-TCP ist es möglich, verschiedene Ideen und Verbesserungsvarianten auf der drahtlosen Strecke auszuprobieren, ohne damit die Stabilität des gesamten Internets zu bedrohen
- die Verzögerung auf der Funkstrecke ist bekannt und von dem Festnetz unabhängig. Ein optimiertes TCP kann diese Kenntnis nutzen und so schnell wie möglich eine Übertragungswiederholung starten
- die Auftrennung in zwei Transportverbindungen ermöglicht die Verwendung weiterer Mechanismen auf der Funkstrecke (z.B. für Datenkompression).

Abbildung 4 zeigt die erreichte Leistungsverbesserung vom I-TCP im Vergleich mit Standard-TCP. Die Messungen wurden mit Bitfehlerrate $1.9\text{x}10^{-6}$ (1 Fehler/65536 Bytes) durchgeführt. Auf der drahtlosen Strecke wurde eine TCP-Version mit der Möglichkeit für selektive Quittungen (RFC 1079) verwendet [BPSK97].

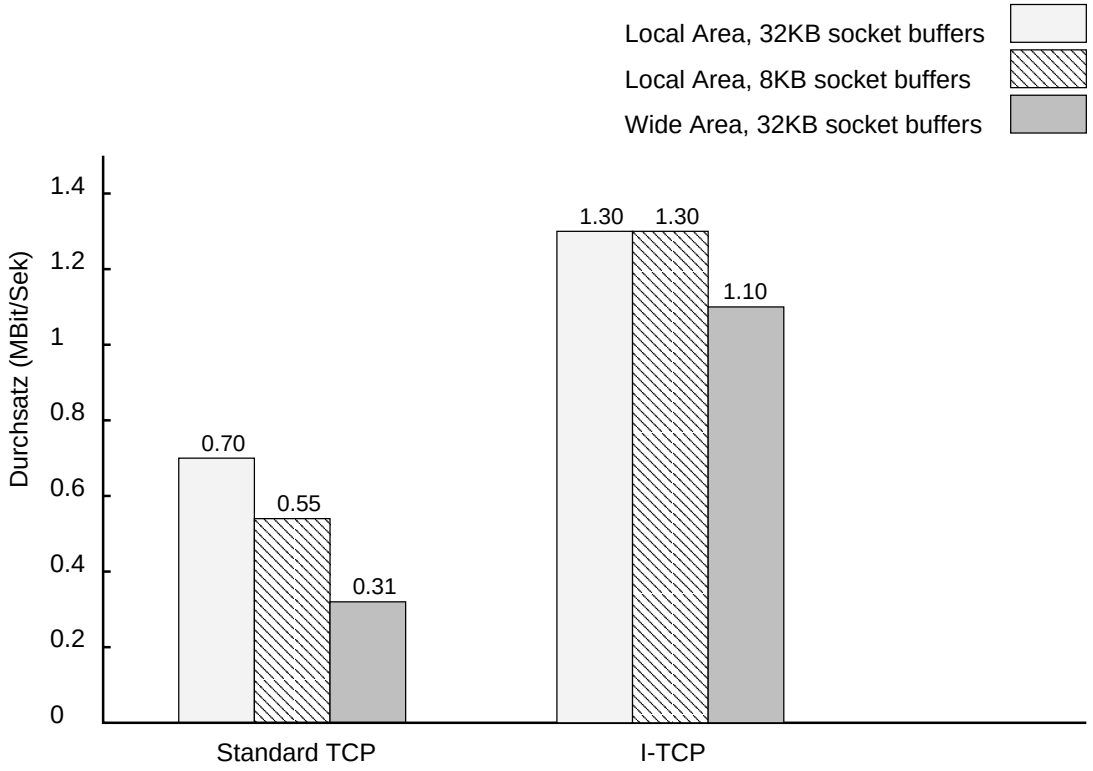

Abbildung 4: I-TCP im Vergleich mit Standard-TCP.

Die Idee der Auftrennung der Transportverbindung hat aber auch Nachteile:

- Verlust der Ende-zu-Ende Semantik von TCP. Nach Erhalt einer Quittung ist der Sender im Festnetz ist davon überzeugt, das der Empfänger sein Paket korrekt empfangen hat. Die Quittung stammt jedoch nicht vom Empfänger, sondern vom Proxy, und es gibt keine Garantie dafür, das er das Paket erfolgreich zum Mobilrechner weiterleitet. Der Sender hat eine inkonsistente Sicht auf den Zustand der Verbindung
- vergrößerte Latenzzeiten, verzögertes Handover. Der Proxy puffert alle Pakete, die er vom Festnetz-Rechner empfängt und entfernt sie aus dem Puffer erst nach Erhalt einer Quittung vom Mobilrechner. Während des Handovers kommen weitere Pakete an. Wenn der Mobilrechner den Kontakt mit dem neuen Proxy aufgenommen hat, kann es eine Weile dauern, bis der alte Proxy die gesamte Menge an gepufferten Paketen zum neuen

Proxy weitergeleitet hat. Wenn der Mobilrechner ein Handover zum nächsten Proxy ausführt, kann es eine Weile dauern, bis der alte Proxy alle gepufferten Pakete zum neuen Proxy weiterleiten kann

- Sicherheitsprobleme. Die Mechanismen des Verschlüsselungverfahrens (falls angewendet) müssen dem Proxy bekannt sein.

4 Snooping TCP

Im Gegenteil zu I-TCP, bei dem die Ende-zu-Ende-Semantik einer TCP-Verbindung verletzt wird, funktioniert Snooping-TCP für die beiden Partner-Instanzen völlig transparent [BSAK95]. Bei diesem Verfahren sind lediglich einige Erweiterungen im Proxy (Basisstation oder *foreign agent*) erforderlich. Das Festnetz, sowie der Mobilrechner sind von den Änderungen nicht betroffen. Die Abbildung 5 stellt das Verfahren schematisch dar.

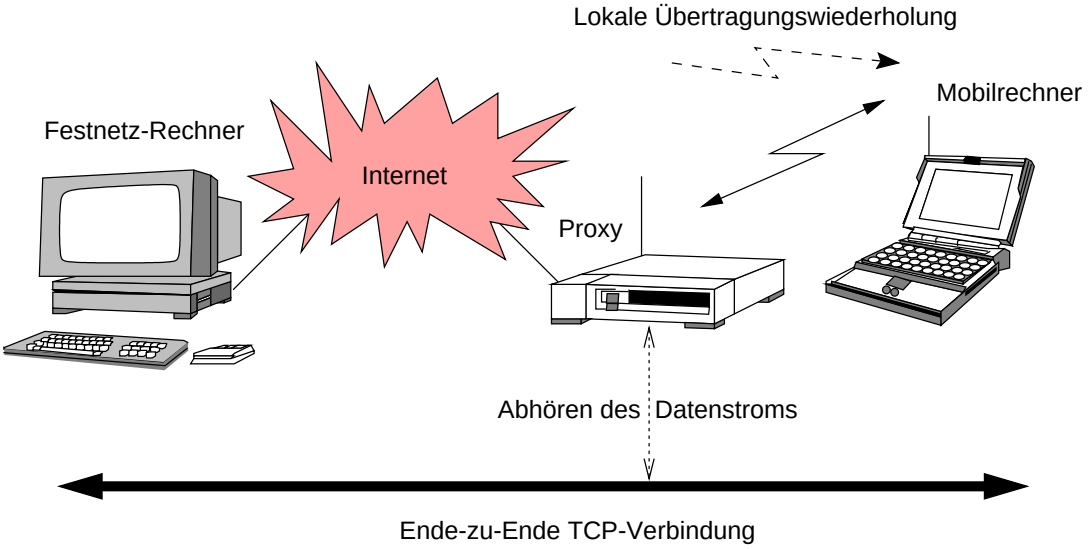

Abbildung 5: Snooping TCP.

Der Proxy überwacht den Datenstrom, der zwischen dem Festnetz-Rechner und dem mobilen Endgerät fließt. Die Software im Proxy kann zwischen verschiedenen Paketarten unterscheiden, insbesondere Sequenznummern und Bestätigungen werden herausgefiltert. So weiß der Proxy, welche Pakete bereits bestätigt wurden und welche noch nicht. Jedes Paket, das in Richtung zum Mobilrechner verschickt wird, wird vom Proxy gespeichert und bleibt im Puffer so lange, bis der Proxy eine Bestätigung für dieses Paket im Datenstrom vom Mobilrechner registriert. Zweck der Paketpufferung ist die Möglichkeit einer lokalen Übertragungswiederholung falls ein Paket auf der Funkstrecke verlorengeht. Wenn der Proxy anhand einer fehlenden oder duplizierten Quittung einen Paketverlust feststellt, überträgt er das entsprechende Paket aus seinem Pufferspeicher sofort erneut. Die lokale Übertragungswiederholung erfolgt viel schneller, als wenn das Paket vom Festnetz-Rechner erneut gesendet und die ganze Strecke über das Festnetz noch mal durchlaufen würde. Die Quittung vom Mobilrechner für das lokal wiederholte Paket erreicht den Festnetz-Rechner noch bevor sein Timer abläuft, er bekommt also nicht mit, dass das Paket auf der Funkstrecke verlorengegangen war. Auch wenn der Snooping-Proxy im Gegensatz zu I-TCP selbst keine Bestätigungen schickt, kann er dennoch Duplikate von Quittungen oder bereits lokal wiederholter Pakete aus dem Paketstrom entfernen, um den überflüssigen Datenverkehr zu vermeiden.

Pakete, die vom Mobilrechner in die Richtung zum Festnetz-Rechner fließen, werden nicht gepuffert. Der Proxy hört jedoch den Paketstrom ab und überprüft die Sequenznummern der

gesendeten Pakete und Quittungen. Eine Lücke in der Paketreihe deutet auf einen Paketverlust auf der drahtlosen Strecke hin. In dieser Situation schickt der Proxy dem Mobilrechner eine negative Quittung (NACK). Der Mobilrechner kann die Übertragung des fehlenden Pakets sofort wiederholen.

Snooping-TCP hat noch einen Vorteil gegenüber I-TCP. Ein Absturz des Proxy im Indirekt TCP führt unmittelbar zum Absturz der Anwendungen auf dem Festnetz-Rechner, die einen zuverlässigen Datentransfer voraussetzen. Die Transparenz des Snooping-TCP ermöglicht eine gewisse Resistenz der Partner-TCP-Instanzen gegen den Proxy-Crash. Wenn der Proxy abstürzt, läuft der Timer auf dem Festnetz-Rechner immer noch weiter und führt im Falle eines Timeouts zu einer Übertragungswiederholung.

Abbildung 6 zeigt die erreichte Leistungsverbesserung vom Snooping-TCP im Vergleich mit Standard-TCP. Die Messungen wurden mit Bitfehlerrate $1.9\text{x}10^{-6}$ (1 Fehler/65536 Bytes) durchgeführt [BPSK97].

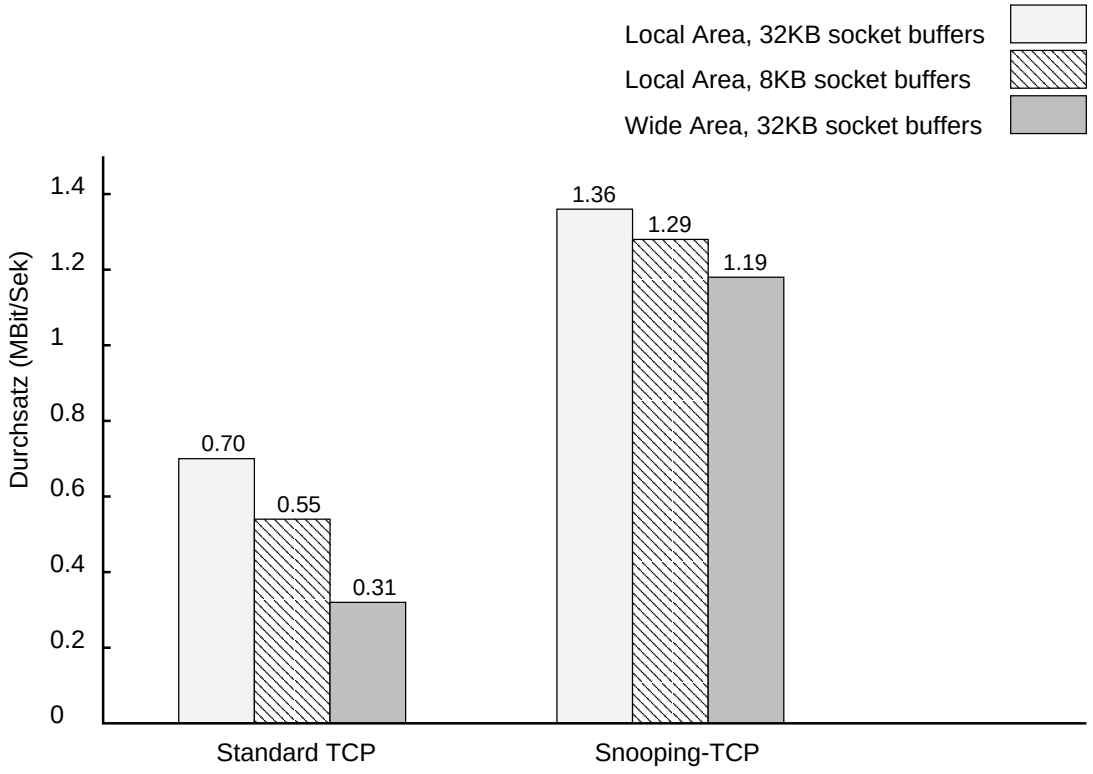

Abbildung 6: Snooping-TCP im Vergleich mit Standard-TCP.

Nun alle Vorteile des Snooping TCP in der Zusammenfassung:

- die Ende-zu-Ende-Semantik einer Transportverbindung bleibt intakt. Auch wenn der Proxy abstürzt, weder mobile Endgerät, noch Festnetz-Rechner haben eine inkonsistente Sicht auf die TCP-Verbindung wie es bei I-TCP möglich ist
- es sind keine Änderungen im Festnetzbereich erforderlich, alle Modifikationen finden im Proxy statt. Wenn Snooping TCP nur für den Datenstrom in Richtung vom Festnetz-Rechner zum Mobilrechner eingesetzt wird, sind sogar keine Änderungen im Mobilrechner nötig
- keine negative Auswirkungen auf Handover. Der alte Proxy muß keine gepufferten Pakete zum neuen Proxy weiterleiten.

Natürlich hat das Verfahren auch Nachteile:

- Snooping TCP isoliert die drahtlose Strecke nicht so gut wie I-TCP. Wenn es wegen Übertragungsfehlern zu lange dauert, bis der Proxy ein gepuffertes Paket erfolgreich übertragen kann, kann der Timer auf dem Festnetz-Rechner überlaufen und eine Übertragungswiederholung auslösen. Das heißt, Probleme der drahtlosen Kommunikation sind jetzt auch für den Festnetz-Rechner sichtbar. Die Qualität der Isolierung, die Snooping TCP bietet, hängt stark von der Qualität der drahtlosen Strecke, Timeout-Werten und anderen Charakteristika ab
- Auf dem Mobilerechner muß eine entsprechende TCP-Version laufen, die negative Quittungen erlaubt. Das Verfahren ist also nicht mit jedem mobilen Endgerät kompatibel
- Abhören des Datenstroms kann nutzlos sein, wenn die Daten in verschlüsselter Form übertragen werden.

5 Mobile TCP

Neben der hoher Bitfehlerrate, die von der schlechten Qualität der drahtlosen Strecke verursacht wird, gehören auch längere und/oder wiederholte Verbindungsunterbrechungen zu den Spezialitäten der Mobilkommunikation. In einem zellularen Mobilfunknetz ohne vollständige Flächendeckung ist es keine Seltenheit, dass mobile Benutzer überhaupt keine Verbindung zur Basisstation aufbauen können. Das traditionelle TCP-Protokoll wurde für diese Besonderheiten nicht ausgelegt und kann deshalb wiederholte Verbindungsunterbrechungen nicht optimal behandeln.

Die Übertragungswiederholungen des TCP-Senders werden vom Wiederholungstimer gesteuert. Der Timer verdoppelt sich bei jedem verunglückten Übertragungsversuch, bis er das Maximum von 1 Minute erreicht hat. Das heißt, dass der Sender versucht jede Minute die Übertragung des unbestätigten Pakets zu wiederholen bis er nach dem 12. Versuch endgültig aufgibt. Selbst wenn die Transportverbindung früher wieder vorhanden ist, werden für die Dauer von einer Minute keine Daten gesendet. Das Wiederholungstimeout ist weiterhin gültig und der Sender muß warten. Außerdem übergeht der Sender in *slow start*, weil er eine Überlastung vermutet (siehe Abschnitt 2.2).

Die zwei bereits vorgestellten Verfahren haben auch ihre Probleme mit Unterbrechungen. Im Falle I-TCP bleibt der Teil der Transportverbindung zwischen dem Festnetz-Rechner und dem Proxy intakt, der Festnetz-Rechner merkt nicht, dass die Funkverbindung unterbrochen ist. Während auf der Funkstrecke kein Datenaustausch mehr stattfindet, sendet der Festnetz-Rechner die Daten munter weiter und der Proxy schickt fleißig Bestätigungen, obwohl er keine Pakete an den Mobilrechner weiterleiten kann. Je länger die Unterbrechung dauert, desto mehr Pakete muß der Proxy puffern. Oft folgt einer Unterbrechung ein Handover, das heißt, die gesamte Datenmenge muß zum neuen Proxy weitergeleitet werden. Das Snooping TCP spezialisiert sich auf das Abhören des Datenstroms und schafft deshalb keine Abhilfe bei Verbindungsunterbrechungen, wenn kein Datenstrom mehr da ist.

Das Mobile TCP (M-TCP) hat das selbe Ziel wie die beiden oben beschriebene Verfahren: das Schrumpfen des Übertragungsfensters beim Sender zu verhindern, wenn nicht Überlastung, sondern Bitfehler oder Unterbrechung Paketverluste verursachen. M-TCP [BrSi97] soll folgende Verbesserungen bringen:

1. Erhöhung der TCP-Leistung für mobile Benutzer

2. Bewahrung der Ende-zu-Ende-Semantik
3. Möglichkeit, Probleme langer und/oder wiederholter Verbindungsunterbrechungen zu behandeln
4. Erweiterungen zur dynamischen Bandbreitenverteilung für bereits hungernde drahtlose Verbindungen
5. Erweiterungen für effizienteres Handover.

Für die Implementierung vom M-TCP wurde das Verfahren der Auftrennung der Transportverbindung wie beim I-TCP gewählt. In zellularen Mobilfunknetzen werden mehrere Basisstationen von einer Netzwerkkomponente, die an das Festnetz angeschlossen ist, gesteuert. Diese Komponente, *supervisor host* genannt, hat im M-TCP die Aufgabe, den Verkehr zwischen dem Festnetz-Rechner und dem Mobilrechner zu überwachen und zu steuern. Zwischen dem Festnetz-Rechner und dem *supervisor host* läuft Standard-TCP, zwischen dem *supervisor host* und dem Mobilrechner wird ein modifiziertes TCP eingesetzt. Der *supervisor host* im M-TCP ist für den Datenaustausch zwischen den zwei Parteien verantwortlich, ähnlich wie der Proxy im I-TCP, siehe Abbildung 3, unternimmt jedoch weder Paketpufferung noch Übertragungswiederholungen. Er schickt selber auch keine Bestätigungen, das bewahrt die Ende-zu-Ende-Semantik vom TCP. Wenn ein Paket auf der drahtlosen Strecke verloren geht, muß es vom Original-Sender erneut übertragen werden.

Das M-TCP-Verfahren nimmt eine relativ niedrige Bitfehlerrate auf der Funkstrecke an. Der *supervisor host* überwacht den Datenstrom auf der Funkstrecke: Pakete, die zum Mobilrechner fließen, sowie deren Bestätigungen. Wenn er eine Zeit lang keine Bestätigung erhält, nimmt der *supervisor host* an, dass die Funkverbindung unterbrochen ist und setzt das Übertragungsfenster beim Sender auf Null. Der Sender übergeht daraufhin in den persistenten Modus, dass heißt, der Zustand der Verbindung bleibt konstant, egal wie lange die Unterbrechung dauert. In diesem Zustand macht der Sender keine Übertragungswiederholungen. Sobald die Verbindung zum mobilen Endgerät wieder da ist, öffnet der *supervisor host* das Übertragungsfenster beim Sender wieder mit dem alten Wert. Der Sender kann jetzt die Übertragung mit voller Leistung fortsetzen. Bei diesem Verfahren sind keine Änderungen des TCP auf dem Festnetz-Rechners erforderlich.

Auf der Seite des Mobilrechners wird ein optimiertes TCP verwendet, das kein *slow start* benutzt. Deshalb ist bei M-TCP ein Bandbreitenmanager nötig, um eine faire Bandbreitenverteilung auf der drahtlosen Strecke zwischen allen Benutzern - mit oder ohne *slow start* - zu sichern.

M-TCP hat folgende Vorteile:

- die Ende-zu-Ende-Semantik bleibt erhalten. Der *supervisor host* sendet selbst keine Quittungen, sondern leitet nur die vom Mobilrechner weiter
- wenn die Verbindung zum mobilen Endgerät unterbrochen ist, vermeidet das Verfahren nutzlose Übertragungswiederholungen, *slow start* oder Verbindungsabbau durch einfaches Schrumpfen des Übertragungsfensters auf Null
- anders als im I-TCP, puffert der *supervisor host* im M-TCP keine Pakete. Deshalb ist im Falle eines Handovers nicht notwendig, gepufferte Pakete zum neuen *supervisor host* weiterzuleiten.

Verzicht auf Pufferung und Modifikationen des TCP auf dem Mobilrechner haben auch einige Nachteile:

- Das Verfahren isoliert die Funkstrecke nicht so gut. Die Übertragungsfehler auf der drahtlosen Strecke breiten sich ins Festnetz aus. M-TCP läuft unter Annahme niedriger Bitfehlerrate, was nicht immer der Fall ist.

- außer Modifikationen im TCP auf dem Mobilrechner, sind neue Netzwerkkomponenten erforderlich, z.B. Bandbreitenmanager.

6 Fast Retransmit / Fast Recovery

Das nächste Verfahren wurde entwickelt um Paketverluste während eines Handovers zu behandeln.

Im TCP bedeutet eine Quittung mit Sequenznummer n, dass der Empfänger alle Pakete bis einschließlich n korrekt erhalten hat. Geht das Paket $n+1$ verloren, bestätigt der Empfänger alle weiteren empfangenen Pakete immer wieder mit einer Quittung mit derselben Sequenznummer n, der Sender erhält also duplizierte Quittungen, siehe Abbildung 7, Teil a. Daraus kann der Sender schließen, dass der Empfänger weiterhin Pakete erhält (sonnst würde er gar keine Quittungen senden), es liegt also kein Stau im Netz vor. Die Lücke im Paketstrom ist nicht wegen Überlastung, sondern aufgrund eines Übertragungsfehlers entstanden. Nun kann der Sender das fehlende Paket erneut übertragen, ohne *slow start* zu starten. Dieses Verhalten des TCP-Senders heißt *fast retransmit*. Wenn das Netz nicht überlastet ist, besteht für den Sender auch kein Grund sich zurückzuziehen und das Überlastungsfenster verkleinern. Der Sender überträgt weiter mit unveränderter Leistung. Dieser Zustand heißt *fast recovery*. Dieser Mechanismus, in der mobilen Umgebung eingesetzt, kann die Effizienz des TCP-Protokolls enorm verbessern.

Wie im Abschnitt 2.1 beschrieben, kann es zu Paketverlusten oder Timeouts kommen wenn sich das mobile Benutzer bewegt und dabei vom alten zum neuen Proxy wechselt. Um die Partner-TCP-Instanzen daran zu hindern, in den *slow start* zu übergehen, muß man sowohl auf dem mobilen, als auch auf dem Festnetz-Rechner das *fast retransmit / fast recovery* Verhalten künstlich erzwingen. Sobald der Mobilrechner Kontakt zum neuen Proxy aufnimmt, beginnt er, duplizierte Bestätigungen an den Festnetz-Rechner zu senden. Nach Erhalt von drei Duplikaten übergeht der Festnetz-Rechner in den *fast retransmit* Modus. Das heißt, kein *slow start* wird gestartet, der Sender setzt die Übertragung mit der selben Leistung fort, wie vor dem Wechsel des Proxy.

Der Mobilrechner übergeht auch in den *fast retransmit* Zustand und wiederholt alle unbestätigten Pakete mit unverändertem Übertragungsfenster, ohne den *slow start* Algorithmus zu beginnen.

Der Vorteil dieses Verfahrens ist seine Einfachheit. Lediglich geringe Änderungen in der Software auf dem mobilen Endgerät sind notwendig, um Leistungssteigerung zu erreichen. Keine Änderungen im Festnetzbereich sind erforderlich.

Der Hauptnachteil des beschriebenen Schemas ist die unzulängliche Isolierung der drahtlosen Strecke. Verwendung vom *fast retransmit* verbessert zwar Effizienz, jedoch müssen Übertragungswiederholungen über das ganze Netzwerk zwischen Mobilrechner und Festnetz-Rechner laufen. Das Verfahren betrachtet nur Paketverluste während Handovers, Übertragungsfehler auf der Funkstrecke, die auch Paketverluste verursachen können, werden nicht berücksichtigt. Darüber hinaus, erfordert das Verfahren mehr Kooperation zwischen Mobile IP- und TCP-Schichten, es ist deshalb schwerer, das eine zu ändern, ohne das andere zu beeinflussen.

7 Transmission / Timeout Freezing

Fast retransmitt / fast recovery ist nützlich bei kurzen Verbindungsunterbrechungen. Im Falle lang anhaltender Unterbrechungen leisten sie aber keine Abhilfe, weil TCP-Instanzen nach einem Timeout die Transportverbindung abbauen. Das kann zum Beispiel passieren, wenn sich ein Benutzer in eine Zelle bewegt, deren Frequenzkapazität erschöpft ist. Oder wenn ein mobiles Endgerät in einem Auto benutzt wird, das gerade in einen Tunnel einfährt.

Ein Mechanismus, der lange Verbindungsunterbrechungen behandelt, erfordert Kooperation zwischen zwei OSI-Schichten: der MAC-Schicht und der TCP-Schicht. Anders als die TCP-Schicht, die in jedem Fall eine Überlastung im Netz vermutet, kennt die MAC-Schicht die wahre Ursache der Unterbrechung. Außerdem, erkennt sie ein Kommunikationsproblem früher, als die darüberliegende TCP-Schicht. Nun kann die MAC-Schicht die TCP-Schicht vor einer drohender Verbindungsunterbrechung warnen oder ihr ggf. mitteilen, das die aktuelle Transportverbindung nicht auf Grund Überlastung unterbrochen wurde. Der TCP-Sender kann jetzt die Übertragung stoppen und den aktuellen Zustand von Übertragungsfenster sowie Timers „einfrieren“. Sobald die MAC-Schicht wieder den Kontakt aufnimmt, signalisiert sie dies der TCP-Schicht. Die TCP-Instanz kann jetzt von jenem Punkt und mit jenem Zustand, wo sie aufgehört hat, weitermachen.

Der Vorteil diese Verfahrens ist, dass es eine Wiederaufnahme der TCP-Verbindung sogar nach langen Unterbrechungen ermöglicht. Darüber hinaus, ist es vom Paketinhalt unabhängig, also ist eine Datenverschlüsselung kein Problem, solange keine zeitabhängige Zufallszahlen verwendet werden. Das Verfahren hat aber auch einige Nachteile. Die nötige Änderungen betreffen nicht nur die Software auf dem Mobilrechner, sondern auch den Festnetz-Rechner. Alle Mechanismen sind von der Fähigkeit der MAC-Schicht bevorstehende Unterbrechungen zu erkennen abhängig.

8 Selektive Übertragungswiederholungen

Wie im Abschnitt 6 kurz beschrieben, verwendet TCP kumulative Quittungen. Beim Verlust eines einzelnes Pakets muß der Sender das verlorene, sowie alle anderen Pakete, die er danach gesendet hat, erneut übertragen. So entsteht eine riesige Menge überflüssiger Daten, die nicht nur in Mobilkommunikation, sondern auch im Festnetz unerwünscht ist. Deshalb wurde im RFC 2018 eine TCP-Version vorgeschlagen, die diese unangenehme Eigenschaft nicht hat: sie erlaubt Quittungen für einzelne Pakete, nicht nur für reihenfolgetreue und lückenlose. Der Sender kann nun genau feststellen, welches Paket fehlt und kann es erneut übertragen. Die Weg-Zeit-Diagramme in der Abbildung 7 machen den Unterschied zwischen verschiedenen Wiederholungsverfahren deutlich.

Der Vorteil dieses Verfahrens ist offensichtlich: der Sender wiederholt nur tatsächlich verlorengegangene Pakete, das spart die Bandbreite und entlastet sowohl den Sender, als auch den Empfänger. Selektive Übertragungswiederholungen können auch im Festnetz eingesetzt werden, nicht nur mobile, sondern auch alle anderen Netzwerkarten würden von dieser TCP-Erweiterung profitieren. Als einen Nachteil könnte man etwas komplexere Software auf der Empfängerseite sowie erhöhten Pufferspeicherbedarf nennen. Das ist jedoch kein richtiger Nachteil, wenn man berücksichtigt, dass CPU-Leistung sowie Speichergröße ein stetiges Wachstum erleben, während sich die Bandbreite für mobile Kommunikationen praktisch nicht ändert.

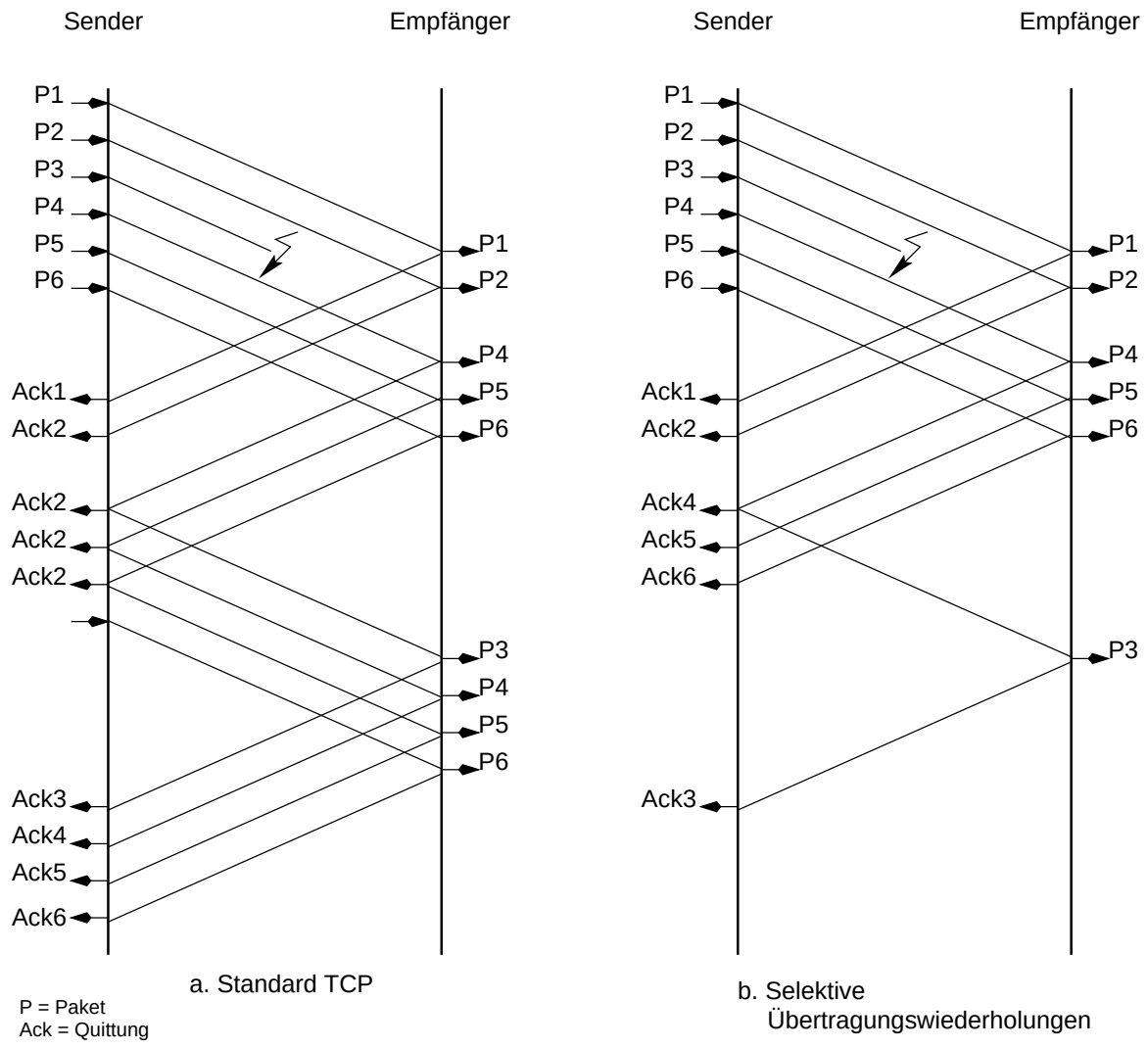

Abbildung 7: Sendewiederholungsverfahren im Vergleich.

9 Transaktionsorientiertes TCP

Es gibt Anwendungen, die miteinander nur ab und zu kurze Nachrichten austauschen. Wenn als Transport-Protokoll TCP eingesetzt wird, wird für jede Nachricht, auch wenn sie nur aus einem Paket besteht, eine extra Transport-Verbindung aufgebaut. Für Auf- und Abbau der Verbindung verwendet TCP 3-Wege-Handshake, d.h. insgesamt 7 Pakete werden benötigt um eine kurze Nachricht zu übertragen. Viel zuviel Overhead für die schmalen drahtlosen Verbindungen!

RFC 1644 schlägt ein transaktionsorientiertes TCP vor, das mit einem Minimum zusätzlicher Daten auskommt. Dieses Verfahren erlaubt, Pakete für Verbindungsaufbau, Daten und Verbindungsabbau zusammenzufassen. Das reduziert die benötigte Datenmenge von 7 auf 2-3 Pakete.

Der offensichtliche Vorteil für bestimmte Anwendungen ist die Verringerung der Datenmenge, die Standard-TCP für die Verbindungsauf- und -abbau benötigt. Genauso offensichtlich ist jedoch auch der Nachteil: das T-TCP ist nicht mehr das Standard-TCP, Änderungen würden alle existierenden TCP-Implementierungen betreffen.

10 Verschiedene Verfahren im Vergleich

Die oben beschriebenen Verfahren wurden vorgeschlagen, um die Effizienz vom TCP-Protokoll in der mobilen Umgebung zu verbessern. Tabelle 1 zeigt einen Überblick über die aufgeführten Mechanismen zusammen mit einigen Vor- und Nachteilen [Schi99]. Manche Verfahren können miteinander kombiniert werden, z.B. könne selektive Übertragungswiederholungen zusammen mit anderen Mechanismen und sogar im Festnetz eingesetzt werden

Verfahren	Mechanismus	Vorteile	Nachteile
Indirektes TCP	Auftrennen in zwei TCP-Verbindungen	Isolation der drahtlosen Strecke, einfach	Verlust der TCP Semantik, erhöhte Latenz
Snooping TCP	Mithören von Daten und Quittungen, lokale Wiederholung	Transparent für Ende-zu-Ende, Integration von MAC	Problematisch bei Verschlüsselung, schlechtere Isolation
M-TCP	Auftrennen der TCP-Verbindung, erzwingen beim Sender einer bestimmten Fenstergröße	Bewahrung der Ende-zu-Ende-Semantik, Behandlung langer und wiederholter Unterbrechungen	Schlechte Isolation der drahtlosen Strecke, overhead durch Bandbreiten-management
Fast Retransmit/ Fast Recovery	Vermeidung von *slow start* nach Verbindungswechsel	Einfach, effizient	Vermischung der Schichten, nicht transparent
Transmission/ Timeout Freezing	Einfrieren des TCP-Zustands bei Unterbrechung	Unabhängig von Dateninhalten, Verschlüsselung	Änderung von TCP, MAC-abhängig
Selektive Übertragungs-wiederholung	Wiederholung nur der echt verlorenge-gangenen Daten	Sehr effizient	Etwas komplexere Empfängersoftware, mehr Speicher
Transaktions-orientiertes TCP	Zusammenfassung von Verbindungsauf/--abbau und Datenpaketen	Effizient	Geändertes TCP, nicht mehr transparent

Tabelle 1: Überblick über verschiedene TCP-Verbesserungen für mobile Umgebung

Literatur

[BaB.95] A. Bakre und Badrinath B.R. I-TCP: indirect TCP for mobile hosts. *Distributed Computins Systems, 1995., Proceedings of the 15th International Conference on*, Dezember 1995, S. 136–143.

[BPSK97] H. Balakrishnan, V.N. Padmanabhan, S. Seshan und R.H. Katz. A comparison of mechanisms for improving TCP performance over wireless links. *Networking, IEEE/ACM Transactions on*, Dezember 1997.

[BrSi97] Kevin Brown und Suresh Singh. M-TCP: TCP for Mobile Cellular Networks. Juli 1997.

[BSAK95] H. Balakrishnan, S. Seshan, E. Amir und R.H. Katz. Improving TCP/IP Performance over Wireless Networks. *ACM Int'l Conf. on Mobile Computing and Networking (Mobicom)*, November 1995.

[Schi99] Jochen Schiller. *Mobile communications*. Addison-Wesley. 1999.

Mobile Ad-hoc Netzwerke

Barbara Pellkofer

1 Grundlagen

Der Bereich der drahtlosen LANs (wireless LANs/WLANs) ist derzeit stark im Wachstum begriffen. Es können dabei zwei verschiedene Arten von Architekturen unterschieden werden: Infrastruktur- und Ad-hoc Netzwerke. Beim Infrastrukturnetz sind WLAN(s) und Festnetz miteinander verbunden. Mobile Teilnehmer können entweder direkt miteinander kommunizieren oder auch über das Festnetz Teilnehmer in weiter entfernten WLANs erreichen. Nach diesem Prinzip funktionieren beispielsweise die Mobilfunknetze. Ad-hoc Netzwerke hingegen können spontan, auch ohne jegliche vorhandene Infrastruktur gebildet werden.

Realisierungen von Ad-hoc Netzwerken gibt es für die Schichten zwei und drei. Bluetooth entwickelt Hard- und Software auf Schicht zwei und Manet Protokolle und Standards für Schicht drei. Das zentrale Problem bei Ad-hoc Netzwerken stellt das Routing dar, da herkömmliche Algorithmen hier nicht effizient arbeiten.[Dani99]

1.1 Kennzeichen von Ad-hoc Netzwerken

Das hervorstechendste Merkmal von Ad-hoc Netzwerken ist, daß sie eigentlich keinerlei Struktur besitzen. Sie können jederzeit spontan gebildet werden und sind dabei unabhängig von vorhandener Infrastruktur. Alle Teilnehmer dürfen sich frei bewegen, so daß sich die Topologie ständig ändern kann. Jeder Knoten kann mit jedem anderen kommunizieren, wenn auch nicht immer unbedingt direkt, und kann die Fähigkeit zum Router haben. Die Verbindungen können sowohl symmetrisch als auch asymmetrisch und sowohl uni- als auch bidirektional sein. Ad-hoc Netzwerke können durchaus auch mit dem Festnetz verbunden sein, allerdings immer nur als End- und niemals als Transitnetz.[Schi99][M. S99b]

1.2 Beispiele für den Einsatz von Ad-hoc Netzwerken

Ein gutes Beispiel für den möglichen Einsatz von Ad-hoc Netzwerken sind Naturkatastrophen. In solchen Fällen ist das Festnetz meist zusammengebrochen, Rettungsmannschaften müssen aber trotzdem irgendwie miteinander kommunizieren können. Hier bieten sich Ad-hoc Netzwerke an, da sie sehr schnell verfügbar sind. Um weitere Strecken zu überbrücken, ist auch eine Kombination mit Satellitenübertragung möglich.

Dasselbe gilt für Konferenzen und Arbeitstreffen an beliebigen Orten, bei denen jeder Teilnehmer seinen Laptop mitbringt, um mit den anderen Teilnehmern Daten auszutauschen.
Ein weiterer Grund für den Einsatz von Ad-hoc Netzwerken ist, daß es teilweise zu teuer und aufwendig wäre, eine Infrasturktur aufzubauen. Das gilt beispielsweise für sehr abgelegene Gegenden.[M. S99b] [Schi99]

1.3 Probleme

Das größte Problem bei der Realisierung von Ad-hoc Netzwerken ist das Routing. Da keine Standardrouter vorhanden sind, sollte jeder Knoten weiterleiten können. Ansonsten muß jeder Knoten mit mindestens einem anderen Knoten in Verbindung stehen, der weiterleiten kann. Dies wird erschwert durch die stark dynamische Topologie des Netzwerks und die damit verbundenen ständigen Änderungen von Verbindungen und Verbindungsqualitäten sowie Anzahl und Aufenthaltsort der Teilnehmer. Auch die Asymmetrie der Verbindungen stellt ein Problem dar. Die meisten herkömmlichen Routingalgorithmen wurden für Festnetze mit symmetrischen Verbindungen entwickelt, so daß erst noch neue Algorithmen entwickelt werden müssen. Schließlich existieren natürlich auch die allgemeinen Probleme des Mobilfunks wie geringe Sicherheit und starke Energie- und Bandbreitenbegrenzung der Endgeräte.[M. S99b]

1.4 Technische Grundlagen

Bei der Realisierung von WLANs können verschiedene Übertragungstechniken und Mehrfachzugriffsverfahren eingesetzt werden.

Für die Datenübertragung gibt es grundsätzlich zwei Möglichkeiten: Funk und Infrarot. Die Infrarottechnik benutzt von IR-Dioden erzeugtes Licht mit Wellenlängen von 850 - 950 nm. Das Licht kann entweder diffus oder gerichtet abgestrahlt werden, wobei in letzterem Fall Sichtkontakt zwischen Sender und Empfänger notwendig ist. Diese Technik hat den Vorteil, daß sie sehr billig und einfach zu realisieren ist. Außerdem werden keine Lizenzen benötigt und die Geräte können leicht abgeschirmt werden. Nachteile sind, daß die Reichweite mit nur 2 - 10 m sehr gering ist und alle Verbindungen auf einen Raum beschränkt sind, da Licht keine Wände durchdringen kann, sondern reflektiert wird. Die Ausbreitung wird auch durch Abschattung und Interferenzen mit anderen Licht- und Wärmequellen beeinträchtigt.

Mit Funk können größere Flächen abgedeckt werden, da auch Wände druchdrungen werden können. Die Abschirmung ist jedoch schwieriger und es entstehen Interferenzen mit Elektrogeräten. Auch ist nur eine geringe Anzahl an Freuenzen für die Nutzung in WLANs freigegeben. Meist wird das lizenzfreie 2,4 GHz-Band genutzt.

Es werden die Bandspreiztechniken FHSS und DSSS verwendet. Bei Bandspreizverfahren wird die zu übertragende Nachricht mit einem Code kombiniert, wodurch sich die Bandbreite gegenüber der ursprünglichen Nachricht stark vergrößert. Bei DSSS (direct sequence spread spectrum) wird die Nachricht mit einer Codesequenz (Chipsequenz) mulitpliziert, bei FHSS (frequency hopping spread spectrum) wird auf schnell wechselnden Frequenzen gesendet, wobei die Frequenzwechsel ebefalls durch eine Codesequenz vorgegeben werden. Beide Verfahren sind resistent gegen Mehrwegeausbreitung und Interferenzen, da sich solche Störungen bei der Übertragung dem gespreizten Signal überlagern und durch das Entspreizen beim Empfänger nicht mehr stark ins Gewicht fallen.

Als Mehrfachzugriffsverfahren wird häufig Zeitmulitplex verwendet. [Sele]

2 Bluetooth

Bluetooth ist ein im Frühjahr 1998 von Ericsson, Intel, IBM, Nokia und Toshiba gegründetes Konsortium verschiedener Firmen und Forschungseinrichtungen, das die Entwicklung eines Chips zum Ziel hat, mit dem unterschiedliche Geräte möglichst billig drahtlos miteinander verbunden werden können. Es handelt sich dabei um Ad-hoc Netzwerke geringer Ausdehnung, sog. Piconetze oder Personal Area Networks [Schi99] [Over99]. Mittlerweile zählt das

Konsortium über 1200 Mitglieder. Das Konsortium schätzt, daß Bluetooth Ende 2001 der Standard in über 100 Mio. Mobiltelefonen und vielen PCs, Laptops u. a. sein wird [snew00]. Es soll sogar teilweise in den IEEE 802.15 Standard übernommen werden.

Extended Systems, ein führendes Unternehmen im Bereich des mobilen Informationsmanagements, hat schon früh den Bluetooth-Standard übernommen und vor kurzem die Entwicklung eines übertragbaren, eingebetteten Protokollsystems nach der Bluetooth Spezifikation R 1.0 fertiggestellt, das Anfang diesen Jahres auf den Markt kommen soll. Es verwendet Infrarottechnik für Soft- und Hardware, die durch Bluetooth ergänzt wird, so daß auch feste Körper durchdrungen werden können. Auch von der Firma IVT ist bereits ein Bluetooth-Protokollstack auf dem Markt.[Dr. 99] [Joan99]

2.1 Anwendungen

Anwendungen sind zum einen die drahtlose Anbindung von Peripheriegeräten wie Tastatur oder Drucker an einen Computer, zum anderen die Verbindung von Netzwerken, z. B. eines lokalen Rechnernetzes über ein Handy mit Bluetoothchip an das globale GSM-Netz. Selbstverständlich wird auch jede andere Art von Ad-hoc Netzwerken unterstützt. Der Vorteil von Bluetooth ist, daß es ein kleiner, billiger Chip ist und somit auch die Geräte klein und billig sein können.[Over99] [Schi99]

2.2 Technik

Bluetooth arbeitet im lizenzfreien 2,4 GHz-Band. Es werden Frequency Hopping und Zeitmulitplexverfahren eingesetzt. Die Sprungrate beträgt 1600 Hops/s. Zwischen zwei Sprüngen, in einem sog. Slot, wird höchstens ein Paket übertragen (ein Paket darf bis zu fünf Slots lang sein). In den meisten Ländern gibt es 79 verschiedene Sprungfrequenzen von jeweils 1 MHz Bandbreite, die zwischen 2,402 und 2,48 GHz liegen. In Japan, Frankreich und Spanien gibt es aufgrund nationaler Beschränkungen nur 23 Sprungfrequenzen. Jede der Frequenzen wird mit der gleichen Wahrscheinlichkeit benutzt. Die Reichweite eines Bluetoothgerätes beträgt bei einer Sendeleistung von 100 mW bis zu 10 m, kann aber bei entsprechend höherer Sendeleistung auf bis zu 100 m ausgedehnt werden.[Over99] [Schi99]

2.3 Netzwerkstruktur

Zwei bis acht miteinander verbundene Geräte, die alle die gleiche Sprungsequenz benutzen, bilden ein Piconet. Eines dieser Geräte, meist dasjenige, welches die Verbindung initiiert hat, ist der Master, die anderen sind Slaves. Der Master bestimmt die Sprungsequenz und synchronisiert das Piconet. Mehrere unabhängige und miteinander nicht synchronisierte Piconets können zu einem Scatternet verbunden werden (s. Abb. 1)..

Die Kommunikation zwischen verschiedenen Piconets in einem Scatternet findet über zwischen verschiedenen Piconets hin- und herwechselnde Geräte statt. Handelt es sich hierbei um Slaves, so müssen sie nur den Master ihres bisherigen Piconets informieren, daß sie vorrübergehend nicht erreichbar sind, und ihre Sprungsequenz mit der ihres neuen Piconets synchronisieren. Will ein Master vorrübergehend sein Netz verlassen, so kann dort während seiner Abwesenheit keinerlei Datenübertragung stattfinden. Im anderen Netz ist er dann nur ein Slave. Wäre er dort auch ein Master, so würden beide Netze dieselbe Sprungsequenz benutzen und es käme zu Kollisionen bei der Datenübertragung.

Jedes Gerät kann sich in einem von acht verschiedenen Zuständen befinden (s. Abb. 2). Zu Anfang befinden sich alle Geräte im Standby-Modus. Jedes Gerät hört alle 1,28 s für

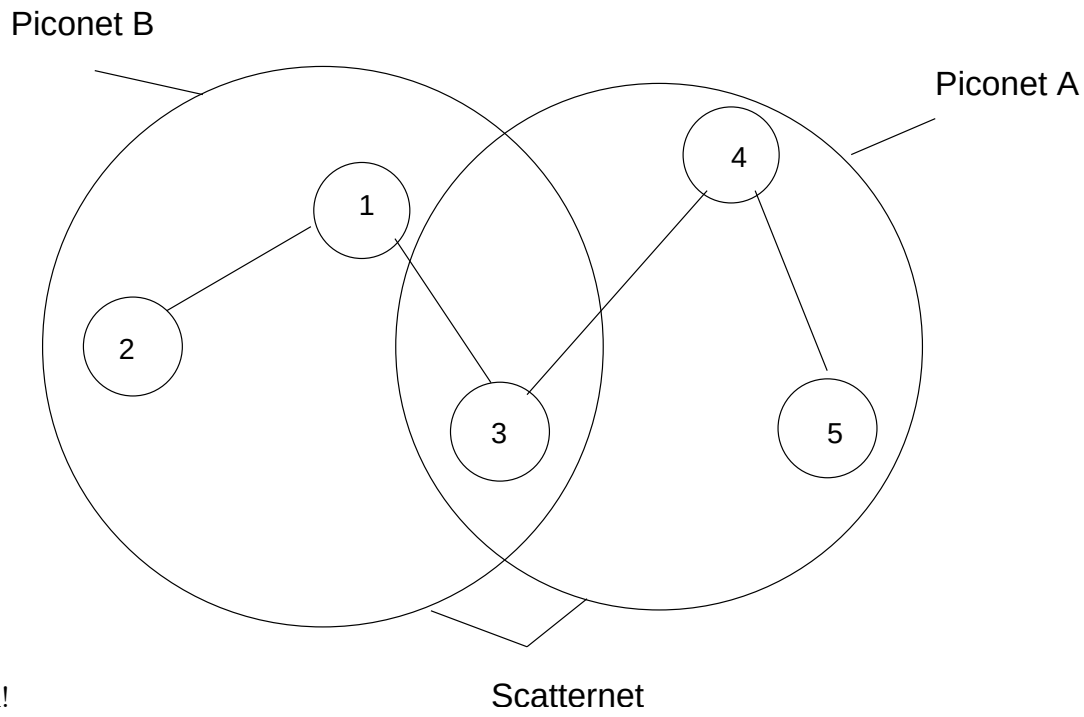

Abbildung 1: Piconets und Scatternet

11,25 ms eine seiner 32 (bzw. 16 in Japan, Frankreich und Spanien) Wake-up Frequenzen ab. Jedes Gerät kann eine Verbindung aufbauen, es wird dann selbst zum Master. Ist die Adresse des gewünschten Kommunikationspartners bekannt, so kann die Verbindung durch ein Page aufgebaut werden. Ist sie nicht bekannt, durch ein Inquiry gefolgt von einem Page. Im Zustand Page sendet der Master 16 identische Pagenachrichten auf 16 der 32 Wake-up Frequenzen des gewünschten Verbindungspartners. Wenn keine Antwort kommt, sendet er nochmals auf den anderen 16 Frequenzen. Die maximale Verzögerung für den Verbindungsaufbau beträgt 2,56 s, im Mittel nur 0,64 s. Im Zustand Inquiry geschieht etwas Ähnliches, allerdings ist es unter Umständen nötig, daß der Master ein drittes Mal seine Anfrage senden muß.

Da Bluetooth-Geräte nicht verkabelt sind und daher ihre Energie aus Batterien beziehen müssen, ist es sinnvoll, wenn die Geräte in einen Energiesparmodus übergehen können, wenn sie gerade keine Daten senden oder empfangen. Bluetooth bietet dafür drei verschiedene Möglichkeiten: den Park-, den Hold- und den Sniff-Modus. Der Park-Modus hat den geringsten Energieverbrauch. Das Gerät gibt seine MAC-Adresse auf, bleibt aber mit dem Piconet synchronisiert. Der Nachrichtenverkehr im Netz wird nur sporadisch abgehört, unter anderem um sich wieder zu synchronisieren. Den nächsthöheren Energieverbrauch hat der Zustand Hold. Die MAC-Adresse wird nicht aufgegeben und es kann jederzeit wieder gesendet werden. Den höchsten Energieverbrauch hat der Sniff-Modus. Das Piconet wird hierbei in vorgegebenen Intervallen abgehört.[Over99] [Schi99]

2.4 Datenübertragung

Bluetooth bietet zwei unterschiedliche Dienste für die Datenübertragung an: einen synchronen, verbindungsorientierten (synchronous connection-oriented, SCO) und einen asynchronen, verbindungslosen (asynchronous connectionless, ACL).

SCO wird hauptsächlich zur Sprachübertragung mit 64 kbit/s genutzt und erfordert eine symmetrische, leitungvermittelte, Punkt-zu-Punkt-Verbindung. Es werden in festen Abständen zwei aufeinanderfolgende Slots gesendet (einer für die Hin- und einer für die Rückrichtung).

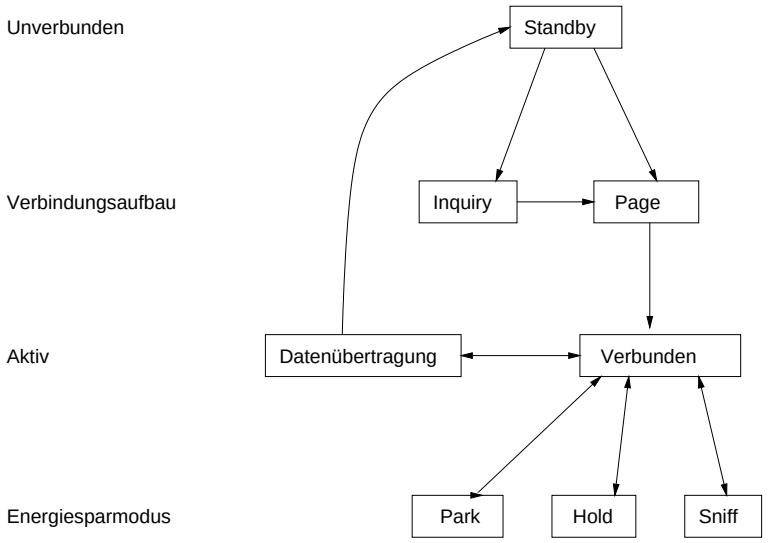

Abbildung 2: Zustände eines Bluetoothgerätes

ACL wird hauptsächlich zur Datenübertragung genutzt und erfordert Paketvermittlung, Punkt-zu-Mehrpunkt-Verbindungen und Polling durch den Master. Die Verbindung kann in diesem Fall sowohl symmetrisch als auch asymmetrisch sein. Bei einer symmetrischen Verbindung sind Datenraten von bis zu 432,6 kbit/s möglich, wofür bis zu fünf aufeinanderfolgende Slots benutzt werden können. Bei einer asymmetrischen Verbindung kann in die eine Richtung mit bis zu 721,0 kbit/s und in die andere mit 57,6 kbit/s gesendet werden, ebenfalls unter Benutzung von bis zu fünf Slots pro Richtung.

Über die pro Link jeweils real zur Verfügung stehende Bandbreite entscheidet der Master. Es können gleichzeitig ein ACL, drei SCOs oder ein ACL und ein SCO existieren. Jedes Master/Slave-Paar eines Piconets kann sich beliebig ändernde, unterschiedliche Arten von Links benutzen. Sowohl SCO als auch ACL unterstützen bis zu 16 verschiedene Paketarten, vier davon (Kontrollpakete) sind für beide gleich.

Alle Links werden von einem Linkmanager verwaltet, der vom Link Controller angebotene Dienste nutzen kann. Linkmanager in verschiedenen Knoten können über das Linkmanagerprotokoll miteinander kommunizieren. Aufgaben des Linkmanagers sind unter anderem Verbindungsauf- und abbau, Senden und Empfangen von Daten, Aushandeln der Linkart (Sprache oder Daten, kann im Laufe der Verbindung auch geändert werden), Herausfinden von Namen und Adresse anderer Knoten sowie Authentifizierung. [Over99] [Schi99]

2.5 Paketaufbau

Abb. 3 zeigt den allgemeinen Aufbau eines Paketes, das in einem Slot übertragen wird. Der Accesscode ist für jede Verbindung einzigartig. Jeder Empfänger vergleicht zuerst den Code eines ankommenden Paketes mit seinem gespeicherten Accesscode. Sind sie nicht identisch, so wird das Paket verworfen. Außerdem wird der Accesscode für die Synchronisation verwendet.

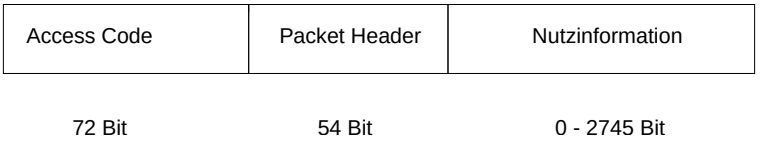

Abbildung 3: Allgemeiner Aufbau eines Paketes

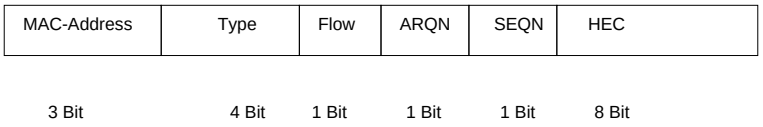

Abbildung 4: Packet Header

Die MAC-Adresse im Paketkopf (s. Abb. 4) ist drei Bit lang, da maximal acht Knoten zu einem Piconet gehören können. Type enthält vier Bit, da es bis zu 16 unterschiedliche Paketarten gibt. Für die Sequenz- und Achnowledgementnummern (SEQN, ARQN) genügt ein Bit, da eine Bestätigung, falls erforderlich, immer gleich im auf die Daten folgenden Slot gesendet wird. Der Kopf wird durch eine acht Bit lange Prüfsumme (Header Error Check, HEC) gesichert. Zusätzlich wird noch 1/3 FEC (Forward Error Correction) angewandt, da der Header wichtige Informationen enthält, die nicht durch ein paar Bitfeher verloren gehen sollen. Dadurch verdreifacht sich allerdings die Länge des Headers auf 54 Bit.

Im allgemeinen entspricht ein Paket genau einem Slot, es kann aber auch drei oder fünf Slots lang sein. Ein Paket wird immer auf derselben Sprungfrequenz gesendet, so daß im Falle eines mehrere Slots langen Paketes zwei bzw. vier Frequenzen in der Sprungsequenz übersprungen werden müssen. Würde die Sprungsequenz nur einfach verzögert, so wäre sie nicht mehr synchron mit dem Rest des Piconets, was zum Zusammenbruch des Netzes führen würde.

Im Falle eines SCO-Links sind alle Pakete immer nur einen Slot lang. Es kann aber zwischen keiner Fehlerkorrektur, 1/3 FEC und 2/3 FEC gewählt werden, wobei allerdings die Vervielfachung der Länge zu berücksichtigen ist. Sprache wird mit einem sehr störsicheren Verfahren codiert, so daß Sprachpakete eigentlich nie wiederholt werden müssen.

Bei ACL sind Pakete einen, drei oder fünf Slots lang und es kann wahlweise 2/3 FEC angewandt werden. Da der Overhead bei FEC aber sehr hoch ist, bietet Bluetooth noch eine andere Möglichkeit: Automatic Repeat Request (ARQ). Hierbei wird jedes Paket gleich im folgenden Slot bestätigt. Geht ein Paket verloren, so kann der Sender es sofort im nächsten Slot nach Erhalt der negativen Bestätigung nochmals senden. [Over99] [Schi99]

2.6 Sicherheit

Eine sichere Übertragung umfaßt zum einen Fehlerfreiheit, zum anderen Abhörsicherheit und Schutz vor unberechtigten Zugriffen. Um Übertragungsfehler zu reduzieren, können FEC oder ARQ verwendet werden (s. Abschnitt 2.5). Um Abhörsicherheit und Schutz vor unberechtigten Zugriffen zu gewährleisten, wurden Authentifizierungs- und Verschlüsselungsroutinen auf der MAC-Schicht implementiert. Zur Verschlüsselung wird eine Stromchiffre mit Schlüssellängen von 0, 40 oder 64 Bit verwendet. Authentifizierung erfolgt mittels Challenge-Response. Für jede Verbindung kann eine Authentifizierung verlangt werden. Die Algorithmen funktionieren nach dem Public-Key-Prinzip mit einem zufällig generierten Sitzungsschlüssel, für jede Transaktion ein Neuer.

Diese Maßnahmen bieten zwar noch keine große Sicherheit, aber stärkere Verschlüsselungsalgorithmen können zusätzlich auf höheren Schichten implementiert werden. [Over99] [Schi99]

3 MANET

MANET ist die Bezeichnung für die IETF (Internet Engineering Task Force) Arbeitsgruppe für Mobile Ad-hoc NETworking. Ziel dieser Arbeitsgruppe ist es, einen effizienten Routingalgorithmus für rein mobile, drahtlose Netze zu entwickeln und zu standardisieren. Außerdem

sollen Schnittstellenstandards definiert werden, die IP-basierte, autonome, mobile Netzwerke unterstützen. [M. S99b]

3.1 Anforderungen an Routingprotokolle für Ad-hoc Netzwerke

Vorerst hat Manet sich zum Ziel gesetzt, ein oder mehrere Unicast- Routingprotokolle für Ad-hoc Nerzwerke und die dafür benötigten Dienste der Schicht drei zu standardisieren , die folgende Anforderungen erfüllen sollen:

- Unterstützung von traditionellem, verbindungslosem IP
- Schnelle Anpassungsfähigkeit an topologische Änderungen und Änderungen des Verkehrsaufkommens ohne Verlust der Leistungsfähigkeit
- Unterstützung möglichst vieler möglicher Charakteristika mobiler Netze

Es sollen auch Adressierung, Sicherheit sowie die Zusammenarbeit mit höheren und tieferen Schichten berücksichtigt werden. Auf lange Sicht sollen weitere Dienste entwickelt werden, die auf das ursprüngliche Routingprotokoll aufsetzen. Zu diesen Erweiterungen sollen beispielsweise Mulitcast und Dienstgütegarantien gehören.

Mobiles Routing auf Schicht drei muß mit verschiedenen Infrastrukturen, d. h. unterschiedlichen Übertragungstechniken, Zugangsprotokollen, usw. zusammenarbeiten. Ein Manet-Knoten ist im Prinzip ein Router, der mit mehreren Hosts über evtl. unterschiedliche Schnittstellen, die unterschiedliche Technologien verwenden, verbunden ist. Ein Knoten der Schnittstellen für unterschiedliche Technologien besitzt, kann zu jedem anderen Knoten eine Verbindung aufbauen, der eine Schnittstelle für eine dieser Technologien besitzt. [M. S99b]

3.2 Manet-Technologie

Ein mobiles Internet in Manet-Technologie besteht aus mobilen Routern und Hosts, von denen jede dauerhaft oder zeitweise den Routern zugeordnet ist. Dabei kann ein Gerät durchaus auch Host und Router in einem sein. Es ist keinerlei Unterstützung durch das Festnetz nötig. Die Router sind entweder gar nicht mit dem Festnetz verbunden oder über mindestens eine Zwischenstation.

Sowohl das Internet als auch ein Manet haben beschränkte Ressourcen, wenn auch unterschiedliche. Im Internet soll in den Routern so wenig wie möglich verarbeitet und gespeichert werden. Daher wird versucht, die vertikale Kommunikation zwischen den verschiedenen Schichten eines Routers durch verstärkte horizontale Kommunikation innerhalb einer Schicht zwischen zwei Routern zu minimieren, was auf Kosten der Bandbreite geht. Im Manet ist es genau umgekehrt. Da nur eine beschränkte Bandbreite zur Verfügung steht, wird die horizontale Kommunikation durch verstärkte vertikale Kommunikation verringert. Die höheren Schichten nutzen hierbei verstärkt Funktionen der unteren Schichten, v. a. von Schicht zwei, um effizienter zu arbeiten. Es wurde vorgeschlagen, wenn möglich RTS/CTS aus dem IEEE-Standard 802.11 zu verwenden, um Informationen über Verbindungen zu Nachbarknoten zu erhalten, wodurch die Leistung erhöht wird. Wenn dies nicht möglich ist, so kann auch CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) verwendet werden.

Eine Schnittstelle zwischen IP und IEEE 802.11 würde es IP-Routingalgorithmen wesentlich einfacher machen, Schicht-2-Dienste wie 802.11 oder in Zukunft vielleicht auch Bluetooth zu verwenden. Das Problem bei diesem Ansatz mit erhöhter vertikaler Kommunikation ist, daß er nicht so einfach und flexibel ist wie die herkömmliche Methode und es Probleme

bei der Zusammenarbeit mit bereits existierenden Internetprotokollen geben kann. Um die Funktionalität von Schicht-4-Diensten wie TCP zu erhöhen ohne die Zusammenarbeit mit existierenden Netzwerken zu gefährden, können nur die untersten drei Schichten stärker integriert werden, um die Leistung zu steigern. Hierbei wird immer noch die Möglichkeit offen gelassen, effizientere integrierte Protokolle speziell für Manets zu entwickeln. [M. S99a]

3.3 Anwendungen für Manet

Die Vorteile von Manet sind: Kosteneffektivität, Flexibilität, Zusammenarbeit mit anderen Systemen und Unabhängigkeit von physikalischen Medien. Wegen der geringen Kapazität können allerdings noch keine Hochgeschwindigkeits- oder Weitverkehrsnetze durch Manet realisiert werden. Vorerst ist Manet vor allem für Netzwerke mit weniger als 100 Knoten geeignet. Aber auch hybride Netze mit WLAN-Technologien wie HiperLAN oder IEEE 802.11, die durch Manet-Routing verbunden sind, sind denkbar. Hierdurch können auch entlegene Gegenden ohne Kommunikationsinfrastruktur an das Internet angeschlossen werden. Auch Einbindung von Satelliten ist möglich.

In der Zukunft müssen vor allem noch die Kapazität und die Sicherheit erhöht werden, sowie die Adress- und Aufenthaltsortsverwaltung der Knoten verbessert werden. Höhere Kapazität kann z B. durch verbesserte Techniken auf den Schichten eins und zwei (z. B. Space Division Multiple Access) erreicht werden. Eine selbstorganisierende Adressverwaltung auf Schicht drei wäre nötig, um Manet allgemein in selbstorganisierenden Netzwerken einsetzen zu können. Eine weitere Herausforderung ist die Zusammenarbeit mit dem Festnetz einschließlich Satelliten und Knoten in der Luft. Auch eine verteilte, bandbreiteneffiziente Sicherheitsarchitektur ist für eine weite Verbreitung notwendig. [M. S99a]

4 Routing

Routing in Ad-hoc Netzwerken ist sehr schwierig. Das Hauptproblem ist die dynamische Topologie mit sich ständig ändernden Verbindungen. Auch ist nicht jeder Knoten von jedem anderen aus direkt zu erreichen, so daß erst ein Weg von einem zum anderen gefunden werden muß. Es muß außerdem jeder Knoten in der Lage sein, Daten für andere Knoten weiterzuleiten. Ein weiteres Problem ist, daß Verbindungen nicht unbedingt symmetrisch sind. D. h. vorhandene Informationen über eine Richtung sagen nichts über die Gegenrichtung aus. Auch die Redundanz stellt ein Problem dar, da sie nicht wie in Festnetzen kontrolliert werden kann. Dies kann bis zu vollvermaschten Netzen gehen, was einen großen Overhead beim Aktualisieren der Routingtabellen bedeutet. Das weiteren kann es zu Interferenzen zwischen einzelnen Links kommen.

In Anbetracht all dieser Probleme können folgende Aussagen über das Routing in Ad-hoc Netzwerken gemacht werden [M. S99b]:

- Für das Festnetz entwickelte Algorithmen sind nicht effizient einsetzbar
- Es werden nicht nur Informationen der Schicht drei, sondern auch von den tieferen Schichten benötigt, beispielsweise über Interferenzen
- Da eine Verbindung nicht lange aufrechterhalten werden kann, arbeiten Ad- hoc Netzwerke verbindungslos. Alle Knoten müssen in der Lage sein, Pakete ungefähr in Richtung Ziel weiterzuleiten.
- Zur Not ist auch Fluten möglich, z. B. bei total unbekannter Topologie. Es ist allerdings sehr uneffizient.

- Hierarchisches Clustern von Knoten kann hilfreich sein, da sich ganze Cluster im allgemeinen weniger schnell bewegen als einzelne Knoten, so daß Routing zwischen Clustern einfacher ist.

4.1 Mögliche Ansätze

Routingalgorithmen für Ad-hoc Netzwerke können entweder nach der Art, auf die die Router ihre Routingingformationen erhalten, oder nach der Art der Informationen die sie benutzen, um Wege zu berechnen, eingeteilt werden. Bezüglich der Art, auf die die Router ihre Informationen erhalten, werden Table-Driven- und On-Demand-Protokolle unterschieden, bezüglich der verwendeten Informationen Link-state und Distance-Vector-Protokolle.

Link-State-Protokolle benutzen Informationen über die Topologie, Dintance-Vector-Protokolle verwenden Entfernungen und Informationen über Pfade zu Zielknoten. Bei On-Demand-Protokollen speichern die Router nur Informationen über Knoten, an die sie wirklich etwas zu senden haben. Kennt ein Router den Weg zu einem Knoten nicht, so flutet er eine Suchanfrage durch das Netz, um die gewünschte Information zu erhalten. Table-Driven-Protokolle verwenden Routingtabellen, in denen Informationen über Wege zu allen bekannten Knoten des Netzes gespeichert werden.[Dr. 99]

4.2 AMRIS

AMRIS steht für Ad-hoc Multicast Routing protocol utilizing Increasing id-numberS [C. W98]. Es handelt sich dabei im Prinzip um einen Baum, bei dem die Kennungen der Knoten umso größer werden, je weiter sie von der Wurzel entfernt sind. Gleichweit entfernte Knoten haben gleiche Kennungen. Die Wurzel heißt Sid (Smallest ID) und hat die kleinste Kennung.

Solch ein Netz kann von irgendeinem beliebigen Knoten erzeugt werden, der ein NewSession sendet, das unter anderem die Kennung und Adresse des Knoten enthalten muß. Dieser Knoten wird dann zum Sid. Alle Knoten, die das NewSession empfangen, erzeugen gemäß einer Funktion F1 ihre eigene Kennung und senden das Paket mit ihrer eigenen Kennung statt der des Sid weiter (s. Abb. 5a). Die mit dem NewSession-Paket erhaltenen Informationen werden in der Nachbarstatustabelle gespeichert. Zu diesem Zeitpunkt sind im allgemeinen die Kennungen der Knoten noch proportional zu ihrer Entfernung vom Sid. Es handelt sich allerdings nicht um aufeinanderfolgende Zahlen, vielmehr liegen große Abstände zwischen den einzelnen Zahlen, um später das Einfügen neuere Knoten zu erleichtern, da auf diese Art nicht immer gleich alle nachfolgenden Knoten umbenannt werden müssen, um die Konsistenz zu wahren (s. Abb. 5b).

Will ein neuer Knoten X in das Netz, so sendet er ein Join-Req entweder an einen Nachbarn der bereits zum Netz gehört und eine kleinere Kennung hat, oder an einen Nachbarn, der zwar eine kleinere Kennung hat, aber noch nicht zum Netz gehört. Diese Anfrage wird entweder mit Join-Ack oder Join-Nack beantwortet, je nachdem, ob der Verbindungswunsch erfüllt werden kann oder nicht. Ein Grund für ein Join-Nack wäre beispielsweise, wenn X eine zu kleine Kennung hat, da dadurch das Netz inkonsistent werden würde (s. Abb. 5c). Im zweiten Fall, wenn der Nachbar selbst noch nicht zum Netz gehört, muß er selbst erst mittels Join-Req eine Verbindung zum Netz aufbauen. Hat X keine Nachbarn, die irgendwie mit dem Netz verbunden sind oder zumindest eine Kennung haben, so sendet X ein allgemeines Join-Req mit einer Lebensdauer von eins und einer Reichweite R. Alle Knoten, die nicht weiter als R von X entfernt sind, erhalten die Nachricht und senden nun ihrerseites Join-Req.passive, wobei R jedesmal um eins erniedrigt wird. Erhält ein Knoten, der bereits zum Netz gehört,

ein Join-Req.passive, so beantwortet er es durch ein Join-Ack.passive, wodurch ein passiver Link erzeugt wird. X erhält nun evtl. mehrere Join-Ack.passive, wählt eines davon aus und beantwortet es mit einem Join-Conf. Dadurch wird aus dem passiven Link ein aktiver. Die übrigen passiven Links werden entweder durch ein Timeout zerstört oder aber von anderen Knoten genutzt (s. Abb. 5c).

Es gibt vier verschiedene Arten von Knoten: Interessierte, uninteressierte, Blätter und innere Knoten. Interessierte Knoten wollen entweder als Sender oder als Empfänger mit dem Netz verbunden sein. Uninteressierte wollen eigentlich nicht, werden aber dazu gezwungen, um einen interessierten Knoten ans Netz anzubinden. Hat ein uninteressierter Knoten keine interessierten Nachfolgeknoten mehr, so löst er sich wieder vom Netz. Blätter sind Randknoten, die keine weiteren Nachfolger haben. Sie können sowohl senden als auch empfangen. Wird ein uninteressierter Knoten zu einem Blatt, so löst er sich vom Baum. Innere Knoten befinden sich im Innern des Baumes, zwischen Sid und Blättern. Sie können sowohl interessiert als auch uninteressiert sein (s. Abb. 5d).

Das Weiterleiten von Paketen geschieht auf folgende Art und Weise: Stellt ein Knoten fest, daß er selbst das empfangene Paket geschickt hat, so verwirft er es wieder. Erhält er es von seinem Vaterknoten, so leitet er es an seine Söhne weiter. Erhält er es von einem seiner Söhne, so leitet er es an seinen Vaterknoten und die anderen Söhne weiter. Stammt das Paket aber von einem beliebigen anderen Knoten, so wird es verworfen (s. Abb. 5e).

Bricht ein Link ab, so geht der Knoten mit der größeren Kennung in den BR-Modus (Branch Reconstruction) über. Ein Knoten X im BR-Modus sendet ein Join-Req mit R = 2. Erhält er darauf keine Antwort, so wird R schrittweise bis hin zu einem Grenzwert erhöht (s. Abb. 5f). Kommt dann immer noch keine Antwort, so existiert das Netz entweder nicht mehr oder es ist geteilt worden. In diesem Fall sendet X eine New-Partition-Id-Nachricht an alle seine Söhne (s. Abb. 5g). Hat er keine, so kann er selbst ein neues Netz aufbauen. Erhält er aber ein Join-Ack, so ist er wieder mit dem Netz verbunden und kann seine Nachbarstatustabelle aktualisieren. Wenn X ein Join-Ack mit der Fehlermeldung, daß seine Kennung zu klein ist, erhält, so erhöht er sie entsprechend.

Werden mehrere Knoten gleichzeitig vom Netz getrennt, so senden nicht alle auf einmal ihr Join-Req, sondern nur ein Teil von ihnen. Die anderen warten ab, ob eine Antwort kommt, um dann gegebenenfalls gleich direkt an den antwortenden Knoten ihr Join-Req zu senden.

Will ein Blatt seinen Vaterknoten verlassen, so muß es ihm dies nicht explizit mitteilen, er merkt dies von alleine, wenn er keine Meldungen mehr von diesem Blatt erhält. Der Blattknoten kann sich sofort zu seinem neuen Aufenthaltsort bewegen und dort versuchen, wieder eine Verbindung zum Netz aufzubauen. Dafür kann es unter Umständen nötig sein, seine Kennung zu erhöhen.

Will ein innerer Knoten seinen Platz verlassen, so bricht er nur die Verbindung zu seinem Vater ab, seine Söhne nimmt er mit. Beim Aufbau einer neuen Verbindung ist zu beachten, daß im Falle einer Änderung der Kennung gegebenenfalls auch die Kennungen der Söhne entsprechend geändert werden müssen (s. Abb. 5h).

Wird ein Netz geteilt, so übernimmt in jedem Teilnetz derjenige Knoten mit der kleinsten Kennung die Stelle eines temporären Sid. Empfängt ein Knoten solch einer Partition ein Signal aus einer anderen Partition, so informiert er umgehend den temporären Sid. Haben die beiden temporären Sids unterschiedliche Kennungen, so können die beiden Partitionen miteinander verbunden werden, wobei derjenige temporäre Sid mit der kleineren Kennung der Sid bleibt. Haben sie aber beide die gleiche Kennung, so ist dies nicht möglich, da nicht eindeutig festgelegt ist, wer der neue Sid wird.

Alle Knoten können das Netz jederzeit verlassen, nur der Sid muß vorher einen Nachfolger bestimmen. Beendet werden kann ein Netz nur vom Sid mittels eines SessionEnd. Bei geteilten

Netzen kann es allerdings vorkommen, daß nicht alle Teile diese Nachricht erhalten, obwohl sie noch eine Weile gespeichert wird. Eine andere Möglichkeit bietet deshalb der Timeout. Ein Netz ist beendet, wenn alle Einträge in seinen Nachbarstatustabellen abgelaufen sind.

Die Nachteile dieses Protokolls sind, daß es bidirektionale Links und periodisch gesendete Nachrichten benötigt. Diese periodischen Nachrichten haben die Funktion zu überprüfen, ob ein Link noch existiert und werden zwischen allen benachbarten Knoten ausgetauscht.

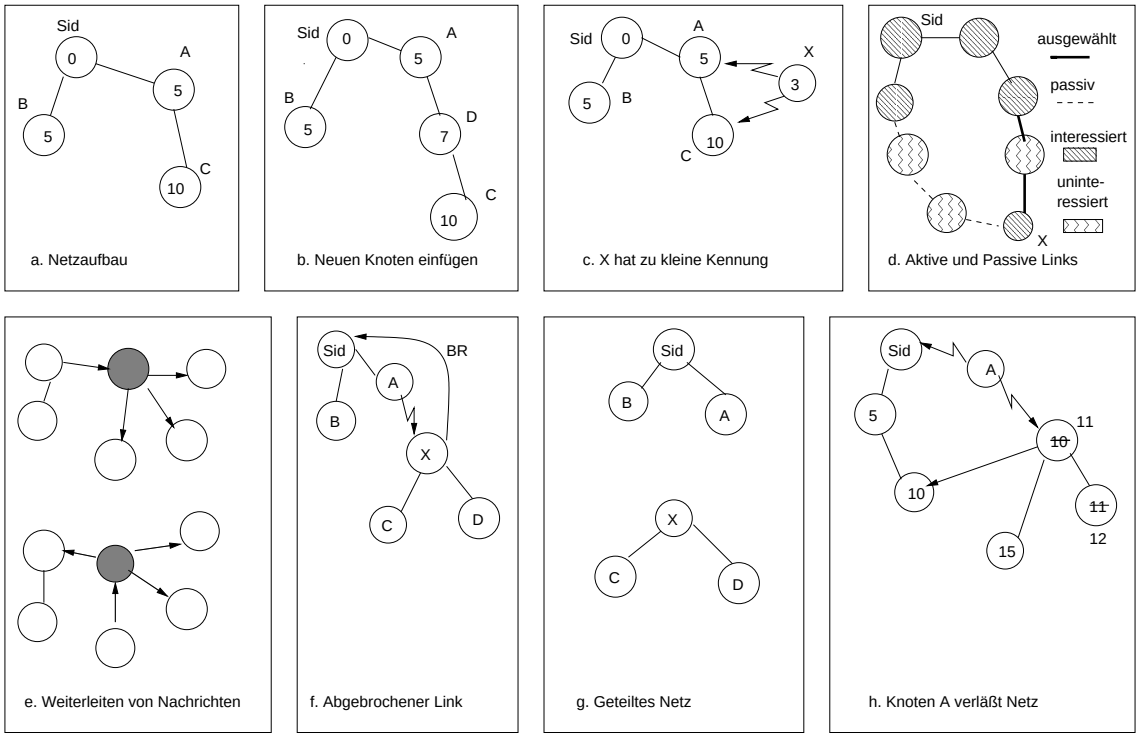

Abbildung 5: Beispiel zu AMRIS

4.3 Dynamic Source Routing (DSR)

Bei diesem Protokoll [Josh99] [Schi99] bestimmt der Sender den kompletten Weg, den ein Paket nehmen muß. Diese Information ist im Kopf des Paketes enthalten, so daß alle Knoten auf dem Weg das Paket richtig weiterleiten können, ohne selbst irgendwelche Informationen über den Weg speichern zu müssen.

Ein weiterer Vorteil ist, daß keine periodischen Nachrichten für das Aktualisieren der Routingtabellen nötig sind. Dadurch werden deutlich der Bandbreite- und Energiebedarf sowie die Kollisionswahrscheinlichkeit zweier Pakete gesenkt.

Dynamic Source Routing paßt sich schnell an Veränderungen der Netzwerktopologie an und verursacht keinen Overhead zu Zeiten in denen sich nichts ändert, da sowieso keinerlei Routiginformationen ausgetauscht werden. Außerdem berücksichtigt DSR auch asymmetrische Links.

DSR teilt das Routing in zwei Bereiche: Route Discovery und Route Maintenance. Route Discovery ist dabei für die eigentliche Wegfindung von der Quelle zum Zeil zuständig. Route Maintenance überprüft den gefundenen Pfad bei jeder Üertragung, ob er noch vollständig intakt ist und existiert. Ist beispielsweise ein Link abgebrochen, so informiert Route Maintenance den Sender sofort, daß der bisher verwendete Weg nicht mehr existiert. Daraufhin kann der Sender irgendeinen anderen ihm bekannten Weg zum Zielknoten verwenden oder aber mit Hilfe von Route Discovery einen neuen Weg finden.

Hierzu schickt er ein Route Request Paket mit seiner eigenen Adresse sowie der des Zielknotens an alle seine Nachbarn. Jeder Knoten der das Paket empfängt hängt, falls er auch keinen Weg zu dem gewünschten Zielknoten kennt, seine Adresse an den bisher zurückgelegten Weg des Paketes an und schickt es an seine Nachbarn weiter.

Kennt ein Knoten einen Weg zum Ziel, so hängt er seine eigene Adresse und den restlichen Weg an den bisher zurückgelegten Weg an und sendet das Paket zurück. Ansonsten wird das Paket solange weitergeschickt, bis es entweder den Zielknoten erreicht, seine Lebenszeit abgelaufen ist oder es zum zweiten Mal denselben Knoten erreicht (jeder Knoten speichert deshalb die Kennungen der letzten Pakete, die er weitergeschickt hat).

Das Zurücksenden der Antwort geschieht bei bidirektionalen Verbindungen einfach entlang des umgekehrten Weges. Bei unidirektionalen Verbindungen muß nochmals mittels Route Discovery ein Weg zurück gesucht werden. Der mit dem Route Request übertragene Weg wird von allen Knoten, die das Paket durchlaufen hat, in einem Route Cache gespeichert. Dadurch wird vermieden, jedesmal wieder Route Discovery aufrufen zu müssen.

DSR bietet kein richtiges Mulitcasting, dafür aber ein kontrolliertes Fluten zu allen Knoten des Netzwerks, die sich innerhalb einer bestimmten Entfernung befinden. Dies geschieht mittels Piggybacking. Die zu übertragenden Daten werden in ein Route Request Paket gepackt, welches als Ziel die Multicastadresse enthält.

Route Maintenance verlangt, daß ein gesendetes Paket sicher beim nächsten Knoten ankommt. Kann der Sender dies nicht sicher überprüfen, so muß er davon ausgehen, daß der Link abgebrochen ist. Überprüfen, ob ein Paket sicher angekommen ist, kann man beispielweise anhand einer expliziten Bestätigung, einer Bestätigung der Schicht zwei oder indem man einfach beobachtet, wie der nächste Knoten es wieder weiterschickt (passive Bestätigung). Route Maintenance ist nicht erlaubt für Route Requests und Bestätigungen, da sonst durch Bestätigungen für Bestätigungen eine Endlosschleife entstehen würde.

4.4 Vergleich zwischen Table-Driven und On-Demand Routingprotokollen

Im Rahmen einer Studie an der Unversity of California und dem Georgia Institute of Technology wurden verschiedene Routingalgorithmen simuliert und ihre Leistungsfähigkeit und Effizienz hinsichtlich Signalisierungsaufwand, benötigter Speicherplatz, Durchsatz sowie Verzögerung getestet. Bei den Algorithmen handelte es sich zum einen um DBF (Distributed Bellman Ford), einen herkömmlichen Table-Driven Algorithmus für das Festnetz. Jeder Knoten hat hierbei eine Routingtabelle mit den Entfernungen zu allen anderen Knoten und tauscht diese Informationen in periodischen Nachrichten oder wenn sich eine Verbindung plötzlich ändert (z. B. abbricht) mit allen seinen Nachbarn aus und berechnet aus den neu erhaltenen Informationen wieder die kürzesten Wege. Dies ist in mobilen Netzwerken extrem ungünstig, da sich hier Verbindungen andauernd ändern und somit sehr viele Nachrichten mit Routinginformation über das Netz geschickt werden müssen.

Der zweite Algorithmus war DSR (s. Abschnitt 4.3) und der dritte ABR (Associativity Based Routing), ebenfalls ein On-Demand Algorithmus für Ad-hoc Netzwerke. Alle Knoten senden periodische Signale aus und je nach dem, wie häufig sie von anderen Knoten solche Signale empfangen wissen sie, ob es sich um eine stabile Verbindung handelt oder ob der andere Knoten nur vorbeiwandert. Stabile Verbindungen werden stets bevorzugt, auch wenn dadurch der Weg weiter wird. Jeder Knoten hat eine Nachbarntabelle in der er die Anzahl der erhaltenen Signale speichert, eine Tabelle in der kürzlich weitergeleitete Nachrichten gespeichert werden, damit sie nicht immer wieder weitergeleitet werden, sollten sie wieder zurückkommen und eine Routingtabelle, in der alle momentan benötigten Wege mit Quelle und Ziel, Vorgänger

und Nachfolger gespeichert sind. Wird ein Pfad unterbrochen, so wird nur das unterbrochene Stück überbrückt, nicht der gesamte Weg neu gesucht.

Die Simulation wurde mit 30 mobilen Hosts mit einem Senderadius von 5 m in einem 20 m x 20 m großen Gebiet durchgeführt. Bei fast allen Versuchen schnitt DBF wesentlich schlechter ab als DSR und ABR, zwischen denen kein großer Unterschied war. Einzig und allein was den Speicherbedarf bei erhöhtem Verkehrsaufkommen betrifft, war DBF besser als die anderen. Es konnte somit gezeigt werden, daß On-Demand Algorithmen für mobile Netzwerke wesentlich besser geeignet sind. ABR hat zwar einen höheren Durchsatz, eine geringere Verzögerungszeit und sendet weniger Kontrollnachrichten, dafür benötigt DSR weniger Speicherplatz und verhält sich absolut still, wenn gerade keine Daten verschickt werden müssen. ABR hingegen sendet weiterhin seine periodischen Signale. Dies kann beispielsweise bei militärischen Anwendungen unerwünscht sein, da dadurch die Entdeckungswahrscheinlichkeit des Senders steigt. [Sung99]

5 Zusammenfassung

Die totale Unabhängigkeit von jeglicher Infrastruktur und die extrem dynamische Topologie sind die hervorstechendsten Merkmale von Ad-hoc Netzwerken, aber sie verursachen auch Probleme bei der Realisierung. Vor allem das Routing ist ein großes Problem, da die herkömmlichen Algorithmen nicht effizient anwendbar sind. Die Manet Working Group hat hierzu mehrerer Protokolle vorgeschlagen mit dem Ziel, einen Standard zu definieren.

Bluetooth arbeitet auf den Schichten eins und zwei. Es handelt sich dabei um eine billige Technik für Ad-hoc Netzwerke geringer Ausdehnung und bietet eine integrierte Hardware-/Software Lösung mit Kompatibilität zu allen anderen Bluetooth-Geräten. Bluetooth wurde von einem Konsortium großer Firmen entwickelt, die so über Marktbeherrschung einen defacto-Standard schaffen wollen. In Anbetracht dessen, daß es teilweise in den IEEE 802.15 Standard aufgenommen werden soll, scheint dies auch gelungen zu sein. Das erste Gerät, daß diese Technik verwendet ist ein Kopfhörer von Ericsson, der über Bluetooth an ein Mobiltelefon angeschlossen werden kann. Er soll ab Mitte des Jahres im Handel erhältlich sein [Rele00].

Manet dagegen arbeitet vor allem auf Schicht drei, unter Verwendung von Diensten tieferer Schichten. Es ist eine reine Softwarelösung und hardwareunabhängig und wurde von einer Arbeitsgruppe der IETF entwickelt. Auch hier ist das Ziel, einen Standard zu definieren. Manet kann auch auf Bluetooth aufbauend verwendet werden oder in hybriden Netzen zusammen mit anderen Technologien. Manet ist zwar momentan noch nicht sehr leistungsfähgig, ist aber im Prinzip auch für Weitverkehrsnetze geeignet. Dafür sind allerdings noch Verbesserungen der Technik auf den beiden untersten Schichten notwendig sowie eine verbesserte Adressverwaltung auf Schicht drei.

Literatur

[C. W98] C.-K. Toh C. W. Wu, Y. C. Tay. Ad hoc Multicast Routing protocol utilizing Increasing id numberS (AMRIS), Functional Specification. *Internet Draft "draft-ietf-manet-amris-spec-00.txt"*, 1998.

[Dani99] Kevin J. Krizman Daniel L. Lough, T. Keith Blankenship. A Short Tutorial on Wireless LANs and IEEE 802.11. *"http//www.computer.org/student/looking/summer97/ieee802.htm"*, 1999.

[Dr. 99] Suzanne Kocurek Dr. Qiang Gao. Bluetooth Press Release. *"http://www.bluetooth.com/v2/news/show.asp?page=pressrelease&id=12"*, Dezember 1999.

[Joan99] Steven Johnson Joanne Taylor. *Bluetooth Press Release*, 1999.

[Josh99] David A. Maltz Josh Broch, David B. Johnson. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. *Internet Draft "draft-ietf-manet-dsr-03.txt "*, Oktober 1999.

[M. S99a] Joseph P. Macker M. Scott Corson. Internet-Based Mobile Ad-hoc Networking. *IEEE Internet Computing* 3(4), Juli 1999, S. 63–70.

[M. S99b] Joseph P. Macker M. Scott Corson. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. Januar 1999.

[Over99] Bluetooth Document: Technical Overview. *"http://www.bluetooth.com/v2/document/default.asp"*, 1999.

[Rele00] Ericsson Press Release. *"http://www.bluetooth.com/v2/news/show.asp?page=news&id=41"*, 2000.

[Schi99] Jochen Schiller. *Mobile Communications*. Addison-Wesley. 1999.

[Sele] Murat Kocaoglu Selen Sonar, Yunus Targu. Wireless Local Area Networks. *"http://www.mis.boun.edu.tr/ulus/courses/mis221/project/wireless-lans/wireless-lans.htm"*.

[snew00] Bluetooth special news. *"http://www.bluetooth.com/v2/specialnews.asp/news=2"*, 2000.

[Sung99] Chai-Keong Toh Sung-Ju Lee, Mario Gerla. A Simulation Study of Table-Driven and On-Demand Routing Protocols for Mobile Ad Hoc Networking. *IEEE Network* 13(4), Juli 1999, S. 48–54.

3GPP - Mobilfunk der 3.Generation

Helen Dittner

Kurzfassung

Mit IMT-2000 läutet die International Telecommunication Union die dritte Generation im Mobilfunk ein, getreu dem Motto „Jedermann, Jederzeit, Überall“. Durch länder- und kontinenteübergreifende Standardisierung wird erstmals internationales Roaming im großen Rahmen möglich. Globale Abdeckung wird durch das Zusammenspiel von terrestrischen und Satellitensystemen erreicht. Mit diesem Standard sollen bessere Qualität, Datenraten bis zu 2 Mbit/s, ein nahtloser Übergang zum Festnetz und damit eine gute Anbindung an das Internet möglich werden. Die mobilen Dienste werden alle bisherigen und zukünftigen Festnetzdienste umfassen, von Telephonie bis hin zu Videokonferenzen.

1 Motivation

Kommt es nicht ab und zu vor, daß man im Sahara-Urlaub eine Videokonferenz abhalten müßte, um die eigene Firma zu retten, während man emails empfängt und gleichzeitig Sonnenbrandcreme über das Internet bestellen möchte? Selbst wenn nicht, das Motto „Jederzeit, Jedermann, Überall“ (ubiquitär) ist in aller Munde und soll durch die sogenannte dritte Generation im Mobilfunk sehr bald ermöglicht werden.

Diese dritte Generation befindet sich momentan in der Aufbauphase. Das Ziel ist es, ab 2002 einen weltweit einheitlichen Standard einzusetzen, mit dem es für den Benutzer vollkommen egal ist, wo, wann und mit welchem Endgerät er telephonieren oder Daten empfangen oder versenden möchte. Zusätzlich sollen dem Benutzer alle möglichen Dienste, die bisher bekannt sind und auch zukünftige, mit besserer Qualität und höheren Datenraten zur Verfügung gestellt werden.

2 Historie

Nach den analogen Anfängen im Mobilfunk entwickelte sich die kommerzielle Branche erst in den letzten Jahren rapide, jedoch mit unterschiedlichen, inkompatiblen Standards und in unterschiedlichen Frequenzbereichen.

Schon 1985 bildete sich eine Arbeitsgruppe, die sich um die Spezifikation eines weltweit einheitlichen Mobilfunkstandards bemühte. Im Jahre 1992 wurden auf einer weltweiten Konferenz (WRC'92) Frequenzbereiche festgelegt, in denen sich der neue Standard bewegen sollte. Zu dieser Zeit fing die International Telecommunications Union (ITU) [ITUn] an, sich um die Standardisierung zu kümmern, und taufte diesen neuen Standard IMT-2000 (International Mobile Telecommunications). Dieser Standard soll in den Jahren 2001, 2002 und 2003 weltweit eingeführt werden.

2.1 Der Weg zur 3. Generation im Mobilfunk

Unter der ersten Generation im Mobilfunk werden die analogen Techniken verstanden, die den allergrößten Teil dieses Jahrhunderts beherrschten. Mit der digitalen Technik kam Anfang der neunziger Jahre die zweite Generation zum Einsatz, die eine bessere Qualität und bessere Ausnutzung des Spektrums lieferte. Zur zweiten Generation gehören u.a. verschiedenste Paging-, Drahtlose-, Mobilfunk- und mobile Satellitensysteme.

In Europa wurde mit GSM (Groupe Speciale Mobile) 1982 die Spezifikation eines Mobilfunkstandards der 2.Generation in Angriff genommen, der europaweit greifen sollte. Im Jahre 1992 wurde dieses System in Deutschland in Betrieb genommen.
GSM, mittlerweile zu „Global System for Mobile Communication“ mutiert, breitete sich nicht nur in Europa aus, sondern wurde auch in 130 Drittländer übernommen. Damit hat GSM bis Ende 1999 weltweit ca. 230 Millionen Benutzer (September 1999 noch 200 Millionen).

GSM sollte sozusagen eine Verlängerung von ISDN werden und war ursprünglich nur zur Sprachübertragung und für minimale Datenübertragungen gedacht. Nach der erstaunlich rapiden Entwicklung und des großen Anklangs des Internets wird jedoch in Zukunft auch ein Hauptaugenmerk auf Datenübertragung gelegt werden. Daher wurde versucht, einen Mobilfunkstandard zu spezifizieren, der diesen Anforderungen genügt. Die Spezifikation fing schon 1985 an und wird seit Anfang der 90er Jahre unter dem Namen IMT-2000 von der ITU standardisiert.

IMT-2000 soll erstmals in Japan in 2001 und in Europa in 2002 spätestens aber in 2003 eingesetzt werden. Das volle Ausmaß wird aber wohl erst 2010 sichtbar werden.

2.2 3GPP/3GPP2 – Third Generation Partnership Project

Eine wichtige Rolle in der Empfehlung und auch Spezifikation des IMT-2000 Standards spielen die beiden Organisationen 3GPP [GPPr], Third Generation Partnership Project, und 3GPP2 [GPP2].

3GPP ist ein Zusammenschluß von mehreren „Standards Developing Organisations“ (SDOs), u.a. mit Japan und Europa (European Telecommunications Standards Institute, ETSI), und Firmen, die über ihre nationalen SDOs daran teilnehmen. Die 3GPP steht für ein IMT-2000 ein, das auf einem weiterentwickelten GSM-Kernnetzwerk basieren soll. Sie sieht ihre Aufgabe darin, weltweit einsetzbare Spezifikationen zu definieren und der ITU vorzuschlagen.

Die USA befürchteten jedoch, daß sich dadurch GSM durchsetzen könnte und gründete 3GPP2, das für ein auf ANSI41 und verwandte CDMA-Technologien basierendes IMT-2000 steht und seine entsprechenden Vorschläge an die ITU weitergibt.

3 Die IMT-2000 Familie

Um einen geeigneten weltweiten Standard zu finden, bat die ITU alle SDOs, Vorschläge für ein solches abzugeben. Es gingen 10 Vorschläge für terrestrische und 5 für Satelliten-Funktechnologien ein. Damit die Standardisierung nicht an dem Markt vorbeidefiniert wird, wurden weltweit alle Standardisierungsorgane der Telekommunikation dazu aufgefordert, Vorschläge für IMT-2000 zu machen. Auch Betreiber von Mobilfunknetzen und Hersteller von Endgeräten konnten über diese Standardisierungsgremien Vorschläge abgeben.

Ein Problem bei der Wahl des Standards sind die unterschiedlichen Frequenzbereiche und die unterschiedlichen Standards, die weltweit verstreut sind, darunter Varianten von CDMA- und

TDMA-Techniken. Daher werden die Frequenzen von der ITU festgelegt und, um die schon vorhandenen Netzwerke weiterverwenden zu können, hat man sich darauf geeinigt, mehrere Techniken zuzulassen, was dazu führt, daß IMT-2000 tatsächlich eine Familie von zueinander kompatiblen Standards ist. Das Ziel ist es u.a. die Unterschiede zwischen den verschiedenen Funkschnittstellen so gering wie möglich zu halten, um die Produktion von Endgeräten zu vereinfachen.

Die Frequenzbereichfestlegung ist jedoch problematisch, da in den Ländern gewisse Bereiche schon längst vergeben sind, z.B. an das Militär, es wird also noch eine Zeit dauern, bis überall auf der Welt exakt dieselben Frequenzbereiche verwendet werden können.

Eine große Rolle in IMT-2000 spielen Satelliten, mit denen überhaupt erst eine globale Abdeckung erreicht werden kann. Somit vereint IMT-2000 verschiedene Mobilfunkumgebungen, von der kleinen terrestrischen Zelle in Gebäuden mit großer Datenübertragungsrate über die große Zelle auf dem Land bis hin zur weltweiten Abdeckung durch Satelliten (siehe Abbildung 4).

3.1 Technische Spezifikationen

Im Dezember 1999 wurden die von der ITU spezifizierten Funkschnittstellen veröffentlicht. Die verwendeten Zugriffsverfahren und die Größe der Bandbreite ergibt sich aus den Anforderungen, die von den Anwendungen an den Mobilfunk gestellt werden. Diese Anwendungen sind in Abbildung 1 dargestellt.

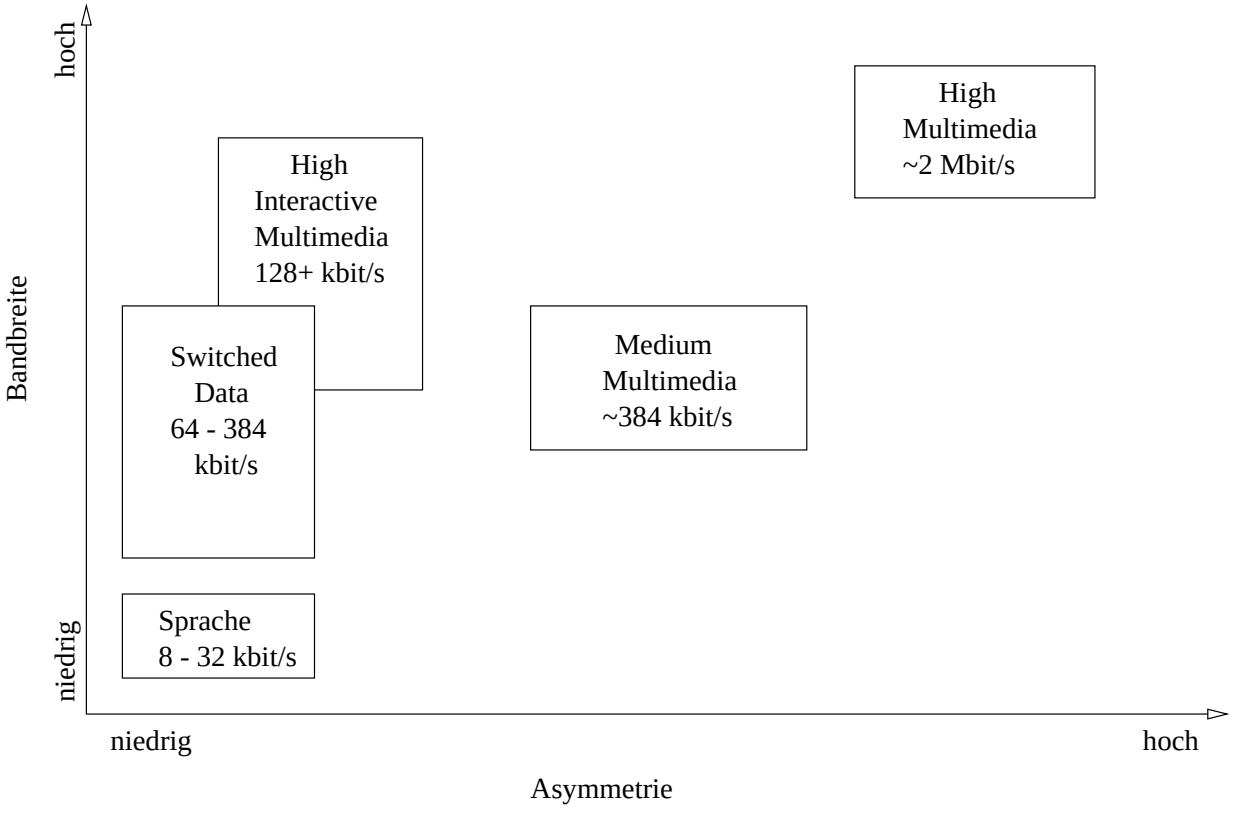

Abbildung 1: Diensteklassifizierung [AlSe99]

Mögliche Zugriffsverfahren laut ITU (siehe auch Abbildung 2):

- Code Division Multiple Access (CDMA), Time Division Multiple Access (TDMA) oder eine Kombination der beiden

- CDMA mit Frequency Division Duplex (FDD) direct sequence, FDD multi-carrier und Time Division Duplex (TDD)
- TDMA mit FDD single carrier, FDD multi-carrier und TDD

In Abbildung 3 sind diese verschiedenen Zugriffsverfahren, die von der ITU für IMT-2000 zugelassen wurden, graphisch dargestellt.

Das Prinzip von Wideband-CDMA (W-CDMA) ist, daß alle Sender das gleiche Frequenzband benutzen und gleichzeitig senden. Das Signal wird mit einer Pseudozufallszahl XOR-verknüpft, der Empfänger kann dann mittles bekannter Sender-Pseudozufallsfolge das Originalsignal wiederherstellen.

Bei Frequency Division Multiple Access (FDMA bzw. Orthogonal FDMA) wird die Kanalbandbreite in Unterkanäle aufgeteilt. Jeder Sender erhält einen Unterkanal.

Bei Wideband-TDMA (W-TDMA) wird jedem Sender zyklisch ein Zeitschlitz zugewiesen, in dem er die gesamte Bandbreite zur Verfügung hat.

Die Kombination von TDMA und CDMA ist in dem vierten Bild in Abbildung 3 abgedruckt.

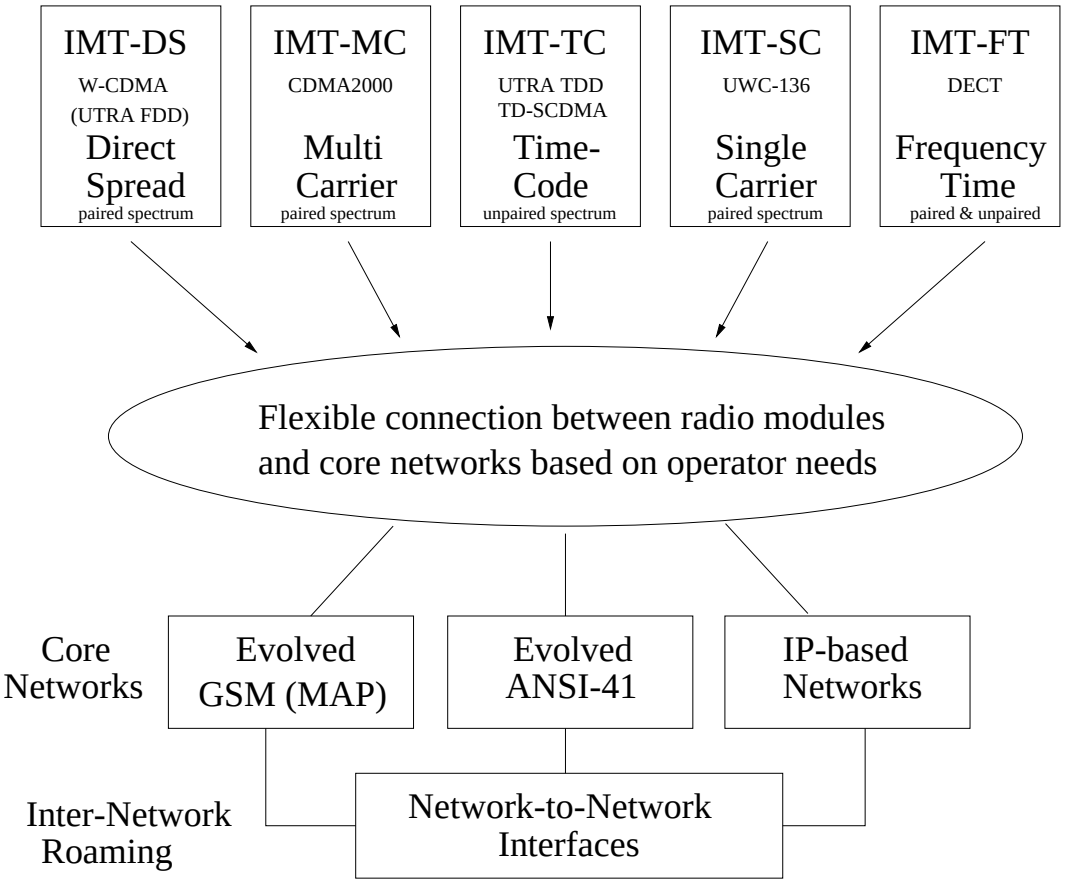

Abbildung 2: Von der ITU spezifizierte terrestrische Funkschnittstellen

Frequenzbereiche:

- terrestrisch: 1885 - 2025 MHz uplink und 2110 - 2170 MHz downlink
- Satelliten: 1980 - 2010 MHz uplink und 2170 - 2200 MHz downlink

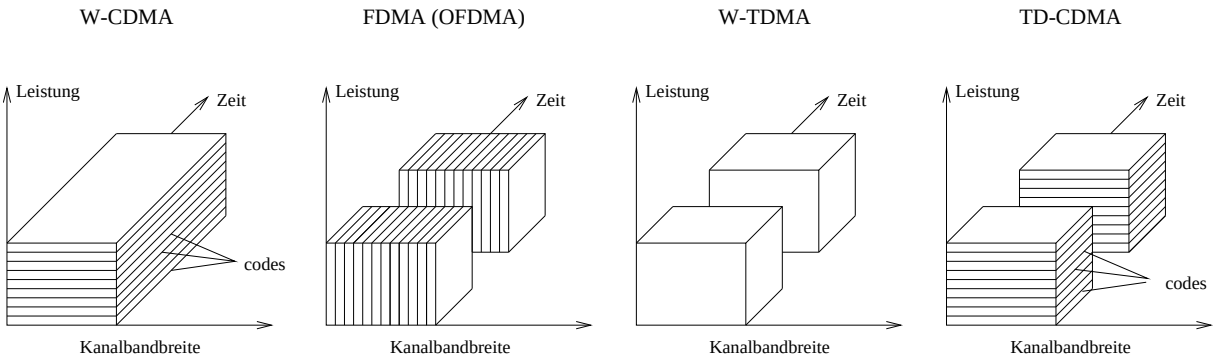

Abbildung 3: Die möglichen Zugriffsverfahren auf das IMT-2000 Medium

Die Vergabe der Lizenzen für die Frequenzen wird den nationalen Regulierungsbehörden überlassen. Diese können sich dann das verwendete Verfahren aussuchen, z.B. ob die Frequenzen versteigert werden sollen, und auch entscheiden, welche Bedingungen der Teilnahme unterliegen.

Datenratenanforderungen (siehe auch Abbildung 4):

- mobile Datenrate (Makrozelle) mit hoher Geschwindigkeit (<500 km/h) 144 kbit/s
- tragbare Datenrate (Makro/Mikrozelle) Fußgänger oder moderate Geschwindigkeit (<120 km/h) 384 kbit/s
- in Gebäuden (Pikozelle) oder in naher Umgebung 2 Mbit/s

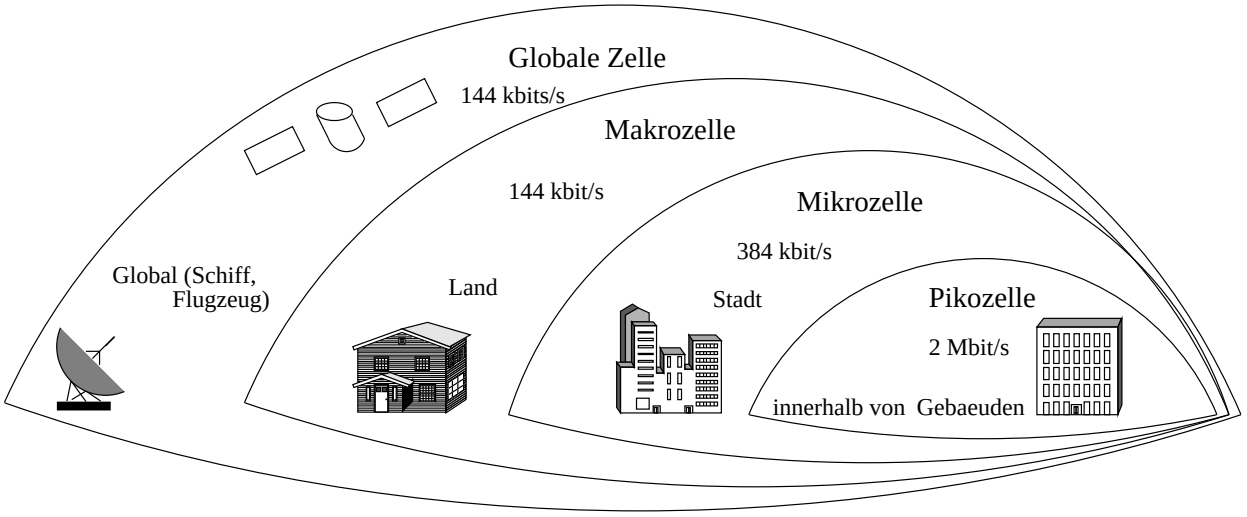

Abbildung 4: Die Zellen in IMT-2000

3.2 Die Dienste

Wie schon mehrmals erwähnt, ist vor allem Kompatibilität zwischen den verschiedenen Mobilfunkstandards auf der Welt ein Hauptanliegen der SDOs und als Folge dessen das internationale Roaming. Um dieses für den Benutzer so angenehm wie möglich zu machen, soll jeder Benutzer ein *Virtual Home Environment (VHE)* haben.

Ein VHE ist eine personalisierte, portierbare Diensteumgebung und macht Dienste unabhängig von dem verwendeten Netz bzw. Endgerät. D.h. unabhängig davon, in welchem Netz und an welchem Endgerät sich der Benutzer befindet, wird er immer seine persönliche Diensteumgebung vorfinden, lediglich etwas den Möglichkeiten des Netzes angepaßt.

Die Dienste werden von netzbetreiberunabhängigen Anbietern, sogenannten *Service-Providern*, angeboten, von denen es dann auch mehrere in einem Netzwerk geben wird. Diese Service-Provider würden lediglich die Dienste aber nicht die Übertragung der Daten anbieten, man hätte dann als Benutzer eine Auswahl an Providern und könnte sich den mit dem momentan passenden Angebot aussuchen oder das Endgerät würde dies für den Benutzer übernehmen. Es gäbe sozusagen virtuelle Netzbetreiber.

Für das Roaming ist vorgesehen, daß es sich nicht nur auf die verschiedenen Mitglieder der IMT-2000 Familie beschränken wird, sondern auch über heterogene Netzwerke wie z.B. Bluetooth, HIPERLAN, Festnetze und DECT, private und öffentliche Netzwerke möglich wird, so daß tatsächlich ein nahtloser Übergang in jedes Netzwerk existieren wird. Für diese Eigenschaft bedarf es aber noch einer Spezifizierung und Standardisierung.

Ein ganz besonderer Vorteil von IMT-2000 wird die flexible Bandbreitennutzung sein. Je nachdem welche Anwendung von dem Benutzer gerade verwendet wird, beansprucht dieser nur die benötigte Bandbreite und nicht mehr. Die Bandbreite wird also dynamisch je nach Bedarf vergeben. Für nur Sprache würde also sehr wenig Bandbreite verwendet werden, wohingegen bei einer Videokonferenz entsprechend mehr Bandbreite genutzt würde.

4 UMTS – 3g in Europa

Der Vorschlag der europäischen Standardisierungsorganisation ETSI [TSIn] für IMT-2000 ist ein System mit Namen Universal Mobile Telecommunication System (UMTS), das auf GSM (Global System for Mobile Communication) basiert. UMTS wurde mittlerweile entsprechend spezifiziert, um ein Mitglied der IMT-2000 Familie werden zu können.

Zumindest in Europa soll hauptsächlich UMTS eingesetzt werden, die ETSI entschied jedoch, daß nicht alle Netzbetreiber in Europa unbedingt UMTS einsetzen müßten, sondern sich einen Standard aus der IMT-2000 Familie aussuchen könnten, sofern mindestens je ein Netzbetreiber pro Land UMTS einsetzt. Natürlich können auch weiterhin alte Netze betrieben werden, vor allem die GSM-Netzbetreiber ohne UMTS-Lizenz haben mit GPRS und EDGE (siehe Abschnitt 4.1.1) gute Möglichkeiten, ihr Netz trotz vorhandenem UMTS für die Kunden interessant zu halten.

4.1 GSM – Die Basis von UMTS

Die Arbeit an dem digitalen Mobilfunksystem Global System for Mobile Communication (GSM), anfänglich „Groupe Speciale Mobile“ genannt, wurde schon 1982 begonnen. Seit ca. 1991 ist es der Standard im Mobilfunk von Europa und wurde mittlerweile in weit über 130 andere Länder übernommen, die Anzahl der Teilnehmer beträgt Ende 1999 weit über 230 Millionen.

GSM, zu der 2.Generation im Mobilfunk gehörend, wurde speziell für Sprachübertragung konzipiert. Zu den Diensten von GSM gehören jedoch auch noch Datendienste wie z.B. Fax und Kurznachrichten (SMS, Short Messaging System).

Die Frequenzbereiche in Europa lagen zunächst nur um 900 MHz, später wurde der Bereich um 1800 MHz hinzugenommen. Das Zugriffsverfahren setzt sich zusammen aus Frequency

Division Duplex (FDD) und einer Kombination aus Time Division Multiple Access (TDMA) und Frequency Division Multiple Access (FDMA). Mit Leitungsvermittlung kann allerdings nur eine Datenübertragungsrate von 9,6 kbit/s angeboten werden. Durch Reduzierung der Fehlerbehebungsmaßnahmen soll bald eine Erhöhung der Datenrate auf 14,4 kbit/s erreicht werden.

Durch Bitratenadaption (V.110) von 64 auf 9,6 kbit/s kann eine Kopplung an ISDN erreicht werden, um zumindest dem Anspruch, eine Art Verlängerung von ISDN zu sein, gerecht zu werden.

4.1.1 Erweiterungen von GSM (HSCSD, GPRS, EDGE)

Da es bis zur Einführung von IMT-2000 noch relativ lange dauert, haben sich Erweiterungen von GSM hervorgetan, die schon jetzt schnellere Datenraten ermöglichen sollen. Mit diesen Erweiterungen haben die Netzbetreiber, die keine UMTS-Lizenz ergattern, trotzdem noch die Möglichkeit, wenigstens annähernd gute Datenraten anzubieten.

Mit High Speed Circuit Switched Data (HSCSD) könnte theoretisch durch Leitungsvermittlung und Zusammenlegen mehrerer Zeitkanäle bis zu 115 kbit/s erreicht werden. Realistischer sind jedoch 64 kbit/s. Jedoch ist selbst diese Datenrate in Netzen mit hoher Last kaum dauerhaft durchsetzbar, vor allem weil dann diese Kanäle so lange für Sprache blockiert sind.

Generic Packet Radio Service (GPRS) ist eine paketorientierte Vermittlung, die bei kurzfristiger Belegung aller 8 Zeitschlitze, die vom GSM-TDMA-Verfahren angeboten werden, bis zu ca. 115 kbit/s liefert.

Enhanced Data rates for GSM Evolution (EDGE) setzt im Gegensatz zu Gaussian Frequency Shift Keying (GFSK) in GSM die QPSK-Modulation (Quaternary Phase Shift Keying) ein (ebenfalls in UMTS eingesetzt), und erreicht damit Datenraten bis zu 384 kbit/s. Fur die Netzbetreiber wäre jedoch der zusätzliche technische Aufwand für dieses System so groß, daß es fraglich ist, ob es sich zu dieser Zeit schon lohnt.

Eigentlich sollte sich damit GPRS durchsetzen, da sich die Kosten gegenüber einer Erweiterung zu EDGE in annehmbaren Grenzen halten und es bessere Datenraten unter besseren Bedingungen liefert als HSCSD. Für die Netzbetreiber ohne UMTS-Lizenz wird wahrscheinlich EDGE auf lange Sicht die beste Lösung sein. Bei den obigen Verfahren sollte jedoch immer bedacht werden, daß die angegebenen Datenraten maximal sind und wohl kaum permanent und überall zu erreichen sind.

E-Plus bietet in Deutschland mittlerweile HSCSD unter dem Namen „High Speed Mobile Data“ (HSMD) an. Laut [Ži00] sind die Datenraten von HSMD in der Praxis auf 43,2 kbit/s begrenzt, erwartet werden 28,8 kbit/s. Damit ist HSMD zwar schneller als GSM, aber immer noch nicht das Gelbe vom Ei, da diese Datenrate natürlich nicht garantiert werden kann.

4.2 UMTS – Universal Mobile Telecommunication System

Mit UMTS/IMT-2000 wird erstmals der Versuch gestartet, sich an das Internet anzupassen. Die Paketvermittlung nicht nur von UMTS, sondern auch schon von GPRS, läßt erahnen, daß hierfür das IP-Protokoll verwendet werden könnte. Die Überlegungen reichen von einem IP-basiert4en Netzwerk über ATM oder IP über ATM. Durch diese Paketvermittlung kann je nach Dienst (z.B. WWW) eine verbesserte Ausnutzung des Spektrums mit einem Faktor von mehreren Hundert erwartet werden.

4.2.1 UTRA – UMTS Terrestrial Radio Access

Als Zugriffsverfahren für UMTS (UTRA) sind zwei Modi vorgesehen: Frequency Division Duplex (FDD) und Time Division Duplex (TDD).

Die optimale Ausnutzung des Spektrums wäre durch die folgende Aufteilung gegeben:

- FDD für symmetrische Dienste
- TDD für asymmetrische Dienste

Allerdings wird selbst mit dieser Aufteilung nicht erwartet, daß breitbandige, asymmetrische Dienste in der Fläche sinnvoll angeboten werden können, sondern daß noch eine Erweiterung von UMTS erforderlich sein wird.

Den Berechnungen des UMTS-Forum zu Folge werden die von der ITU festgelegten Frequenzbereiche nicht ausreichen. Es dürften im Jahr 2005 spätestens in 2010 ca. 187 MHz an terrestrischer Bandbreite und 30 MHz für Satelliten fehlen, daher hat die ITU eine Allokierung von weiteren 160 MHz zugesagt.

Hier eine Bewertung von Kombinationen von Frequenzen für die Netzbetreiber laut:

- sehr begrenzte Dienste:
  - 2 x 5 MHz gepaart
  - 2 x 5 MHz gepaart und 5 MHz ungepaart
  - 2 x 10 MHz gepaart
- etwas begrenzte Dienste:
  - 2 x 10 MHz gepaart und 5 MHz ungepaart
  - 2 x 15 MHz gepaart
- volle Dienste:
  - 2 x 15 MHz gepaart und 5 MHz ungepaart (wird als Optimum betrachtet)
  - 2 x 20 MHz gepaart
  - 2 x 20 MHz gepaart und 5 MHz (auch 10 MHz möglich) ungepaart

Diese Aufzählung basiert auf den Aspekten von asymmetrischem Datenverkehr, hohe Effizienz des Spektrums und Quality of Service [Foru].

Wie schon erwähnt, ist die Lizenzvergabe den einzelnen Ländern überlassen. In Deutschland werden die Lizenzen versteigert, über den Sinn eines solchen Verfahrens gibt es verschiedene Meinungen. Das UMTS-Forum [Foru], dessen Mitglieder Endgerätehersteller, Organisationen und Netzbetreiber sind, die Empfehlungen für UMTS machen, befürchtet, daß durch Versteigerungen zu hohe Kosten entstehen, die dann auf den Benutzer weitergeleitet werden und damit die Entwicklung der neuen Diensten und des Netzes gehemmt wird. Weiterhin könnten die Versteigerungen den Wettbewerb behindern.

Laut der Regulierungsbehörde für Telekommunikation und Post in Deutschland [fTuP] werden die bundesweiten Lizenzen in einem ersten Abschnitt in 5 Lizenzen zu je 2 x 10 MHz (gepaart) und in einem zweiten Abschnitt 5 Frequenzblöcke zu je 1 x 5 MHz (ungepaart) versteigert. Bleiben vom ersten Abschnitt Lizenzen übrig, so werden diese in Blöcke zu je 2 x 5 MHz

(gepaart) im zweiten Abschnitt angeboten. Am zweiten Abschnitt dürfen nur die teilnehmen, die im ersten Abschnitt Lizenzen ersteigert haben.

Fazit: Die Grundausstattung einer bundesweiten UMTS/IMT-2000 Lizenz an Funkfrequenzen beträgt 2 x 10 MHz (gepaart).Laut der obigen Bewertung wird die Grundausstattung kaum ausreichend für einen Netzbetreiber sein.

In England werden wohl auch nur jeweils 2 x 10 MHz gepaart und 5 MHz ungepaart vergeben werden. In Finnland hingegen ist festgelegt, daß jeweils 2 x 15 MHz gepaart (FDD) und 5 MHz ungepaart (TDD) an die Interessenten vergeben werden sollen. Diesem Beispiel werden voraussichtlich die Niederlande folgen.

Mittlerweile wird UMTS nicht nur in Europa eingeführt, sondern auch in vielen anderen Ländern wie z.B. Japan. Hier werden 3 Lizenzen zu je 2 x 20 MHz verteilt.

4.2.2 S-UMTS – Die Satellitenkomponente von UMTS

Ein sehr wichtiger Aspekt von UMTS/IMT-2000 ist die Satellitenkomponente (S-UMTS), die überhaupt erst das „Überall“ ermöglichen wird. Es lohnt sich für die Netzbetreiber, nur in gut besiedelten Gegenden Basisstationen zu errichten. D.h. aber daß es sehr viele Flächen geben würde, die praktisch ohne Netzabdeckung wären. Als Lösung dieses Problems werden Satelliten benutzt, die in Zukunft immer mehr Bandbreite anbieten werden können.

Mit Satelliten kann eine globale Abdeckung und damit weltweites Roaming stattfinden. Zusätzlich wird mit neuen Satelliten eine schnelle und kosteneffiziente Bereitstellung in großen Regionen möglich.

Es gibt drei Sorten von Satelliten:

- LEO (Low Earth Orbit) Satelliten in einer Höhe von 1.000 km und einer Umlaufzeit von 2 Stunden. Vorteilhaft sind die kleine Signaldämpfung und die kleine Signallaufzeit, jedoch sind für eine globale Versorgung viele Satelliten erforderlich und daher viele Bodenstationen oder Intersatellitenlinks.
- GEO (Geostationäre) Satelliten in einer Höhe von 36.000 km und einer Umlaufzeit von 24 Stunden. Diese haben eine konstante Position und eine große Versorgungszone, jedoch muß man eine hohe Signaldämpfung und lange Signallaufzeiten (Antwortzeit ca. 0,5 sec) in Kauf nehmen.
- MEO-Satelliten in einer Höhe von 10.000 km und einer Umlaufzeit von 6 Stunden

Die bisherigen Satellitensysteme wie z.B. Iridium 1998 (LEO) oder Globalstar 1999 (LEO) bieten lediglich 2,4 kbit/s bzw. 9,6 kbit/s, eindeutig zu wenig, um UMTS/IMT-2000 flächendeckend zu verwirklichen. Weitere noch ausstehende Satelliten wie z.B. Horizons 2002 (GEO) könnten Multimedia und High-Speed Daten mit bis zu 144 kbit/s (bei mehreren Kanälen: 432 kbit/s) unterstützen. Zukünftige breitbandige Satellitennetze sollen nahtlos in terrestrische Mobilfunknetzen integriert sein. Einige dieser breitbandigen Satellitensysteme sind in Abbildung 5 dargestellt. Das Zusammenspiel mit dem Kernnetz und dem Benutzer ist in Abbildung 6 dargestellt. Mit bis zu 2 Mbit/s wird auf den Einsatz von z.B. SkyBridge 2002 (80 LEOs) gehofft. Es sind viele weitere Satellitensysteme in Planung, die in naher Zukunft entstehen sollen und zwischen 144 kbit/s und 2 Mbit/s unterstützen. Bisher blieb der kommerzielle Erfolg aus, dies wird sich jedoch voraussichtlich mit der Einführung von IMT-2000 ändern.

Anfänglich wird für S-UMTS eine Datenrate von 144 kbit/s erwartet. Anfang November 1999 wurden die Frequenzbereiche für S-UMTS auf 1980 - 2010 MHz uplink und 2170 - 2200 MHz

System	Satelliten	Region	Dienstrate (kl. Terminal)	Status, Betriebsbeginn
SkyBridge (Alcatel)	80 LEOs bei 1457 km	global	bis 2 Mbit/s	2002
Astrolink	9 GEOs	global	bis 384 kbit/s	2002
EuroSkyWay (Alenia)	5 GEOs	Europa	auf: 128 kbit/s ab: 2Mbit/s	2001
Spaceway (Hughes)	8 GEOs	global	bis 384 kbit/s	2001
Teledesic	288 LEOs bei 1350km	global	auf: <= 2 Mbit/s ab: <= 64 kbit/s	2003
WEST (Matra Marconi)	1 GEO + 9 MEOs	global	384 kbit/s	GEOs: 2001 MEOs: 2003
Expressway (Hughes)	14 GEOs	global	1,5 Mbit/s	1997 beantragt
GS-40 (Globalstar)	80 LEOs bei 1440km	global	1,2 Mbit/s	1997 beantragt
Orblink (Orbital Sciences)	7 aequat. MEOs bei 9000km	Aequator	1,5 Mbit/s	2002
Sky Station	250 aeron. Plattf. bei 21km	Ballungs-geb. mobil	64 kbit/s .. 2 Mbit/s	2001

Abbildung 5: S-UMTS Nachfolgesysteme: Breitbandige Satellitennetze [Lutz99]

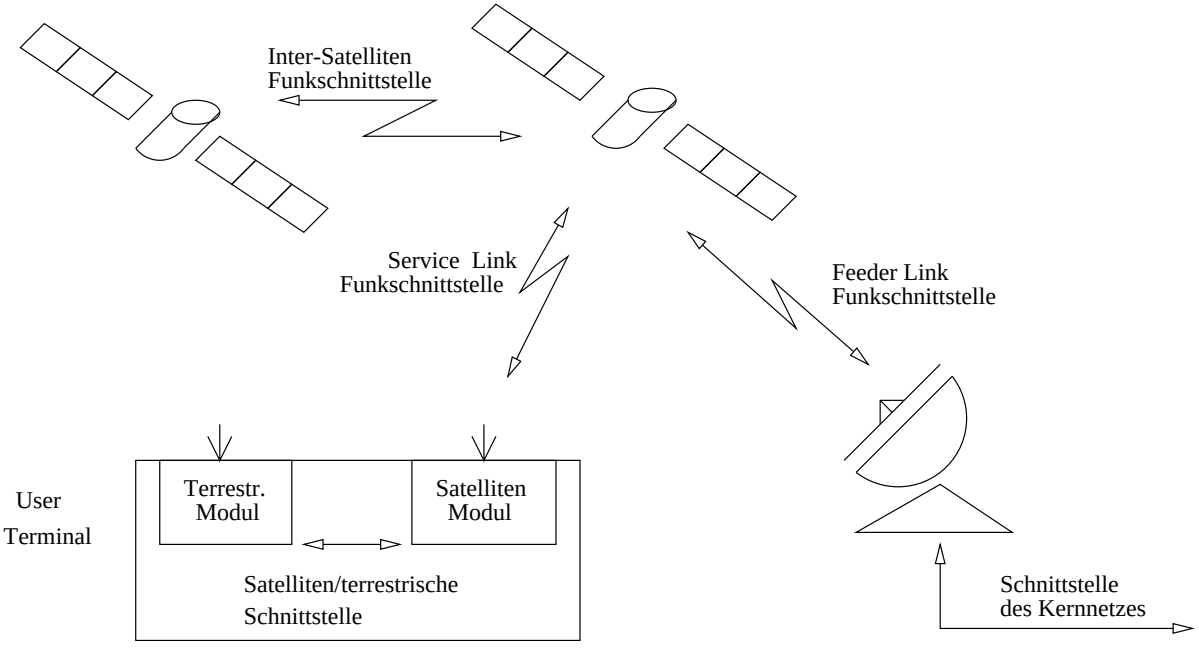

Abbildung 6: Schnittstellen in der Satellitenkomponente von UMTS/IMT-2000

downlink festgelegt. In den USA und Kanada werden 1990 - 2025 MHz uplink und 2160 -

2200 MHz downlink verwendet. Auf lange Sicht werden jedoch einheitliche Frequenzbereiche angestrebt.

5 Anwendungen

Die Hauptanwendung in IMT-2000 wird wohl weiterhin die Sprachübertragung sein. Jedoch werden mit Datenraten bis zu 2 MBit/s Videokonferenzen eine mögliche Alternative werden, vor allem für Geschäftsleute und Firmenvertreter.

Es werden durch verschiedene Service-Provider je nach Netz verschiedene Dienste angeboten. Man könnte sich Informationen automatisch zuschicken lassen (Pushdienste) oder diese explizit anfordern (Pulldienste). Anwendungen, Skripte und ausführbare Programme lägen bei dem Service-Provider bereit, um auf das Endgerät des Kunden heruntergeladen zu werden.

Wenn man in einer neuen Stadt wäre, könnte man sich alle Orte der Sehenswürdigkeiten anzeigen lassen oder die Straßenkarte der jeweiligen Stadt automatisch empfangen. Die aktuellen Programme der lokalen Kinos, Theater und ähnliches würden zum Abruf bereitstehen.

Ein weiteres Vorhaben ist, daß über IMT-2000 mit Hilfe von Bluetooth verschiedene persönliche Endgeräte miteinander kommunizieren. Voraussichtlich wesentlich verbesserte Sicherheitsvorkehrungen werden e-commerce über den Mobilfunk endlich akzeptabl machen.

Emails müßten nicht nur in Textform, sondern könnten auf verschiedener Art und Weise z.B. als Videomessages verschickt werden. Netzwerk-Spiele könnten über Mobilfunk gespielt werden.

Diese Dienste stellen nur einen Bruchteil derer dar, die existieren werden, sie werden zum Teil gleichzeitig abgewickelt werden können.

6 Sicherheit in IMT-2000

Über die Sicherheitsvorkehrungen in IMT-2000 wurde zwar schon nachgedacht, jedoch noch nichts konkretes und neues entwickelt. Voraussichtlich werden ähnliche Sicherheiten wie bei GSM eingebaut werden, wie z.B. die SIM-Karte mit PIN und verschlüsselnde Algorithmen. Es ist aber klar, daß diese nicht ausreichen werden, um Dienste wie e-commerce sicher anbieten zu können. Es werden also noch wesentliche Verbesserungen erwartet, wie diese aussehen werden, ist noch unklar.

7 Abrechnungen (Billing)

Durch die Paketvermittlung werden völlig neue Möglichkeiten zur Abrechnung der „online“-Dauer eröffnet. Es wird wohl nicht mehr nur nach Minuten abgerechnet werden, sondern vielleicht pro gesendetem Paket, pro versendeter Email, pro bit, pro Sitzung, pro Spiel oder sogar pro Munitionseinheit im Spiel. Es wird jedoch erwartet, daß das mobile Telephonieren nur unwesentlich billiger wird. Natürlich wird die Struktur von „was man bei wem zahlt“ komplizierter, da ein Netzbetreiber natürlich den Wert eines Bits nicht erkennen kann, außerdem werden die Dienste ja von Service-Providern angeboten. Also wird auch in dieser Rubrik noch einiges zu erwarten sein.

8 3g Endgeräte

Aufgrund des Unterschiedes zwischen terrestrischer und Satellitenübertragung und der unterschiedlichen Medienzugriffsverfahren in der IMT-2000 Familie, werden die Endgeräte der 3. Generation (3g Endgeräte) Multimode-Endgeräte sein müssen, um eine breitere Netzwerkabdeckung auszunutzen und ein verbessertes internationales Roaming zu ermöglichen.

Endgültige Ziele für 3g Endgeräte müssen noch festgelegt werden, da die Spezifikation der Funkschnittstellen erst zur Zeit der Fertigstellung dieser Arbeit festgelegt wurde und wird. Jedoch hat ETSI eine Anzahl Mindestanforderungen an diese definiert [Noki]:

- USIM (UMTS Subscriber Identity Module) Funktionalität und Rückwärtskompatibilität zu GSM
- Netzwerk Registrierung
- Location Update
- Authentifizierung
- Unterstützung von sowohl verbindungsorientierter als auch verbindungsloser Dienste
- Identifikation des Endgeräts
- grundlegende Identifikation der Möglichkeiten des Endgerätes
- Endgeräte, die Notrufe absenden können, sollten dies auch ohne USIM ermöglichen
- die Endgeräte müssen einen UMTS-Träger entdecken, die Dienstmöglichkeiten vom Broadcastkanal der unterschiedlichen Netzwerke lesen, das Netzwerk auswählen und zum richtigen Netzwerk verbinden können

In diesem Zusammenhang werden die Endgeräte dann auch die im aktuellen Netzwerk genutzte Variante von GSM erkennen.

Auch die Benutzerschnittstelle soll für UMTS so einfach wie möglich gemacht werden mit den folgenden Eigenschaften:

- der Ablauf zwischen Endgerät und Netzwerk wird nahtloser
- es sollen Anwendungen, Skripte und Programme auf das Endgerät heruntergeladen werden können
- neue Technologien wie z.B. Spracherkennung werden zum Einsatz kommen
- das VHE bietet dem Benutzer einen konsistenten Blick auf die Benutzerschnittstelle seines Endgeräts, egal in welchem Netzwerk er sich befindet

Damit werden die Endgeräte also „intelligenter“ werden als bisher, um dem Benutzer ein einheitliches Netzwerk vorzuspiegeln. Dazu soll laut ETSI ein Betriebssystem (Terminal Operating Environment, TOE), wie beim PC, sorgen, auf das die Anwendungen aufgesetzt werden können. Nokia [Noki] hat sich mit anderen Firmen zusammengeschlossen, um ein „open operating system“ zu definieren, damit Anwendungen auf unterschiedlichen Endgeräten laufen können, ohne daß Kompatibilitätsprobleme dabei entstehen.

Die Funktionen eines Endgeräts müssen nicht in einem Gerät untergebracht sein, sondern könnten auf verschiedene Geräte aufgeteilt sein, die dann über z.B. Bluetooth interagieren.

Das Endgerät müßte auch nicht unbedingt alle Programme beinhalten, sondern könnte nur die grundlegenden Dienste anbieten, wie es im Moment der Fall ist, die eigentliche „Intelligenz“ könnte dann vom Netzwerk als nahtlos eingefügter Dienst kommen.

Durch den Video-Dienst wird eine wesentlich größere Prozessorleistung in den Endgeräten vorhanden sein müssen. Mit Bluetooth können dann Videos zum Endgerät übertragen werde.

Es existieren sogar Spekulationen, wonach zukünftige Endgeräte holographisch arbeiten sollen. Fest steht bisher eigentlich nur, daß die Endgeräte GPS-fähig und farbig, nicht mehr schwarz/weiß sein werden. Es existieren zwar bereits Studien von der Firma Nokia, wie ein solches Endgerät aussehen könnte. Diese sind jedoch wirklich nur Studien, die tatsächlichen Endgeräte könnten ganz anders aussehen als die in Abbildung 7 dargestellten.

Abbildung 7: 3g Endgeräte Studie von Nokia [Noki]

9 Zusammenfassung

Im Zeitalter der „Globalisierung“ und in Anbetracht der Tatsache, daß zwischen September 1999 und Januar 2000 weltweit alleine für GSM 30 Millionen neue Benutzer hinzukamen und die Internetgemeinschaft nicht viel weniger schnell wächst, wird es tatsächlich an der Zeit, diese beiden Bereiche miteinander zu verbinden. Der Standard der 3. Mobilfunkgeneration IMT-2000 wird eine solche Verbindung ermöglichen.

Das Zusammenspiel von breitbandigen Satelliten und verbesserten terrestrischen Komponenten versprechen Datenraten bis zu 2 Mbit/s und weltweites Roaming. Der Mobilfunk wird nahtlos in das Festnetz und damit in das Internet übergehen. Videokonferenzen, e-commerce und von mehreren Service-Providern angebotene Anwendungen werden mit „intelligenten“ Endgeräten benutzt werden können.

Momentan existieren noch viele Spekulationen, wie diese neue Mobilfunkumgebung genau aussehen und funktionieren wird. Fest steht bisher eigentlich nur, daß sie kommt und zwar sehr bald.

Literatur

[Alca] Alcatel. www.alcatel.com.

[AlSe99] Marc-Peter Althoff und Peter Seidenberg. Dienstekonzept und Spektrumsbedarf für UMTS. In *UMTS-Systeme der 3. Mobilfunkgeneration*, Congress Center Düsseldorf, April 1999.

[Beck99] Helmut Becker. Fixed/Mobile Convergence and UMTS. In *UMTS-Systeme der 3. Mobilfunkgeneration*, Congress Center Düsseldorf, April 1999.

[Data] GSM Data. www.gsmdata.org.

[Date] Datenfunk. www.dafu.de.

[Eric] Ericsson. www.ericsson.se.

[Foru] UMTS Forum. www.umts-forum.org.

[fTuP] Regulierungsbehörde für Telekommunikation und Post. www.regtp.de.

[GMSA] Global Mobile Suppliers Association. www.gsassociation.org.

[GPP2] Third Generation Partnership Project 2. www.3gpp2.org.

[GPPr] Third Generation Partnership Project. www.3gpp.org.

[ITUn] International Mobile Telecommunications (IMT-) 2000 International Telecommunications Union. www.itu.int/imt/.

[Lutz99] Dr. Erich Lutz. S-UMTS, Die Satellitenkomponente von UMTS. In *UMTS-Systeme der 3. Mobilfunkgeneration*, Congress Center Düsseldorf, April 1999.

[Naß99] Markus M. Naßhan. UTRAN TDD. In *UMTS-Systeme der 3. Mobilfunkgeneration*, Congress Center Düsseldorf, April 1999.

[Noki] Nokia. www.nokia.com.

[Schi99] Jochen Schiller. Mobilkommunikation Vorlesung, 1999.

[Skö99] Johan Sköld. The UTRA FDD Mode. In *UMTS-Systeme der 3. Mobilfunkgeneration*, Congress Center Düsseldorf, April 1999.

[TSIn] European Telecommunications Standards Institute. www.etsi.org.

[Ži00] Dušan Živadinović. Wellensalat satt, Daten-Mobilfunk holt Modems ein. *c't*, Februar 2000.

[Will99] Ingo Willimowski. UTRAN – UMTS Terrestrial Radio Access Network. In *UMTS-Systeme der 3. Mobilfunkgeneration*, Congress Center Düsseldorf, April 1999.

[Worl] GSM World. www.gsmworld.com.

[Zoll99] Dr. Ernst Zollinger. The Radio Channel in UMTS-Systems. In *UMTS-Systeme der 3. Mobilfunkgeneration*, Congress Center Düsseldorf, April 1999.

Betriebssysteme mit kleinem footprint

Christina Schmidt

Kurzfassung

Die vorliegende Seminarausarbeitung trägt der wachsenden Bedeutung von mobilen Kommunikationseinheiten und eingebetteten Systemen Rechnung und stellt nach einer Einleitung in die Thematik sowie der Beschreibung des Aufbaus eines typischen mobilen Betriebssystems drei verbreitete Betriebssysteme vor, die speziell für diese Anwendungsbereiche entwickelt wurden: Epoc32, PalmOS und WindowsCE. Diese werden dann einem zusammenfassenden Vergleich unterzogen, der die prägnanten Unterschiede und jeweiligen Schwächen und Stärken herausstellen soll. Aufgrund seiner zunehmenden Verbreitung wird auf WindowsCE respektive die Systemarchitektur in einem gesonderten Abschnitt genauer eingegangen, ein Einsatzbereich - WindowsCE in industriellen Anwendungen - detailliert aufgezeigt und schließlich ein Ausblick gewagt auf die zukünftigen Entwicklungen im Bereich der mobilen Betriebssysteme.

1 Einleitung

Mobile Kommunikationseinheiten erfreuen sich dank ihrer kleinen Ausmaße und der wachsenden Funktionalität bereits heute einer immensen Beliebtheit im Bereich der Telekommunikationsanwendungen, und dieser Trend wird sich dank der rasch fortschreitenden Entwicklung noch weiter verstärken.

Schon jetzt werden Geräte mit vielfältigen Fähigkeiten und Kommunikationsmöglichkeiten angeboten, der Benutzer erhält Zugang zu Telefon- und Fax-Diensten und zum Internet.

Grundsätzlich lassen sich nach [Lawt99] zwei Arten dieser mobilen Kommunikationseinheiten unterscheiden:

- „Intelligente Handies“: Dies sind Mobiltelefone, die über zusätzliche Informations-Funktionen verfügen, Zugang zum World Wide Web anbieten und außerdem noch mit Adreßbuch- und Terminplaner-Anwendungen aufwarten.
- „Communicators“: Diese sind schwerpunktmäßig auf Informations-Funktionen ausgerichtet und bieten zusätzlich noch Internet-Zugang, Telefon-, E-Mail-, Pager- und Fax-Dienste an.

Die zunehmende Vielfalt an Personal Digital Assistants (PDAs), intelligenten Handies und Embedded-Systemen erfordert jedoch auf sie zugeschnittene, leistungsfähige Betriebssysteme mit geringem Speicherbedarf, oft müssen auch Automatisierungsaufgaben mit Echtzeitanforderungen bewältigt werden, weshalb es hier eine große Herausforderung von den Anbietern solcher Betriebssysteme zu meistern gilt.

Wurden die erforderlichen speziellen Betriebssysteme früher meist noch firmenintern entwickelt, so erwies sich über kurz oder lang outsourcing wegen der kurzen Produktlebensdauer als vorteilhaft. Dies führte zur Entwicklung verschiedener Betriebssysteme für die genannten Anwendungsbereiche, und damit zwangsläufig zur Konkurrenz um Marktanteile.

Der Konkurrenzkampf rührt nicht nur daher, daß der Hersteller des erfolgreichsten Betriebssystems natürlich einen ansehnlichen Gewinn zu erwarten hat, sondern auch aus der Position, die damit erlangt wird: Er kann dann stark beeinflussen, in welche Richtung sich die Technologie auf dem Gebiet der mobilen Kommunikationseinheiten bewegen wird.

Drei der Hauptkonkurrenten auf diesem heiß umkämpften Markt sind Symbian mit Epoc32, Palm Computing mit seinem PalmOS sowie Microsoft mit WindowsCE, welche nun im folgenden betrachtet werden sollen.

2 Vergleich von Epoc32, PalmOS und WindowsCE

2.1 Aufbau eines typischen mobilen Betriebssystems

Wegen der geringen Größe der mobilen Kommunikationseinheiten, welche beschränkten Speicherplatz und begrenzte Energieversorgung zur Folge hat, sind die Anforderungen eines mobilen Betriebssystems deutlich geringer als die eines Desktop-Betriebssystems.

Die Architektur eines typischen mobilen Betriebssystems ist der folgenden Grafik aus [Lawt99] zu entnehmen:

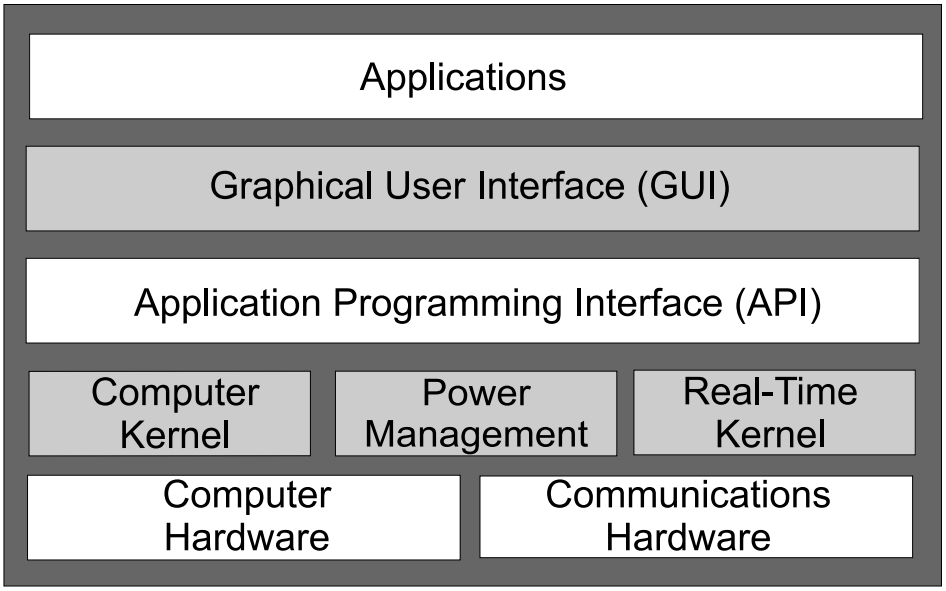

Abbildung 1: Systemarchitektur eines typischen mobilen Betriebssystems

Ein mobiles Betriebssystem kann als Stapel („Stack“) veranschaulicht werden, mit der Computer- und Kommunikations-Hardware als Fundament. Die nächste Schicht umfaßt den Computer-Kernel, den Echtzeit-Kernel (welcher die Verbindungen des Kommunikationsbereiches verwaltet und in einem Standard-Desktop-Betriebssystem nicht vorhanden ist) und die Power-Management-Komponente, die ebenfalls nicht in Standard-Desktop-Betriebssystemen vorkommt. Die meisten mobilen Betriebssysteme benötigen die separaten Kernel,

um die Computer- und die Kommunikations-Hardware zu betreiben. Anwendungen nutzen die API-Schnittstelle, um Informationen auf der grafischen Benutzeroberfläche darzustellen. (Diese unterliegt in mobilen Betriebssystemen stärkeren Einschränkungen als in Desktop-Betriebssystemen.)

Ein mobiles Betriebssystem hat im allgemeinen einen footprint von 68 KByte bis zu 2 MByte, benötigt 2-4 MByte ROM und 2-16 MByte RAM, wobei der Arbeitsspeicher als dauerhafter Speicher fungiert, wenn das Gerät ausgeschaltet wird.

Im Vergleich dazu hat ein Desktop-Betriebssystem einen footprint von 4-16 MByte und benötigt 16-64 MByte RAM sowie mehrere GByte für das dauerhafte Speichern von Daten.

Die kritischen Punkte bei einem mobilen Betriebssystem sind ähnlich denen, die bei einem Desktop-Betriebssystem auftreten, wenngleich sie oft schwieriger zu handhaben sind aufgrund von größenabhängigen Faktoren.

Diese kritischen Punkte beinhalten Kompatibilität, Portabilität, Skalierbarkeit, die Unterstützung mehrerer Prozessoren, die Qualität der Benutzerschnittstelle und die Leistung in Echtzeitanwendungen (beispielsweise bei Sprachverbindungen). Ein anderer wichtiger Punkt ist die Fähigkeit der mobilen Kommunikationseinheit, Verbindung mit PCs und LANs aufzunehmen.

2.2 Epoc32

Mitte 1998 schlossen sich Ericsson, Motorola, Nokia und Psion in einem Joint Venture zusammen und gründeten ein Unternehmen namens Symbian, um Software- und Hardwarestandards für die nächste Generation von mobilen Kommunikationseinheiten zu entwickeln und im Markt der mobilen Betriebssysteme konkurrieren zu können.

Die Gründe für diese Zusammenarbeit waren nach [Futu99] teilweise defensiv - die teilnehmenden Unternehmen wollten verhindern, daß WindowsCE das de facto Standard-Betriebssystem für Mobilgeräte wird -, teilweise offensiv - es sollte die Entwicklung eines Standard-Betriebssystems für mobile Kommunikationseinheiten angeregt werden.

Symbians mobiles Betriebssystem, Epoc32, wurde entwickelt, um eine offene Plattform für intelligente Handies zu schaffen. Das System hat einen footprint von 1 MByte und einen im Betriebssystem eingebauten Hardware-Fehler-Schutz, welcher jedoch einen erheblichen overhead benötigt. Epoc32 beinhaltet nach [Alt98a] viele der features eines Desktop-Betriebssystems, einschließlich echtem Multitasking, geschützter Speicherbereiche und dem Einbetten von Dokumenten ineinander. Zusätzlich ist es schnell, robust und effizient in der Energienutzung, da es speziell für seinen Anwendungsbereich, intelligente Handies und Communicators, entworfen wurde. Dies zeigt sich unter anderem in der Speicherverwaltung, die in [Ltd.99a] dokumentiert ist: In typischen Epoc32-Umgebungen ist Speicherplatz knapp, so daß Programme mit Ressource-Engpässen zu rechnen haben und damit umgehen können müssen. Epoc32 stellt einen Rahmen zur Verfügung, der die Programme dabei unterstützt, mit sehr geringem overhead und kompaktem Code: So wird durch Überprüfung aller GUI-Programme sichergestellt, daß deren Speichernutzung im zulässigen Rahmen liegt. Muß das System eine Anwendung wegen mangelnden Speicherplatzes schließen, so wird dafür gesorgt, daß kein Datenverlust entsteht.

Der Kernel von Epoc32 hat sehr geringe Ausmaße, so daß nur ein minimaler Teil des Codes im privilegierten Modus laufen muß und hohe Leistungsfähigkeit gewährleistet wird.

Epoc32 stellt, wie in [Ltd.99b] aufgeführt, eine ganze Reihe von Anwendungen und eine Programmiersprache zur Verfügung, eine Laufzeit-Umgebung für Java wird ebenfalls angeboten; Schriftenerkennung ist kein Standard-Zubehör, kann aber durch Dritt-Software hinzugefügt werden. Das Betriebssystem ist kompatibel zu verschiedenen Gerätetypen, Prozessoren und Hardware-Konfigurationen, die Benutzeroberfläche ist an verschiedene Anwendungsgebiete anpaßbar, außerdem bietet Epoc32 Telefonie-Unterstützung. Die Synchronisation mit PCs und Servern ist möglich, zwischen Geräten, die jeweils unter Epoc32 laufen, kann per Infrarot-Schnittstelle kommuniziert werden. Epoc32 ermöglicht ebenfalls den Anschluß eines Druckers und Kommunikation über das Internet via TCP/IP.

Im März 99 wurde das Ericsson R380 Mobiltelefon mit Epoc32-Betriebssystem vorgestellt, welches über die zusätzlichen Funktionen eines intelligenten Handies verfügt, z.B. Unterstützung von WAP, E-Mail und Organizer-Funktionalität.

Insgesamt wird der Erfolg von Symbian wohl davon abhängen, inwieweit seine Gründungsmitglieder es fertigbringen werden, Epoc32 bei anderen PDA- und Telekommunikations-Anbietern zu lizensieren. (Bereits lizensiert wurde Epoc32 durch Philips, Sun und NTT DoCoMo.)

2.3 PalmOS

Im Jahr 1998 erreichte 3Com, wozu Palm Computing gehört, einen Anteil von 72% am weltweiten Handheld-Computer-Markt, während Konkurrent Microsoft mit WindowsCE auf 15% Marktanteil kam. 1999 wird Microsoft seinen Marktanteil voraussichtlich verdoppeln, während der Anteil von Palm auf 64% zurückgehen wird (Quelle: [Futu99]).

Trotz dieser Entwicklungen setzen die Palm-PDAs immer noch den Standard für leichtgewichtige Kleincomputer, denn nicht umsonst ist PalmOS hauptsächlich bekannt als Betriebssystem für PalmPilot-Organizer. Es wird jedoch auch in anderen mobilen Kommunikationseinheiten eingesetzt, z.B. intelligenten Handies, denn nachdem PalmOS jahrelang für den Einsatz in Organizern entwickelt wurde, kann das Betriebssystem nun mit minimalen Ressourcen eine brauchbare Anzahl von Anwendungen zur Verfügung stellen.

PalmOS wurde speziell für den Einsatz in kleinen, kostengünstigen Computern in Hemdtaschengröße entworfen, mit dem Ziel, schlicht und einfach zu bedienen zu sein, was gleichzeitig zu höherer Leistungsfähigkeit verhilft.

Daher benötigt das Betriebssystem relativ wenig Ressourcen und ist somit ideal für die Anwendung in billigen Geräten mit mäßiger oder minimaler Funktionalität.

Das normalerweise in einem aufrüstbaren ROM gespeicherte Betriebssystem hat einen footprint von 1 MByte, ist stabil und hat einen sehr geringen Energieverbrauch. Das wird unter anderem durch eine klar gestaltete Benutzeroberfläche erreicht und durch eine stift-basierte Eingabe, die auf einem einfachen Schriftenerkennungsprogramm beruht. Diese zwingt den Benutzer zwar dazu, sich einen bestimmten Stil anzueignen, beansprucht dafür jedoch weniger Prozessorleistung und spart Batteriekraft ein.

Der kleine footprint rührt aus dem einfachen Anwendungs-Design her und der Taktik, Anwendungen im Hintergrund bei Bedarf zu schließen, wobei die Daten dieser Programme gespeichert werden.

PalmOS verfügt über keine spezielle Speicheraufräum-Unterstützung, da dies ohnehin in regelmäßigen Intervallen vorgenommen wird, weil Anwendungen sehr häufig geschlossen werden. Aus diesem Entwurf rührt jedoch auch laut [Ltd.99a] eine Schwäche von PalmOS her: Das PalmOS-Konzept der Speicherverwaltung ist nur schwer auf größere Umgebungen zu übertragen, die Multitasking unterstützen, ohne Hintergrund-Anwendungen zu schließen.

Außerdem unterliegt das Betriebssystem einer weiteren Beschränkung: Es läuft nur auf dem Motorola 68000-Prozessor.

PalmOS ist gedacht für Mobilgeräte, die regelmäßig mit Desktop- oder Laptop-Computern synchronisiert werden, und bietet deshalb keine eigenständige Datenarchivierung, Backup- und Druck-Funktionen. Entsprechend ist die Synchronisation mit Desktop-Systemen unkompliziert, und Informationen können zwischen Geräten auf PalmOS-Basis per Infrarot-Schnittstelle ausgetauscht werden.

Gegenüber WindowsCE wirkt PalmOS schlichter und etwas veraltet, aber genau diese Schlichtheit wird von den Benutzern geschätzt, da dies auch einen Teil der Leistungsfähigkeit von PalmOS ausmacht. In einer Hinsicht hat PalmOS jedoch mit der Version 3.5 - beschrieben in [Comp99] - Boden gutgemacht: Es unterstützt nun ebenfalls Farbdarstellung.

Die Hauptpartner von Palm sind IBM mit WorkPad und Symbol, die ihre Produkte mit PalmOS als Betriebssystem anbieten.

Um dem zunehmenden Konkurrenzdruck durch Microsoft entgegenzuwirken und die Unterstützung von Palm-Produkten zu sichern, engagiert sich 3Com stark in Bündnissen mit anderen Firmen, zu denen beispielsweise Qualcomm und Mitsubishi zählen.

Erfolgversprechend ist, wie in [Alt98b] nachzulesen, ebenfalls der Einsatz von PalmOS in PDAs von 3Com und anderen Firmen, welche vielfältige Kommunikationsdienste wie E-Mail, Fax, Pager und Telefon anbieten.

2.4 WindowsCE

WindowsCE kann als Konglomerat zweier eingestellter Microsoft-Projekte angesehen werden: „WinPad“ sollte als ein Handheld-Computer auf den Markt kommen, während das andere Projekt, „Pulsar“, als Pager-System entwickelt wurde. Diesen Projekten war jedoch kein Erfolg beschieden, und 1994 wurden beide Projekte aufgelöst und kombiniert, woraus dann WindowsCE entstand.

WindowsCE ist ein selbständiges, multitaskingfähiges Betriebssystem mit kompatiblen Programmierschnittstellen zu Windows95/98 bzw. NT.

Grundlage von WindowsCE ist ein Betriebssystem-Kernel, der eine Untermenge von Win32 darstellt und der auf RISC-Prozessoren (60-80 MHz laut [ScRi98]) und seit der Version 2.0

(beschrieben in [Greh98]) auch auf x86-Prozessoren eingesetzt werden kann.

Das Betriebssystem hat einen footprint von bis zu 2 MByte, wobei in der minimalen Konfiguration zwar nur 200 KByte ROM benötigt werden, eine angemessenere Konfiguration, bestehend aus Kernel, Dateisystem und Stapel für die Kommunikation, jedoch nach [Greh98] ca. 0,5 MByte ROM und 256 KByte RAM erfordert, laut [Woll98] sind für ein Minimalsystem 512 KByte ROM und 350 KByte RAM vonnöten. Ein Handheld-PC, der typische Anwendungen geladen hat, benötigt schließlich ca. 4MByte ROM und 2 MByte RAM.

WindowsCE ist damit das größte und auch das langsamste mobile Betriebssystem, erfährt aber gleichzeitig die breiteste Unterstützung durch Anbieter von Mobil-Geräten und Anwendungen. Es wird eingesetzt auf diversen Handheld-PCs von Casio, Compaq, Hitachi und anderen und findet auch sonst Anwendung in verschiedensten Produkten des Marktes für Mobil- und Kleingeräte: in stiftgesteuerten Kleincomputern (Palm-size PCs), in Handheld-PCs und auch in Industrie-Terminals. Ungefähr ein Dutzend Firmen wie z.B. Casio, Philips, Compaq und Sharp haben bereits Mobilgeräte - mit farbigen Displays - herausgebracht, die auf dem WindowsCE-Betriebssystem basieren. Zusätzlich hat Microsoft WindowsCE in diversen nicht-mobilen Anwendungen lizensiert, z.B. in Videospiel-Konsolen und Decodern für digitales Fernsehen, und Siemens hat sich, wie in [ScRi98] erwähnt, bei seinen technischen Geräten für die Anwendung von WindowsCE in eingebetteten Systemen entschieden.

WindowsCE ist nicht einfach eine abgespeckte Version von früheren Win32-basierten Betriebssystemen wie NT oder 95/98, sondern ein von Grund auf neu gestaltetes System, welches speziell für eingebettete Systeme entwickelt wurde. Bei genauerer Betrachtung findet man eine weitreichende Modularisierung vor, und eine verhältnismäßig überschaubare hardwareabhängige Zwischenschicht ermöglicht es, mit geringem Aufwand Lösungen für die unterschiedlichsten technischen Gegebenheiten zu finden. Besonderer Wert wurde beim Entwurf auf die effiziente Ausnutzung der Systemressourcen, vor allem des Arbeitsspeichers, gelegt - unter WindowsCE sind sogar Systeme ohne Display möglich.

WindowsCE ist trotz seiner separaten Entwicklung in vielerlei Hinsicht ähnlich zu anderen Windows-Versionen, was von der Verwendung einer Untermenge der Win32-API herrührt. Dies hat den Vorteil, daß Anwendungsentwickler, die bereits mit der Windows-API vertraut sind, einfacher damit arbeiten können.

Gleichzeitig jedoch läßt sich aus der Verwendung der Win32-API eine Schwäche von WindowsCE ableiten: Die Win32-API wurde nicht für eine allumfassende Fehlerbehandlung und eine saubere Speicherbereinigung entworfen, was jedoch für Betriebssysteme auf mobilen Kommunikationseinheiten mit beschränkten Systemressourcen eine nicht zu vernachlässigende Bedeutung besitzt. So kann es unter WindowsCE passieren, daß bei zuwenig freiem Speicher und nicht erfolgter Freigabe von Speicherplatz durch den Benutzer (durch Schließen einer Anwendung) bzw. einer Anwendung selbst das System selbständig Anwendungen schließt, wobei dann die Gefahr von Datenverlust besteht. Die Speicherverwaltung von WindowsCE ist also nicht auf die Bedürfnisse kleiner Systeme zugeschnitten.

Beim WindowsCE-Konzept wird großer Wert auf die Verwendung von Threads gelegt, was dem System bessere Taskwechselzeiten ermöglicht, als dies bei Windows95/98 und NT der Fall ist. WindowsCE kann Mehrfachprozesse unterstützen, seine wahre Multitasking-Stärke liegt jedoch im Multithreading: Die Anzahl der erlaubten Threads ist nur durch die zur Verfügung stehenden Speicherressourcen begrenzt.

Die Ausführung eines Interrupts blockiert bei diesem Betriebssystem andere Interrupts nur für eine möglichst kurze Zeit, was WindowsCE in vielen weichen Echtzeit-Anwendungen einsetzbar macht. Der Nachteil dieses recht simplen Interrupt-Handling-Konzepts war bisher jedoch, daß prinzipiell Interrupts verloren gehen konnten und die Reaktionszeit sehr stark mit dem Systemzustand variierte. Daher war WindowsCE bisher kein Betriebssystem für harte Echtzeit-Anwendungen. In WindowsCE 3.0 wurden jedoch Verbesserungen am Interrupt-Handling-Konzept vorgenommen, auf die in Abschnitt 3.1.1 genauer eingegangen wird.

Eine Stärke von WindowsCE ist seine ausgeprägte Kommunikationsfähigkeit, denn es eröffnet breite Möglichkeiten beim Datenaustausch und der Synchronisation zwischen PDAs unter WindowsCE und Desktop-PCs: Laut [ScRi98] bietet die Hardware einen IrDA-kompatiblen Infrarotanschluß, eine serielle Schnittstelle, einen PC-Card-Slot sowie bei manchen Modellen FlashCard-Einschub, auf der Softwareseite stehen Protokolle wie SMTP, SLIP, POP3 und TCP/IP zur Verfügung.

Ein weiterer Pluspunkt von WindowsCE ist seine Lauffähigkeit auf einer Vielzahl von Prozessoren sowie seine Eignung für Anwendungen, bei denen Visualisierung und Kommunikation im Vordergrund stehen. Dies birgt jedoch gleichzeitig auch einen Nachteil von WindowsCE in sich: Durch die aufwendig gestaltete Benutzeroberfläche und das Farb-Display hat das System einen hohen Stromverbrauch, worauf in [ScRi98] hingewiesen wird.

WindowsCE zielt neben intelligenten Handies und Communicators auch auf andere mobile Kommunikationseinheiten. Jedoch stellt sich hierbei das Problem, daß das Betriebssystem aufgrund seiner Größe zwar bei einer großen Bandbreite an Produkten eingesetzt werden kann, aber Mobilgeräte, die klein sein und effizient mit ihrer Energieversorgung haushalten müssen, möglicherweise vor schwer zu überwindende Hindernisse stellt.

Zudem war WindowsCE vor Version 3.0, wie bereits erwähnt, noch nicht geeignet für harte Echtzeit-Anwendungen, die Echtzeit-Unterstützung soll jedoch in WindowsCE 3.0 verbessert sein.

2.5 Zusammenfassender Vergleich

Stellt man nun die drei betrachteten mobilen Betriebssysteme gegenüber, so lassen sich durchaus einige Unterschiede feststellen:

Während Epoc32 speziell für intelligente Handies und andere kleine Mobilgeräte entworfen wurde, ebenso wie PalmOS, ist WindowsCE bei einer Vielzahl von Produkten, sowohl im Bereich der mobilen Kommunikation als auch in eingebetteten Systemen, wofür es speziell entwickelt wurde, einsetzbar: Durch seine Modularisierung kann eine Anpassung an das jeweilige Anwendungsgebiet mit geringem Aufwand vorgenommen werden.

Das hat natürlich Konsequenzen für die Ausgestaltung der einzelnen Betriebssysteme: Epoc32 kann ebenso wie PalmOS mit einem kleinen footprint von 1 MByte aufwarten, der kleine Kernel sorgt bei beiden Betriebssystemen für eine hohe Leistungsfähigkeit, beide sind stabil und effizient in der Energienutzung, wogegen WindowsCE mit einem footprint von bis zu 2MByte deutlich größer und auch langsamer ist sowie durch eine aufwendig gestaltete Benutzeroberfläche und Farb-Display mit einem hohen Stromverbrauch zu kämpfen hat.

Daher leuchtet ein, daß WindowsCE in seiner jetzigen Ausgestaltung für kleine Mobilgeräte mit minimalen Ressourcen und hart kalkulierter Energieversorgung nur sehr schlecht geeignet ist, während Epoc32 und PalmOS hier das für sie gedachte Einsatzgebiet vorfinden.

Auftrumpfen kann WindowsCE jedoch, wenn es um die Entwicklung geht: Durch die Verwendung von Teilen der Win32-API in WindowsCE wird Entwicklern, die bereits mit der Win32-API vertraut sind, die Einarbeitung stark erleichtert, und es ist auch möglich, Programme, die für andere Windows-Systeme entwickelt wurden, ohne allzu großen Aufwand auf WindowsCE zu portieren.

Echt multitaskingfähig sind sowohl WindowsCE als auch Epoc32, während PalmOS hier Abstriche machen muß. Dies rührt vom Speicherverwaltungskonzept her, das bei PalmOS verwendet wird (siehe Unterabschnitt 2.3), welches zwar bei Kleingeräten sehr effizient ist, sich jedoch für größere Umgebungen mit Multitasking-Unterstützung kaum eignet.

Bei der Speicherverwaltung zeigt sich ein Nachteil, der sich aus der Verwendung der Win32-API in WindowsCE ergibt: Da diese nicht für Systeme mit beschränkten Ressourcen gedacht ist und daher bei der Speicherverwaltung von WindowsCE unerwünschte Effekte auftreten können (siehe Unterabschnitt 2.4), wird hierdurch die Eignung von WindowsCE für die Bedürfnisse kleiner Systeme in Frage gestellt; Epoc32 steht hier mit seiner in Unterabschnitt 2.2 ausgeführten ausgeklügelten Speicherverwaltung deutlich besser da.

Nachteilig ist bei PalmOS die Unterstützung nur eines Prozessors, während WindowsCE hier mit der Lauffähigkeit auf diversen Prozessoren aufwarten kann und Epoc32 zu mehreren Prozessoren kompatibel ist.

Zusammenfassend läßt sich sagen, daß Epoc32 und PalmOS gegenüber WindowsCE deutlich besser für kleine Mobilgeräte geeignet sind, dafür aber auch weitestgehend auf diesen Bereich beschränkt bleiben, während WindowsCE auf einer großen Bandbreite an Produkten einsetzbar ist und auch den Bereich der Echtzeitanwendungen im Visier hat.

Aufgrund der weitreichenden Einsatzgebiete von WindowsCE und seiner zunehmenden Verbreitung wird im nächsten Abschnitt genauer auf dieses Betriebssystem eingegangen.

3 WindowsCE

Ausgangspunkt für die vertiefende Betrachtung von WindowsCE soll die Systemarchitektur dieses Betriebssystems sein, welche aus der folgenden Grafik ersichtlich ist.

Ziel dieses Abschnitts ist es, die Hauptmodule von WindowsCE zu untersuchen: Den Kernel, den Object-Store, das Graphics Windowing Event System (GWES), den OEM Adaptation Layer (OAL) (Anmerkung: OEMs sind die Originalentwickler oder Hersteller von Produkten, welche unter verschiedenen Handelsmarken laufen), das Core-System-Interface sowie das Communication-Modul. Grundlegende Literatur hierzu ist [Murr98].

Anschließend wird auf den Einsatz dieses Betriebssystems in industriellen Anwendungen eingegangen, wo sich WindowsCE gegenüber den bisher dort verwendeten Betriebssystemen immer mehr durchsetzt.

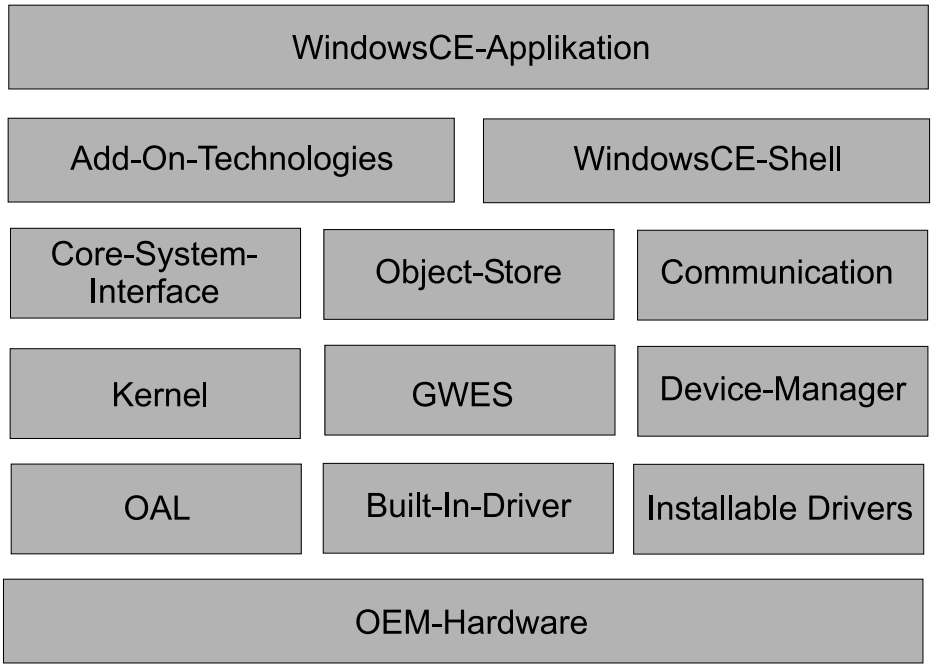

Abbildung 2: Modularer Aufbau von WindowsCE

3.1 WindowsCE - Systemarchitektur

3.1.1 Der Kernel

Der Kernel ist der innerste Kern eines Betriebssystems und verantwortlich für das Scheduling und die Synchronisation von Threads, für die Ausführung von Ausnahmen und Unterbrechungen, das Laden von Anwendungen und die Verwaltung des virtuellen Speichers.

Der WindowsCE-Kernel (in C programmiert) unterstützt die Ausführung direkt im ROM oder Seitenabruf in den Programmspeicher im RAM. Er wurde unter dem Aspekt größtmöglicher Portabilität entworfen, weshalb er auch auf einer Vielzahl von Prozessoren lauffähig ist. Diese müssen - nach aktuellem Stand - 32-Bit little endian Prozessoren sein und außerdem einen Translation Look-Aside Buffer unterstützen.

WindowsCE unterstützt das Win32-Prozeß- und Thread-Modell und verwendet einen nach dem Round-Robin-Prinzip arbeitenden, prioritätsgesteuerten Scheduler, wobei hochpriorisierte Threads niemals von niedrigpriorisierten unterbrochen werden; verschachtelte Interrupts sind jedoch seit Version 3.0 möglich, was dem Verlust und der Verzögerung von hochpriorisierten Interrupts vorbeugt und das Echtzeit-Verhalten von WindowsCE verbessert. Auf eine diesbezügliche Leistungssteigerung zielt auch die erweiterte Anzahl von Thread-Prioritäten: Bisher lagen acht diskrete Prioritäten vor, seit WindowsCE 3.0 sind es 256.

Für die Synchronisation von Threads stellt die Win32-API eine Vielzahl von Objekten zur Verfügung, durch die eine kritische Sektion, gegenseitiger Ausschluß („mutual exclusion“) und System-Ereignisse realisiert werden können; seit Version 3.0 werden auch Semaphoren unterstützt.

In WindowsCE laufen die meisten Prozesse im Benutzer-Modus ab, und nicht im Kernel-Modus - so sind zum Beispiel das Windows-Management-System, das Grafik-System und

das Datei-System selbständige Untersysteme und unabhängig vom Kernel. Damit wird der Situation vorgebeugt, daß ein Fehler in einem dieser Systeme bei einem Crash den Kernel beschädigt.

Die Behandlung von Unterbrechungen („interrupt handling“) wird in WindowsCE von zwei Software-Komponenten vorgenommen: einer traditionellen Interrupt-Service-Routine (ISR) und einem Interrupt-Service-Thread (IST). Wird eine Unterbrechung ausgelöst, wird die ISR angestoßen.

Ziel war es, die ISR möglichst klein und schnell zu machen. Daher wird die ISR einfach aktiviert, gibt ein Signal an das Betriebssystem ab und wird dann beendet. Dieses Signal wiederum alarmiert den entsprechenden IST. Durch diesen wird dann die Unterbrechung tatsächlich behandelt. Da der IST nichts anderes als ein gewöhnlicher Thread ist, kann die Anwendungs-Software die Prioritäts-Level für den IST steuern und so das Gleichgewicht zwischen Ressourcenverbrauch und Durchsatz sicherstellen. Durch diese ISR/IST-Kombination kann WindowsCE in vielen weichen Echtzeitanwendungen eingesetzt werden, und mit den in Version 3.0 vorgenommenen Verbesserungen soll auch die Anwendung in harten Echtzeitsystemen möglich sein.

In [Corp99] beschriebene Tests des Echtzeit-Verhaltens von WindowsCE 3.0 auf einem Pentium II 350 MHz (Pentium 90 MHz) zeigen für die ISR eine Latenzzeit von 3,3-5 μs (3,3-7,5 μs) bei einem Durchschnittswert von 3,5 μs (4,5 μs) sowie für den IST eine Latenzzeit von 10-14,2 μs (23,4-42,7 μs) bei einem Durchschnittswert von 12,1 μs (29,8 μs). Hierbei ist unter der Latenzzeit der ISR die Zeitspanne vom Auslösen einer Unterbrechung bis zum Start der ISR zu verstehen, während die Zeitspanne von der Beendigung der ISR bis zum Start des IST die Latenzzeit des IST darstellt.

3.1.2 Der Object-Store

Die dauerhafte Speicherung der Benutzerdaten erfolgt bei WindowsCE durch das Object-Store-Modul. Es werden dabei drei verschiedene Möglichkeiten unterstützt, um Daten zu sichern: Dazu gehören das File-Handling aus dem Win32-Funktionsumfang, die Registry, die mit derjenigen von Windows95/98/NT identisch ist, und schließlich eine WindowsCE-eigene API für Datenbanken, welche sich besonders zum Speichern von Datenblöcken, wie sie typischerweise in Terminplaner- oder Adreßverwaltungen anfallen, eignet.

Da viele eingebettete Systeme keinen Festspeicher beinhalten, wird in WindowsCE die Datenspeicherung auf einem internen Heap, der sich im RAM, im ROM oder in beiden befindet, aufgebaut. Der interne Heap stellt ein Transaktions-Modell zur Verfügung, welches die Datenintegrität sicherstellt.

Für Datei-Systeme stellt WindowsCE sein eigenes geschütztes Datei-System zur Verfügung, das auf dem internen Heap aufgebaut ist. In Zukunft sollen auch CD-ROM- und DVD-Laufwerke sowie ATA-Festplatten unterstützt werden.

WindowsCE bietet zudem einen sogenannten File System Driver (FSD) Manager an, um die Entwicklung neuer Datei-Systeme zu vereinfachen. Außerdem stellt Microsoft eine Implementation eines bedeutenden installierbaren Datei-Systems zur Verfügung: das FAT Datei-

System.

Von der Registry macht WindowsCE umfassenden Gebrauch, um konfigurierbare Systemeinstellungen herauszustellen. So können Entwickler von eingebetteten Systemen bestimmte Registry-Einträge nutzen, um spezielle Konfigurations-Informationen anzubieten.

Der Object-Store befindet sich im Datenspeicher, welcher vom Programmspeicher unterschieden wird. WindowsCE erlaubt den Entwicklern von eingebetteten Systemen, die relativen Ausmaße dieser beiden Speicherbereiche zu verändern. Die maximale Größe für ein RAM-Datei-System ist 256 MByte, bei einer maximalen Dateigröße von je 32 MByte.

3.1.3 Das Graphics Windowing Event System (GWES)

Das GWES ist verantwortlich für die gesamte Bedienoberfläche, es integriert das Win32-User- und GDI-Subsystem in einer Komponente.

Das User-System verarbeitet Eingaben von der Tastatur und vom Eingabestift. Es beinhaltet außerdem den Event-Manager, der für Meldungen zuständig ist, und den Window-Manager.

Der Window-Manager wurde speziell für kleine Geräte optimiert, indem die Minimierungs- und Maximierungs-Zustände entfernt wurden, da davon ausgegangen werden kann, daß die meisten Anwendungen den ganzen Bildschirm in Anspruch nehmen.

Das Grafiksystem (GDI, Graphical Device Interface) wurde durch ein kleineres, aber leistungsfähiges Multiplattform-GDI (MGDI) ausgetauscht, das den Anforderungen im Bereich eingebetteter Systeme besser gerecht wird.

Als weitere optionale Module kann das GWES mit Schnittstellen für diverse Eingabegeräte wie Tastatur und Maus bis hin zum Touchscreen aufwarten.

3.1.4 Der OEM Adaptation Layer (OAL)

Der OAL ist eine hauchdünne, OEM-spezifische und auf die jeweilige Hardware angepaßte Zwischenschicht, welche zwischen der Hardware des OEM und dem WindowsCE-Kernel vermittelt.

Der OEM oder der Entwickler eines eingebetteten Systems implementiert dabei den Boot-Loader, den sogenannten OEM Adaptation Layer (OAL), und die Gerätetreiber. Diese Treiber werden normalerweise in vier Kategorien unterteilt: WindowsCE native Treiber, WindowsCE stream interface Treiber, NDIS Netzwerk-Treiber und Universal Serial Bus (USB) Treiber. Andere Treiber-Modelle können in zukünftigen WindowsCE-Versionen unterstützt werden.

Die OAL-Funktionen kommunizieren zwischen dem Kernel und der Geräte-Hardware, wie zum Beispiel Timern, seriellen Ports, Parallel-Ports und Ethernet-Hardware. Inbegriffen sind auch Power-Management-Schnittstellen und Hardware-Unterbrechungs-Behandlung.

Der OAL wurde so entworfen, daß es sehr einfach zu implementieren ist und eine große Anzahl von Geräten unter WindowsCE einsatzfähig macht.

Dadurch, daß die OEMs die untersten Stufen des Betriebssystem steuern können, ist WindowsCE auch in (weichen) Echtzeitanwendungen einsetzbar.

3.1.5 Das Core-System-Interface (CSI) und das Communication-Modul

Über das CSI können Applikationen direkt auf Dienste des Betriebssystems zugreifen. Diese sind zum Beispiel der lokale Heap, Speicherzugriffe, serielle Schnittstellen oder das TAPI (Telephony-API, eine Programmierschnittstelle für automatisiertes Telefonieren, beispielsweise Wählen aus Datenbanken).

Interessant sind in diesem Zusammenhang auch die Dienste des Communication-Moduls. Die Hardware-Unterstützung reicht von der seriellen Datenübertragung via Modem bis hin zu kabellosen Verbindungen wie beispielsweise der Infrarot-Schnittstelle. Besonders erwähnenswert sind hier noch die Funktionen der TAPI sowie der Netzwerkdienste: Das Modul WinInet unterstützt ftp und http 1.0 sowie den WinSock-Sicherheits-Layer gemäß SSL.

Als Standardprotokoll wird in WindowsCE TCP/IP verwendet, darüber hinaus können Rechner unter WindowsCE als RAS-Clients (PPP) betrieben werden.

Durch das modulare Design von WindowsCE wird es OEMs und Entwicklern ermöglicht, in einfacher Weise Treiber und Protokolle hinzuzufügen sowie die unterstützten Kommunikations-Komponenten zu erweitern und sogar zu ersetzen.

3.2 WindowsCE in industriellen Anwendungen

Bisher waren grafische Benutzersysteme dem Consumer-Bereich vorbehalten, im industriellen Bereich dagegen, wo weniger die Systemleistung als der Kostenfaktor im Vordergrund steht, kamen häufig Betriebssysteme zum Einsatz, deren grafische Fähigkeiten deutlich unter dem heutigen Standard liegen.

Dies soll sich nun, wie in [Scha99] beschrieben, durch den Einzug von WindowsCE in multifunktionale Betriebskonsolen ändern, denn wo noch vor kurzer Zeit auf teure Speziallösungen zurückgegriffen werden mußte, besteht nun die Möglichkeit, eine Applikation gemäß den jeweiligen Anforderungen gezielt zu erstellen.

Dies wird durch den bereits erwähnten modularen Aufbau von WindowsCE ermöglicht: Das System, das in einem komplizierten Buildprozeß erstellt wird, kann durch eine Vielzahl von Konfigurationsmöglichkeiten auf die Hardware abgestimmt werden. Das birgt allerdings insofern Nachteile in sich, als daß der Entwickler sowohl die Hardware, die eigene Applikation als auch das CE-Betriebssystem sehr genau kennen muß, um eine optimale Integration vornehmen zu können.

Daher wurde diese Aufgabe durch Microsoft an sogenannte CE-Systemintegratoren übertragen, die auch die nötigen Treiber für spezielle Hardware liefern. So sind zum Beispiel im Lieferumfang von WindowsCE keine Treiber vorhanden, die es ermöglichen, Daten persistent auf Festplatte oder Floppy-Disk abzuspeichern.

Bisher waren 50 000 DM und mehr durchaus keine Seltenheit für eine WindowsCE-Integration, doch nun stehen kostengünstige Treiberpakete zur Verfügung, die den Integrationsaufwand erheblich reduzieren, was natürlich die weitere Verbreitung von WindowsCE-Systemen begünstigt.

Eine Systemintegration hat aber immer noch eine enge Zusammenarbeit mit einem CE-Systemintegrator zur Folge. Zusätzlich werden Schulungen angeboten, um durch Vermittlung von Grundwissen im Umgang mit WindowsCE den Einstieg zu erleichtern.

Einer solchen Integration geht die Auswahl der Komponenten voraus, wobei sich hierbei eine x86-Plattform aus mehreren Gründen anbietet:

- Es ist ein standardisiertes System mit bekannten und dokumentierten Ressourcen.
- Die verschiedenen CPU-Klassen von 80386 bis Pentium können bei gleichbleibender Software skaliert werden.
- Hochintegrierte Lösungen bieten eine direkte Grafikschnittstelle für LC-Displays.
- Zusätzliche Komponenten wie Watchdog und Matrixtastatur verringern den eigenen Designaufwand.
- Das System kann sehr leicht neuen Erfordernissen angepaßt werden.
- Die Softwareentwicklung kann auf einem Standard-PC erfolgen, man ist damit also nicht auf teure Entwicklungstools, zum Beispiel für spezielle Mikrocontroller, angewiesen.
- Die Verfügbarkeit und die Preisregulierung solcher Systeme wird durch eine Vielzahl von Second-Source-Anbietern gewährleistet.
- Durch die Verwendung eines Standard- bzw. Stückzahlenproduktes reduzieren sich die Kosten, und es ist umfassender Treibersupport vorhanden.

Bei Applikationen, die Grafik verwenden, bietet sich ein System auf 486er-Basis an, wobei auch durchaus 386er-Systeme zum Einsatz kommen können, nämlich in Black-Box-Anwendungen (zum Beispiel TCP/IP-Router).

Ein ganz spezieller und geradezu idealer Markt für den Einsatz von WindowsCE ist der Bereich der abgesetzten Terminals. Eine Vielzahl verschiedenartiger Bedienkonsolen können nun mit grafischen Eigenschaften versehen werden, die bisher Betriebssystemen wie Windows95/98 oder WindowsNT vorbehalten waren.

Denn wo bisher zum Beispiel für eine Windows95-Installation 45MByte nötig waren, kann nun unter WindowsCE schon mit unter 5MByte volles Windowing betrieben werden. Hinzu kommt hier noch die eigene Applikation.

Ein weiterer Pluspunkt für WindowsCE ist, daß bei bereits vorhandenen Windows-Applikationen (95/98 oder NT) viele Code-Fragmente der bestehenden Applikation direkt übernommen werden können. Allerdings stehen unter WindowsCE freilich, wie bereits in Unterabschnitt 2.4 erwähnt, nicht alle Funktionen der Win32-API zur Verfügung, so daß man sich ggf. mit Ersatzfunktionen behelfen muß.

Insgesamt läßt sich sagen, daß beim Umstieg auf WindowsCE in industriellen Anwendungen oft noch einige Hürden überwunden werden müssen, dennoch stehen jetzt schon wie erwähnt leistungsfähige Treiberpakete zur Verfügung, mit Hilfe derer eine Integration so einfach als möglich gemacht werden kann.

Laut [Scha99] wird WindowsCE nicht nur im Bereich der Embedded-Control-Anwendungen in Bedienterminals Einzug halten, sondern kann sich vielmehr mit den verbesserten Echtzeit-Eigenschaften von WindowsCE 3.0 durchaus zu einem ernstzunehmenden Betriebssystem im Bereich der Echtzeitanwendungen entwickeln.

Abschließend soll nun noch ein Ausblick auf kommende Entwicklungen gegeben werden, mit besonderem Schwerpunkt auf zu erwartende Marktdominanzen der betrachteten Betriebssysteme.

4 Ausblick

Nach der Aussage von Marktforschern in [Lawt99] ähnelt der Markt für mobile Kommunikationseinheiten in vielerlei Hinsicht dem Markt für PCs vor 20 Jahren: Eine Anzahl von Anbietern drängt auf den Hardware-Markt, mit dem gleichen Betriebssystem als Grundlage (DOS für PCs, WindowsCE für Mobilgeräte). Diese Betriebssysteme konkurrieren gegen die Systeme von Hardware-Anbietern (MacOS von Apple konkurrierte gegen DOS, PalmOS konkurriert gegen WindowsCE).

Trotz des zunehmenden Erfolges von WindowsCE hoffen die Anbieter mobiler Betriebssysteme verhindern zu können, daß Windows in diesem Bereich die gleiche Dominanz erreicht wie bei den Desktop-Betriebssystemen.

Als ein positiver Aspekt des harten Konkurrenzkampfes wird übereinstimmend die dadurch vorangetriebene Produktentwicklung gesehen.

Marktforscher prognostizieren laut [Lawt99], daß WindowsCE eventuell zumindest in einigen intelligenten Handies genutzt werden wird, jedoch erst nach einer Reduktion des Speicher- und Energiebedarfs. Bis dies jedoch erreicht ist, wird die Konkurrenz hart daran arbeiten, ihre Produkte zuerst in die Mobilgeräte zu bringen, und so könnte Microsoft gezwungen werden, auf einem bereits etablierten Markt zu operieren.

Microsoft wird jedoch möglicherweise trotz dieses Hindernisses in einer starken Position sein, denn WindowsCE wird wahrscheinlich de facto zum Standard für mobile Betriebssysteme von Unternehmen werden. Grund dafür ist, daß WindowsCE einfacher in bereits existierende Windows-basierte Umgebungen der Firmen zu integrieren ist, und Entwickler einfacher damit arbeiten können. Falls dies eintreten sollte, müßten Entwickler und Dritt-Software-Anbieter WindowsCE unterstützen, um im Unternehmensmarkt erfolgreich zu sein, was für Microsoft eine erneute Stärkung der Marktposition darstellen würde. Für WindowsCE wurden zudem bereits eine große Anzahl an Anwendungen entwickelt, und es liegen ebenfalls eine ganze Reihe an Hardware-Lizenzen vor.

WindowsCE ist außerdem das einzige mobile Betriebssystem, welches auf Produkten in den meisten Kategorien des Marktes für Mobilgeräte läuft. Auf intelligenten Handies ist

WindowsCE wie gesagt noch nicht zu finden, was für Symbian die Chance bietet, diese Marktnische für sich zu erobern.

Trotzdem ist die Dominanz von Microsoft unübersehbar, und um mit Microsoft konkurrieren zu können, werden Firmen bei gleicher Funktionalität preisgünstiger sein und Allianzen mit Anbietern von mobilen Kommunikationseinheiten eingehen müssen.

Auf dem Markt für Geräte, die auf der in Nordamerika weit verbreiteten CDMA-Technologie (CDMA: code-division multiple-access) basieren, wird eine weitgehende Marktbeherrschung von WindowsCE erwartet, da Microsoft mit Qualcomm, dem Entwickler der CDMA-Technologie und einem führenden Hersteller von digitalen mobilen Kommunikationseinheiten, jüngst eine Absatzabsprache vorgenommen hat. Jedoch darf nicht vernachlässigt werden, daß PalmOS, im Gegensatz zu WindowsCE, bereits in einem CDMA-basierten Produkt angewendet wurde und außerdem schon seit längerem auch in anderen Arten von Kleingeräten genutzt wird.

Nach [Lawt99] könnte Symbian den Markt für Mobilgeräte beherrschen, die auf dem GSM-Standard (GSM: Global System for Mobile Communication) beruhen. Dies läßt sich darauf zurückführen, daß zu seinen Eigentümern Ericsson, Motorola und Nokia (siehe Unterabschnitt 2.2) gehören, die mehr als 60% des Weltmarkts für GSM-basierte intelligente Handies und Communicators auf sich vereinigen. Andere Anbieter von GSM-basierten Mobilgeräten könnten sich ebenfalls für Symbian entscheiden, um Kompabilität zu garantieren.

Andere Quellen ([Futu99]) sind hier pessimistischer: Die in Symbian zusammenarbeitenden Firmen sind eher technologie-orientiert als auf die Anwendungsentwickler und Benutzer ihrer Produkte ausgerichtet, so daß mangelnde Unterstützung der Entwickler dazu führen könnte, daß diese sich von Symbians Epoc32 ab- und WindowsCE zuwenden.

Für PalmOS werden in [Futu99] bessere Zukunftschancen vorhergesagt, mit dem Hinweis darauf, daß PalmOS von den meisten Endbenutzern verwendet wird und zusätzlich eine rasante Produktentwicklung vorweisen kann. Palm operiert allerdings nach wie vor in einer Unterkategorie des Marktes für drahtlose Kommunikationsgeräte, nämlich den Organizern wie dem PalmPilot, und konzentriert sich stark auf die Benutzer seines PalmOS und deren Support, beispielsweise in Form von regelmäßigen Updates.

Insgesamt läßt sich sagen, daß mobile Kommunikationseinheiten weiterhin eine rasante Entwicklung durchmachen und immer mehr und immer komplexere Funktionen anbieten werden. Daher wird für den Erfolg der jeweiligen mobilen Betriebssysteme von entscheidender Bedeutung sein, wie gut sie diese neuen Funktionen und Dienste unterstützen.

Literatur

[Alt98a] AltOS. *Epoc32 Guide - Summary*, 1998.

[Alt98b] AltOS. *PalmOS Guide - Summary*, 1998.

[Comp99] Palm Computing (Hrsg.). PalmOS Software 3.5 Overview. Technischer Bericht, Palm Computing, 1999.

[Corp99] Microsoft Corporation (Hrsg.). Designing and Optimizing Microsoft Windows CE 3.0 for Real-Time Performance. Technischer Bericht, Microsoft Corporation, Juni 1999.

[Futu99] FutureFoneZone (Hrsg.). Wireless Operating Systems. White Paper, FutureFoneZone, 1999.

[Greh98] Rick Grehan. Windows CE - new software force in embedded? *Computer Design*, Januar 1998.

[Lawt99] George Lawton. Vendors Battle over Mobile-OS Market. *Technology News*, Februar 1999.

[Ltd.99a] Symbian Ltd. (Hrsg.). Approaches to memory management. Technischer Bericht, Symbian Ltd., 1999.

[Ltd.99b] Symbian Ltd. (Hrsg.). Epoc Overview: Summary. Technischer Bericht, Symbian Ltd., Juni 1999.

[Murr98] John Murray. *Inside Microsoft Windows CE*. Microsoft Press. 1998.

[Scha99] Michael Schanz. Bedienkonsole - Windows CE in industriellen Anwendungen. *ElektronikPraxis* (5), März 1999.

[ScRi98] Dr. Thomas J. Schult und Dr. Jürgen Rink. Das Imperium sticht zurück - Microsofts Feldzug für ein mobiles Windows. *c't* (8), 1998.

[Woll98] Dr. Jörg F. Wollert. Zwergen Anatomie - Aufbau von und Entwicklung für Windows CE. *c't* (8), 1998.

Weg mit dem Flaschenhals im Rechner

Rainer Vogt

Kurzfassung

Mit der wachsenden Bedeutung des Einsatzes verteilter Systeme und des Internets selbst wird auch die Frage, wie Daten am effektivsten übertragen werden können immer wichtiger. Um dies zu erreichen ist es notwendig, bessere und schnellere Netzwerkkarten zu entwickeln, die bisher noch einen Flaschenhals in der Datenübertragung darstellen.

1 Einleitung

Kommunikation wird in der modernen vernetzten Welt immer wichtiger. Zum einen wird es bald niemanden mehr geben, der noch nicht im Internet ist, zum anderen sind Netzwerke in der Industrie nicht mehr wegzudenken. Hinzu kommt, daß die Computertechnologie in der jüngeren Vergangenheit enorme Fortschritte verzeichnet hat und dies immer noch tut. Man denke nur an die heutigen Prozessorraten oder auch an die Gigabit/Sekunde-Netzwerke. Leider ist es bisher so, daß die Netzwerkkarten diese rasante Entwicklung nicht mitmachen konnten. Sie stellen auf den jetzigen Kommunikationswegen einen Flaschenhals dar, der die ganze Kommunikation massiv ausbremst. Um also den heutigen Anforderungen gerecht werden zu können, sind die Entwickler gefordert, schnellere und effizientere Netzwerkanbindungen zu entwickeln. Um dieses Ziel zu erreichen, kann man an verschiedenen Punkten ansetzen. Neben der eigentlichen Netzwerkkarten-Hardware, die die Daten überträgt, Schutz garantiert und Kommunikationsereignisse erzeugt liegt ein besonderes Augenmerk vor allem auf der Netzwerk-Software, die das Netzwerk verwaltet und Kommunikationsprotokolle implementiert. Diese beiden Faktoren können aber sehr unterschiedliche Ausprägungen haben, abhängig davon, welche Anwendungen man im Auge hat, welche Leistungsansprüche sich daraus ergeben und mit welcher Netzwerktechnologie man es zu tun hat. Die Netzwerkanbindung kann man in drei große Bereiche aufteilen:

- PCs, verbunden über ein LAN oder WAN
- PCs in einem industriellen SAN (System Area Network) oder
- parallele Prozessoren in einem Kundenspezifischen Netzwerk.

Jeder dieser drei Bereiche hat seine individuellen Eigenschaften: LAN´s und WAN´s bieten keine verläßliche Datenübertragung an, daher ist es in solchen Netzwerken notwendig, dies durch Netzwerk-Software zu gewährleisten, z. B. durch ein TCP/IP Protokoll. Die Kosten für so einen Protokollturm sind allerdings enorm hoch. Dadurch zielt die Forschung in diesem Bereich hauptsächlich darauf ab, zuverlässige Protokolle mit kürzeren Datenwegen zu entwickeln und die Datenbewegung zwischen Host und Netzwerkkarte zu optimieren. Im SAN-Bereich sind die Anforderungen von vorneherein höher. Bald wird man Bandbreiten von 10 Gbps und Latenzzeiten im Bereich von 10 ns haben. Außerdem bieten diese Netzwerke schon zuverlässige Datenübertragung an, so daß man hier eher auf leichtgewichtige Netzwerkprotokolle

wie Fast Messages oder Active Messages setzen kann. Außerdem wird dadurch der Einsatz von fortgeschritteneren Techniken für das Erkennen ankommender Nachrichten und für das Durchsetzten von Schutzmechanismen möglich. Zu guter letzt muß man sich noch überlegen, inwiefern sich eine veränderte Computerarchitektur positiv auf die Leistung der Netzwerkanbindung auswirken kann und in welchem Maße man die Kommunikation in den Berechnungsprozeß selbst einbinden kann. Im Folgenden wird darauf eingegangen, wo grundsätzlich die Ursachen für die ungenügende Leistung der Netzwerkanbindung liegen, welche Techniken entwickelt wurden, diese zu verbessern, und welche Leistungssteigerung die einzelnen Techniken mit sich bringen. Anschließend wird ein neuer Standard für Netzwerkschnittstellen vorgestellt, der VIA, der viele der im Abschnitt davor angesprochenen Techniken verwendet. Danach folgt ein Ausblick darauf, was mit einer veränderten Computerarchitektur alles zu erreichen ist und es wird versucht, das Besprochene in Bezug zu den realen Anwendungsbereichen zu bringen.

2 Gründe für den Flaschenhals

Wie in der Einleitung angesprochen, wirken sich vor allem die aufgeblähten Protokolltürme negativ auf die Geschwindigkeit der Datenübertragung aus aber auch bei der Verwendung von Host-Routern, die gerne als flexiblere Alternative zu speziellen Routern verwendet werden, gibt es noch einiges zu verbessern. Dies ist darauf zurückzuführen, daß bei beiden Punkten die gleichen oder zumindest ähnliche Mechanismen eine Rolle spielen und somit die Gründe der mangelhaften Leistung bei diesen zu suchen sind.

Um nun die eigentlichen Schwächen besser erkennen zu können, wird im Folgenden kurz eine traditionelle Architektur des Netzwerk-Subsystems und ein Modell für den herkömmlichen Weiterleitungsvorgang auf Host-Routern vorgestellt. Das vorgestellte Architektur wird auch in dem in der Forschung gerne eingesetzte Myrinet Netzwerk verwendet. Dieses Netzwerk zeichnet sich dadurch aus, daß es eine programmierbare Netzwerkkarte hat, die es ermöglicht, die verschiedensten Protokolle auszutesten und miteinander zu vergleichen. Auf diesem Weg sind auch viele später besprochene Neuerungen entwickelt worden. Im Anhang findet sich eine genauere Beschreibung der Architektur des Myrinet Netzwerks.

2.1 Traditionelle Architektur des Netzwerk-Subsystems

Abbildung 1 beschreibt die Funktionsweise der traditionellen Architektur des Netzwerk-Subsystems. Dabei ist das Protokoll schon soweit optimiert, daß der Eingriff des Betriebssystems bei der Datenübertragung vermieden wird.

Zunächst startet ein Benutzerprozeß auf Host A ein Sende-Primitiv, um ein Datenpaket zu übertragen. Da der maximale Nutzdatenanteil hier auf 256 Bytes beschränkt ist, müssen die zu übertragenden Daten zuvor fragmentiert werden, um sie auf die einzelnen Pakete aufzuteilen. Das Senden umfaßt grundsätzlich zwei Aktionen: Zunächst kopiert die CPU des Hosts, wie in Schritt 1 der Abbildung gezeigt, die zu sendenden Daten aus dem User-Space in den Kernel-Space. Auf diesen Bereich kann die Netzwerkkarte über einen direkten Speicherzugriff (DMA) zugreifen. Im zweiten Schritt schreibt der Host-Prozessor eine Sendeanfrage auf einen Deskriptor im Speicher der Netzwerkkarte und setzt ein Flag, um die CPU der Netzwerkkarte davon in Kenntnis zu setzen. Dabei sitzen mehrere Deskriptoren auf einem Ringpuffen (Sendering) der Netzwerkkarte. Der aktuelle Deskriptor enthält nach diesem Vorgang die Adresse des Zielrechners der Datenübertragung, die Größe der Nutzdaten auf dem Paket und den Offset des Paketes im DMA-Bereich des Hosts. Die CPU der Karte überprüft nun regelmäßig den Flag darauf, ob eine neue Anfrage vorliegt (polling). Ist dies der Fall, so greift sie auf den ersten freien Deskriptor zu und berechnet die physikalische Adresse des Pakets im

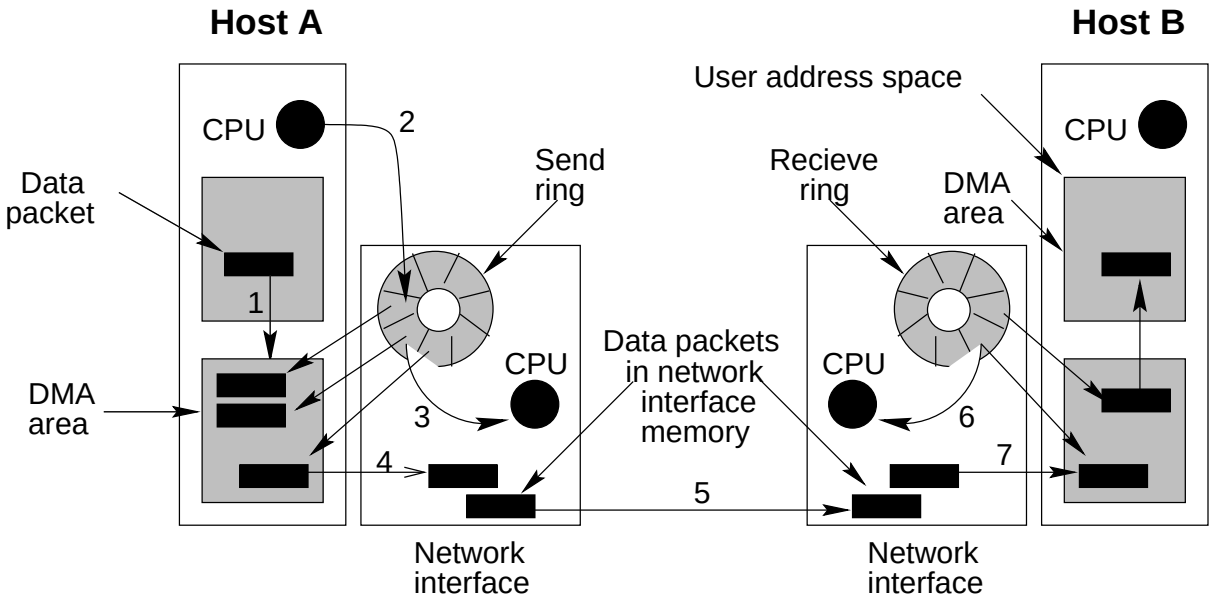

Abbildung 1: *Funktionsweise der traditionellen Architektur des Netzwerk-Subsystems. (1) Host A kopiert Daten in seinen DMA-Bereich und schreibt (2) einen Deskriptor auf dem Sendering. Die Netzwerkkarte liest (3) den Deskriptor und kopiert die Daten in den eigenen Speicher (4). Nach dem erfolgenden Transfer (5), liest die empfangende Karte ihren Empfangsring um einen freien Speicherplatz im DMA Bereich von Host B zu finden (6), und kopiert die Daten dorthin (7). Optional kann der Host noch die Daten in einen Speicherbereich der Anwendungsebene kopieren (8).*

DMA-Bereich sofort aus Startadresse und Paket-Offset (Schritt 3). Nun kopiert die Netzwerkkarte das Paket mit einem direkten Zugriff auf den Host-Speicher über den Ein-/Ausgabebus in den eigenen Speicher (Schritt4). Danach liest die CPU die Zieladresse aus dem Deskriptor dieses Vorgangs. Im 5. Schritt, der eigentlichen Übertragung, wird das Paket über das physikalische Netz in den Speicher der Zielnetzwerkkarte übertragen. Anschließend wird der benutzte Beschreiber zur Wiederverwendung für den Host freigeben. Zu beachten ist, daß bei der Übertragung mehrerer Pakete die Übertragung eines Pakets vom Host-Speicher zur Netzwerkkarte überlappt zur Übertragung eines anderen Paketes zur Zielkarte erfolgen kann. Auf der Empfängerseite besitzt die Netzwerkkarte auch einen Pufferspeicher mit Beschreibern, den Empfangsring. Diese Beschreiber zeigen nun aber auf freien Speicher im DMA-Bereich von Host B. So weiß die Netzwerkkarte, wo sie ankommende Pakete ablegen kann (Schritt 6). Gibt es momentan keinen freien Puffer im Host-Speicher, so muß das Paket verworfen werden. Ansonsten werden in Schritt 7 die Nutzdaten in den DMA-Bereich des Hosts geschrieben. Dabei wird wieder ein Flag gesetzt, das auf diese Aktion hinweisen soll. Der Host überprüft den Flag des nächsten Puffers wiederholt, bis ein neues Paket angekommen ist. Ist dies der Fall, so wird eine Funktion des Anwenders aufgerufen, die das Paket liest und gegebenenfalls im 8. Schritt dessen Inhalt in einen eigenen Speicherbereich kopiert.

2.2 Weiterleiten bei einem Host-Router

Ein Router sorgt grundsätzlich dafür, daß über ihn gesendete Pakete auf dem richtigen Weg weitergesendet werden. Dazu hat er zum einen die Aufgabe die Pakete zu puffern, oder auch zu verwerfen, um auf den Datenfluß reagieren zu können und zum anderen muß die eigentliche Weiterleitung (forwarding) durchgeführt werden. Ein Router besteht dazu aus einem Host und mindestens zwei Netzwerkkarten. Weiterhin hat der Host einen I/O Controller, der einen direkten Speicherzugriff auf die Netzwerkkarten über einen Ein/Ausgabe-Kanal ermöglicht.

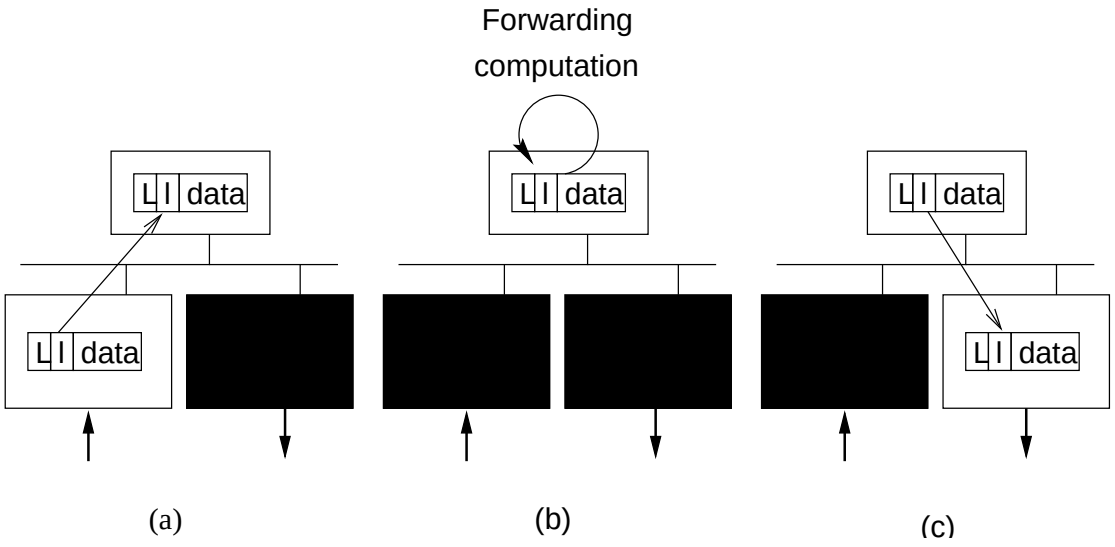

Abbildung 2: *Drei Schritte bei der Weiterleitung von Paketen. a)Das Paket wird komplett über DMA in den Host-Speicher geholt; b) Der Host berechnet die für die Weiterleitung erforderlichen Daten und modifiziert die Einträge für IP und die Verbindung im Paketkopf; und c) das Paket wird über DMA auf die weiterleitende Netzwerkkarte kopiert*

Jede der Netzwerkkarten hat einen DMA-Controller, ein wenig Speicher und natürlich eine Anbindung an das Netzwerk. Zu erwähnen bleibt noch, daß durch die Möglichkeit der direkten Speicherzugriffe auch hier auf eine Beaufsichtigung durch das Betriebssystem verzichtet werden kann.

In Abbildung 2 wird der Weiterleitungsprozeß beschrieben. Zunächst kommt ein Paket bei einer Netzwerkkarte an. Diese signalisiert dies dem Host und das Paket wird über einen direkten Speicherzugriff komplett in den Speicher des Hosts kopiert (a). Nun erfolgt die Weiterverarbeitung des Pakets, der eigentliche Weiterleitungsschritt. Unter anderem werden die Einträge im Kopf des Paketes für die IP-Schicht untersucht und so die Netzwerkkarte ermittelt, die auf einen entsprechenden Weg führt. Dann wird der Kopf der MAC-Schicht entsprechend modifiziert (b). Als letzter Schritt wird das Paket nun über einen weiteren DMA auf die richtige Netzwerkkarte kopiert und dort letztendlich auf seinen weiteren Weg geschickt (c). Sollte der Host selber das Ziel des Pakets sein, so wird das Paket nach Schritt b natürlich aus dem Speicherbereich des Betriebssystemkerns in den Speicherbereich der zugehörigen Anwendung kopiert und weiterverarbeitet.

2.3 Entwurfsaspekte für den Entwurf neuer Netzwerkkarten und Protokolle

Die Geschwindigkeit der Kommunikation über ein Netzwerk hängt von mehreren Aspekten ab. Die wichtigsten dabei sind:

- Betriebssystem
- Geschwindigkeit der Datentransfermechanismen
- Effizienz der Adreßumrechnung
- Schutz
- Erkennung von ankommenden Paketen

- Zuverlässigkeit
- Multicast

Dabei werden hier die Punkte 2-7 deswegen notwendig, weil diese Aspekte normalerweise durch das Betriebssystem abgedeckt werden, man auf dessen Eingreifen aber aus Geschwindigkeitsgründen verzichten will. Deshalb wurde auch schon in den obigen Beispielen für das Protokoll und das Routing von Mechanismen ohne den Einsatz des Betriebssystems gesprochen. Doch auch hier kann man noch einige Schwachstellen entdecken.

Beim Forwarding-Beispiel läßt sich kritisieren, daß einige teure Datenkopien notwendig sind, um das Paket weiterzuleiten. Zum einen von der Eingangs-Netzwerkkarte in den Host-Speicher und zum anderen, von dort auf die Ausgangs-Netzwerkkarte. Außerdem wird die eigentliche Weiterleitung im Betriebssystemskern durchgeführt, es gibt Ansätze, die dies direkt in die Anwendung oder gar auf die Netzwerkkarte verlegen wollen, um das Betriebssystem ganz auszuschalten.

Der Kritikpunkt mit den vielen Datenkopien und somit auch Datentransfers gilt genauso auch für die traditionelle Architektur des Netzwerk-Subsystems. Alle Datentransfers gehen über die DMA-Bereiche des Hosts, dadurch werden zusätzliche Datenkopien von Speicherbereichen von entfernt sendenden oder empfangenden Anwendungen notwendig. Außerdem kann es vorkommen, daß Seiten des DMA-Bereichs, der ja auch ein Teil des Hauptspeichers ist, durch Auslagerungsstrategien des Betriebssystems ausgelagert werden. Außerdem gibt es hier noch das Problem, daß die Netzwerkkarte sämtliche physikalischen Adressen von Seiten im DMA Bereich kennen muß, auf die sie schreiben oder von denen sie lesen will und das Betriebssystem diese Informationen nicht an Anwender weitergibt. Ein weiterer Schwachpunkt ist, daß dieses Protokoll keinen Schutz mehr gewährleistet. Alle Pakete werden durch den Speicher der Netzwerkkarte geschleust, haben nun mehrere Anwender Zugriff auf die Karte, so können fremde Pakete gelesen und geändert werden und es ist sogar möglich, durch eine Änderung des Kontrollprogramms der Netzwerkkarte Zugriff auf den DMA-Bereich eines anderen Hosts zu bekommen. Zu Bemängeln ist auch das Einsetzen von polling auf der Empfängerseite der Datenübertragung, bei der wiederholt der Zustand eines Flags abgetestet wird. Es stellt sich hier die Frage nach der geeigneten Pollfrequenz. Wird der Test des Flags zu häufig vorgenommen, so hat man einen großen Zusatzaufwand. Verpaßt man dagegen das Ankommen eines Paketes wegen zu spätem Prüfen, so kann es sein, daß nicht mehr genug Zeit bleibt, das Paket abzunehmen, bevor es verworfen wird. Als Alternative zum Polling käme ein Interrupt in Frage, der das Betriebssystem beim Eintreffen eines Paketes direkt benachrichtigt. Dies ist aber sehr teuer. Unabhängig von der Zuverlässigkeit des Netzwerkes, ist das Protokoll selbst nicht zuverlässig. Wenn z. B. schneller Pakete gesendet werden, als sie auf der anderen Seite abgenommen werden können, so kann der Puffer auf der Empfangsseite überlaufen und Pakete müssen verworfen werden. Als letzter Punkt bleibt noch anzumerken, daß dieses Protokoll lediglich das Übersenden von Nachrichten Punkt-zu-Punkt erlaubt und somit Multicast nicht direkt unterstützt. Zwar kann man Multicast auch auf der Grundlage von mehreren Punkt-zu-Punkt Verbindungen realisieren aber der Aufwand steht dabei in keinem Verhältnis zum Nutzen. Maßzahlen für die Leistung eines Netzwerkes sind hierbei der Durchsatz, die Latenzzeit und die Dauer eines Rundlaufs (round trip).

2.4 Techniken zur Optimierung

Wie im vorangegangenen Abschnitt deutlich wurde, gibt es noch einiges, das bei den traditionellen Datenübertragungssystemen zu verbessern ist. Es werden nun einige Techniken vorgestellt, die dies leisten sollen.

2.4.1 Verbesserung des Weiterleitens

Wie schon angedeutet, wird beim herkömmlichen Weiterleiten der Datenpfad zwischen CPU des Hosts und dem Speicher der Netzwerkkarte durch möglicherweise unnötige Übertragungen belastet. Um nun die Anzahl der Übertragungsvorgänge zwischen Hostspeicher und dem der Karte zu minimieren gibt es nun 2 wesentliche Ansätze:

- Hardware streaming Bei diesem Ansatz werden die Daten zwischen Quelle und Senke ohne Hilfe der CPU übermittelt. Dadurch wird ein unmittelbares Verarbeiten des Pakets zwischen dem eigentlichen Datentransfer, wie es sich im Weiterleiten und dem Einstellen der Pakete in eine Warteschlange findet, vermieden. Somit werden auch unnötige Kopien in den Speicherbereich des Betriebssystemkerns vermieden. Es findet dabei also eine Kommunikation zwischen zwei Peripheriegeräten statt, wie sie auch bei video-to-disc Kommunikation zu finden ist.
- Kernel-level streaming Hier werden die Daten zwischen Quelle und Senke über den Systembus übertragen aber ohne einen Anwendungsprozeß in den Datenpfad einzubeziehen. Allerdings werden die Daten wieder durch den Speicher des Hosts geschleust. Diese Technik verläßt sich dabei auf die V Streams Schnittstelle des Systems.

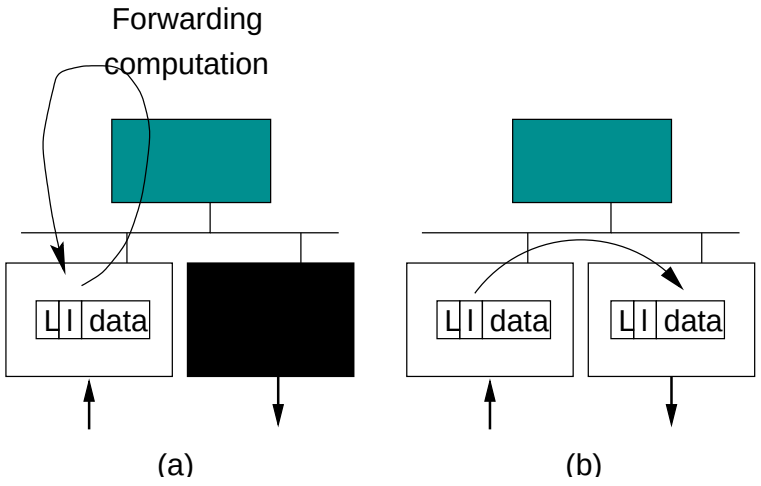

Abbildung 3: *Beim in-place forwarding bleibt das Paket auf der hereinführenden Netzwerkkarte. Dort werden in a) zunächst die notwendigen Berechnungen durchgeführt und der IP-Kopf und der Kopf für die Verbindung verändert. In b) wird dann der direkte DMA zwischen den beteiligten Karten durchgeführt.*

Ein anderer Ansatz vermeidet den Einbezug des Betriebssystems dadurch, daß Anwendungen der direkte Zugang zur Netzwerkkarte gewährt wird. Damit erreicht man niedrige Latenzzeiten und eine große Bandbreite. Ermöglicht wird dies, indem die Karte virtuell gemacht wird. Dies erfordert verschiedene Funktionen der Netzwerkkarte, wie eigene Puffer, DMA und eigene Koprozessoren. Der Koprozessor der Karte erlaubt es, den Paketkopf an Ort und Stelle zu untersuchen und das Paket direkt mittels DMA in den Speicherbereich der Anwendungsebene zu übermitteln. Dieser Ansatz wird z. B. bei U-Net verwendet, die darauf aufbauend daran arbeiten, eine Untermenge der Java Virtual Machine für Myrinet zu implementieren, was es möglich machen soll, das Verarbeiten der Pakete direkt auf der Netzwerkkarte auf eine an die jeweilige Anwendung angepaßte Art und Weise durchzuführen.

Schließlich gibt es noch eine Anzahl von Techniken, die das traditionelle Routing-Modell zu optimieren versuchen. Dabei werden die Daten direkt zwischen den beteiligten Netzwerkkarten über einen direkten Speicherzugriff übertragen. (Peer-DMA Forwarding).Das Weiterleiten

geschieht bei dieser Methode entweder auf der Karte selbst oder auf einem eigenen Prozessor. Der genaue Ablauf des peer-DMA forwarding ist wie folgt: Das erklärte Ziel dieser Technik ist es, den größten Teil des ankommenden Pakets solange auf der hereinführenden Netzwerkkarte zu belassen, bis sein Ziel bestimmt ist. Danach wird das Paket dann über einen direkten Zugriff auf den Speicher der Zielnetzwerkkarte übertragen. Dabei gibt es wieder zwei verschiedene Strategien, das in-place forwarding und das header-copy forwarding.

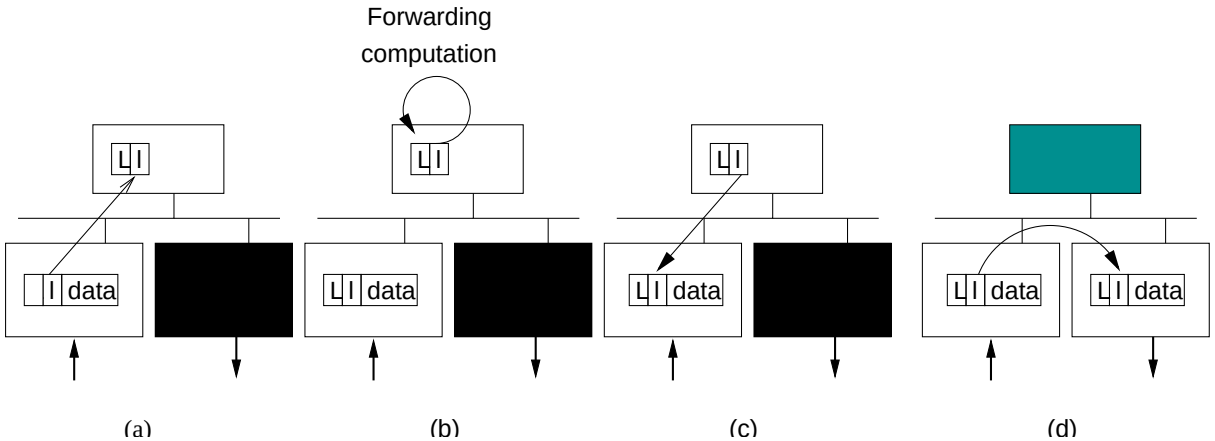

Abbildung 4: *Beim header-copy forwarding wird in a) zunächst der Verbindungskopf und der IP-Kopf über ein DMA in den Speicher des Hosts übertragen. In b) werden dann die Änderungen an den Köpfen vorgenommen und in c) der gesammte Kopf über den alten kopiert. In d) wird dann anschließend der Datentransfer über einen DMA zwischen den Netzwerkkarten vorgenommen*

In Abbildung 3 wird das Weiterleiten an Ort und Stelle beschrieben. Das gesamte Paket bleibt auf der Karte und der Host liest die benötigten Felder des Paketkopfes direkt dort über den Bus. Dabei wird jedes Feld separat über einen gemeinsam genutzten Speicherbereich gelesen. Danach wird direkt zwischen den beteiligten Netzwerkkarten übertragen. Diese Methode nimmt an dem eigentlichen Algorithmus nur wenige Veränderungen vor. Zeiger in der Routine zeigen lediglich auf gemeinsam genutzten Speicher einer Netzwerkkarte anstatt auf Hauptspeicher. Ein Nachteil dabei ist, daß die Zugriffe auf die Paketkopffelder einen Zugriff auf den Peripheriebus bedeuten, der mit einer geringeren Taktrate läuft als die, mit der die CPU auf den Hauptspeicher zugreifen könnte. Außerdem kann dieser Zugriff, da er auf einen geteilten Speicherbereich erfolgt, nicht im Cache gespeichert werden. Somit steigt der Busverkehr an.

In Abbildung 4 ist die zweite Strategie beschrieben, das header-copy forwarding. Wie der Name schon ahnen läßt, werden hier der IP- und Verbindungskopf in den Hauptspeicher des Hosts kopiert und dort über lokale Speicherzugriffe manipuliert. Danach wird der neue Paketkopf über den alten im Speicher der Netzwerkkarte kopiert. Auch hier wird anschließend das gesamte Paket per DMA zur Zielkarte übertragen. Ähnlich wie bei der ersten Strategie ist auch beim Weiterleiten durch Kopieren des Paketkopfes der Aufwand die Treiber umzuprogrammieren recht gering. Wegen des zusätzlichen Zurückkopierens des Paketkopfes belastet diese Methode den Peripheriebus unwesentlich höher als die herkömmliche Methode.

Grundsätzlich hat das direkte Weiterleiten der Pakete zwischen den beteiligten Netzwerkkarten folgende Vorteile:

- Die Latenzzeit, die durch den Router verursacht wird ist geringer
- Der Ein/Ausgabekanal wird weniger belastet, da der Bus nur einmal pro Paket benutzt wird statt zweimal.
- Es wird kein Speicher im Host benutzt und somit Ressourcen freigesetzt

Ein Nachteil stellt allerdings noch dar, daß für die Pakete der Internetprotokolle ARP und ICMP spezielle Mechanismen bereitgestellt werden müssen. Das Problem bei solchen Paketen ist, daß sie beim Empfang eine direkte Beantwortung auslösen, die dadurch geschieht, daß die Pakete überschrieben und direkt zurückgesendet werden. Das würde bedeuten, daß die hereinführende Netzwerkkarte einen DMA auf sich selbst zu starten versucht, was als Operation meist nicht unterstützt wird. Fängt man solche Pakete ab kann man den DMA durch eine Zeigermanipulation verhindern. Das Information Sciences Institute hat beide Ausprä-

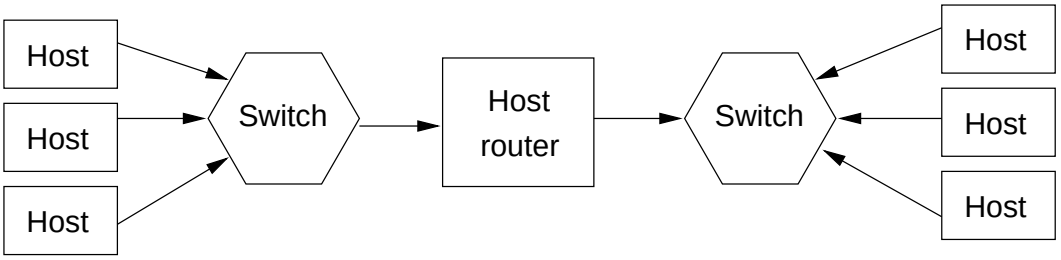

Abbildung 5: *Topologie des Testsystems. 3 Quell-Hosts verbunden über ein Myrinet-Switch, der dann über einen Host-Router mit einem anderen Switch und 3 Senken verbunden ist*

gungen des peer-DMA forwarding untersucht und mit der traditionellen Methode verglichen [AACMu98]. Die Versuche wurden dabei auf einem System vorgenommen, dessen Topologie in Abbildung 5 zu sehen ist. Das System benutzt ein Myrinet LAN (siehe auch Anhang) und ATM mit 155 Mpbs zur Verbindung von 53 PCs mit 200 MHz Pentium Pro als Prozessor und 33-MHz, 32-Bit breitem PCI-Bus. Beim Test wurde ein Softwarewerkzeug verwendet mit Namen Netperf, das Pakete des User Datagram Protocol (UPD) erzeugt, um somit die Leistung des Netzes zu messen. Der Test umfaßte dabei Paketgrößen zwischen 128 Byte und 8KByte und wurde mit unterschiedlicher Anzahl an Quellen und Senken durchgeführt. Eine Quelle kann dabei maximal 300-Mbps UPD Pakete ohne Kontrollsummen und 275-Mbps Pakete mit Kontrollsummen senden. Die einzelnen Testergebnisse sind in der Abbildung 6 zu sehen.

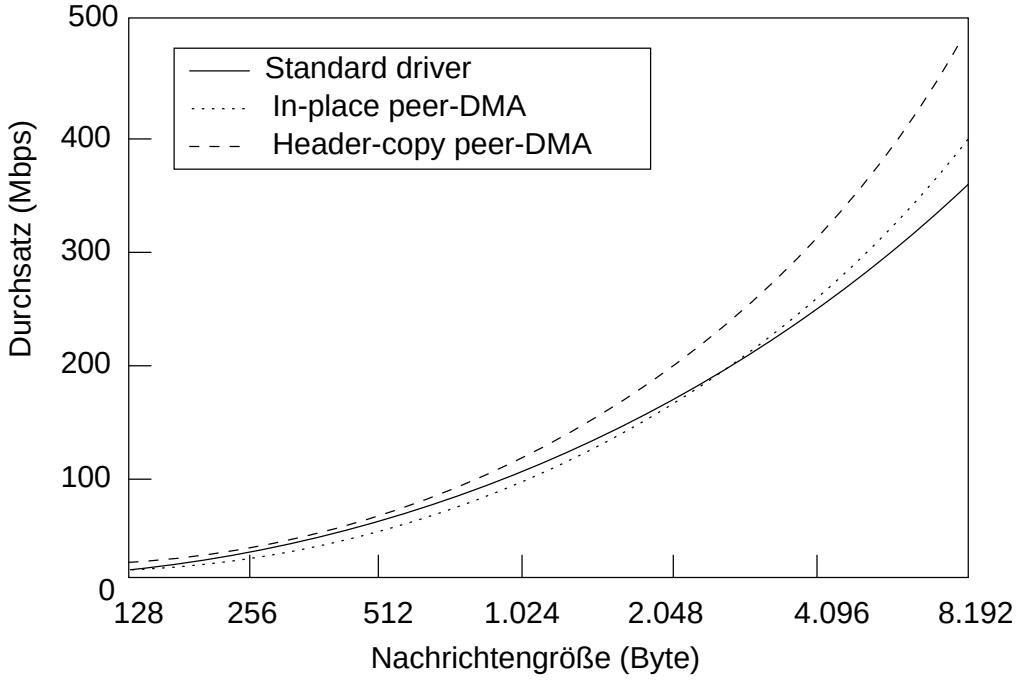

Abbildung 6: *UPD Bandbreite für die drei verschiedenen Treibertypen. Man kann sehen, daß unter 1Kbyte die Leistung aller Treiber gleich sind, sich bei größeren Nachrichten die peer-DMA Technik aber durchsetzen kann.*

Beim Test mit einer Quelle und einer Senke stellte sich heraus, daß ohne einen vermittelten Router oder mit einem solchen weder die Paketrate noch die Bandbreite unterschiedlich waren. Die CPU des Routers wurde dabei beim Standardprotokoll weniger belastet (37 Prozent) als beim in-place Treiber (65 Prozent) oder dem header-copy Treiber (56 Prozent). Dies kommt daher, daß aufgrund dessen, daß das System ansonsten keine Last zu tragen hatte, sorgten die peer-DMA Treiber nur für zusätzlichen Aufwand durch die Paketkopfkopie oder das Lesen über den PCI-Bus gesorgt haben.

Im Falle von zwei Quellen und zwei Senken dagegen schaffte der Standardtreiber nur einen Durchsatz von 335 Mbps während der in-place Treiber 419 Mbps und der Treiber für die zweite Strategie 441 Mbps Durchsatz erreichen konnte. Das hat den Grund, daß nun mehr Last anfiel, im Ganzen 600 Mbps an angebotenen Paketen und sich so die Treiberstrategien bemerkbar machen konnten. Dabei schnitt die zweite Strategie vermutlich deswegen besser ab, weil das Kopieren des gesamten Paketkopfes wohl doch schneller geht, als die einzelnen Felder über den PCI-Bus zu lesen. Die CPU-Auslastung betrug wie zu erwarten bei fast allen Paketgrößen und Strategien an die 100 Prozent. Nur bei der Strategie mit den Paketkopfkopien sank die Belastung bei einer Paketgröße von 8-KByte auf 65 Prozent.

Bei drei Quellen und ebensovielen Senken konnte der Standardtreiber seinen Durchsatz nicht mehr steigern, der inplace-Treiber dagegen kam auf 472 Mbps und der andere gar auf 482 Mbps. Der Test zeigt also, daß die Weiterleitung mittels einem direkten Speicherzugriff zwischen den beteiligten Netzwerkkarten und der Strategie des Kopierens der Paketköpfe den Durchsatz deutlich steigern kann. Es hat sich allerdings ergeben, daß der Durchsatz, sollte der Host selber das Ziel sein nicht höher ist als der des Standardtreibers. Daher sollte eine solche Technik nur dort angewandt werden, wo die Datentransfers hauptsächlich durch den Router und nicht zum Router hin führen.

2.4.2 Verbesserung der Architektur des Netzwerksubsystems

In diesem Abschnitt werden nun zu den oben angesprochenen Aspekten, die sich auf die Leistung eines Netzwerks auswirken, verbesserte Techniken vorgestellt und untersucht, welche Leistungssteigerungen so erzielt werden können.

- Datentransfer: Beim der oben vorgestellten traditionellen Architektur waren 5 Datenübertragungen für ein Paket notwendig. Optimierte Protokolle kommen dagegen mit 3 Übertragungen pro Paket aus: Eine Übertragung vom Host zur Netzwerkkarte, von dort eine weitere zur Zielkarte und schließlich noch die Übermittlung zum Zielhost. Für den effizienten Ablauf dieser Übermittlungen gilt es eine Wahl zu treffen zwischen verschiedenen Techniken. Beim Transport des Paketes zwischen Host und Netzwerkkarte und genauso auf der anderen Seite von der Zielkarte zum Zielhost hat man die Wahl, die Übertragung per direktem Speicherzugriff oder über eine programmierte Ein/Ausgabe-Routine vorzunehmen. Bei der programmierten Routine liest der Prozessor die Daten vom Hauptspeicher des Hosts und schreibt pro Zugriff etwa 1-2 Worte auf den Speicher der Netzwerkkarte. Somit kommt es hier zu sehr vielen Zugriffen und damit auch zu viel Busverkehr. Der DMA dagegen benutzt eine spezielle DMA-Maschine, die es ermöglicht ein gesamtes Paket auf einmal zu transportieren und daneben noch den Vorteil bietet, daß dieser Vorgang parallel zu sonstiger Prozessortätigkeit ablaufen kann. Der Verdacht liegt also nahe, daß ein so durchgeführter Übertragungsvorgang immer der ersten Technik überlegen ist. Das stimmt aber in diesem Maße nicht. Beim Pentium Pro gibt es z. B. eine Methode, die mehrere Schreibanforderungen über diese programmierbaren Routinen zu einer Bustransaktion zusammenfaßt. (write-combining-Buffer) Dadurch wird die Leistung dieser Technik um einiges gesteigert. Hinzu kommt, daß es

unter Umständen auch zu Problemen beim entfernten Speicherzugriff kommen kann. Da der Übertragungsvorgang hier asynchron abläuft, völlig unabhängig vom Betriebssystem und sowohl vom Host als auch von der Karte angestoßen werden kann, kann es passieren, daß einzelne Seiten dieses DMA-Bereichs durch Betriebssystemstrategien ausgelagert werden können, was die Daten natürlich verfälschen würde. Um dies zu vermeiden, kann man nun natürlich alle benötigten Seiten einzeln im Speicher festschreiben (pinnen). Dies ist allerdings mit einem teuren Systemaufruf verbunden. Außerdem hängt diese Möglichkeit auch stark davon ab, ob genügend Speicher zur Verfügung steht und ob dem Festschreiben nicht andere Strategien des Betriebssystems entgegenwirken. Um dieses Problem zu vermeiden gibt es in vielen Systemen einen reservierten, im ganzen festgeschriebenen DMA-Bereich, wodurch allerdings eine extra Kopie des Paketes innerhalb des Speichers nötig wird. Eine beim Schimps-System verwendete Technik baut auf eine spezielle Hardware auf, die einen DMA ohne Festschreiben zuläßt, dies aber nur für eine Übertragung, die durch den Host angestoßen wird. Somit ist diese Technik auf der Zielseite nicht einsetzbar. Man sieht also, daß dadurch die programmierte I/O wieder an Attraktivität gewinnt. Bei dieser Technik ist es nicht notwendig Seiten festzuschreiben, da ausgelagerte Seiten über einen Seitenfehler automatisch wieder eingelagert werden. In Abbildung 7 kann man somit sehen, daß diese Technik, zusammen mit dem Puffern mehrerer Schreibanforderungen bei Paketpuffern bis 1.024 Byte schneller ist als der entfernte Speicherzugriff. Das liegt an den hohen Kosten, die entstehen, den Speicherzugriff startbereit zu machen. Da nun beide Techniken ihre Vor- und Nachteile haben wird bei vielen Systemen auch ein Mix beider Techniken verwendet. So benutzen die Systeme FastMessages (FM) und Link-Level Flow Control (LFC) die programmierte I/O für den Verkehr zwischen Host und Netzwerkkarte und DMA auf der anderen Seite der Übertragung und Active Messages II (AMII) verwendet diese Technik bei kleinen Nachrichten, um sich so das Einrichten des entfernten Speichezugriffs zu sparen. Allgemein ist die Entscheidung welche Technik man nun einsetzt davon abhängig, welche CPU, welche DMA-Maschine einem zur Verfügung stehen und wie die verwendete Paketgröße ist. Bei dem eigentlichen Datentransfer zwischen den zwei beteiligten Netzwerkkarten könnte man auch wieder zwischen den gerade beschriebenen Techniken wählen. Da aber die Prozessoren der Netzwerkkarten nicht sehr schnell sind, scheidet die programmierbare Ein-/Ausgabe hier bisher aus. Nun gibt es hier aber noch das Problem, daß man entweder eine Flußkontrolle für den Datenfluß zwischen den zwei Karten braucht, oder die Empfängerseite mit dem Abnehmen der übertragenen Daten wirklich schnell genug sein muß. Ansonsten kommt es zu einem Datenstau und es müssen entweder Pakete verworfen werden oder der Sender wird blockiert um ein endgültiges Blockieren der Übertragung zu vermeiden.

- Adreßübersetzung: Für den Zugriff auf gemeinsam genutzten Speicher ist es auch erforderlich, daß die Netzwerkkarte die physikalische Adresse jeder Seite kennt von der sie liest oder auf die sie zu schreiben versucht. Nun wird die Netzwerkkarte aber direkt von der Benutzerebene und nicht mehr über das Betriebssystem verwaltet. Da aber das Betriebssystem die obige Information nicht an Benutzer weitergibt, muß man sich eine andere Lösung einfallen lassen, wie die Netzwerkkarte die notwendigen Informationen erhält.. Hat man die oben angesprochenen DMA-Bereiche eingeführt, so muß beim Reservieren dieses Bereiches der Netzwerkkarte nur einmal die physikalische Startadresse und die Größe des Speichers übergeben werden. Die Position der einzelnen Pakete kann dann später über den Paket-Offset errechnet werden. Eine andere Möglichkeit ist es, daß die benötigten Seiten über eine spezielle Anwendung oder eine Bibliothek dynamisch festgeschrieben und freigegeben werden. (Dies entspricht in Abbildung 7 der Kurve für den cache coherent DMA). Da die Netzwerkkarten nicht alle Adressen im Cache-Speicher halten kann muß es z. B. über ein einfaches Modul im Kern des Be-

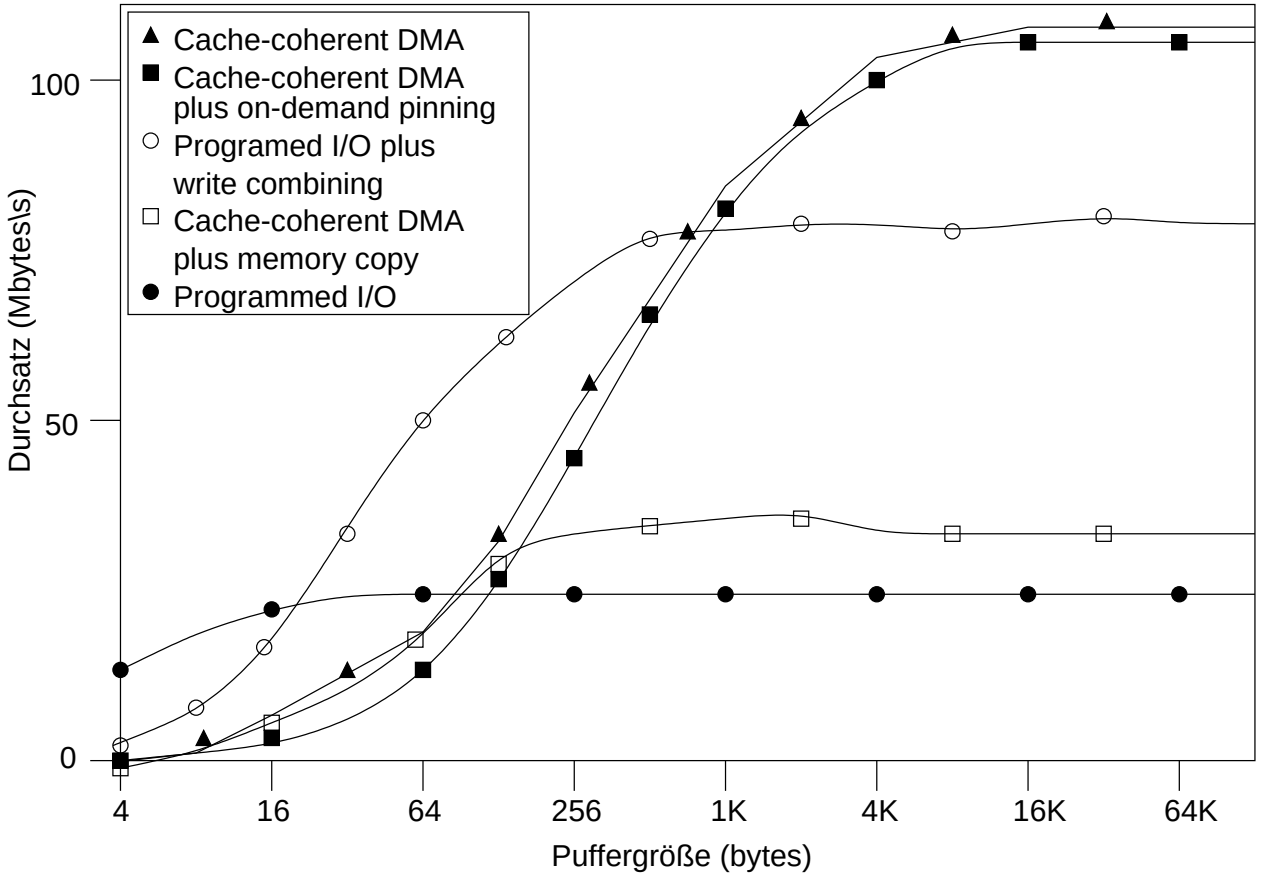

Abbildung 7: *Der Durchsatz bei einem Datentransfer zwischen Host und Netzwerkkarte bei verschiedenen Techniken. Bei Puffern bis 1.024 Byte ist die programmierte Ein/Ausgabe schneller als DMA*

triebssystems, die Möglichkeit geben, die Adresse der benötigten Seite zu erfahren. Der Nachteil dieser Methode ist, daß die Netzwerkkarte keine Möglichkeit hat, die Informationen nachzuprüfen und sich somit auf die Zuverlässigkeit dieser Methode verlassen muß.

- Schutz: Wie oben erwähnt, ist es sehr kritisch, daß mehrere Anwender gleichzeitig Zugriff auf den Speicher der Netzwerkkarte haben und somit die Gefahr des Mißbrauchs besteht. Um dies nun zu vermeiden, kann man natürlich einfach nur jeweils einem Benutzer den Zugriff erlauben, was allerdings nicht sehr effizient wäre. Eine andere Möglichkeit ist es, jedem Anwender nur den Zugriff auf einen Teil des Speichers zu erlauben, d. h. ein virtuelles Speichersystem zu verwenden. Dazu kopiert das Betriebssystem jedem Anwender die Adreßumrechnung auf seinen Teil des Kartenspeichers in seinen Bereich im Hostspeicher, wenn er auf die Netzwerkkarte erstmals zugreift. Zur Datenübertragung schreibt nun jeder Benutzer auf seinen zugewiesenen Speicherbereich. Die Netzwerkkarte muß nun also den Bereich jedes Anwenders wiederholt abfragen um neue Übertragungsanfragen zu bemerken. Dabei wird auch gleich geprüft, ob die Anfragen sich auch wirklich nur auf den richtigen Bereich beziehen und nur dann ausgeführt. Da nun auch hier wieder der Speicher nur begrenzt groß ist, wird z. B. bei AM-II eine Art von Seitenverwaltung eingeführt. Dabei werden alle zur Zeit aktiven Kommunikationsendpunkte im Speicher der Netzwerkkarte gehalten und alle anderen in den Hostspeicher ausgelagert. Gibt es nun eine Empfangsanfrage oder einen Sendewunsch eines inaktiven Endpunktes, so kooperieren die Netzwerkkarte und das Betriebssystem um diesen Endpunkt zu aktivieren. Dabei wird wohl eine anderer Endpunkt ausgelagert werden müssen. Die selben

Überlegungen, den Schutz betreffend gelten natürlich auch für den DMA-Bereich im Hostspeicher. Auch hier wird das Problem durch die Einführung eigener Bereiche für die Anwender gelöst.

- Übertragungskontrolle Ein wichtiger Punkt ist, wie der Zielhost darüber informiert wird, daß ein neues Paket angekommen ist. Da ein Interrupt sehr viel Zeit kostet (10 μs), kommt eigentlich nur das Abtesten eines Flags in Frage (polling). Dabei gibt es grundsätzlich die Möglichkeit, ein Flag im Speicher der Netzwerkkarte zu setzen, dabei müßte aber jeder Anfragezugriff des Hosts über den Ein/Ausgabebus erfolgen, was sehr langsam ist (500 μs) und außerdem auch anderen Verkehr über diesen Bus, unter anderem auch den Transfer der Pakete zwischen Karte und Host behindern würde. Da ist es schon besser, daß die Netzwerkkartensteuerung ein Flag direkt im Speicher des Hosts setzt. Dadurch wird das Abtesten zu einem normalen Speicherzugriff und benötigt bei einem negativen Ergebnis nur 5 μs und bei einem erfolgreichen Testen etwa 125 μs, da es durch das Beschreiben durch die Karte zu einem Seitenfehler kommt und der Host aus dem Speicher lesen muß. Auch hier verwenden wieder einige Systeme eine Verknüpfung mehrerer Techniken. Oft gibt es auf der Empfängerseite grundätzlich die Möglichkeit, beim Eintreffen eines Paketes einen Interrupt auszulösen. Diese Funktionalität kann nun ein- oder ausgeschaltet werden. Die Verwendung des Interrupts kann zum einen vom Empfänger veranlaßt werden oder der Interrupt wird dann verwendet, wenn der Sender im Paketkopf einen entsprechenden Flag gesetzt hat. Eine weitere Möglichkeit wurde von LFC implementiert. Durch einen sogenannten Polling-Watchdog wird nach setzen des Flags durch die Netzwerkkarte ein Timer aktiviert, der dafür sorgt, daß wenn der Host das Flag nicht in einem gewissen Zeitrahmen abprüft, automatisch ein Interrupt angestoßen wird.

  Man sieht also, daß sich die Entwickler in vielen Prototypen einige Verbesserungen einfallen lassen haben. Wie diese Verbesserungen nun in der Praxis verwendet werden können, wird im folgenden Abschnitt geklärt.

2.5 Die Entwicklung von VIA

Vor kurzem haben die drei großen Gesellschaften Microsoft, Intel und Compaq einen neuen Standard für den SAN-Bereich auf den Mark gebracht. Es handelt sich hierbei um VIA, was für Virtual Interface Architecture steht [vEVo98]. Die Architektur dieses Systems kann man in Abbildung 11 sehen. Wie der Name schon vermuten läßt, handelt es sich hierbei um einen Standard für Netzwerkkarten, die direkt über die Anwendungsebene kontrolliert werden können und somit das Betriebssystem von der Datenübertragung ausschließen. Wie dies aussieht, kann man in Abbildung 8 nachvollziehen. VIA verwendet im Großen und Ganzen die Techniken, die im vorhergehenden Abschnitt angesprochen worden sind.

Hier folgt nun eine kurze Auflistung der einzelnen Meilensteine in der Entwicklung dieses Standards. Die Wurzeln von Netzwerkkarten der Anwendungsebene liegen in den traditionellen Modellen der Nachrichtenweiterleitung bei mehreren Computern. In diesen Modellen wurde durch den Sender die Position der zu sendenden Daten im Speicher und der Zielrechner angegeben. Die Daten wurden dann in einen Zielspeicherbereich übertragen. Durch diese Sende- und Empfangsoperationen war es notwendig, die einzelnen Nachrichten zu puffern oder ein kompliziertes Handshake-Verfahren zwischen Sender und Empfänger für jede Nachricht auszuführen. Das führte in jedem Fall zu einigem Verwaltungsaufwand. Auf der Grundlage dieses Problems wurde Active Message entwickelt. Die Grundidee von diesem System war es, in jeder Nachricht die Adresse eines speziell dafür spezialisierten Handlers anzugeben, der dann durch die Netzwerkkarte aktiviert wird. Damit wurden die Nachrichten auf der Empfängerseite wesentlich schneller weiterverarbeitet und man braucht keine Pufferung der Nachrichten

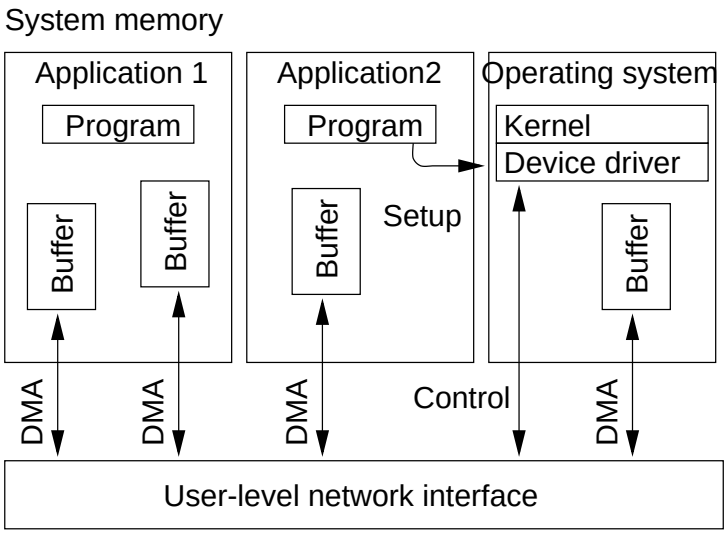

Abbildung 8: *Beispiel wie zwei Anwendungen auf das Netzwerk über eine Netzwerkkarte des Anwendungsbereichs zugreifen. Die Netzwerkkartenhardware wird über einen gewöhnlichen Gerätetreiber im Betriebssystem betrieben und sorgt für das Einrichten der direkten Speicherzugriffe auf die Karte für die Anwendungen (setup). So können die Anwendungen dann Datenübertragungen zur Karte direkt über DMA-Zugriffe zwischen Kartenspeicher und ihren eigenen Pufferbereichen vornehmen.*

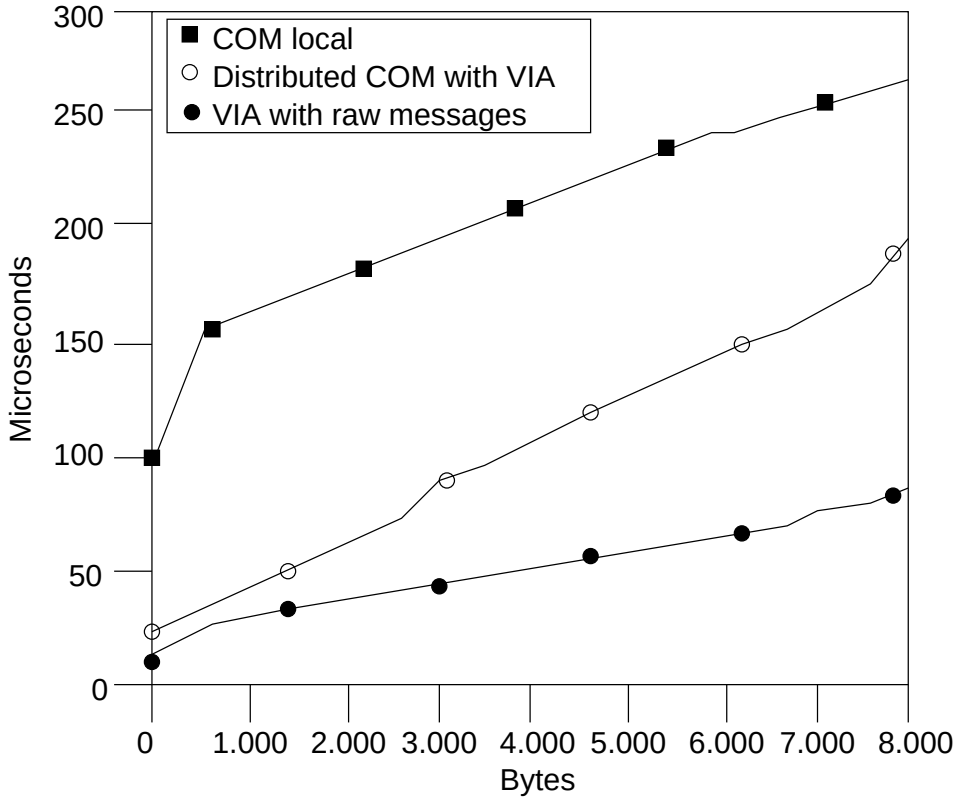

Abbildung 9: *Zeit für einen round-trip für einen lokalen entfernten Prozeduraufruf unter NT verglichen mit einem entfernten Prozeduraufruf unter der Verwendung von DCOM und VIA. Da der Betriebssystemkern umgangen wird ist der Aufruf im zweiten Fall sogar schneller als der lokale Gegenpart.*

vornehmen. Durch die veränderte Prozessorarchitektur mit tiefer werdenden Pipelines wurde es immer schwieriger, die Handler direkt nach dem Ankommen eines Paketes zu aktivieren. Somit wurden beim nachfolgenden Fast Message die Benachrichtigung des Handlers dahin verändert, die Nachrichten zu puffern und die Weiterverarbeitung durch explizites Abfragen (polling) anzustoßen. Somit kann das Starten der Handler verzögert werden, ohne daß das Netzwerk ausgebremst wird. Nach Fast Message und Active Message gab es mit U-Net

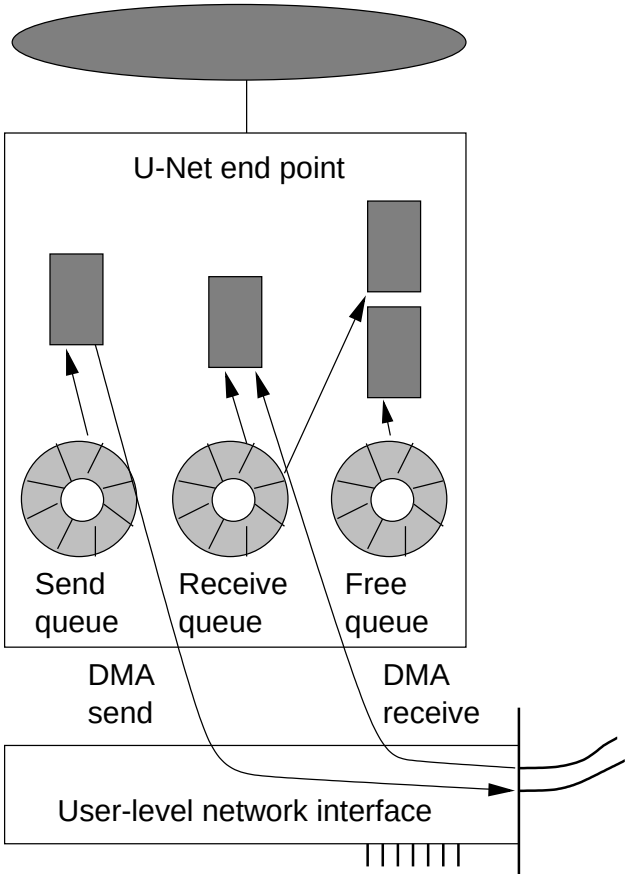

Abbildung 10: *Architektur von U-Net. Jede Anwendung greift über Endpunkte auf das Netzwerk zu. Dieser besteht aus einem Sende-, Empfangs- und Freipuffer, die auf durch die Anwendung angeforderten Speicher zeigen. Somit kann die Netzwerkkarte die Entsprechenden Speicherbereiche für entfernten Speicherzugriff beim Senden und Empfangen ermitteln.*

erstmals eine sich von den Architekturen der Vorgänger in wesentlichen Punkten unterscheidende Architektur. Diese Architektur ist näher an die Funktionsweise angelehnt, wie sie typischerweise bei LAN-Netzwerkkarten zu finden ist. Es werden keine Pufferspeicherbereiche im Hostspeicher angefordert und keine impliziten Nachrichtenpufferungen durchgeführt. Anstatt eine Menge von Aufrufen an die Anwenderschnittstelle zur Verfügung zu stellen wird hier die Operationsweise der Hardware spezifiziert und den Anwendungen somit eine standardisierte Schnittstelle zur Netzwerkkarte geliefert. U-Net stellt der Anwendungsschicht sogenannte Endpunkte für die Handhabung der Netzwerkfunktionalität bereit. Ein Endpunkt enthält drei zirkuläre Warteschlangen, die Beschreiber von Nachrichtenpuffern enthalten. Dabei gibt es für das Senden Puffer (send queue), Puffer, die für das Empfangen von Nachrichten frei sind (free queue) und Puffer, die bereits empfangene Nachrichten enthalten (receive queue), wie in Abbildung 10 zu sehen ist. Für das Senden enthalten die Beschreiber jeweils die Zeiger auf einen Puffer, dessen Länge und die jeweilige Zieladresse der Nachricht. Die Netzwerkkarte nimmt diese Beschreiber auf, überprüft die Zieladresse und überträgt die Nachricht über einen Zugriff auf den gemeinsam genutzten Speicher. Beim Empfangen einer Nachricht sucht sich

die Netzwerkkarte den richtigen Endpunkt aus nimmt einen entsprechenden Beschreiber aus der free-queue, übersetzt die virtuelle Adresse in die physikalische Adresse des Zielspeicherbereichs und schreibt die Nachricht über DMA an die richtige Stelle. Danach wird ein neuer Beschreiber in die Warteschlange der empfangenen Nachrichten eingestellt. Die Anwendungen können nun eingehende Nachrichten durch einfaches Überprüfen der received Queue erkennen. Dabei wird Schutz dadurch gewährleistet, daß ein Endpunkt genau einer Anwendung zugeordnet ist. Active Message II (AMII) und Virtual Memory Mapped Communication (VMMC) bieten Kommunikationsprimitive an, die auf die Verwendung von geteiltem Speicher beruhen. Es gibt eine put und eine get-Primitive, die der sie auslösenden Instanz erlauben, die Adressen von Daten am anderen Ende anzugeben. Damit wird die zusätzliche Kopie durch die Verwendung eines dazwischengelagerten Pufferbereichs vermieden und die Übertragung erfolgt vollkommen ohne die Beteiligung der Partnerinstanz. VMMC stellt auf der Empfangsseite Funktionalität zur Verfügung, die den Host von ankommenden Nachrichten in Kenntnis setzt. Die Erweiterung VMMC2 dagegen stellt die Nachrichten in einen Puffer zur Verfügung, der Adreßübersetzungen zu Zielspeicherbereichen enthält, die vor dem Vorgang angefordert werden müssen (user-managed translation look-aside buffer). Außerdem gibt es einen Default-Pufferspeicher, für den Fall, daß man ohne vorherige Anforderung von Speicher senden will.

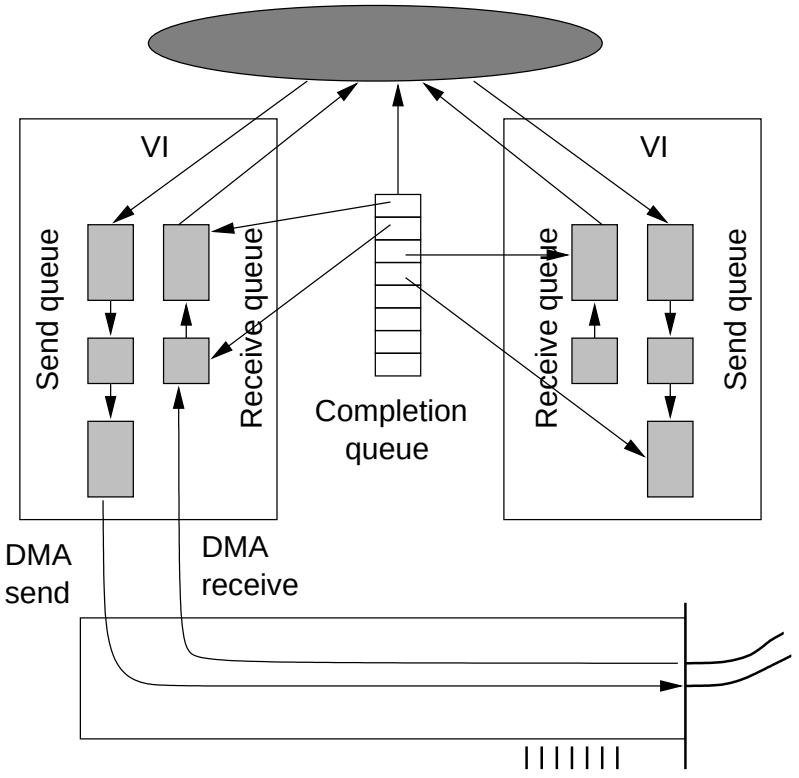

Abbildung 11: *Die Architektur von VIA.*

Bei VIA wird nun die grundsätzliche Funktionsweise von U-Net verwendet und zusätzlich der gerade beschriebene Übertragungsmechanismus über Kommunikationsprimitive hinzugefügt. Ein Prozeß öffnet eine virtuelle Schnittstelle (VI) um das Netzwerk handhabbar zu machen, die im wesentlichen dem Endpunkt bei U-Net entspricht. In Abbildung 11 sieht man die wesentlichen Unterschiede zur U-Net Architektur. Jede VI hat eine Sende- und eine Empfangswarteschlange in denen wieder Beschreiber sitzen. Zum Senden werden nun einfach neue Sendebeschreiber in die Sendewarteschlange und für das Empfangen neue Beschreiber in die Empfangswarteschlange eingestellt, die jeweils wieder auf Speicherbereiche zeigen. Nach der Abarbeitung beim senden setzt VIA einfach ein Bit im entsprechenden

Beschreiber (completion-bit), worauf die Anwendung den Beschreiber entfernt. Jede VI repräsentiert dabei genau eine Verbingung zu einer entfernt liegenden oder lokalen anderen VI. Bei U-Net konnte ein Endpunkt dagegen noch mehrere Kanäle zusammenfassen. Bei VIA gibt es außerdem die Möglichkeit, daß ein Prozeß eine oder mehrere Warteschlangen für bereits empfangene Nachrichten einrichtet und in diese auch die Nachrichten von mehreren Sende- und/oder Empfangswarteschlangen verschiedener VI´s einstellen kann. Zusätzlich bietet VIA auch die direkte Übertragung zwischen lokalem und entferntem Speicher, wie bei den oben beschriebenen put und get-Operationen an. Um hier für mehr Schutz zu sorgen, kann ein Prozeß den dafür vorgesehenen Speicherbereich registrieren lassen. Dieses Registrieren liefert gleichzeitig eine Handhabung für die Adreßumrechnung.

Abschließend bleibt noch zu sagen, daß dieser neuer Standard kein revolutionäres Konzept anbietet, das alles dagewesene in den Schatten stellt. Es bietet manche Vorteile, ist aber in seiner Leistungsfähigkeit nicht unbedingt besser wie die Vorgängerkonzepte. VIA ist vielmehr ein Ansatz der einen Standard für Netzwerkkarten der Benutzerebene setzen soll um somit einen Einsatz in realen Systemen erst zu ermöglichen. Je nach Netzwerk, in dem so ein System eingesetzt werden soll wird es sicherlich zu verschiedenen Ausprägungen kommen oder sich auch die anderen Systeme als besser erweisen.

2.6 Zusammenfassung

Es wurde nun viel darüber diskutiert, daß heute Netzwerkkarten eingesetzt werden, die für die zukünftigen Anforderungen an die Kommunikation nicht mehr geeignet sind, da sie einfach jetzt schon einen Flaschenhals zwischen den schneller gewordenen Prozessoren und den Hochleistungsnetzwerken darstellen. Es wurde auch angesprochen woran dies im einzelnen liegt und wie man diesen Problemen zu Leibe rücken kann. Dabei war z. B. häufig die Rede, die Latenzzeit durch das Umgehen des Betriebssystems zu verringern und Netzwerkkarten direkt aus der Anwendung heraus zu steuern. Zusätzlich dazu mußte festgestellt werden, daß dies alleine vielleicht noch gar nicht ausreichend ist, sondern die Zukunft der Kommunikation vielleicht eher in der Verwendung von neuen, speziell auf den Einsatz in Netzwerken zugeschnittenen Rechnerarchitekturen liegen muß. Alles in allem muß man diese Überlegungen aber in einem etwas realistischeren Licht betrachtet. All diese Aspekte zielen auf Bereiche ab, in denen eine schnelle Kommunikation wirklich auch notwendig ist. Diese sind z. B. in der Forschung oder in der Industrie. Zwar wünscht sich natürlich auch der private Anwender einen möglichst schnellen Übertragungsweg zu anderen Anwendern, doch kann man durchaus auch auf dem Standpunkt sein, daß die Netzwerktechnologie für diesen Bereich wirklich immer noch ausreichend gut ist und ein privater Nutzer niemals überhaupt die hohen Übertragungsraten wird ausnutzen können, die in den Besprochenen Ansätzen angestrebt werden. Diese Meinung kommt auch in einer Studie über die Übertragungsverzögerungen bei einer Anbindung ans Netz mit Windows NT zu Tage, die sich mit den herkömmlichen Mechanismen durchaus zufrieden gibt [JoRe99]. Das ganze Thema ist also immer auch eng an den Bereich geknüpft, in dem die Kommunikation stattfindet, und welche Anforderungen tatsächlich an eine Netzwerkkarte gestellt werden. Die technische Entwicklung geht auf jeden Fall weiter und früher oder später wird auch der private Anwender Nutzen aus der Forschung ziehen können, die in diesem Bereich für industrielle Anwendungen betrieben wird.

Literatur

[AACMu98] Mark D. Hill Andrew A. Chien und Shubhendu S. Mukherjee. Design Challenges for High-Performance Network Interfaces. *IEEE Communications*, November 1998, S. 42–45.

[JoRe99] Michael B. Jones und John Regehr. The Problems You're Having May Not Be the Problems You Think You're Having: Results from a Latency Study of Windows NT. *IEEE Computer Society*, März 1999.

[vEVo98] Thorsten von Eicken und Werner Vogels. Evolution of the Virtual Interface Architecture. *IEEE Communications*, November 1998, S. 61–68.

RMON und RMON2 - effizientes „remote network monitoring“

Matthias Rieber

Kurzfassung

Hier werden die RMON und RMON2 MIBs des IETF beschrieben. Sie wurden entworfen, um das Netzwerkmanagement großer Netze zu vereinfachen und zu dezentralisieren. Viele proprietäre Lösungen ermöglichen zwar, einzelne Subnetze zu überwachen, jedoch erschweren die vielen verschiedenen Standards, sich ein Bild über das gesamte Netzwerk zu machen, da sie meist von unterschiedlichen Managementprogrammen gesteuert werden müssen. Durch die vorwiegend in den RFC-1757 und RFC-2021 definierten MIBs lassen sich die einzelnen Probes von einem Managementprogramm steuern und überwachen und eröffnen dadurch Möglichkeiten der Fehlererkennung und Vermeidung, sowie einer gezielten Netzwerkplanung.

1 SNMP MIBs

1.1 SNMP Protokoll

Das SNMP Protokoll wurde Mitte der 80er Jahre entwickelt, als klar wurde, daß man für die immer schneller wachsenden Netze ein Verfahren benötig, das in der Lage ist, diese zu überwachen und zu steuern. Dabei war das SNMPv1 Protokoll zunächst nur als Zwischenlösung gedacht, bis man ein neues, besseres Protokoll entwickelt hat. Allerdings haben sich die anderen Protokolle auf Grund ihrer Komplexität nicht durchsetzen können.

SNMPv1 besitzt ein recht simples Design. Es dient zur Kommunikation zwischen einem Manager und einem beliebigen, SNMP fähigen Gerät in einem Netzwerk.(Abb. 1)

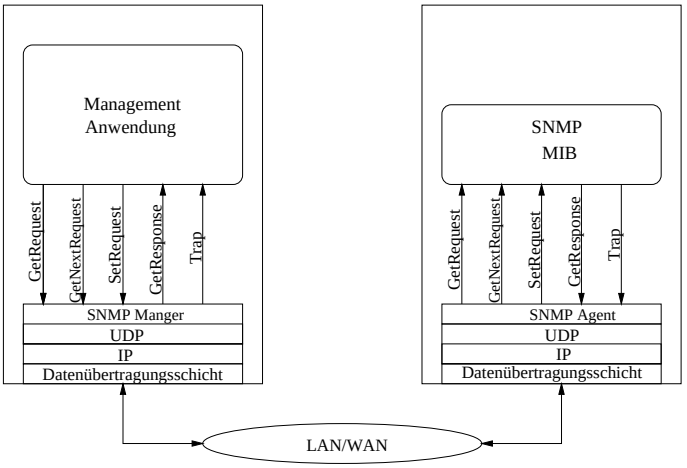

Abbildung 1: SNMP Kommunikationsmodell

Dies kann zum Beispiel ein Router oder Switch sein, von dem dann der Manager mit Hilfe dieses Protokolls, verschieden MIB Variablen, wie z.B. die Systemlaufzeit, abfragen kann. Es stehen vier verschiedene Befehle zur Kommunikation zwischen einer Managementstation und dem Gerät zur Verfügung. Mit dem Befehl `GET` werden Variablen ausgelesen. Durch `GETNEXT` wird die darauf folgende Variable ausgelesen. Dies ist eine wichtige Funktion für das Auslesen von Tabellen, da sich einzelne Zeilen einer Tabelle nicht direkt ansprechen lassen. Mittles `SET` werden Variablen gesetzt. Eine `TRAP` PDU dient dazu, den Manager über eine definierte Ausnahmesituation zu informieren

Auf Grund der geringen Komplexität hat das Protokoll den Vorteil, daß es leicht und kostengünstig zu implementieren ist, und sich problemlos weitere Variablen den MIBs hinzufügen lassen. Allerdings besitzt die Version 1 des Protokolls einige Schwächen. So wird jede Variable in einer einzelnen PDU verschickt, dadurch entsteht, vor allem wenn wenige Oktetts übertragen werden müssen, ein großer Protokolloverhead. Weitere Nachteile existieren im Bereich der Sicherheit. Daher wurden die größten Sicherheitslücken im neuen SNMPv2 geschlossen. Es wurde eine *Verschlüsselung* der Daten eingeführt, die zuvor im Klartext übertragen wurden, bessere *Authentifizierungsmöglichkeiten* erhöhen zusätlich die Sicherheit, da zuvor der *Community* Name als Authentifizierung genügte. Außerdem lassen sich nun *benutzerabhängige* Rechte vergeben, so daß nicht jeder Benutzer alles ändern und lesen kann, sondern nur die für ihn relevanten Variablen. Schließlich wurden noch zwei neue Befehle eingefügt. Zum einen `GetBulkRequest`, der es dem Manager ermöglicht, mehrere GetNextRequests zu kombinieren, um den Overhead bei der Übertragung zu verringern. Zum anderen `InformRequest`, der der Kommunikation zwischen Managern dient, damit beispielsweise eine Managementstation eine andere auf eine bei ihr vorliegenden Situation aufmerksam zu machen, die unter Umstanden Auswirkungen auf den anderen Manager haben kann. Daher wurde auch eine spezielle *Manager to Manager Management Information Base* definiert.[Dive00]

Da diese Erweiterungen jedoch nie in einem Standard festgelegt wurde und sich auf ca. 12 RFCs verteilen, konnten sie sich nie kommerziell durchsetzen. Nur bestimmte Varianten, wie zum Beispiel SNMPv2p, die den neuen Sicherheitsmechanismus implementiert, werden von einigen managebaren Geräten unterstützt. Um diese Dilemma zu beseitigen, wird das SNMPv3 Protokoll entwickelt, das SNMPv2p und SNMPv2u(Benutzer-basierter Sicherheitsmechanismus) kombiniert. [Dive00]

1.2 Die MIBs

Die MIB(Management Information Base) ist eine Art Datenbank, welche alle durch das IETF definierten Objekte enthält, die ein managebares Gerät enthalten kann. Die einzelnen Definitionen einer MIB sind in einer Baumstruktur (Abb. 2) angeordnet. Die obersten Ebenen dieser Hierachie werden für die Zuweisung der Berechtigung an einzelne Gruppen verwendet und enthalten in der Regel keine sinnvollen Daten. Erst mit der OID iso.org.internet.dod.mgmt erscheinen administrierbare Objekte für TCP/IP. Unter dod.internet.experimental befinden sich die in der Entwicklung und Standardisierung befindlichen MIBs. Diese werden bei Verabschiedung des Standards nach dod.internet.MIB... verschoben. Unternehmen können sich unter dod.internet.private.enterprises einen Namensraum für ihre proprietären Entwicklungen reservieren. [Klei98] [Perk93]

Die Daten werden in den MIBs in der Beschreibungssprache ASN.1[1] abgelegt. Mit dieser Sprache lassen sich die einzelnen Datenobjekte in einer allgemeinen Form speichern. So sieht zum Beispiel eine Definition aus der RFC1213 (MIB-II) wie folgt aus:

```
sysUpTime OBJECT-TYPE
```

[1] Abstract Syntax Notation One

```
SYNTAX  TimeTicks
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
        "The time (in hundredths of a second) since the
        network management portion of the system was last
        re-initialized."
::= { system 3 }
```

SYNTAX: beschreibt den Typ des Objekts, wie INTERGER, SEQUENCE OF oder komplexere Objekttypen

ACCESS: beschreibt wie auf ein Objekt zugegriffen werden kann, z.B. lesend, schreibend, erzeugend und verschiedene Kombinationen dieser Typen

STATUS: beschreibt ob das Objekt gültig, veraltet, oder in den nächsten Versionen nicht mehr unterstützt wird

DESCRIPTION: beschreibt was für Daten das Objekt eigentlich enthalten soll

::= system 3 beschreibt die Position des Objekts

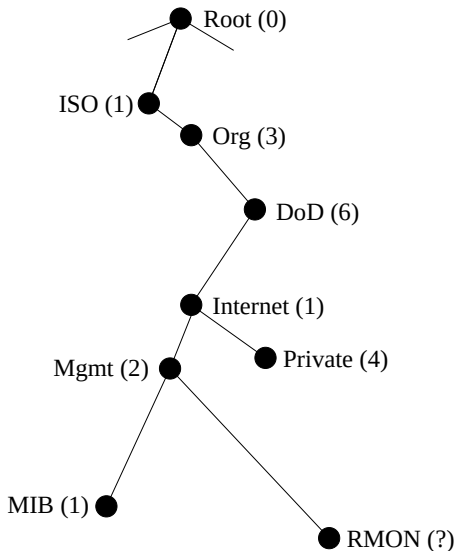

Abbildung 2: Die Struktur einer MIB

1.3 Nachteile der SNMP MIBs

Die hier gezeigte Netzwerkmanagementlösung besitzt, neben den bereits erwähnten Mängeln im SNMP Protokoll, weitere Nachteile. Viele wichtige Funktionen sind in den SNMP MIBs I und II definiert, allerdings haben die Hersteller ihren Geräten weitere, an ihren Funktionsumfang angepaßte, MIB Variablen hinzugefügt. Damit ist es nicht mehr oder nur mit großem Aufwand möglich, mehrere SNMP fähige Geräte mit einer Managementsoftware zu überwachen. Dazu kommt noch, daß die MIBs eher im Hinblick auf die Überwachung von einzelnen Geräten und nicht des Netzwerkes entwickelt worden sind.

Diese Verfahren stellen ein zentralisiertes Netzwermanagement dar, da die Geräte ständig von der Managementstation abgefragt werden, und keine Zwischenspeicherung gesammelter

Daten vorgesehen ist. Dadurch ensteht bei häufigen Abfragen auf Grund der ineffizienten Übertragung eine hohe Netzwerkbelastung, und sollte wegen einer Netzwerkstörungen die Verbindung zwischen Agent und Manager abbrechen, so gehen die in dieser Zeit gesammelten Daten verloren. Daher bietet sich eine denzentralisierte Lösung an, bei der die einzelnen Agenten im Netz autonom arbeiten, die erfassten Daten zwischenspeichern und durch einen Manager abgerufen werden können. Genau diese Möglichkeiten bieten die RMON MIBs, die wesentlich umfangreichere Überwachungs- und Konfigurationsmöglichkeiten bieten.

2 Die RMON MIBs

2.1 Die Entwurfsziele

- Betriebskonzept (Offline, Multiple Managers)

  Das Betriebskonzept unterscheidet sich grundlegend von der SNMP MIB Variante. Die einzelnen RMON Geräte, auch *Probe* genannt, sind meist spezialisierte Geräte, die nur dem Zweck dienen, das Netzwerksegment zu überwachen, allerdings sind zum Beispiel, um RMON Funktionalität erweiterte, Router oder Switche denkbar. Diese Probes verfügen über mehr Speicher und einen leistungsfähigeren Prozessor, als vergleichbare SNMP MIB Geräte, da sie von der Managerstation unabhängig arbeiten müssen und ihr Ergebnisse, bis zum Abruf speichern. Deswegen ist für eine lückenlose Erfassung der Daten keine ständige Verbindung zum Manager erforderlich. Durch den besonderen Aufbau dieser MIBs ist es möglich, daß *verschiedene* Manager die Probe auf vielfältige Art und Weise konfigurieren und diese die unterschiedlichen Aufgaben erfüllen kann. Somit wird ein einheitliches Remote-Management möglich.

- Speicherkonzept (Proactive Monitoring)

  Wie bereits erwähnt, speichern die Probes ihre Messdaten, Analyse- und Diagnoseergebnisse ab, dadurch ist es möglich, über einen längeren Zeitraum die Entwicklung eines Parameters zu beobachten, später auszuwerten und gegebenenfalls, bei unerwünschter Entwicklung, Maßnahmen zu ergreifen. Ein weiterer Vorteil der Speicherung ist der geringere Messbias, denn würden die Messergebnisse ständig übertragen werden, so könnte sich das Ergebnis der Messung durch die zusätzliche Netzbelastung verfälschen. Durch einen besseren Aufbau der Tabellen ist es nun möglich, nur die seit dem letzten Aufruf veränderten Teile der Tabelle neu zu übertragen und dadurch eine unötige Übertragung bereits bekannte Daten zu vermeiden. Außerdem ermöglicht die Speicherfunktion eine Filterung und Protokollierung bestimmter Pakete, die schließlich später wiederum untersucht werden können. [Moll98]

2.2 RMON 1

Die RMON 1 MIBs definieren, im Gegensatz zu den RMON 2 MIBs, Statistik- und Kontrollobjekte für den Layer 1 und 2 (Abb. 3). Im Gegensatz zu den alten MIB-I und MIB-II, die hauptsächlich auf die Überwachung des Gerätes ausgerichtet sind, ist es nun möglich, das Netzwerksegment zu überwachen und auch Daten über andere Geräte zu sammeln. So gibt es eine Reihe von neuen Funktionen, die nun ein Proaktives Netzwerkmanagement ermöglichen.

Voraussetzung ist auch die Möglichkeit ohne eine Managerstation auszukommen, und die Daten, die die Probe sammelt, zu speichern und selbstständig zu filtern bzw. auszuwerten. Die Fähigkeit, von mehreren Managern gleichzeitig gesteuert zu werden, birgt weitere Probleme. So sind folgende Konfliktsituationen denkbar:

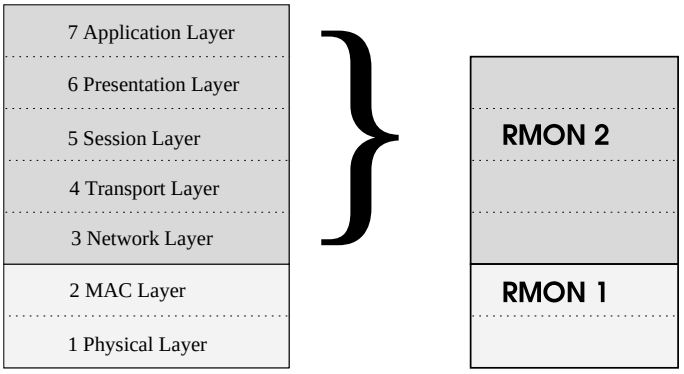

Abbildung 3: Aufgabenteilung zwischen RMON 1 und 2

- Zwei Management Stationen wollen gleichzeitig auf Resourcen zugreifen, dies übersteigt aber die Fähigkeiten der Probe.
- Eine Management Station benötigt über einen langen Zeitraum viel Rechenzeit.
- Eine Management Station stürtzt ab und gibt die Resourcen nicht wieder frei, so daß sie von anderen benutzt werden könnten.

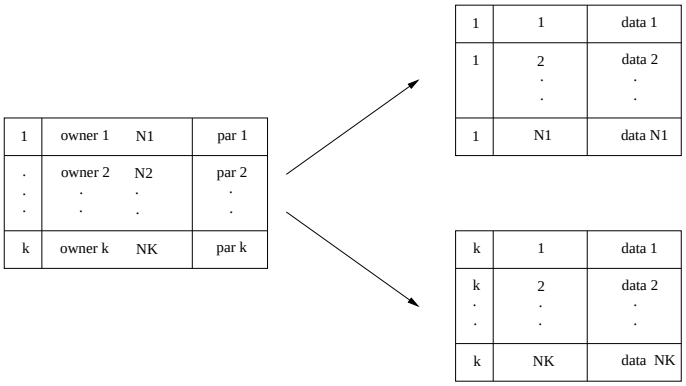

Abbildung 4: Aufbau der Kontroll- und Datentabellen

Um Probleme zu vermeinden, die durch den Zugriff mehrerer Manager entstehen, ist ein spezieller Kontrollmechanismus eingeführt worden. Dieser wird durch die folgenden Tabellen (Abb. 4) realisiert. In den *Kontrolltabellen* wird pro Auftrag ein Eintrag erzeugt. Ein solcher Eintrag besitzt bei allen Gruppen einen ähnlichen Aufbau. Dieser beginnt mit einem Index, der die Tabelle durchnumeriert, aber nicht fortlaufend sein muß, und zugleich eine Verbindung zwischen Kontrolleintrag und der entsprechenden Datentabelle schafft. Dann wird in einem String, der den Besitzer des Eintrages identifiziert, gespeichert. Zusätzlich werden noch die genauen Parameter des RMON Auftrages abgelegt. Diese Tabelle dient also vor allem der Allokation von Resourcen an den Manager. Der Vorgang wird in Abbildung 5 durch die verschiedenen Zustände eines Tabelleneintrages während seiner Erzeugung verdeutlicht. Nachdem der Monitor den Auftragswunsch mit der selbstgewählten ID, den Ownerstring und den gewünschten Parameter (`requestedPar`) abgeschickt hat, wechselt der Zusand des Tabelleneintrages von invalid nach createRequest, falls diese ID frei sein sollte, andernfalls gibt es einen Fehler. Dann wechselt der Status unter Voraussetzung ausreichender Resourcen nach underCreation. Nun wird der Eintrag von dem Manager konfiguriert und danach auf valid gesetzt. Danach beginnt die Probe mit der Ausführung des Auftrages und die Ergebnisse können ausgelesen werden. Wird der Eintrag nicht mehr benötigt, sollte vom Manger der

Zustand wieder auf invalid gesetzt werden. Anhand dieser Einträge ist es der Probe nun möglich, Störungen des Managers zu erkennen. Befindet sich ein Eintrag zu lange 'underCreation' oder wurden die Datentabellen eines gültigen Eintrags lange Zeit nicht ausgelesen, können diese Einträge automatisch durch Setzen auf invalid gelöscht werden, da im ersten Fall der Manager vermutlich abgestürzt ist und im zweiten vergessen wurde, den Eintrag zu löschen.

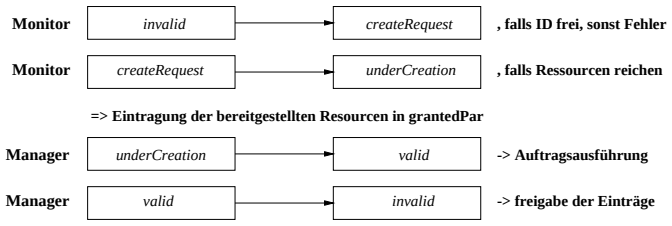

Abbildung 5: RMON Synchronisation

In den *Datentabellen* werden die von den einzelnen RMON Aufträgen erzeugten Werte gespeichert. Die nun folgenden RMON Gruppen sind optional und müssen nicht alle implementiert werden, um dem RMON Standard zu genügen. Es müssen allerdings aus der MIB-II die `system` und die `interface` Gruppe implementiert werden. Diese beiden Gruppen ermöglichen dem Agenten Befehle zur Steuerung der Objekte.

2.2.1 Ethernet Statistics Group

Diese Gruppe ermöglicht einfache Statistiken über jede in der Probe eingebauten Netzwerkschnittstelle. Dabei handelt es sich bei diesen Funktionen um Counter, die bei bestimmten Paketen den Zähler erhöhen. Das Objekt `etherStatsPkts` speichert beispielsweise die Zahl aller im Netzwerk übertragenen Pakete. Ein weiteres interessantes Objekt ist das `etherStatsDropEvents`. Dies gibt die Zahl der Pakete an, die durch Resourcenmangel verlorengegangen sind. Dies kann bei hohem Netzwerkverkehr passieren, da die Probe alle Pakete, die in ihrem Netzwerksegment übertragen werden, untersuchen muß. Ist dieser Wert im Vergleich zu der Anzahl der übertragenen Pakete groß, ist es möglich, daß die Meßergebnisse dieser und anderer Gruppen fehlerhaft oder sehr ungenau sind. Um Aussagen über die Übertragungsqualität zu treffen, bietet sich das Objekt `etherStatsCRCAlignErrors` an. Dort wird die Anzahl der Pakete mit fehlerhafter Rahmenprüfsumme gespeichert.

2.2.2 History Control Group

In dieser Gruppe werden die Kontrolleinträge für die Ethernet History Group erstellt. Dabei wird neben der Datenquelle auch die Intervallänge und die Anzahl der Intervallwiederholungen konfiguriert. Da die Probe vorher prüft, ob sie diese speicherintensive Funktion ausführen kann, wird die tatsächliche Anzahl der Wiederholungen in `historyControlBucketsGranted` gespeichert.

2.2.3 Ethernet History Group

Hier wird für jeden History Kontrolleintrag eine Tabelle erzeugt, die für jedes Samplingintervall eine Zeile enthält. Dort sind nahezu alle Statistiken, die auch in der Ethernet Statistic Group existieren, enthalten. Ein neues wichtiges Objekt ist `etherHistoryUtilization`, das die Auslastung in Prozent an dem überwachten Interface angibt. Die durch diese Funktionen gewonnenen Daten stellen eine wertvolle Hilfe zur Problembekämpfung vieler Netzwerkprobleme dar.

2.2.4 Alarm Group

Diese Gruppe ermöglicht die Überwachung von Variablen des ASN.1 Primitivtyps `INTEGER` (`INTEGER, Counter, Gauge` oder `TimeTicks`). Wird ein bestimmter Schwellwert über- bzw. unterschritten, so wird ein *Event* ausgelöst. Um eine unnötige Netzwerkbelastung zu vermeiden, wird dieser Event nur ein einziges mal ausgelöst.(Abb. 6)

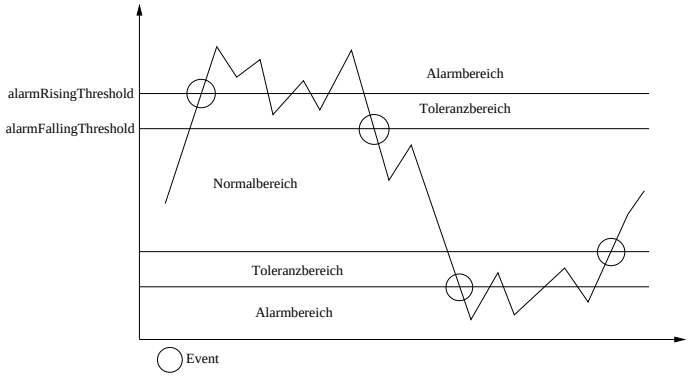

Abbildung 6: Hysterese Mechanismus

Es ist nicht nur eine Überwachung von absoluten Werten möglich, sondern auch Differenzen zwischen zwei Messwerten können überwacht werden, um eine zu schnellen Anstieg eines bestimmten Messwertes zu erkennen. Ein Beispiel für eine solche Anwendung ist das Erkennen von bestimmten Denial of Service Angriffen, die sich durch eine stark ansteigende Zahl von kleinen Pakete im Netzwerk bemerkbar machen. Dazu wird einfach durch die Alarm Group die OID der Variable `etherStatsPkts65to127Octets` überwacht und bei einer schnellen Zunahme ein Alarm ausgelöst.

2.2.5 Host Group

Diese Gruppe baut anhand der MAC Quell- und Zieladressen, die es in gültigen Paketen findet, zwei Tabellen auf. Zum einen die `hostTable`, diese Tabelle enthält alle Hosts nach der MAC Adresse sortiert. Zum anderen die `hostTimeTable`, die die gleichen Hosts, allerdings nach dem Zeitpunkt der Entdeckung sortiert, enthält.

Die `hostTimeTable` hat zwei entscheidende Vorteile. Dadurch, daß die Tabelle von 1 bis zur ihrer Größe durchnumeriert ist, sind diese Werte vorhersehbar und somit mit dem SNMP Protokoll effizient zu übertragen. Da der letzte Tabelleneintrag bekannt ist, kann man bei einer Veränderung der Host Tabelle einfach die neuen Tabelleneinträger herunterladen, ohne daß wieder die gesamte Liste übertragen werden muß. Ein Neuladen ist nur nötig, wenn der Agent auf Grund eines Resourcenmangels alte Tabelleneinträge löschen muß, und deshalb den Index der Host Tabellen neu erstellt, da es sonst zu Inkonsistenzen in der Tabelle der Managementstation führt.

In diesen Tabellen werden, neben den MAC Adressen, die gesendeten und empfangenen Pakete und Oktetts gespeichert, sowie die Multicast, Broadcast und fehlerhafte Pakete.

2.2.6 Host TopN Group

Diese Gruppe basiert auf der `Host Group`. Sie dient zum Erstellen von Top N Listen. Die Managementstation bestimmt einen Zeitraum und einen Startzeitpunkt, an dem die Messung durchgeführt werden soll. Dabei können Statistiken über gesendete und empfangene Pakete

und Oktetts, sowie über Multicast, Broadcast und fehlerhafte Pakete erstellt werden, die dann in absteigender Reihenfolge nach einem dieser Zähler sortiert wird.

2.2.7 Matrix Group

Diese Gruppe erstellt für bestimmte Netzwerkschnittstellen jeweils zwei Tabllen. Diese enthalten die Adressen aller Verbindungen, die die Probe aus den Header der gültigen Pakete entnimmt. Dabei werden diese Tabellen einmal nach der Quelladresse und nach der Zieladresse sortiert. Außer den Verbindungen enthalten diese Tabellen die Zahl der übertragenen Pakete und Oktetts, sowie die fehlerhaften Pakete. Mit diesen Daten kann eine Managementstation eine Grafik erstellen, die die Kommunikation zwischen den einzelnen Rechner aufzeigt, und dabei Verbindungen mit viel Verkehr oder auch vielen fehlerhaften Pakete anzeigen kann.

2.2.8 Filter Group

Diese Gruppe stellt vielfältige Filterfunktionen zur Verfügung. Es gibt zwei verschiedene Arten von Filtern. Der *Datenfilter* basiert auf einem Bitmuster, das bei einem festgelegten Offset überprüft wird. Die Konfiguration erfolgt durch drei Bitmasken. Dabei werden die relevanten Bits festgelegt, dann das Muster, der es entsprechen muß und die Bits, die nicht gesetzt sein dürfen. So ließe sich ein Muster wie z.B. 01x10, wobei das Bit x egal ist, wie folgt filtern.

Möglichkeit	filterPktDataMask	filterPktData	filterPktDataNotMask
1	11011	01x10	00x00
2	11011	01x10	10x01

Entsprechend funktioniert dies bei den *Statusfiltern*

Verschiedene Filter können zu einem *Channel* zusammengefasst werden. Dabei ist eine `or` oder `and` Verknüpfung möglich. Zusätzlich werden die gefilterten Pakete gezählt und es können auch *Events* ausgelöst werden. Einzelne Channel lassen sich auch per Event an- und ausschalten. Wichtig ist auch noch, daß jeder Filter mindestens einem Channel zugewiesen wird, da hier das Netzwerkinterface festgelegt wird, bei dem die Filterregeln gelten sollen.

2.2.9 Packet Capture Group

Mit der Packet Capture Gruppe kann man die Pakete, die den Bedingungen eines Channels entsprechen, speichern. Diese können dann zu einem späteren Zeitpunkt durch die Managementstation heruntergeladen werden. Dabei kann das Verhalten sehr genau definiert werden. So müssen die Pakete nicht komplett gespeichert werden, sondern es ist möglich, ab einem bestimmten Offset, eine definierte Zahl von Oktetts zu speichern. Tritt ein Pufferüberlauf auf, kann man die neuen Pakete zu verwerfen, oder, nach dem FIFO Prinzip die alten Pakete zu verwerfen, um die neuen aufzunehmen.

2.2.10 Event Group

Events dienen zur Signalisierung von Ereignissen und können auch andere Ereignisse auslösen. Es gibt vier verschiedenen Event Typen. Ein Event kann keine Reaktion auslösen, er schreibt einen Eintrag in eine Log Tabelle, er generiert eine SNMP Trap oder er erzeugt beides.

Events können beispielsweise durch die Überschreitung eines Alarmgrenzwertes erzeugt werden. Ein solcher Event könnte dann wiederum einen Channel aktivieren, um bestimmte Pakete zu filtern.

2.3 RMON 2

Die RMON 1 MIBs beseitigen viele Einschränkungen der alten MIBs. Es bleiben jedoch einige grundlegende Probleme. So besteht immer noch ein hohes Datenaufkommen, da häufig komplette Tabellen übertragen werden müssen. Die Beschränkung auf den MAC-Layer erschwert die Untersuchung verschiedener Sachverhalte. Unter anderem ist es nur schwer möglich, den Netzverkehr einzelner Protokolle zu untersuchen. Daher wurden die RMON 2 MIBs entwickelt, die diese Probleme beheben sollten. Zusätzlich zu den neuen Funktionen wurden die alten RMON 1 MIBs in ihrer Funktionalität erweitert. Alle Tabellen enthalten nun einen Eintrag für *Dropped Frames* und *LastCreateTime*. Dabei hat LastCreateTime den Zweck, daß man nicht immer alle Zeilen einer Tabelle übertragen muß. Der Offset der Filtermaske ist nun nicht mehr fest, sondern hängt von dem Protokolltyp des Pakets ab, und auch das Statusbit der Filterfunktion enthält mehr Zustände.

Ein wichtiger Hinweis noch zum Verständnis der RMON 2 MIBs ist die Bedeutung des *Application Layer*. Dies bedeuted nicht, wie man meinen könnte, den OSI Layer 7, sondern alle Layer ab 4 bis einschließlich 7. Der *Network Layer* beschreibt den OSI Layer 3. [Moll98]

2.3.1 Protocol Directory Group

In dieser Gruppe sind alle Protokolle eingetragen, die die Probe erkennen und dekodieren kann. Diese Protokolle sollten auch dann eingetragen sein, wenn für diesen Protokolltyp noch keine Implementierungen in den anderen Gruppen existieren. Es kann eingestellt werden, ob der Benutzer von einer Protokollart weitere Protokolltypen ableiten kann, zum Beispiel von `ip.udp` zu `ip.udp.snmp` oder `ip.udp.dns`. Es werden auch die Fähigkeiten beschrieben, die die anderen Gruppen betreffen, zum Beispiel ob ein Adressmapping (→ 2.3.3) möglich ist, ob der Host (→ 2.3.4) bestimmt werden kann oder ob der Aufbau einer Verbindungsmatrix (→ 2.3.5) möglich ist.

2.3.2 Protocol Distriubution Group

Hier kann man sich für jedes Interface der Probe eine Tabelle erstellen lassen, die eine sehr rudimentäre Statistik der übertragenen Pakete enthält. Es wird für jedes Protokoll ein Eintrag erzeugt, der die Zahl der übertragenen Pakete und Oktetts enthält.

2.3.3 Address Mapping Group

Hier wird für jedes Interface die erkannten Netzwerkadressen (IP-Adresse) und die entsprechenden Hardwareadresse abgespeichert.

2.3.4 Network/Application Layer Host Groups

In der `NlHostEntry`[2] Tabelle wird für jeden Host, der erkannt wird, die Anzahl der empfangenen und gesendeten Pakete und Oktetts gespeichert.

Die `AlHostEntry`[3] Tabelle entspricht der NlHostEntry Tabelle, abgesehen davon, daß nur ein bestimmtes Protokoll berücksichtig wird.

[2] Nl → Network layer
[3] Al → Aplication layer

2.3.5 Network/Application Layer Matrix Group

Dies entspricht den RMON 1 Gruppen Host TopN (→ 2.2.6) und Matrix (→ 2.2.7). Wie bei RMON 1 lassen sich also wieder Matrix und TopN Listen erstellen, allerdings jetzt auf der Netzwerkebene, bzw. für bestimmte Protokolle.

2.3.6 User History Group

Dies entspricht der History Group (→ 2.2.2) der RMON 1 MIBs. Ergänzend kommt hinzu, daß man benutzerdefinierte MIB Variablen erstellen kann, und die Variablen zu Gruppen zusammengefaßt werden.

2.3.7 Probe Configuration Group

Hier lassen sich viele Parameter der Probe konfigurieren. So läßt sich über `probeResetControl` ein Warm- oder auch ein Kaltstart der Probe durchführen. Über `probeCapabilies` kann man sich über die unterstützten RMON MIBs, auch RMON 1 MIBs, informieren. So ist es für die Managementsoftware einfacher, sich auf die Fähigkeiten des Agenten einzustellen. Firmwareupdates sind mittels der `probeDownload*` Befehle durch einen Download von einem TFTP Server möglich. Wichtig ist natürlich, daß sich sämtliche Schnittstellen konfigurieren lassen. Dies schließt neben den eigentlichen Netzwerkinterfaces die Serielle und auch eventuell angeschlossene Modems mit ein. Über ein solches Modem ließe sich der Agent auch ohne eine funktionierende Netzverbindung konfigurieren. Dies stellt im Falle einer Fehlkonfiguration der Netzschnittstelle, beispielsweise eine falsch gesetzte Netzmaske, eine ideale Möglichkeit dar, das Problem zu beheben, ohne den Ort der Probe aufsuchen zu müssen.

3 Anwendungsmöglichkeiten einer RMON Probe

3.1 Beispiel 1

Es soll überprüft werden, wieviele Broadcast-Pakete der Computer versendet, der am meisten ausgehende Pakete erzeugt.

Dazu sind in RMON1 folgende Schritte nötig. Man erstellt eine hostTopN Tabelle mit dem Sortierkriterium „hostTopNOutPkts“ und definiert ein Intervall, in dem die Messung erfolgen soll. Nach Abschluß der Diagnose kann das Ergebnis in Form einer Tabelle übertragen werden. Damit läßt sich dann der gesuchte Rechner festellen und einen Filter definieren, der eine entsprechende Bitmaske enthält, um die Broadcast Pakete zu erfassen. Diesen Filter verbindet man schließlich mit einem Kanal, der mit dem Interface verbunden wird, an dem der zu untersuchende Host hängt. Nun kann man in `channelMatches` die Zahl der erfaßten Broadcast Pakete auslesen. Will man die Pakete noch genauer untersuchen, ist es möglich, durch einen Eintrag in der Packet Capture Group, die Pakete, bzw. Teile davon, zu speichern.

3.2 Beispiel 2

Der Administrator soll benachrichtig werden, wenn zu viele http(Port 80) Zugriffe stattfinden.

Dies stellt mit den RMON2 MIBs kein Problem dar. Man muß nur den IP.www (definiert in der Protocol Directory Group) Verkehr überwachen und kann mit der Alarm Group eine Trap auslösen, der in der Event Group festgelegt wurde.

Mit den RMON1 MIBs ist diese Aufgabe etwas schwieriger zu lösen, da man keine bestimmten Protokolle beobachten kann. Deshalb muß man über die Filterfunktion die http Ports aller Server einzeln beobachten und die Anzahl der Pakete aufsummieren. Das Aufsummieren geschieht auf der Managementstation, und würde bedeuten, daß das Netzwerk unnötig belastet wird, da ständig Pakete übertragen werden müßten, um eine Überschreitung des Grenzwertes rechtzeitig festzustellen.

4 Sicherheit und der Einsatz im Unternehmen

4.1 Sicherheit

Ein nicht zu unterschätzendes Problem der RMON Probes ist die Sicherheit. Wie man in Bild 7 sieht, befinden sich die RMON 1 Probes innerhalb eines Subnetzes und sehen nur diesen Netzwerkverkehr. RMON 2 Probes hingegen sehen einen wesentlich umfangreicheren Bereich eines Netzwerkes, da sie an Ports eines Routers oder Routing Switches sitzen, an dem der Netzwerkverkehr mehrerer Teilnetze sichtbar ist. Die Fähigkeit, Pakete zu filtern und zu speichern, birgt große Gefahren, sollte ein Angreifer Kontrolle über die Probe erlangen. Mit dieser Funktion lassen sich dann recht einfach Passwörter abfangen, indem z.B. Pakete mit dem Zielport 21(ftp) oder 23(telnet) gespeichert werden. Dies läßt sich noch optimieren, indem man die Pakete speichert, nachdem durch die Filter Group der String „LOGIN:“ o.ä. erkannt wurde. Also wie man sieht, ein sehr praktisches Hackerwerkzeug.

Auf Grund der recht schwachen Authentifizierungsmöglichkeit der RMON MIBs bzw. des SNMP Protokolls und der Möglichkeit der RMON Probes Datenpakete zu speichern und zu übertragen, empfiehlt es sich den SNMP Zugriff auf diese Funktionen einzuschränken. Auch spezielle Ports für RMON2 Probes stellen ein Sicherheitsproblem dar, da dort Unbefugte den kompletten Netzwerkverkehr abhören können.

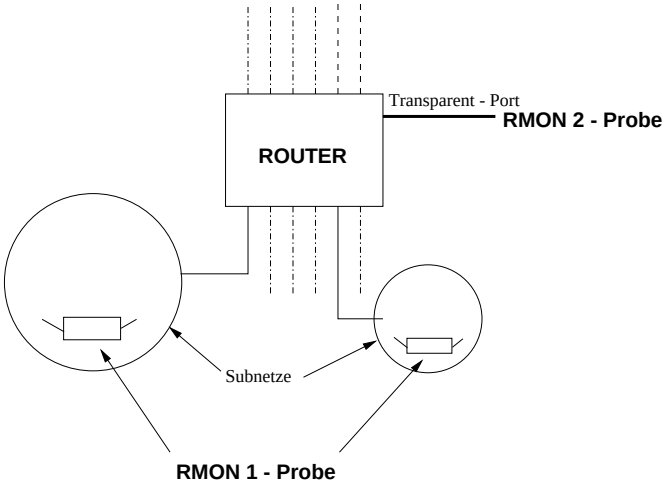

Abbildung 7: Aufbau eines Netzwerkes mit RMON-Probes

4.2 RMON Einsatz im Unternehmen

Nun stellt sich natürlich für Unternehmen die Frage, ob es für ihr Netzwerk lohnenswert ist, eine auf RMON basierende Überwachung einzuführen. Dagegen sprechen zunächst einmal die hohen Anschaffungskosten. Eine RMON 1 Probe kostet in etwa zwischen 1000-2000DM.

RMON 2 Probes schlagen je nach Funktionsumfang mit etwa 6000-20000DM zu Buche, haben aber den Vorteil, daß sie nicht für jedes einzelne Ethernetsegment benötig werden (siehe 4.1). Die dazu benötigte Managementsoftware bewegt sich in preislich ähnlichen Regionen. Der größte Dorn im Augen der Systemadministratoren dürfte die mangelnde Sicherheit sein. Solange die Kommunikation zwischen den Probes und dem Manager unverschlüsselt von statten geht, sind Angriffe, zumindest von innerhalb des eigenen Netzes, sicherlich möglich. Hier bieten sich verbesserte Protokolle, wie das SNMPv3 an, die über ein ausgefeilteres Sicherheitssystem verfügen. Diesen Punkten stehen allerdings auch große Vorteile gegenüber. Durch den kompletten Überblick über das Netz, lassen sich drohende Fehler schneller feststellen und die Ursachen beheben oder falls doch ein Fehler auftritt, ist das Problem schneller zu finden und zu beheben. Ein weiterer wichtiger Vorteil ist die Möglichkeit, Engpässe im Netz zu erkennen, um dadurch das Netzwerk gezielter und damit kostengünstiger auszubauen, denn es ist sehr ärgerlich, Geld zu investieren, das sich dann nicht in einer gestiegenen Leistungsfähigkeit bemerkbar macht. Denkbar ist auch, daß sich Personalkosten einsparen lassen, da die zentrale Zusammenfassung der Daten weniger arbeitsintensiv ist, als dezentrale Auswertung der einzelnen Subnetze. Als Fazit bleib also die Frage, ob man für eine Reduzierung der DownTime des Netzes, die zweifellos Geld kostet, ein höheres Risiko eines Angriffs in Kauf nimmt.[Unko98]

Literatur

[Dive00] Diverse. *Lehr- und Übungsbuch TELEMATIK*. Gerhard Krüger, Dietrich Reschke. 2000.

[Klei98] Jürgen Klein. SNMP Allgemein. *Internet*, 1998.

[Moll98] Wolfgang Moll. Tutorial Netzwerkmanagement. Technischer Bericht, Institut für Informatik, Rheinische Friedrich-Wilhelms-Universität Bonn, November 1998.

[Perk93] David T. Perkings. Understanding SNMP MIBs. Technischer Bericht, September 1993.

[Unko98] Unkown. RMON2 Management Software And Probe Features. *www.NetworkComputing.com*, 1998, S. 68,72.

Computer Telefon Integration - CTI

Fenghui Chen

Kurzfassung

Computer Telefon Integration (CTI) spielt eine bedeutende Rolle bei der Verbindung von auf Computer basierenden Informationsdiensten und auf Telefon basierenden Beratungsdiensten bei Call-Centern. Das Seminar befaßt sich mit der CTI Architektur, den Bestandteilen, der Funktionsweise und CTI-bezogenen Standards. Am Ende werden die CTI-fähige Produkte verglichen und eine CTI-Anwendung vorgestellt.
Schlüsselwort: CTI, TAPI, First-Party-Architektur, Third-Party-Architektur

1 Motivation

Immer mehr Kundenkontakte erfolgen heute über das Telefon. Call-Center bieten die entscheidende Hilfe für die notwendigen Qualitätssteigerungen im Kundendienst. Die Call-Center sind Telefonzentralen, die durch entsprechende Technik, wie automatische Anrufverteilung oder Ansagedienste, sehr viel mehr Anrufe entgegennehmen können und eine gleichmäßigere Auslastung der Agenten ermöglichen. Call-Center verbessern die Effizienz des Telefonverkehrs, und steigern damit die Erreichbarkeit des Unternehmens enorm und tragen erheblich zur Kundenzufriedenheit bei.

Die verschiedenen Dienste werden in einem Call-Center angeboten. Z. B. ist die Problemlösungsunterstützung für Kunden ein Dienst, bei dem die Informationsdienste und Beratungsdienste zugleich angeboten werden müssen. Informationsdienste und Beratungsdienste sind die beiden Extreme in diesem Kontinuum. Informationsdienste sind kostengünstig, asynchron zugänglich und betreffen einfache Probleme, für die Standardhilfe angeboten werden kann, während die Beratungsdienste die menschliche Komponente darstellen, den kooperativen Dialog zur Lösung komplexer und schwieriger Probleme von Kunden ermöglichen und somit synchron zugänglich und zwar kostenintensiv sind aber individuelle Hilfe anbieten können.

Die Verschmelzung der Informationsdienste und Beratungsdienste bei Call-Centern hat das Potential, die existierenden Dienste hinsichtlich Kosteneffizienz und Dienstgüte zu verbessern und sogar neue ienstleistungsformen zu schaffen.

Außerdem gibt es schon umfassende Rechnerunterstützung im Call-Center, sie bezieht sich bisher jedoch allein auf die effiziente Handhabung hoher Gesprächsaufkommen mit Technologie wie Automatic Call Distribution (ACD) oder Interactive Voice Response (ICR) bezieht. Die Verschmelzung der Informationsdienste und Beratungsdienste bei

Call-Center selbst basiert auf der Computer-Telefon-Integration (CTI) Technologie. Der Seminarbeitrag dient zur Studie der CTI Architektur und Funktionsweise.

2 Architektur von CTI

2.1 ACD-Anlage und Grundbegriffe der CTI

Die ACD-Anlage (Automatic Call Distribution) ist das Kernstück des Call-Centers. Die ACD-Anlage hat die folgenden Aufgaben oder Funktionalitäten:

- Eingehende Anrufe an alle Mitarbeiter im Call-Center zu verteilen,
- Anrufe zu verwalten, die nicht sofort bearbeitet werden können,
- unterschiedliche Prinzipien für die Bearbeitung der Warteschlange anzuwenden,
- Statistiken zu führen,
- frei programmierbare Rufverteilung (Call Flow), Skill-based-Routing zu unterstützen,
- Reportingfunktion für die Steuerung von Call-Center zu übernehmen,
- Status zu überwachen.

Herkömmliche Telekommunikation (TK)-Anlagen können die obigen Aufgaben nur bedingt erfüllen. Man unterscheidet drei Arten von ACD-Systemen : (1)adaptiertes System (2) Integriertes System (3) CT-System.

Bei den adaptierten Systemen ergänzt der ACD-Server die vorhandene TK-Anlage und steuert sie. Der Vorteil ist eine leichte Integration in das System. Einen möglichen Nachteil stellt die Verbindung zwischen TK-Anlage und ACD-Server dar. Manche ACD-Server sind speziell für eine TK-Anlage ausgelegt und von demselben Hersteller entwickelt worden. Integrierte Systeme implementieren die ACD-Funktionen direkt als Erweiterung der TK-Anlage. Dazu müssen meistens spezielle Endgeräte eingesetzt werden.

Bei CT-Systemen handelt es sich um PC-basierte Systeme, die neben Netzwerkadaptern keine spezielle Hardware benötigen, also auch keine TK-Anlage. Der Einsatz ist aus Kostengründen dann sinnvoll, wenn noch keine TK-Anlage vorhanden ist. Diese Systeme sind sehr flexibel und leicht zu integrieren.

Innerhalb eines Call-Centers unterteilt man ein System in 2 Teile, einen TK-Teil und einen IT-Teil. Die ACD-Anlage ist das Kernstück des TK-Teils. Kernstück der IT- Komponente ist das "Customer Interaction System" (CIS), normalerweise eine Software, die auf dem Arbeitsplatzrechner der Agenten läuft. Am Arbeitsplatz spiegelt sich diese Trennung in zwei Endgeräte wider: dem Telefon und dem PC. Die Verbindung dieser beiden Komponenten wird als Computer-Telefon-Integration (CTI) bezeichnet, die eine Reihe von nützlichen Anwendungen ermöglicht. Ein leicht vorstellbares Beispiel ist das "Screen-Pop", wenn ein Anruf eintritt, erscheint am PC Bildschirm eine Maske, die die Kundendaten unter der Anrufnummer enthält.

2.2 Architektur von CTI

Nachdem wir die Grundbegriffe der CTI beschrieben haben, wird die gesamte Architektur der CTI schichtweise dargestellt. CTI wird in 2 Architekturen realisiert, nämlich First-Party-Architektur und Third-Party-Architektur.

2.2.1 First-Party-Architektur der CTI

Abbildung 1 beschreibt die logische Verbindung von Komponenten in der First-Party-Architektur der CTI. Die Eigenschaft der First-Party-Architektur ist, daß die CTI-Verbindung nur zwischen einem Rechner und einem Telefon existiert. Die benötigten Komponenten bei der First-Party-Architektur sind TK-Anlage, Telefon, Rechner(meistens PC), PC-Schnittstelle zum Telefon und die entsprechende Software wie TAPI und Treiber der Schnittstelle.

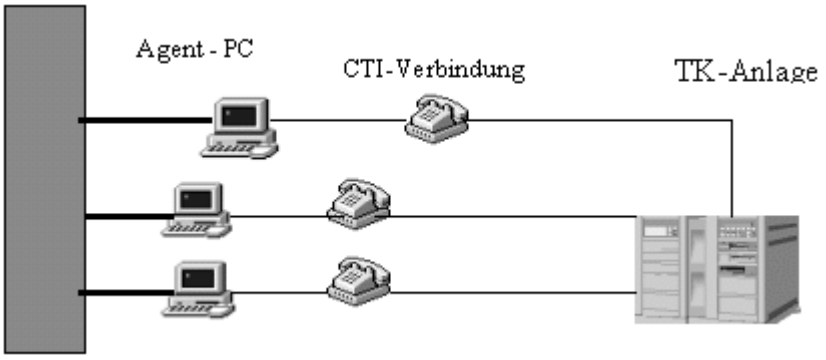

Abbildung 1: First-Party-Architektur und logische Verbindung von CTI-Komponenten

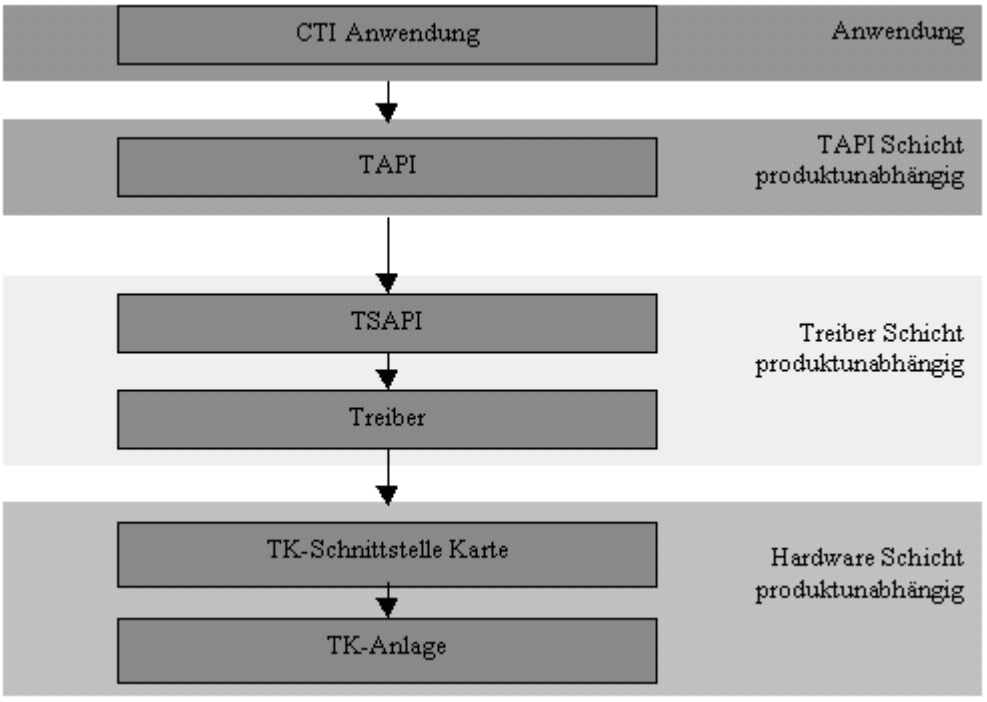

Abbildung 2: Architektur von CTI Software

Abbildung 2 beschreibt die geschichtete Architektur der Software der CTI in der First-Party-Architektur. CTI ist eigentlich eine Anwendung, die mittels TAPI(Telefon Application Programm Interface) die Funktionalität der TK-Anlage in Anspruch nimmt.

Komponenten und Funktionsweise:

Um den PC mit einer Telefonleitung zu verbinden, wird eine PC-Schnittstelle (nämlich eine Karte im PC) benötigt. Sie stellt eine Schnittstelle zur Signalisierung dar (z. B. Eintreffen eines Anrufs, Beenden eines Gespräches und Steuerung des Telefons).

Es ist wichtig zu wissen, daß der PC bei der First-Party-Architektur sehen und tun kann, was das Telefon sehen und tun kann. TK-Anlage ist dem PC nicht bekannt. So hängt die CTI-Anwendung von der Funktionalität, die die TK-Anlage dem Telefon anbietet, ab. Einem modernen Telefon bietet eine TK-Anlage schon viele erweiterte Funktionalitäten an. Und zwischen Telefon und TK-Anlage werden neue Protokolle für die Signalisierung verwendet, zum Beispiel kann man mit einem entsprechendem Telefon einfach einen eingehenden Anruf durch den Druck eines Buttons an einen anderen Agent innerhalb einer Gruppe umleiten, sogar den Zustand eines anderen Telefons in dieser Gruppe ermitteln, z. B. ob das Telefon besetzt oder frei ist. Um das Telefon von Software(in diesem Fall die CTI-Anwendung) steuern zu können, wird noch ein Treiber, der direkt die Schnittstelle steuert, benötigt. Solche Treiber sind hardwareabhängig oder produktabhängig.

Um die Entwicklung einer CTI-Anwendung zu vereinfachen, wird eine einheitliche Anwendungsschnittstelle wie die TAPI entwickelt. Sie übernimmt die Anpassung der Hardware verschiedener Hersteller. Auf diese Weise braucht die CTI-Anwendung nicht auf die verschiedene Treiber zuzugreifen. Die TAPI, die TSAPI und die Treiber werden noch später im Detail erklärt.

Hier wird die Funktionsweise der CTI in zwei Vorgängen - Eintreffen eines Anrufs und Initieren eines Anrufs beschrieben.

Eintreffen eines Anrufs

(1) Ein Kunde ruft einen Agenten an.
(2) Die TK-Anlage leitet diesen Anruf an das Telefon des Agenten via Signalisierung weiter.
(3) Die PC-Schnittstelle erkennt die Signalisierung. Die Telefonnummer wird auch erkannt.
(4) Der Treiber erzeugt ein Ereignis(Die Rufnummer ist im Ereignis enthalten).
(5) Das Ereignis wird nach oben geliefert und durch TAPI an ein Standardereignis angepaßt.
(6) Die CTI-Anwendung erkennt ein Ereignis und löst eine Aktivität aus.
(7) Ein Fenster wird geöffnet, in dem die Kundendaten unter dieser Rufnummer wie Name, Adresse, etc. dargestellt werden.
(8) Der Agent kann verschieden darauf reagieren, z. B. das Gespräche anzunehmen, den Anruf umzuleiten oder den Anruf zu verweigern.
(9) Wenn das Gespräche angenommen wird, ruft die CTI-Anwendung eine Funktion der TAPI auf, um der TK-Anlage die Annahme zu signalisieren.
(10) Die TAPI paßt den Funktionsaufruf einem Funktionsaufruf des Treibers an.
(11) Der Treiber treibt durch die Schnittstelle die Hardware. Und die Hardware signalisiert. Und damit ist eine Verbindung aufgebaut.
(12) Das Gespräch beginnt.
(13) Für Umleitung oder Verweigerung werden andere Funktionen der TAPI aufgerufen.

Initiieren eines Anrufs

(1) Der Agent wählt einen Kunden von einer Liste aus (oder er gibt die Rufnummer direkt ein) und klickt dann einen Button.
(2) Der Mausklick löst einen Aufruf einer TAPI Funktion aus.
(3) Die TAPI paßt diesen Funktionsaufruf in die Treiber an.
(4) Der Treiber steuert die Hardware, um zu signalisieren.
(5) Die CTI-Anwendung geht in eine normaler Event-Behandlungs-Schleife.
(6) Der Angerufene hebt den Hörer ab und signalisiert das durch Telefonnetz zurück.
(7) Die TK-Anlage signalisiert dem Telefon des Anrufers.
(8) Die Schnittstelle erkennt die Signalisierung.
(9) Der Treiber erzeugt ein Ereignis.
(10) Das Ereignis wird nach oben geliefert und durch die TAPI an ein Standardereignis angepaßt.

(11) Die CTI erkennt das Ereignis und löst eine Aktivität aus. (z. B. den Zeitpunkt zu vermerken, den Ton von Telefon an Lautsprecher zu leiten.)
(12) Das Gespräch fängt an.

Hier soll noch etwas auf den Audiostrom vom Telefon eingegangen werden. Prinzipiell steuert die PC-Schnittstelle nur Signalisierung von Telefon. Der Audiostrom kann wie üblich direkt an das Telefon (Hörer und Mikrofon) geleitet werden. Ein moderner PC hat jedoch normalerweise eine Sound-Karte. So kann der Audiostrom des eingehenden Anrufs in die Sound-Karte bzw. in den Lautsprecher geleitet werden. Umgekehrt wird der Audiostrom des Agenten durch das Mikrofon (nicht Mikrofon des Telefons) der Sound-Karte auf die Telefonleitung geleitet (dabei entsteht keine Rechenzeit). So besteht die Möglichkeit, daß der eingehende Audiostrom durch die CTI-Anwendung bearbeitet (z.B. in eine Datei aufgenommen) und gesteuert werden kann (Zum Beispiel Einstellung der Lautstärke).

2.2.2 Third-Party-Architektur(Client/Server Architektur)

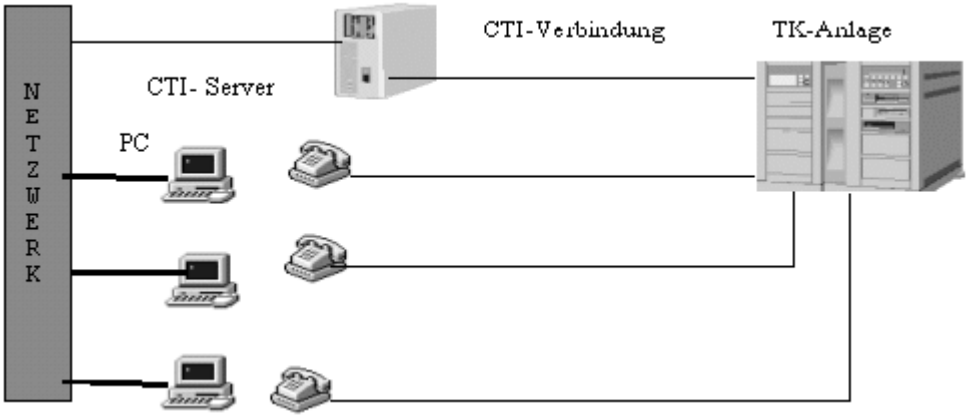

Abbildung 3: Third-Party-Architektur und logische Verbindung von Komponenten

Abbildung 3 stellt die Third-Party-Architektur und die logische Verbindung ihrer Komponenten dar. Die Third-Party-Architektur wird auch als Client/Server Architektur bezeichnet. Sie besteht aus TK-Anlage, CTI-Server, Agent-PCs und einem LAN, das den CTI Server mit den Agent-PCs verbindet. Bei der Third-Party-Architektur steuert der CTI-Server die TK-Anlage und schickt die Signale bezüglich des Telefons durch das LAN an die Agent-PCs. Zugleich stellt er allen Agenten gemeinsame Ressourcen wie eine Datenbank (für z.B. Kundenadreßbuch) zur Verfügung. So kann man sagen, daß der CTI-Server die Funktionalität von Rechner und Telefon tatsächlich zusammen integriert. Im folgenden wird die Architektur von CTI-Server und CTI-Client, Funktionsweise und Ablauf beschrieben.

2.2.2.1 Die Architektur des CTI-Servers

Abbildung 4 zeigt die Softwarearchitektur des CTI-Servers. Hier werden die Bestandteile erklärt.
CSTAPI, ein Software Schnittstelle. Sie implementiert das Protokoll CSTA (Computer Supported Telefon Application) und bietet für Anwendungen die Schnittstelle an, die auf die Media-Call-Control Schicht zugreift.
Media Call Control, eine Call Control Software, die solche Funktionen wie Hold, Transfer, Forward, Divert, Konferenz und Grundfunktionen der Audiobearbeitung (Audiostrom abspielen, aufnehmen) anbietet. `Media Driver Control`, Software Schnittstelle zwischen Media Call Control und Hardware Treibern von verschiedenen Ressourcen wie Line Card, DSP Card, Telefon Network Card. Diese Schnittstelle trennt die Entwicklung der CTI Server Software

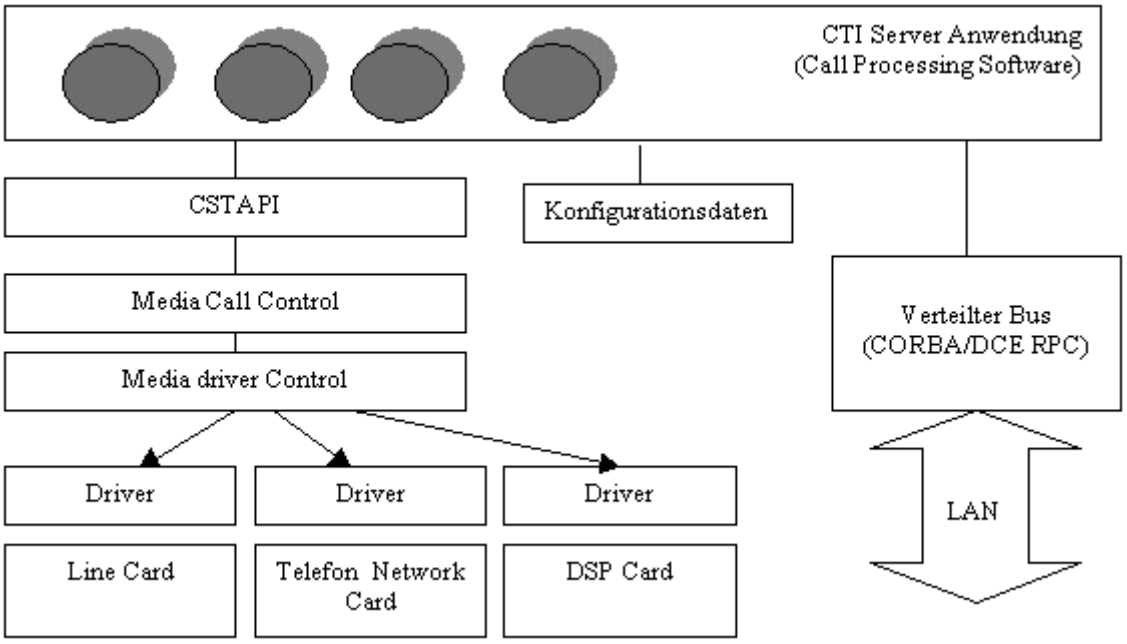

Abbildung 4: Die Architektur von CTI-Server Software

von individuellen Telefon-Kommunikationskarten (Solche Karte werden als Switching Ressource gezeichnet, die zwischen ihnen umgeschaltet weren kann. Der Entwickler von CTI Server Software braucht nicht zu wissen, wie die verschiedenen Kommunikationskarten funktionieren. Media Driver Control unterhält mit Telefon-Kommunikationskarten mittels MVIP (Multi Vendor Integration Protokoll) Bus.

Line Card, eine PC Karte, die einem Verbindungspunkt für ein analoges Telefon anbietet. Sie enthält die Switching Hardware. Das Media Driver Control kann dieses Switching steuern und die Verbindung zwischen individuellen Telefons aufbauen.

DSP Karte, eine PC Karte, die die Möglichkeit anbietet, Ton Nachrichten vorzuspielen und aufzunehmen, sowie Text zu Ton zu übersetzen und Konferenz Rufe zu realisieren.

Telefon Network(Trunk) Karte, eine PC Karte, die eine Schnittstelle zum externen Telefon Network anbietet. Diese Karte enthält Switching Hardware. Das Media Driver Control verwendet das Switching zum Aufbau der Verbindung zwischen externer Telefonleitung und internem Telefon.

Verteilter Bus, der Bus ermöglicht dem CTI-Client, auf die Media-Call-Control Schicht im CTI-Server zu zugreifen. Das heißt, die Funktionen, die Aufruf von Methoden der CSTAPI enthalten, werden in die verteilten Objekte (oder Remote Procedure Call beim DEC RPC) gekapselt. Die verteilten Objekte werden via verteiltem Bus durch CTI-Client aufgerufen.

2.2.2.2 Die Architektur des CTI-Clients in der Third-Party-Architektur

Die Abbildung 5 zeigt die Schichtenarchitektur des CTI-Clients. Die Telefon Anwendung bietet mittels GUI Funktionalitäten wie Umleitung von Anrufen, Wählen, Antworten auf Anruf an. Diese Anwendungen können z. B. in Visual Basic, C++ oder Java implementiert werden.

TAPI: Sie ist eine einheitliche Programmierschnittstelle für Telefon-Funktionalität. Diese wird im nächsten Abschnitt erklärt.

TAPI Service Provider: Mittels TAPI Service Provider werden die Funktionen der TAPI in die CSTAPI (Client Server Telefon API) Requests umwandelt. D. h. der TAPI Service Provider trennt den Programmier von Hardware und Middleware.

CSTAPI: Sie ist gleich wie die in Kap. 2.2.2.1 beschriebene. Man sollte darauf achten, daß die CSTAPI der TAPI nicht bekannt ist.

Verteilter BUS: Er wird bereits in Kap. 2.2.2.1 beschrieben. Er ermöglicht dem CTI-Client, die Methode im CTI-Server für Anrufsteuerung aufzurufen. Damit kann der CTI-Client (Agent) den Anruf manipulieren.

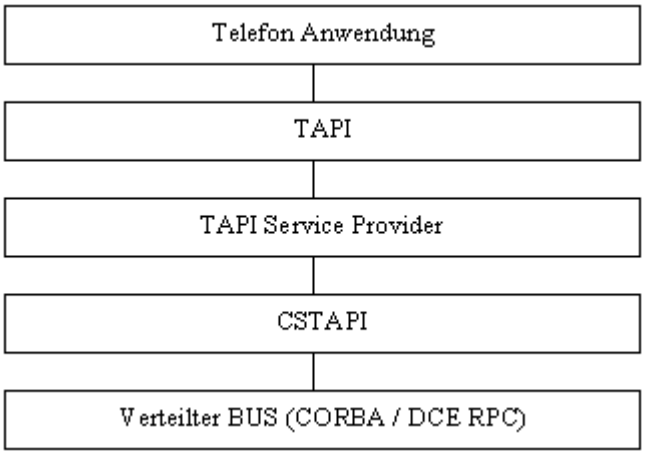

Abbildung 5: Die Architektur von CTI-Client Software

2.2.2.3 Funktionsweise und Eigenschaften

Der CTI-Server spielt eine zentrale Rolle. Er kann alle Gespräche kontrollieren und überwachen. Aber er bermittelt nur die Signalisierung des PCs. Er nimmt nicht an den einzelnen Gesprächen teil. D.h. der Audiostrom von Gesprächen wird nicht via Netzwerk an den PC übermittelt. Physikalisch existieren mehrere Telefone und PCs. Wie wird die Signalisierung von irgendeinem Telefon an einen bestimmten PC per Netzwerk weiter geleitet? Bei der Client-Server Architektur besteht keine explizite Verbindung zwischen Agent-PC und Telefon. Aber im CTI-Server gibt es eine Tabelle, in der die Beziehungen zwischen Agent-PCs und Telefonen definiert werden. Man spricht von einer logischen Verbindung. Mittels dieser Tabelle kann der CTI-Server entscheiden, an welchen PC eine Signalisierung geleitet werden soll, wenn eine Signalisierung von der TK-Anlage durch den CTI-Server erkannt wird. Es ist erforderlich, daß die TK-Anlage in der Lage ist, die Signalisierung an den PC zu liefern und der PC eine spezielle Schnittstelle besitzt, um die Signalisierung abfangen zu können. D. h. die Implementierung der CTI-Verbindung ist für CTI-Server und TK-Anlage bekannt. Die CTI-Anwendungen bleibt eigentlich noch bei den PCs. Aber der Audiostrom wird nicht mehr in die Sound-Karte im PC weitergegeben.

Zusammengefaßte Eigenschaften der Third-Party-Architektur:

- Es gibt nur eine zentrale CTI-Verbindung zwischen CTI-Server und TK-Anlage.
- Zwischen Agent-PCs und Telefon gibt es keine physikalischen CTI-Verbindungen mehr, sondern eine logische Verbindung.
- Die TK-Anlage muß CTI-fähig sein.
- Die CTI-Anwendungen bleiben noch bei den PCs.

- Zwischen CTI-Server und Agent-PC läuft nur die Signalisierung über das Telefon, der Audiostrom von Gesprächen läuft nicht mehr durch den CTI-Server, sondern direkt von der TK-Anlage zum Agent-Telefon.

Folgend wird der Ablauf der CTI Funktionalität beschrieben.

Eingehender Anruf
(1) Ein Anruf von außen ist bei der TK-Anlage eingetroffen.
(2) Der CTI-Server hat mittels TK-Schnittstellenkarte im CTI-Server dieses Signal erkannt und weiß, zu welchem Agent dieser Anruf geht.
(3) Der CTI-Server schickt eine Nachricht über das Eintreffen an den Agent-PC durch das LAN ab. Zugleich klingelt das Agent-Telefon. Die Nachricht enthält die Rufnummer und einige Optionen, wie Antwort, Zurückschicken einer Sprachnachricht, Übergeben an einen anderen Agenten oder Ignorieren.
(4) Der CTI-Client im Agent-PC reagiert auf diese Nachricht. Der CTI-Client stellt die Rufnummer (bzw. die Kundendaten, wenn die Kundendaten unter dieser Nummer gespeichert sind) auf dem Bildschirm dar.
(5) Wenn der Agent den Anruf beantwortet (er hebt den Hörer ab), entsteht bei der TK-Anlage ein Signal, und die TK-Schnittstellenkarte im CTI-Server erkennt das Signal.
(6) Der CTI-Server schickt eine andere Nachricht hinsichtlich dieses Signals an den Agent-PC durch das LAN ab. Diese Nachricht enthält auch einige Optionen, wie z. B. Bereithalten dieses Anruf (das bedeutet, die Verbindung bleibt, aber der Anrufer kann den Angerufen nicht hören, normalerweise wird eine Musik vorgespielt.), Übergeben des Anrufs an einen anderen Agenten.
(7) Der CTI-Client reagiert auf diese Nachricht. Z. B. kann er einfach einen Knopf drücken, um den Anruf bereitzuhalten, oder eine Nummer eines anderen Agenten eingeben und dann den Knopf "Umleiten" drücken. Wenn z. B. der Agent den Knopf "Umleiten" drückt, wird eine Funktion im CTI-Server aufgerufen, dabei ruft diese Funktion via CSTAPI die Funktion der Media-Driver-Control auf, die Media-Driver- Control steuert die TK-Anlage mittels der Schnittstellenkarte im CTI-Server, um die Umleitung zu erledigen.
(8) Der Agent hat das Gespräch beendet und legt den Hörer auf. Dadurch entsteht auch bei der TK-Anlage ein Signal. So wird das Signal wieder vom CTI-Server an den Agent-PC geschickt.
(9) Der Agent reagiert darauf. Z. B. werden einige Funktionsknöpfe ausgeblendet oder einige lokalen Funktionen werden aufgerufen, um z. B. Gesprächsdauer, Anfangszeit, Endzeit bzw. Kundendaten in der Logdatei zu speichern.

Abgehender Anruf
(1) Der Agent nimmt den Hörer ab.
(2) Er gibt die zu rufende Nummer ein oder wählt einfach einen zu rufenden Kunden aus. Die Telefonnummer des Kunden ist schon vorher gespeichert und er drückt einen Knopf "Anruf". So wird eine Funktion zum Anruf im CTI-Server via verteiltem Bus aufgerufen. Die Telefonnummer und z. B. der Agent-Name werden als Parameter mitgeschickt. Diese Funktion sucht die Agent-Telefonnummer in einer Telefonnummer-Agent Tabelle unter dem Agentnamen. Dann ruft sie die Funktion der Media-Access- Control mittels CSTAPI auf. Die zurufende Nummer und die Agent-Nummer werden als Parameter übergeben. Somit weiß die TK-Anlage, zu welchem Agent-Telefon die Verbindung später aufgebaut werden soll. Durch die TK-Schnittstellenkarte wird die TK-Anlage gesteuert und ein Anruf wird durchgeführt, das gleiche passiert, wenn der Agent selbst die Telefonnummer auf der Telefontastatur auswählt und dann wartet die TK-Anlage auf das Ergebnis.
(3) Es gibt drei Möglichkeiten. Der Kunde ist momentan besetzt, die Vermittlung ist belegt, oder die Verbindung wird aufgebaut und das Kundentelefon klingelt. Bezüglich verschiedener Ergebnisse entstehen auch verschiedene Signale. Darauf schickt der CTI-Server auch verschiedene Nachrichten an den entsprechenden CTI-Client bzw. Agent.

(4) Wenn z. B. das Kundentelefon klingelt, trifft eine Nachricht vom CTI-Server beim CTI-Client ein. Der CTI-Client zeigt an dem Bildschirm "Kunde wird angerufen" dar.
(5) Wenn der Kunde den Hörer abhebt, entsteht auch ein Signal bei der TK-Anlage. Entsprechend schickt der CTI-Server eine Nachricht wieder an den CTI-Client. Somit kann der CTI-Client merken, wann das Gespräch ganz genau beginnt, zugleich kann der Agent auch diesen Kunden hören.
(6) Das Gespräch ist beendet. Der Agent legt den Hörer auf. Dadurch entsteht auch ein Signal bei der TK-Anlage. So wird das Signal wieder vom CTI-Server an diesen Agent-PC geschickt.
(7) Der CTI-Client reagiert darauf. Eine Funktion wird lokal durchgeführt, um Gesprächsdauer, Anfangszeit, Endzeit bzw. Kundendaten in den Logdaten zu speichern.

2.3 Vergleich von First-Party-Architektur und Third-Party-Architektur

	First-Party-Architektur	Third-Party-Architektur
Verbindung zwischen PC und Telefon	Physikalisch und explizit	Logisch und im CTI-Server gespeichert.
Wo läuft CTI-Anwendung?	Auf dem PC	Beim PC
Wird Signalisierung gesteuert ?	Ja	Ja
Wohin kann der Audiostrom geleitet werden?	Auf dem PC, wenn Multimediaausstattung vorhanden.	Nur bis zum CTI-Server, nicht bis zum PC

3 CTI Funktionalitäten und Vorteile

3.1 Funktionalitäten

Hier wird eine Übersicht über die CTI-Funktionalität vorgestellt. Normalerweise soll die CTI die folgenden Funktionalitäten anbieten.

Call-Control Service Es handelt sich um die Unterstützung des Verbindungsaufbaus und -abbaus. Dadurch wird die bildschirmorientierte Anrufsteuerung möglich. Beispiel: Der Aufbau einer Verbindung aus einer Kundendatenbank durch Auswahl des gewünschten Gesprächspartners.

- Screen-Pop-up Mit Screen-Popup werden die Möglichkeiten bezeichnet, die Daten des Anrufers wie die Rufnummer, den Namen des Anrufers und andere Informationen auf dem Bildschirm anzuzeigen. Die Rufnummer des Anrufers kann drüber hinaus als Identifikation genutzt werden, um auf diese Art weitere spezifische Daten über den Anrufer, zum Beispiel den Name und die Adresse aus der Kundendatenbank, bereitzustellen. Der Vorteil dieser Funktionalität liegt darin, daß der Anwender vom CTI-System schnell wichtige Informationen erhalten kann und sich somit die Reaktion auf eingehende, aber auch abgehende Rufe beschleunigen läßt.
- Call Monitoring und Management Diese Funktionalität umfaßt die Überwachung von Telefonen. Sie dient zum einen dazu, ankommende Gespräche zu verteilen (Automatic Call Distribution) respektive dazu, auch abgehende Rufe zu verwalten (Outbound Call Management, Power Dialing, Predictive Dialing). Beispiel: Unter Berücksichtigung, welche Agenten zur Zeit überhaupt verfügbar sind, welche davon gerade frei sind und ob Wartezeiten vor dem nächsten Anruf für die Nachbearbeitung von Gesprächen eingehalten werden sollen, werden ankommende oder abgehende Rufe auf die verfügbaren Agenten verteilt.

- Call Log in Call Login bietet die Möglichkeit, allgemeine Informationen bei Abwesenheit zu sammeln beziehungsweise generell alle Kommunikationsvorgänge in einem Rufjournal zu protokollieren. Beispiel: Alle Kommunikationsvorgänge können mit weiteren Informationen, zum Beispiel Dauer des Gesprächs, verursachte Gebühren, Datum und Zeit des Gesprächs in einem Rufjournal protokolliert werden und dienen zum Beispiel bei einer Hotline zur Leistungsverrechnung für die Anrufer.

- Voice and Data Call Association Es geht darum, simultan zur Weiterleitung des Anrufs auch dessen Daten an den entsprechenden Agenten im Call-Center weiterzugeben. Somit muß der Agent die bereits erfaßten Daten nicht nochmals vom Anrufer abfragen. Beispiel: Ein Kunde hat bei einer Produkt-Hotline angerufen und ist bei dem für das gewünschte Produkt zuständigen Agenten gelandet und hat um die Zusendung von Informationen gebeten. Danach möchte der Kunde noch Informationen über ein weiteres Produkt. Dieses Produkt wird aber von einem anderen Agent betreut. Nun kann der Kunde an diesen Agenten weitergeleitet werden und dem neuen Agenten liegen bereits alle erfaßten Daten des Kunden vor.

- Intelligent Routing Mit dieser Funktionalität läßt sich der Anrufer automatisch zu einem für sein Anliegen kompetenten Agenten weiterleiten. Meist unterstützen IVR-Systeme (Interactive Voice Response Units) Intelligent Routing. Beispiel: In einem Call-Center werden mehrere Produkte oder Dienstleistungen betreut. Der Anrufer läßt sich mit Hilfe von Spracheingaben an den richtigen Agenten weiterleiten.

3.2 Vorteile von CTI

CTI-Anwendungen haben eine Reihe von Vorteilen wie folgt
(1) Produktivitätssteigerung durch professionellere, rationellere und erfolgreichere Telefonate,
(2) Kostenreduktion durch den Ersatz manueller Arbeitsschritte und -prozesse,
(3) Zeiteinsparung durch schnellere Suche nach Nummern und schnellere Durchführung von Wählvorgängen und Komfortfunktionen. Der PC hilft den Leuten einen abgehenden Anruf zu tätigen. So kann man Zeit sparen, und der Anruf geht an den richtigen Ort. Die relevanten Daten der Anrufer werden automatisch angezeigt, so wird die Effizienz verbessert,
(4) Höhere Kundenzufriedenheit dank verbessertem Service am Telefon,
(5) Flexibilität durch offene Schnittstelle zur freien Wahl von TAPI-konformen Anwendungen.
(6) Der Agent kann sehen, was andere Agenten in derselbe Gruppe machen und kann gut entscheiden, wohin ein Anruf umgeleitet werden soll,
(7) Effiziente Verwendung von Ressourcen (Kunden Datenbank, Kommunikation innerhalb einer Gruppe). Die oben genannten Funktionalitäten sind nur als grober Überblick zu verstehen. Zusammenfassend wird festgelegt, daß das wichtigste Merkmal von CTI die Verknüpfung von Rufnummer und Daten ist. Die Telefonnummer des Anrufers ist der Schlüssel zur richtigen Anrede des Anrufers und der sofortigen Verfügbarkeit von wichtigen Informationen. Aber auch die angerufene Nummer kann als Entscheidungskriterium genutzt werden. Anhand dieser gewählten Nummer können zum Beispiel Mitarbeiter in einem Call-Center erkennen, welche Informationen der Anrufer benötigt oder aus welchem Grund er anruft.

4 Standards bezüglich CTI

Während der letzten Jahre sind im Umfeld von CTI einige Standards entstanden. Bei der Vielzahl von Möglichkeiten und den unterschiedlichen Sichtweisen fällt es schwer, den CTI-Standard auszumachen. Im Zusammenhang mit CTI-Standards werden oftmals zunächst

CSTA, TSAPI oder TAPI genannt. Sie spielen eine wichtige Rolle am Markt. Abbildung 6 stellt einige CTI-Standards und ihre Abhängigkeit dar.

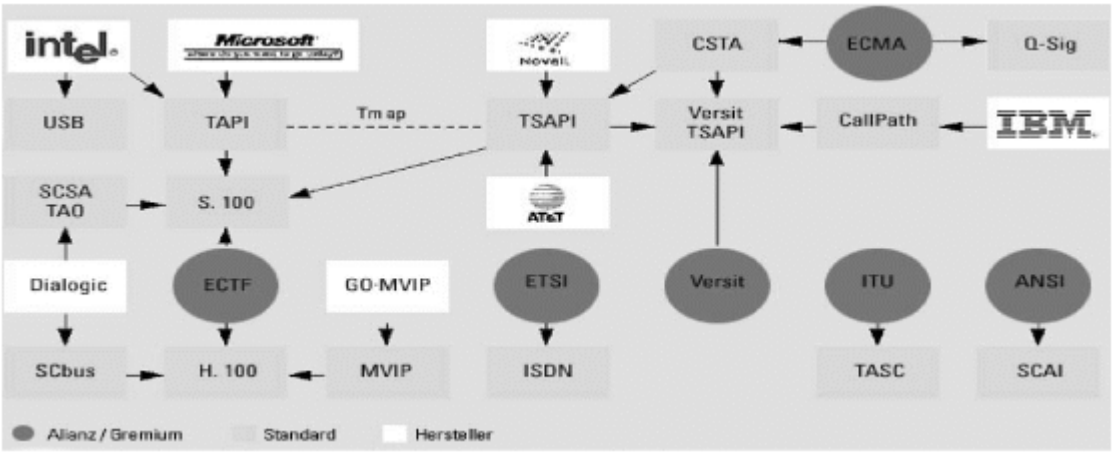

Abbildung 6: CTI-Standards und ihre Abhängigkeit

4.1 CSTA

CSTA (Computer Supported Telecommunications Applications) wurde gemeinsam von Dialogic und der ECMA (European Computer Manufactures Association) als Standard für Computer- und TK-Anlagenhersteller initiiert. Im Gegensatz zu anderen Standards ist CSTA keine spezifizierte Schnittstelle, sondern vielmehr ein Leitfaden zur standardisierten Implementierung der CTI-Funktionalität. Demzufolge ist CSTA als Schnittstelle zwischen TK-Anlagen und DV-Systemen allgemein akzeptiert, aber nur selten vollständig implementiert. CSTA ist systemunabhängig und sowohl für den Desktop- als auch den Systembereich definiert und beinhaltet CTI-Modelle, Protokolle und Funktionen.

4.2 TAPI

Die TAPI 2.0 (Telephony API) wird von Microsoft gemeinsam mit Intel als Telefonie-Schnittstelle für Windows entwickelt. TAPI läßt sich über Modems, ISDN-Karten oder spezielle Adapter an Telefonen ansprechen. Viele Hersteller von TK-Anlagen haben bereits die Unterstützung dieses Standards zugesagt. Die TAPI-Software-Architektur besteht auf der Client-Seite aus einer TAPI.DLL, die die Anwendungsschnittstelle zur Verfügung stellt und einem TAPI-Service-Provider, der die TAPI an die endgerätespezifischen Eigenheiten anpaßt. Bei der Third-Party-Lösung kommuniziert der TAPI-Service-Provider mit einem TAPI- Server, lauffähig unter Windows NT, der wiederum über einen PBX-Treiber an die Funktionalität der eingesetzten TK-Anlage anpaßt wird. Der PBX-Treiber ist deshalb wieder abhängig von der TK-Anlage und kommt meist vom TK-Anlagenhersteller. Bei TAPI sind die Funktionen in vier Gruppen unterteilt. Assisted Telephony, Basic Telephony, Supplemental Telephony und Extended Telephony Services.

4.3 TSAPI

Novell und AT&T haben gemeinsam TSAPI (Telephony Services API) entwickelt, die Anwendungsschnittstelle der Novell Telephony Services (NTS). Die NTS stellen eine Third-Party-Lösung unter Umsetzung von Funktionen und Protokollen nach CSTA Phase 1 zur Verfügung und sind verwaltungstechnisch vollständig in die Novell Directory Services integriert.

Die TSAPI-Softwarearchitektur besteht aus einem Telephony-Server-NLM, einem Telephony-Client für Windows, OS/2, System 7 und Unixware sowie einem CSTA-PBX-Treiber, der die TSAPI-Funktionalität an die Gegebenheiten der eingesetzten TK-Anlage anpaßt. Dieser CSTA-PBX-Treiber ist somit abhängig von der TK-Anlage und wird in der Regel vom TK-Anlagenhersteller bereitgestellt.

4.4 Andere

Es gibt auch andere Standards wie SCSA (Signal Computing System Architecture), MVIP (Multi Vendor Integration Protocol), S.100 und H.100, Java Telephony API, Telas4.0

5 Eine Produktübersicht bezüglich CTI und Call-Center

In der CTI-Architektur gibt es drei wichtige Bauteile - ACD-Anlage, TK-Anlage, CTI Software (einschließlich CTI-Server, CTI-Client für Third-Party Modell, oder einfach CTI Anwendung für First- Party Modell). Es gibt im Weltmarkt verschiedene Unternehmen, die ACD-Anlage, CTI Software herstellen wie Siemens, Nortel, Alcatel, HP.

5.1 Produkte von Siemens

Im Weltmarkt steht das deutsche Unternehmen Siemens in der führenden Position in diesem Bereich. Hier wird beispielhaft ein Produkt von Siemens vorgestellt.

COMManager V 3.0: COMManager Ver 3.0 ist eine CTI-Software. Er kann nicht nur bei der Third-Party-Architektur sonder auch bei der First-Party-Architektur verwendet. Er hat die folgenden Merkmale: 32 Bit-Technologie(lauffähig unter Windows NT/95), kontextabhängige Bedienung der Telefoniefunktionen, TAPI als Schnittstelle zur Telefoniewelt sowie lokale und zentrale Adreßbücher unter Nutzung der MAPI-Schnittstelle.

Er wird dadurch weitgehend unabhängig von den jeweils verwendeten Telefonnetzanschlüssen. So bestehen die Alternativen der direkten Kopplung mit dem Telefon (First-Party-Architektur, wenn der LAN-Zugang nicht nötig oder nicht möglich ist.) oder der Nutzung eines zentralen Telefonieservers (CTI-Server, Third-Party-Architektur, die Benutzer bekommt Zugang zu ihren Telefonen über einen zentralen Telefonieserver). Erreicht wird diese Unabhängigkeit des ComManager durch das Zusammenspiel sogenannter Service-Provider mit der offenen Telefonieschnittstelle TAPI.

Eine immer größere Notwendigkeit beim Einsatz von CTI-Anwendungen ist die Unabhängigkeit von den verwendeten Adreßressourcen. In der Regel sind sie bereits als Datenbank vorhanden, und es soll auf diesen vorgegebenen Datenbestand zugegriffen werden. Der ComManager V3.0 ist in der Lage, diese Datenbestände über die zentrale MAPI-Schnittstelle des Windows-Betriebssystems zu nutzen und gestattet zugleich den parallelen Einsatz lokaler benutzerspezifischer Adreßbücher. Durch die Möglichkeit, unterschiedlichste zentrale Datenbestände in die Suche einzubeziehen, kann eine ständige Aktualität der Adressen gewährleistet bleiben. Beispielsweise ist es auch möglich, den ComManager an unternehmensweite Datenbestände zu koppeln, die von X.500-basierten Systemen wie Org-D (Siemens Nixdorf), Lotus Notes oder Microsoft Exchange zur Verfügung gestellt werden.

Integration der CTI Server in die LAN-Umgebung: CTI Server - Telas Server stellt ein intelligentes Server-Konzept dar. SimplyPhone, der Komforttelefonmanager, bietet

vielfältige Kommunikationsfunktionen. Über die TAPI-Schnittstelle lassen sich beliebige weitere Anwendungen in die TELAS-Umgebung integrieren. TELAS-DDE-Schnittstelle ermöglicht die Integration der Telefonfunktionen in vorhandene Anwendungen auf Basis von Visual Basic. Das Prinzip ist, daß die Entwicklungsumgebung TELAS-SDK (Software Development Kit) eine C-Library für individuelle Anwendungsentwicklung. Sie ermöglicht die Steuerung aller wichtigen Funktionen der TK-Anlage. Mit dem Toolkit SimplyPhone Web lassen sich CTI-Anwendungen im Intranet aufbauen. Der Anwender kann damit ohne einen CTI-Client direkt aus dem Web-Browser heraus wählen. Die Integration wird in Abbildung 7 dargestellt.

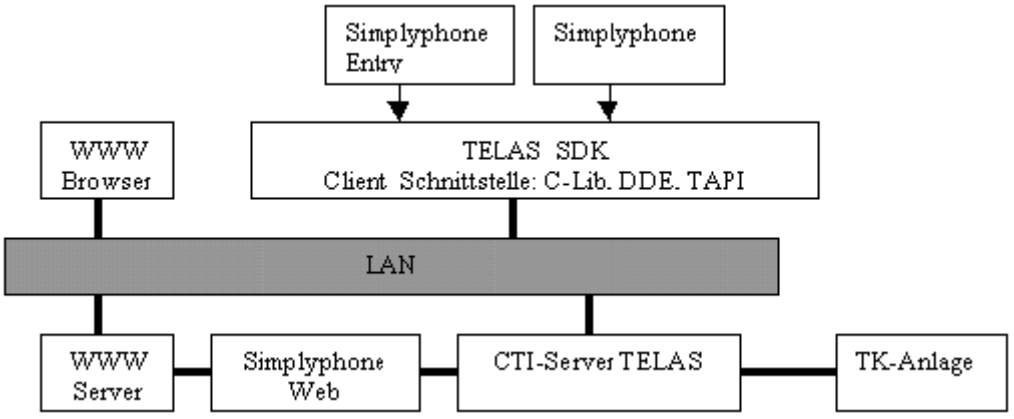

Abbildung 7: Schnittstellen zur Integration in bestehende Systeme

5.2 Produkte von COM4 Bussiness AG

COM4 Bussiness AG ist eine deutsche Firma. Die von ihr vertriebene Anwendung ist XPhone, das unter Windows 95/NT läuft.

Funktionalitäten von Xphone: XPhone hat die folgenden wesentlichen Funktionalitäten. (1) POP-UP Screen. Wenn ein Anruf eintrifft, werden die Kundeninformation nach der Anrufnummer in Kundendatenbank ermittelt und im Fenster angezeigt. (2) Gespräche Kontrollieren durch Software. Durch einfachen Mausklick kann man Hörer abheben bzw. auflegen. Somit die Gespräche wird gestartet und gestoppt. (3) Gespräche überwachen. man kann die Gesprächsdauer überwachen und protokollieren. (4) Man kann unterschiedliche Ereignisskategorien definieren, um eine fremde Anwendung zu starten.

Schnittstelle und Standards: XPhone unterstützt die folgende Telefonschnittstelle bzw. Produkte. (1) TAPI . Durch TAPI kann XPhone auf ISDN Karte zugreifen, damit wird die Kontrolle der TK-Anlage ermöglicht. (2) Serielle Schnittstelle. Durch die serielle Schnittstelle des PC kann XPhone direkt auf Siemens TK-Anlage HiCom Produkte zugreifen.

Datenbankverbindung: Kundendaten sind in einer Datenbank gespeichert. XPhone verwendet ODBC, um auf die Datenbank zuzugreifen. So kann XPhone auf entweder die lokale Datenbank oder die entfernte Datenbank zugreifen. Siehe Abbildung 8.

Integration, Anbindung der fremden Anwendungen: In XPhone kann man bestimmte Ereignisskategorien definieren, um eine bestimmte Aktion auszulösen. Eine sinnvolle Aktion ist, eine externe Anwendung zu starten. Dadurch wird die externe Anwendung in XPhone integriert. Typische Ereignisse und auszulösende Aktionen sind z. B.:

Beginn Klingeln -> Datenbankmaske des entsprechenden Anrufers öffnen
Beginn Klingeln -> Eine externe Anwendung starten,
Ende einer Verbindung -> Protokolleintrag in Logdatei,
Kontext-Menü -> frei konfigurierbare e-Mail mit Gesprächsnotiz (z.B. mit Name, Telefonnummer und Anrufzeitpunkt) verschicken,
Beispiel für eine Ereignisdefinition: Wenn ein Anruf zwischen 08:00 Uhr und 18:00 Uhr eintrifft, wird eine Anwendung APP1 gestartet. Wenn ein Anruf außerhalb dieser Zeit eintrifft, wird eine Anwendung APP2 gestartet. Das ist sinnvoll im Call-Center. Wenn ein Anruf innerhalb der Dienstzeit ankommt, soll etwas (z. B. eine Trouble-Ticket wird automatisch im Agent-PC geöffnet) durch APP1 erledigt werden. Und Xphone funktioniert wie ein Auslöser. Im Zusammenhang entsteht eine Softwarearchitektur von XPhone wie Abbildung 8.

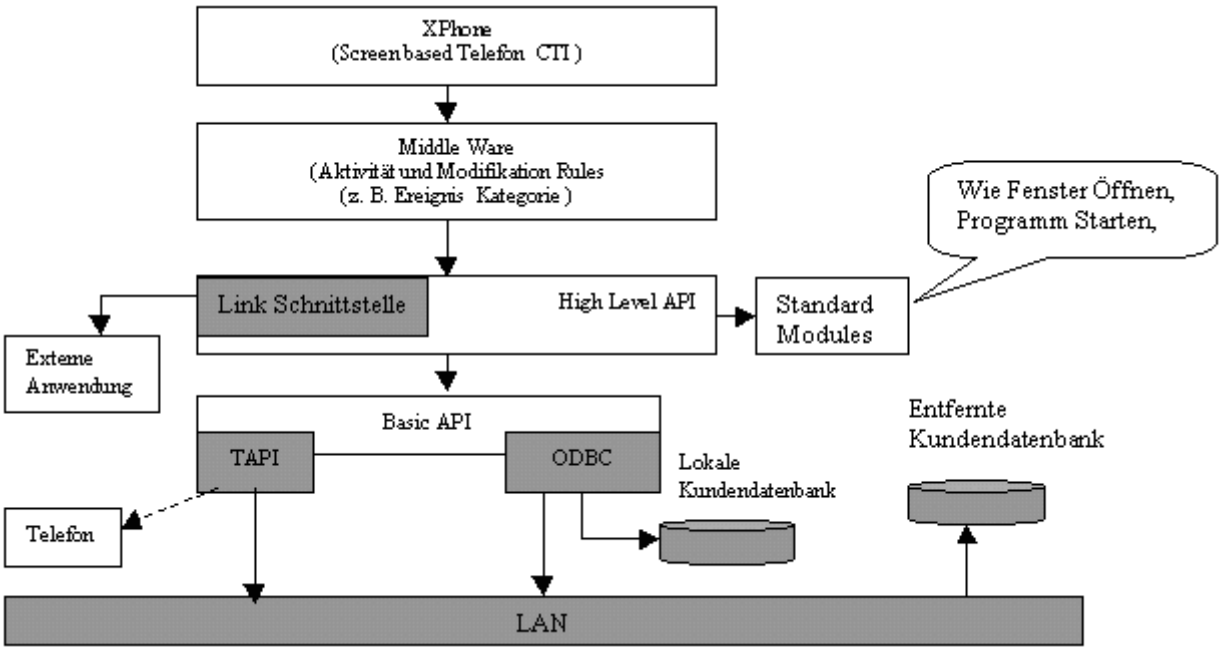

Abbildung 8: Softwarearchitektur von XPhone und Integration

6 Zukunft der CTI

Es scheint, daß sich die drei wichtigsten Werkzeuge für die moderne Gesellschaft - Computer, Telefon, Netzwerk endlos weiter entwickeln. Ihre Funktionalitäten nehmen immer mehr zu. CTI integriert diese drei Werkzeuge. Somit wird die Funktionalität des gesamten Systems noch stärker. Deshalb wird CTI als eine Integrationstechnologie oder Integration Plattform bezeichnet. Das Telefon bzw. Telekommunikationssystem fungiert als Übertragungskanal für Audiostrom, der Computer fungiert als Integrationswerkzeug auf funktionaler Ebene, das Netzwerk dient als allgemeiner Übertragungskanal (Daten oder Audio). Wegen der Begrenzung der Bandbreite ist das Netzwerk nicht immer geeignet für Audioübertragung. Deshalb hat die CTI Technologie die Übertragungen von Daten und Audio organisch vereint. Um CTI aufzubauen und zu entwickeln, spielen die Schnittstellen und Standards eine wichtige Rolle. Das heißt, CTI braucht eine saubere Architektur und Industriestandards, die die Hardware und Software funktionell standardisieren. So wird CTI mit bestehenden Systemen besser verschmolzen und leichter erweitert. Microsoft hat eine gute Strategie geschaffen, die Standard - TAPI, die unter Windows verwendet wird. Die TAPI hat die CTI Software von der

TK-Hardware getrennt. Weitere Informationen zur CTI finden sich unter: [Walt97], [uKom], [Münc], [Deut], [GmbH].

Literatur

[Deut] Alcatel Deutschland. *www.alcatel.com*.

[GmbH] SiKom Software GmbH. *www.sikom.de*.

[Münc] Com 4 Business AG München. *www.c4b.de*.

[uKom] Siemens AG Information und Kommunikation. *www.icn.siemens.com/siemensonline/siemensonline.html*.

[Walt97] Rob Walters. *CTI in Action*. John Wiley and Sons Ltd. 1997.

Konvergenz der Netze

Tim Gölz

Kurzfassung

Konvergenz in Netzen ist vielschichtig. Diese Arbeit befasst sich mit unterschiedlichen Konvergenzprozessen und versucht teilweise deren Zukunftschancen einzuschätzen. Sie vermittelt einen groben Überblick über das was „Konvergenz der Netze“ bedeutet und welche Interessen, Kräfte und Institutionen bei solchen Entwicklungen Einfluß nehmen. Besondere Aufmerksamkeit erfährt dabei die Konvergenz von Sprach- und Datennetzen, als ein Beispiel von Vielen für die Konvergenz von Netzen. An ihm werden in dieser Arbeit Gründe und Voraussetzungen für konvergente Prozesse erklärt und schließlich eine konkrete Realisierung, VoIP, mit ihren Problemen und Lösungsansätzen vorgestellt.

1 Einleitung

„Wie geht es in Zukunft weiter?“ ist eine Frage, die sich jeder ab und zu stellt. Die Antworten darauf sind meistens nur ungenau und die Voraussagen treffen in den seltensten Fällen exakt ein. Bestes Beispiel hierfür ist die Wettervorhersage. Halbwegs zuverlässige Aussagen sind nur für die nächsten zwei bis drei Tage möglich. Alles darüber hinaus ist sehr spekulaiv. Ähnlich wie den Meteorologen geht es Telematikern werden sie nach den Netzen von morgen gefragt. Ob sich eine Technik durchsetzt oder nicht ist von so vielen Faktoren abhängig (Kosten, Zuverlässigkeit, Verfügbarkeit, Konkurrenzprodukte, Werbung, etc ...), daß es fast unmöglich ist gute, langfristige Vorhersagen zu machen. Zu behaupten, in Zukunft gibt es nur noch ein auf **ATM** basierendes Universalnetz ist sicherlich genauso falsch wie die These, daß es die nächsten zwei Wochen nicht regen wird. Gerade **ATM** ist ein passendes Beispiel. Vor einigen Jahren noch als der Durchbruch schlechthin auf einem Weg zum Einheitsnetz gefeiert, hat es sich bis heute nur im Backbone-Bereich durchgesetzt.

Deshalb kann und soll die vorliegende Seminararbeit auch kein Zukunftszenario erstellen. Sie soll vielmehr sich bereits schon vollziehende Konvergenzprozesse aufzeigen sowie die Möglichkeiten und Grenzen solcher Enwicklungen ausloten. Bei der Erstellung diente der Tagungsband *„Konvergenz der Netze“* als Grundlage. [Eber99a] [Fisc99] [Char99] [Eber99b]

Zum Aufbau ist zu sagen, daß in *Abschnitt 2* die Gründe für die Konvergenz von Netzen beschrieben werden. Außerdem enthält er eine Reihe von Beispielen, die einen ersten Eindruck von der Thematik vermitteln sollen. Diese werden in *Abschnitt 3* teilweise aufgegriffen, um die Effekte, die bei solchen Entwicklungen auftreten, zu erläutern. *Abschnitt 4* befasst sich ausschließlich mit aktuellen Beispielen und Entwicklungen, anhand derer auch die Probleme von bzw. Voraussetzungen für ein Zusammenwachsen von Netzen diskutiert werden. Großes Augenmerk wird dabei auf Implementierungen von QoS-Mechanismen gelegt, welche unerläßlich für realzeitkritische Anwendungen wie Telefonie oder Video sind.

2 Gründe für Konvergenz

Konvergenz von Netzen kann unterschiedliche Gründe haben. Die meisten Entwicklungen sind sicherlich ökonomisch motiviert, sie können aber auch technische Ursachen haben. Dieser Abschnitt enthält einige Beispiele, die das belegen. Gleichzeitig soll er aber auch einen Eindruck davon vermitteln wie vielseitig die Problematik ist. Oder anders: Auf welchen Ebenen überall Konvergenz stattfinden kann.

2.1 Ökonomische Gründe

Dieser Unterabschnitt behandelt anhand von zwei Beispielen (2.1.1 Konvergenz von Sprach- und Datennetzen, 2.1.2 Konvergenz von Datennetzen und Fernsehen) die unterschiedlichen ökonomischen Gründe bzw. Motivationen, die für die Konvergenz von Netzen verantwortlich sein können.

2.1.1 Konvergenz von Sprach- und Datennetzen

Das **Internet Protocol** kurz **IP** hat sich im weltumspannenden *„Netz der Netze“* durchgesetzt und ist somit nahezu überall verfügbar. Da es auch die Möglichkeit bietet, Sprache als Daten zu transportieren (näheres dazu siehe *Abschnitt 4.1*) ist eine Verschmelzung von Sprach- und Datennetzen technisch durchaus denkbar. Desweiteren nimmt der Datenverkehr gegenüber dem Sprachverkehr stark zu (Siehe Abb. 1). Da stellt sich die Frage, ob es nicht sinnvoll ist, ein großes, leistungsfähiges Datennetz zu errichten und die dann im Verhältnis recht kleinen Sprachmenge auch darüber zu verschicken. Unterscheidet man der Einfachheit halber grob in zwei unterschiedliche Firmentypen (Betreiber klassischer Telefonnetze vs. Datennetzbetreiber), so stellt sich die Situation wie folgt dar:

Telefonnetzbetreiber: Konzerne wie die deutsche Telekom erzielen mit dem *„herkömmlichen“* Telefonnetz den größten Umsatz und den höchsten Gewinn (12,5 Mrd. Euro bzw. 2 Mrd. Euro– Das entspricht etwa der Hälfte des Konzernumsatzes und zwei Dritteln des Konzernergebnisses vor Steuern). [AG99] Dies hat die Firma nicht zuletzt dem Umstand zu verdanken, daß sie als einzige in Deutschland flächendeckend über die sogenannte *„letzte Meile“* verfügt, also eine Anbindung an den Endnutzer. Trotzdem müssen solche „sprachlastigen“ Anbieter, wollen sie langfristig am Markt bestehen, der sich verändernden Situation Rechnung tragen und ihre strategischen Planungen entsprechend ausrichten. Sie sind gezwungen sich das nötige *„Daten-Know-How“* verschaffen. Dies geschieht in aller Regel durch Fusionen und Übernahmen (Beispiele dazu siehe weiter unten).

Datennetzbetreiber: Diese Firmengattung steht vor einem Problem. Sie hat außer Bandbreite nicht viel anzubieten und deren Vermietung ist kein besonders lukratives Geschäft. Der Einstieg ins Sprachgeschäft bietet da die Möglichkeit, die bereits getätigten Investitionen wie verlegte Kabel, angeschlossene Router und sonstige Hardware für ein neues Geschäft zu nutzen und sich gleichzeitig attraktiv gegenüber der Konkurenz zu machen.

Aus der Sicht des Endkunden sieht das dann so aus: Er schließt nur noch einen Vertrag mit einem Netzanbieter ab. Der versorgt ihn dann mit Internet- und Telefondiensten. Der Kunde erhält nur noch eine Rechnung, was natürlich wesentlich bequemer ist als zwei oder mehrere. Wirtschftlich besonders interessant für die Anbieter sind vor allem Großkunden. So schreibt die „Neue Züricher Zeitung“ in ihrer Ausgabe vom 28. Oktober 1999: [Weis98]

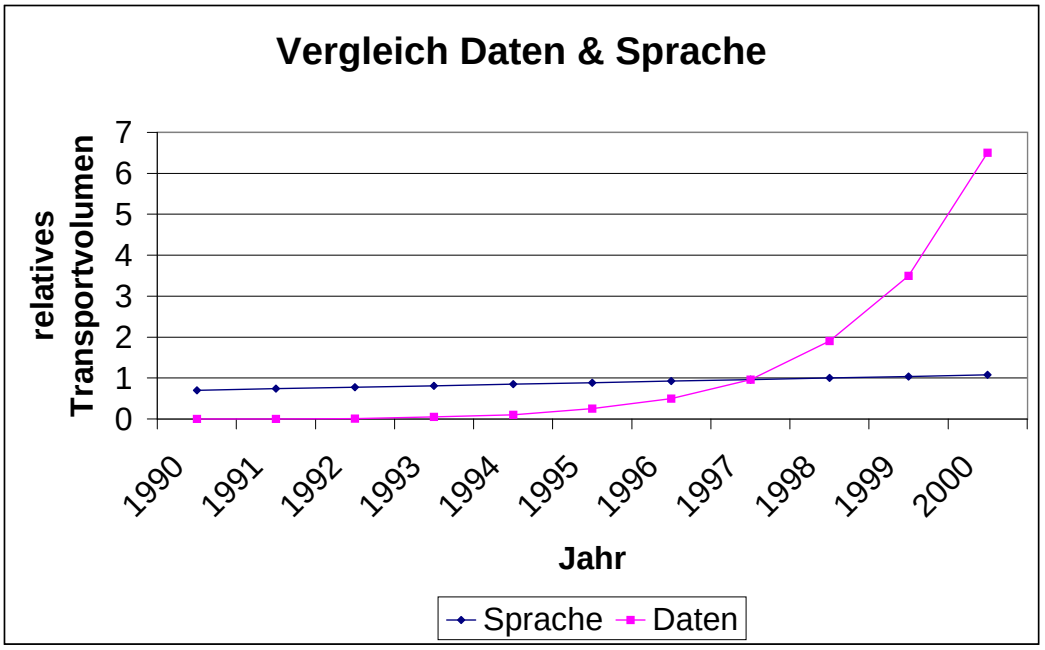

Abbildung 1: Sprache vs. Daten

> „Viele Großanwender wollen für ihre Firmennetze, in denen letztlich ebenfalls Konvergenz angestrebt wird, am liebsten einen Hauptlieferanten für deren sogenannte *End-toEnd*- Lösungen."

Bedingt durch diese Erkenntnis ist seit geraumer Zeit ein Großeinkauf in der Telekommunikationsbranche im Gange. Hier nur zwei Beispiele:

- Worldcom (Telefonanbieter aber auch Besitzer von 20000 Meilen Glasfaser-Backbone in den USA) hat für 37 Milliarden $ MCI gekauft. Diese Firma verfügte ebenfalls über ein großes Backbone (25000 Meilen). (Firmenangaben)
- In etwas kleineren Dimensionen ging der Zusammenschluß von Nortel (Northern Telecom) und Bay Networks vonstatten. Nortel, traditioneller Lieferant von Telefon- und Weitverkehrsystemen mußte Aktien im Wert von 7 Milliarden $ tauschen, um sich den Spezialisten für Datenkommunikationsausrüstung einzuverleiben.

Doch liegt der wirtschaftliche Vorteil eines verschmolzenen Sprach- und Datennetzes nicht nur im Mehrangebot. Durch die einheitliche Struktur lassen sich die Kosten für das Management und Instandhaltung reduzieren. Dadurch kann auch der Gewinn größer ausfallen.

2.1.2 Konvergenz von Datennetzen und Fernsehen

Eine nicht unwichtige Entwicklung zeichnet sich auch im TV-Bereich ab. Bereits seit einigen Jahren bieten Fernsehsender zusätzliche Informationen zu Ihrem Programm in Form des

Videotextes an. Dies führt zu einer größeren Attraktivität und in letzter Konsequenz zu höheren Werbeeinahmen. Ist hier die Konvergenz vielleicht noch nicht ganz zu erkennen – schließlich wurde das herkömmliche Fernsehen nur um einen Dienst erweitert– so wird das ganze interessant, wenn man den umgekehrten Fall betrachtet:

Fernsehen im Internet. Mittlerweile verfügt nahezu jeder Sender über einen eigenen Webauftritt. Hier wird, ähnlich dem Videotext ein erweiterter Service geboten. Doch das ist nicht alles; unter „www.tagesschau.de“ z.B. sind bereits gesendete Beiträge abrufbar. Man hat also schon in Ansätzen eine Art interaktives Fernsehen. Spinnt man den Gedanken mit einem Blick auf die marktwirtschaftlichen Interessen der TV-Sender weiter, so kann man sich in Zukunft Web-TV mit einfachen Pay-per-View Angeboten vorstellen:

Einfach ein Click auf die Sendung, die man sehen möchte, Eingabe der persönlichen Kennung, und am Monatsende liegt die Rechnung im Briefkasten.

Weiterer Vorteil: Beliebige Sender sind erreichbar. Auch außerhalb des Heimatnetzes und ohne Kabel/Satellitenanschluß. Doch darf man bei aller Euphorie die großen Probleme von Pay-TV in Deutschland nicht außer acht lassen:

Bisherige Versuche diese Art des Fernsehens populär zu machen sind nahezu alle gescheitert. Als warnendes Beispiel steht steht allen voran das DF-1 Projekt der Kirch-Gruppe. Es fuhr über die gesamte Laufzeit nur Verluste ein und endete letztendlich in einem Zusammenschluß mit dem Konkurrenten Premiere (neuer Name: Premiere-World). Der wirtschaftliche Mißerfolg ist aber nicht das einzige Problem von Pay–TV bzw. Pay-per-View. Die Akzeptanz für derartige Projekte ist in Deutschland noch nicht besonders groß. Wichtige Sportveranstaltungen wie z.B. Fußball-Weltmeisterschaften, die hohe Einschaltquoten und damit große Einnahmen ermöglichen würden, können auf Grund großer Widerstände von Seiten der Bevölkerung nicht im bezahlten Fernsehen ausgestrahlt werden. Fußball wird als eine Art Allgemeingut angesehen für das nicht gesondert bezahlt werden darf.

2.2 Technische Voraussetzungen

Bei allen ökonomischen Vorteilen, die eine Konvergenz von Netzen bringen kann, darf man aber nicht vergessen, daß gewisse technische Voraussetzungen gegeben sein müssen:

- Das neu entstandene Konvergenzprodukt darf nicht oder nur unwesentlich komlpizierter sein als die ursprünglichen Systeme. Das muß sowohl für die Anwender- als auch die Betreiberseite gelten.
- Die Qualität der neuen Dienste muß mindestens die gleiche wie bei den Vorgängern sein.
- Erst entsprechende Komprimierungsverfahren ermöglichen den effizienten Einsatz von Sprache und Video in IP-Netzen. (Nähere Details dazu finden sich in Abschnitt 4)

2.3 Zusammenfassung von 2.1 und 2.2

Faßt man die beiden vorherigen Abschnitte zusammen, so ist ein Zusammenhang besonders wichtig:

Eine konvergente Entwicklung wird sich nur durchsetzen, wenn die technischen Voraussetzungen dafür geschaffen sind und gleichzeitig genügend Marktkraft vorhanden ist, um einen wirtschaftlichen Erfolg zu erzielen. Man könnte auch sagen die technische Machbarkeit ist

das *notwendige*-, das ökonomische Interesse das *hinreichende Kriterium* für eine erfolgreiche Konvergenz.

Diese Erkenntis ist nicht neu, man sollte sie sich aber immer dann in Erinnerung rufen, wenn es darum geht Verschmelzungsprozesse bzw. ihre Zukunftschancen einzuschätzen. Dazu zwei Beispiele:

- Vor ca. einem Jahr verkündete die Firma Mobilcom, sie werde als erster Internet-Provider überhaupt in Deutschland einen Flatrate-Zugang anbieten. Das Ergebnis fiel sehr ernüchternd aus. Die Server waren hoffnungslos überlastet, man hatte sich total überschätzt. Nun ist dies zwar kein Beispiel für einen Konvergenzprozess, es zeigt aber, daß bei neu gestarteten Projekten eine sorgfältige Verbereitung und das Abwägen aller Eventualitäten wichtig ist. Man sollte nicht bewährte Konzepte wie z.B das **PSTN** (Public Switched Telephone Network) zugunsten eines vermeintlich besseren Internet mit allen Schikanen sofort über Bord werfen.
- Als **ATM** (Asynchronus Transfer Mode) neu entwickelt wurde, war der Jubel anfänglich sehr groß. Man glaubte die letzte Hürde in Richtung Universalnetz genommen zu haben. Daß dem nicht so war zeigt sich heute. Die Internetgemeinde nahm **ATM** nicht an und bis heute wird es trotz aller technischen Brillianz fast ausschließlich im Backbone-Bereich eingesetzt.

3 Begriffsdefinitionen

Dieser Abschnitt befasst sich mit den Begriffen, die bei (technischen) Konvergenzprozessen auftreten. [Eber99a] Zuerst wird eine Definition für technische Konvergenz selbst gegeben, danach werden die Begriffe Kollision, Substitution und Integration erklärt, allesamt Effekte die bei der Konvergenz von Netzen auftreten können.

Konvergenz: Ist der Prozess bei dem Technologien für die Lieferung von Diensten sich zu einm einzigen vereinigen. (RealNetworks Inc., 1990)

Kollision: Zur Kollision kann es kommen, wenn zwei Technologien konvergieren aber gleichzeitig auch konkurrieren (z.B. verbindunsorientiert vs. verbindungslos). Konvergenz kann zu Kollisionen führen. Oft sind diese hilfreich, da sie häufig einen fördernden Einfluss auf die Entwicklung von Innovationsprozessen haben.

Substitution: Bei der Substitution werden alte Technologien komplett durch neue ersetzt. (z.B. **BTX** durch **WWW**, oder, gerade im Gange: **Fax** durch **Email**)

Integration: Kommt es zur Integration, dann verschmelzen zwei Technologien zu einer einzigen, und sind von außen nicht mehr unterscheidbar. (z.B. Computer-Telefon-Integration)

4 Aktuelle Entwicklungen und Probleme

In Abschnitt 2 wurde gezeigt, daß ein Verschmelzen von Sprach- und Datennetzen ökonomisch durchaus sinnvoll sein kann, daß aber auch bestimmte technische Voraussetzungen erfüllt sein müssen. Dieser Abschnitt beschreibt mit Voice over IP, kurz VoIP, einen Ansatz für ein solches Verschmelzen. In 4.1 geht es dabei um die grundlegende Idee von **VoIP** und die bei der Umsetzung zu überwindenden Probleme. 4.2 behandelt dann unterschiedliche Konzepte, um das Hauptproblem von **VoIP**, die Garantie einer gewissen Dienstgüte, zu lösen.

4.1 VoIP

Unter **VoIP** (Voice over IP) versteht man im Allgemeinen die Sprachübertragung in IP-Netzen. Die Entwicklung der zugrundeliegende Technologie wird auf breiter Front vorangetrieben. Die Gründe hierfür liegen auf der Hand. Die Technik bietet Internet-Providern die Möglichkeit, ohne große Investitionsausgaben neue Geschäftsfelder zu erschließen, etablierte Telekommunikationsfirmen hingegen können zusätzliche Dienste anbieten. (Vergleiche Abschnitt 2.1)

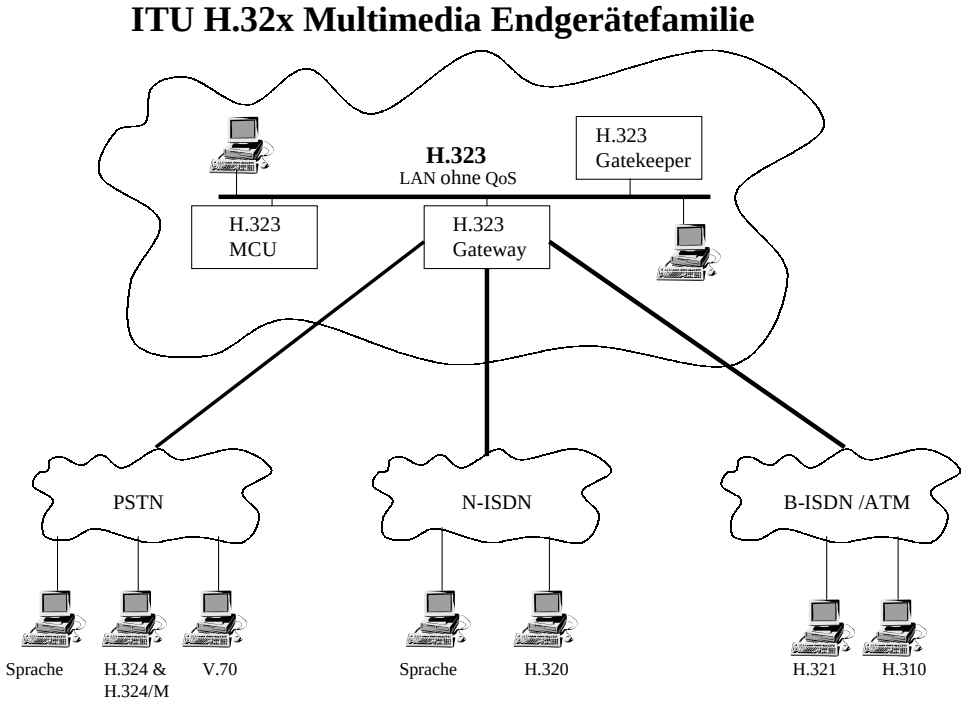

Abbildung 2: H.32x Quelle: 3Com/Computer TELEPHONY

Die Voraussetzungen für eine Verschmelzung von Sprach- und Datennetzen sind scheinbar geschaffen:

- Mit dem Internet existiert ein weltumspannendes IP-Netz
- Endgeräte, z.B. IP-Telefone, existieren und sind auch auf dem Markt erhältlich
- die **H.323**-Empfehlung der **ITU** (International Telecommunications Union) definiert eine Schnittstelle zwischen herkömmlichen Telefon- und den neuen IP-Netzen. Sie berücksichtigt Sprach- aber auch multimediale Dienste. (Siehe auch Abb. 2)
- Bandbreite im Backbone-Bereich ist durch die Glasfsertechnik ausreichend vorhanden

Trotzdem gehen die Meinungen über den Zeitraum der Verschmelzung weit auseinander. Tom Evslin, ehem. Präsident von AT&T's WorldNet Services im Juli 1997:

> „Ich sage voraus, daß es in fünf Jahren im Carrierbereich kein getrenntes Netzwerk für Telefonie– und IP-Daten mehr geben wird. Ein IP Netzwerk mit fortgeschrittenen Dienstmerkmalen im Bereich Sicherheit und Servicequalität –für private und geschäftliche Nutzung– stellt die Basis für alle Dienstangebote dar...“

Die „Neue Züricher Zeitung“ [Weis98] hingegen schreibt:

> „Umstritten ist unter Kennern der Szene das „Wann“ der Konvergenz. Während das eine Lager vom Ende des **PSTN** in vier, fünf Jahren ausgeht, sprechen andere von weiteren 20 Jahren mit **PSTN**. Die Mehrzahl der Netzwerkanalysten geht von einem langfristigen Trend aus. Etwa 21 Prozent aller internationalen Gespräche würden im Jahr 2003 über das Internet abgewickelt schätzt beispielsweise eine Studie des Londoner Beratungshauses Schema.“

Diese vorsichtige Einschätzung hat verschiedene Gründe:

- Das **Internet Protocol** arbeitet *paketvermittelnd* und nicht *verbindungsorientiert*. Dies hat zur Folge, daß die Sprachdaten weder *reihenfolgetreu* noch *zeitsychron* ankommen. Gerade aber ersteres ist für eine verständliche Sprachübertragung unerläßlich.
- Die Vermittlung im Internet erfolgt nach dem **„Best-Effort“** Prinzip, es gibt kein **QoS**. Vergleiche dazu Abschnitt 4.2.
- Durch die Liberalisierung des Telefonmarktes in Deutschland gibt es für den Endanwender fast keinen finanziellen Anreiz mehr auf Internet-Telefonie umzusteigen.

4.1.1 Internet-Telefonie und die Bandbreite

Die Breite des Datenstroms, in dem Sprache verpackt wird, hängt vom sogenannten *codec* (Codierer/Decodierer) ab. Die Standards **G.711- G.729** arbeiten mit unterschiedlichen Kompressionsraten. Sie sind in Abb. 3 aufgeführt und erläutert. Da das TCP/IP-Protokoll die Sprachdaten nicht kontinuierlich überträgt, sondern zu Paketen verarbeitet, kommt es in Kombination mit der Kodierung zu einem minimalen *Delay* (= die Verzögerung mit der die Sprache beim Empfänger ankommt).

Verzögerungen, die kleiner als 30 Millisekunden sind, nimmt das menschliche Ohr nicht wahr. Verzögerungen unter 100 Millisekunden werden noch nicht als störend empfunden. Im (öffentlichen) Internet gelten alledings derzeit Verzögerungen von einer Sekunde als tolerierbar, alles darunter ist gut. [Koss99] Will man also ein Verschmelzen von Sprach- und Datennetz auf globaler Ebene realisieren, dann ist eine Ausbesserung dieses Problems unumgänglich. Es müssen sogenannte **QoS** (Quality of Service, siehe Abschnitt 4.2) Mechanismen implementiert werden. In firmeninternen Netzen ist das Problem meist nicht vorhanden. Es steht fast immer genug Bandbreite zur Verfügung. In diesem Bereich sind die Erfolgschancen für IP-basiertes Telefonieren auch am größten, da ein gehöriges Einsparpotential besteht:

Bei der Neuinstallation müssen keine zwei Netzwerke eingerichtet werden, was kurzfristig hohe Investitionsausgaben schmälert und langfristig die Unterhaltungskosten gering hält.

Für weitere Informationen zum Thema Internet-Telefonie vergleiche mit [Kuri99] und [Koss99];

Kodierungsverfahren für Sprache			
Kodierung	**Erforderliche Bandbreite pro Gespräch**	**minimales Delay**	**Bemerkungen**
G.711A	64 kBit/s	5 ms	Ohne Komprimierung, beste Sprachqualität, deutsches Verfahren zur Tondigitalisierung (ISDN)
G.711U	64 kBit/s	5 ms	wie G.711A, aber US-Verfahren zur Tondigitalisierung
G.723-53	5,3 kBit/s	30 ms	Sprachqualität geringfügig schlechter als bei analogem Telefon
G.723-63	6,3 kBit/s	30 ms	Sprachqualität entspricht etwa analogem Telefon
G.729A	8 kBit/s	10 ms	Beste Sprachqualität der komprimierenden Verfahren, geringes Delay

Abbildung 3: Kodierungsverfahren

4.2 Exkurs: QoS

Abschnitt 4.1 zeigt, daß ein globales Verschmelzen des Telefonnetzes mit dem Internet, ohne die Garantie von **QoS** (Quality of Service) nicht möglich ist. Die Abschnitte 4.2.1 - 4.2.4 stellen unterschiedliche Konzepte vor, die genau das gewährleisten sollen.

QoS heißt übersetzt Dienstgüte. Es geht bei **QoS** um spezielle Parameter (maximale Verzögerung, minimale Bandbreite, maximaler Paketverlust, etc. . .), die man versucht bei einer Datenübertragung einzuhalten. Wie in Abschnitt 4.1.1 erwähnt ist die Verwirklichung von **QoS** im Internet ein aktuelles Problem und es gibt dazu verschiedene Ansätze:

4.2.1 ATM

ATM (Asynchronous Transfer Mode) ist ein *verbindungsorientierter* Transfermodus, der absolute **QoS** garantiert. Das ist möglich, da **ATM** während des Verbindunsaufbaus Dienstgüteparameter (Zellverlustrate, Zellverzögerung und deren Schwankung, etc) vereinbart werden, die jeder Knoten (**ATM-Switch**) einhält. Bei **ATM** werden die zu übertragenden Daten in 53 Byte große Pakete zerlegt, die **ATM-Zellen**. Wegen der geringen Größe der Pakete eignet sich **ATM** hervoragend zur Sprachübertragung.

> „Stehen unterschiedliche Daten zur Übertragung an, so benutzt **ATM** einen statistischen Zeitmultiplex-Mechanismus und erzeugt einen konstanten Zellstrom. Die Asynchronität bezieht sich auch nur auf diese Multiplexen und sagt nichts über das darunterliegende Medium aus.“ (Aus Lehr- und Übungsbuch Telematik [Dive00])

Will man nun IP-Pakete über ein ATM-Netz verschicken, so muß zwecks Anpassung der Paketgröße eine Zwischenschicht, die **AAL** eingefügt werden. Sie übernimmt unterschiedliche Funktionen (Nach ITU-Empfehlung I.362):

- Behandlung von Übertragungsfehlern, weil die Prüfsumme im Header nur den Zellenkopf testet.
- Die Segmentierung und und Wiederherstellung der Nutzdaten. ATM-Zellen haben einen Nutzdatenanteil von 48 Byte.
- Verlorengegangene Zellen wiederherstellen. Das ist notwendig, da die kleinen Blöcke unter Umständen einem großen Nutzdatenblock entstammen können. Dieser wäre andernfalls unbrauchbar.

Abbildung 4 zeigt den entstandenen Schichtaufbau. Wie im Bild zu erkennen, sitzen **AAL** und **ATM** zwischen Schicht 1 und 3. Beide übernehmen auch typische Schicht 2 Aufgaben wie Flußsteuerung und Übertragungssicherung.

Die von der **ITU-T** einmal vorgesehenen Dienstklassen stellten sich als zu komplex und redundant heraus. Durchgesetzt haben sich die folgenden Vorschläge des ATM-Forums, einer weiteren Standardisierungsorganisation, die teilweise in Konkurrenz zur **ITU-T** steht (nach [Dive00]):

CBR: (Constant Bit Rate) Hat konstante Bitrate und ermöglicht so einen Übergang zum gewöhnlichen Telefonnetz.

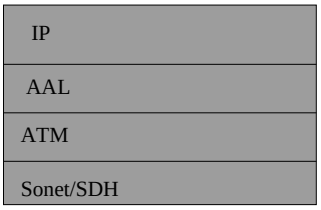

Abbildung 4: ATM im Schichtenmodell

RT-VBR: (Real Time Variable Bitrate) Für Anwendungen mit variabler Bitrate sind die VBR-Dienstklassen zuständig. RT-VBR garantiert eine zusätzlich maximale Verzögerungszeit. Ein Anwendungsbeispiel sind Videokonfernzen.

NRT-VBR: (Non Real time Variable Bitrate) Ist die Anforderung an die zeitliche Verzögerung nicht ganz so strikt (z.B.) bei digitalem Fernsehen, dann nutzt man NRT-VBR.

ABR: (Available Bit Rate) Eine Dienstklasse mit geringen Anforderungen. Garantiert eine minimale Übertragungsrate, nutzt aber nur die verbliebenen Ressourcen. Ein Sender kann in diesem „Modus“ gedrosselt werden.

UBR: (Unspecified Bitrate) Diese Dienstklasse nutzt nur noch Restkapazitäten eines Mediums. Keine Garantien, ähnlich dem IP-Verkehr.

ATM konnte sich bisher nur in Weitverkehrsnetzen (Backbone-Bereich) durchsetzen. Für das Problem aus Abschnitt 4.1.1 stellt es also noch keine Lösung dar.

4.2.2 Differentiated Services

Differentiated Services ist der im Augenblick am meisten diskutierte Ansatz für **QoS** im Internet. Die Idee ist folgende:

Man versieht den IP-Header mit einer Kennung, die etwas über die Prioriät bzw. die Güteklasse des Paketes aussagt. Kommt nun ein Paket bei einem Router an, so prüft dieser erst die Kennung und entscheidet dann über das weitere Verfahren. Ist die Priorität hoch (z.B. entstammt das Packet einer Videokonferenz zwischen Arzt und OP) so wird sofort weitervermittelt. Andernfalls kommen die Dateneinheiten in eine Warteschlange oder werden gar verworfen.

Diese Technik ist für Internet-Provider sehr interessant. Sie können mit Ihren Nachbarn genau spezifizierte Durchlaßraten vereinbaren und so dem Kunden wenigstens eine relative **QoS** garantieren. Damit läßt sich viel Geld verdienen, denn für garantierte Dienstqualitäten lassen sich höhere Einnahmen erzielen als für den herkömmlichen Internet-Zugang.

Um aber hohe Dienstgüte zu garantieren, sind Zugangskonntrolle und Resourcen-Reservierung unumgänglich. Wichtig für *services on demand* (Dienste auf Anfrage) hingegen ist das verschicken von Nachrichten die eine bestimmte Bandbreite anfordern. Für diese Zwecke wurde

das Konzept des *Bandwidth Broker* (BB) ersonnen. Der BB ist ein spezieller Knoten in einem Netz mit implementierten **Differentiated Services**. Er kümmert sich um die o.g. Aufgaben und könnte außerdem die Abrechnung der geleisteten Dienste übernehmen. (nach [Dive99a])

Eine Herausforderung für die IT-Ingenieure ist dann, die Router mit dieser Technik auszustatten und den reibungslosen Ablauf zu garantieren. Denn ein Versagen der Dienstgüte, sprich eine Verstopfung im Netz, führt zwangsläufig zu Regressansprüchen auf der Kundenseite.

4.2.3 QoS durch „unendlich“ Bandbreite

Die gedanklich einfachste Methode **QoS** zu garantieren, wäre einfach nahezu unendlich Bandbreite zur Verfügung zu stellen. Könnte man das ermöglichen, bräuchte man sich um nicht mehr viel kümmern. Alle Pakete würden schnell und in bewährter IP-Manier weitergeleitet. Obwohl gewisse Parallelen in der PC-Entwicklung erkennber sind, hier werden seit Jahren die Rechner aufgerüstet und nicht an der Effizienz der Software gefeilt, hat diese Idee vorerst wohl kaum Chancen. Es ist einfach zu teuer, zu jedem Haushalt eine Glasfaser zu legen.

4.2.4 MPLS

Zum Schluß soll an dieser Stelle noch ein sehr aktuelles Thema angeschnitten werden. **MPLS** (= Multiprotocol Label Switching) hat derzeit den Status *Draft* (siehe auch [Dive99b]), was der erste Schritt auf der RFC-Leiter in Richtung Standard ist. Dabei handelt es sich um eine Technik, die entwickelt wurde (wird), um das Routen in paketvermittelnden Netzen zu beschleunigen.

In seinem Wesen ähnelt **MPLS** der **ATM**-Technologie. Es arbeitet ebenfalls mit virtuellen Verbindungen und benutzt zum Weiterleiten kleine Felder im Kopf der weierzuleitenden Datagramme, die sogenannten „label“. Das bringt einen entscheidenden Vorteil mit sich:

Weil Label kürzer als IP-Adressen sind, erreicht man eine höhere Durchsatzrate. [Huit00] Das allein rechtfertigt aber noch nicht die Existenz von **MPLS**. Etwas anderes wird den (wahrscheinlich) zu erwartenden Erfolg von **MPLS** ausmachen: Die Tatsache, daß es einfach in gewöhnliches IP-Routing eingebunden werden kann.

> „Die Idee dabei ist, daß man Switching- und Routing-Funktionalität in eine einzelne Komponente, den Integrated Switch Router (ISR) integriert.“ (nach [Brau99])

Der ISR entscheidet dann darüber, ob ein ankommendes IP-Paket geswitched oder geroutet wird. Da die zugrundeliegende Switching-Technologie meistens der von **ATM** entspricht, können die gleichen Vorteile in Bezug auf **QoS** genutzt werden (siehe Abschnitt 4.2.1). Ein weiterer, wichtiger Aspekt ist, daß die IP-Router die Switches kontrollieren. Sie nutzen die durch Routing- und Kontrollprotokolle eingehenden Informationen um die darunterliegenden Switches zu konfigurieren. Dies kann z.B. über SNMP (Simple Network Management Protocol) realisiert werden.

Abbildung 5 zeigt zwei ISRs und drei Datenströme. Der Strom von Quelle 1 zu Ziel 1 nimmt den „herkömmlichen“ IP-Weg durch die Router. Die beiden anderen (mit identischem Ziel) werden durch die Switches als „switchbar“ erkannt und entsprechend weitergeleitet. Damit das passieren kann müssen die Switches aber vorher durch die Router konfiguriert worden sein.

Ist **ATM** noch an seiner Komplexität und dem Widerstand der „Internet-Community“ gescheitert, so könnte **MPLS**, das auch **IP-Switching** (Nomen est Omen) genannt wird, den Durchbruch schaffen. Damit wäre eine wichtige Hürde in Richtung **QoS** im Internet genommen und der Weg frei für neue gewinnbringende Dienste.

5 Zusammenfasung und Ausblick

Es hat sich gezeigt, daß „Konvergenz von Netzen“ ein sehr vielseitiges Thema ist. Die Konvergenz kann auf vielen Ebenen stattfinden und es gibt unzählige Beispiele die hier keine Berücksichtigung gefunden haben. Ein Konvergenzprozess mit recht tiefgreifenden Auswirkungen ist dafür aber sehr ausführlich untersucht worden: Die Konvergenz bzw. Verschmelzung von Sprach- und Datennetzen. An ihm wird deutlich was für alle Netzkonvergenzen gilt. Daß Konvergenz fast immer durch ökonomische Interessen vorangetrieben wird. Daß sie häufig große technische Herausforderungen mit sich bringt. Und daß nicht selten außer der Vereinigung zweier alter bzw. der Verdrängung eines Einzelnen ganz neue Dienste entstehen können, die den Leistungsumfang der „neuen alten“ Netze erheblich vergrößern.

Es bleibt abzuwarten wie sich der traditionelle Telefonmarkt und der rasant wachsende Datenmarkt weiter entwickeln. Wann beide endgültig zu einem verschmelzen ist derzeit noch nicht zu erkennen. Daß es passiert scheint aber aus heutiger Sicht absolut sicher.

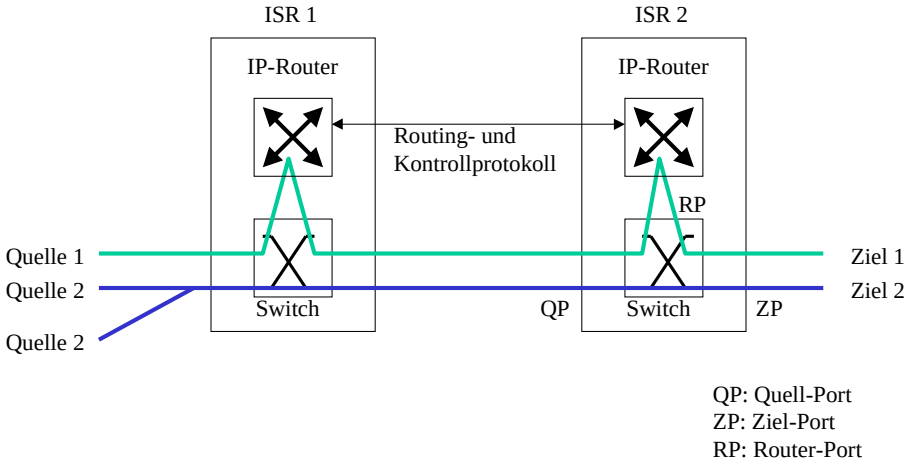

Abbildung 5: Integrated Switch Router

Literatur

[AG99] Deutsche Telekom AG. Quartalsbericht III/99. Technischer Bericht, Deutsche Telekom AG, 1999.

[Brau99] Torsten Braun. *IPnG: Neue Internet-Dienste und Virtuelle Netze*. dpunkt.verlag. 1999.

[Char99] Dr.-Ing. Joachim Charzinski. Verkehrsaspekte im Internet. In *Konvergenz der Netze*, 1999.

[Dive99a] Diverse. DiffServ in the Web. Technischer Bericht, Institut für Telematik, Universität Karlsruhe, 1999.

[Dive99b] Diverse. Framework for IP Multicast in MPLS. Technischer Bericht, MPLS Working Group Internet Draft, 1999.

[Dive00] Diverse. *Lehr- und Übungsbuch TELEMATIK*. Gerhard Krüger, Dietrich Reschke. 2000.

[Eber99a] Jörg Eberspächer. Wächst Zusammen, was (Nicht) Zusammengehört. In *Konvergenz der Netze*, 1999.

[Eber99b] Markus Eberspächer. Ansatz Zu Einer Verallgemeinerten Netzarchitektur Für Integrierte Dienste. In *Konvergenz der Netze*, 1999.

[Fisc99] Wolfgang Fischer. Next Generation Networking. In *Konvergenz der Netze*, 1999.

[Huit00] Christian Huitema. *Routing in the Internet*. Prentice Hall PTR. 2. Auflage, 2000.

[Koss99] Axel Kossel. Netzgespräche. *c't magazin*, 1999, S. 230–234.

[Kuri99] Jürgen Kuri. Sprache in Päckchen. *c't magazin*, 1999, S. 220–209.

[Weis98] Manfred Weise. Ein Netzwerk und ein Protokoll Für Alle Fälle. *Neue Züricher Zeitung*, Dezember 1998.