

Messung von Benutzerzugriffszeiten auf Internetseiten
entfernter Rechner

Studienarbeit im Fach Informatik

Vorgelegt von:
Wolfgang Wiese

Angefertigt am:
Institut für Mathematische Maschinen und Datenverarbeitung der
Friedrich-Alexander-Universität Erlangen-Nürnberg Lehrstuhl für
Rechnerarchitektur und -Verkehrstheorie

Hochschullehrer: Prof. Dr. Ing. U. Herzog
Betreuer: Dr. Ing. W. Dulz

Erlangen, Mai 1998

Erklärung

Ich versichere, daß ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und daß die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Erlangen, den 27. Mai 1998

Wolfgang Wiese
Karlsbader Str. 3
91058 Erlangen

E-Mail: xwolf@xwolf.com

URL: <http://www.xwolf.com/>

Übersicht

Lange Zeit galt das WWW („*World Wide Web*„) als langsam und wurde spöttisch als „*World Wide Wait*„ bezeichnet. Bei Zugriffen auf Internetseiten kam es zu langen Wartezeiten. Und auch nachdem im Laufe der Zeit die Zahl der Verbindungen zwischen den weltweiten Informationsnetzen stark zugenommen hat, ist Geduld immer noch ein Gebot, wenn man auf Seiten entfernter Rechner zugreifen will.

Die Studienarbeit „*Messung von Benutzerzugriffszeiten auf Internetseiten entfernter Rechner*“, im folgendem kurz „*Internet Speed*“ genannt, nimmt sich dieser Problematik an. Sie versucht nicht, die bereits bekannten Gründe für die schlechten Übertragungsraten zu untersuchen; Das Ziel ist es vielmehr, eine möglichst objektive Aussage darüber zu machen, ob und wo das Netz (das Internet) langsam ist und welche Rechner betroffen sind. Dabei wird diese Arbeit es zum erstenmal ermöglichen, ein Maß über die aktuelle, mittlere Geschwindigkeit des Internets zu geben.

Die Messungen und Ergebnisse werden dabei automatisch so aufbereitet, daß sie aus dem Netz abgerufen werden können. Es wird während der gesamten Arbeit bezug auf die Seiten genommen, die sich im Netz unter folgender URL abrufen lassen: <http://speed.xbaer.com/>.

Inhaltsverzeichnis

ERKLÄRUNG.....	2
ÜBERSICHT.....	3
INHALTSVERZEICHNIS.....	4
ABBILDUNGS- UND TABELLENVERZEICHNIS.....	6
1. EINLEITUNG.....	7
1.1 MOTIVATION UND ZIELSETZUNG.....	7
1.2 AUFGABENSTELLUNG UND ÜBERBLICK.....	8
2. THEORETISCHER ÜBERBLICK.....	10
2.1 GESCHICHTE DES INTERNETS	10
2.2 INTERNET-PROTOKOLLE UND -TOOLS.....	12
2.2.1 <i>Einleitung</i>	12
2.2.2 <i>PING und TRACEROUTE</i>	14
2.2.3 <i>TELNET, FTP, SMTP und FINGER</i>	16
2.2.4 <i>HTTP</i>	19
2.3 MATHEMATISCHE GRUNDLAGEN.....	21
2.3.1 <i>Einleitung</i>	21
2.3.2 <i>Mittelwert</i>	22
2.3.3 <i>Varianz (Standardabweichung)</i>	22
2.3.4 <i>Mittlerer quadratischer Fehler</i>	22
2.3.5 <i>Fehlerfortpflanzung</i>	23
2.4 PROGRAMMIERTECHNISCHE GRUNDLAGEN.....	24
2.4.1 <i>Einleitung</i>	24
2.4.2 <i>Shell-Skripte</i>	24
2.4.3 <i>CRONTAB-Einträge</i>	24
2.4.4 <i>Perl</i>	26
3. LÖSUNGSANSÄTZE.....	27
3.1 EINLEITUNG.....	27
3.2 ANSATZ DER „C'T“.....	27
3.3 ANSATZ VOM „INTERNET WEATHER“	29
3.4 ANSATZ DIESER ARBEIT.....	31
<i>Mittel und Fehler</i>	34
4. UMSETZUNG.....	35
4.1 EINLEITUNG.....	35
4.2 KONZEPTIONELLE UMSETZUNG UND PLANUNG.....	36

4.2.1 Die Lokale Meßstation.....	36
4.2.2 Die Zentrale Sammelstelle.....	37
4.2.3 Die Verarbeitung der Daten und Berechnung der Statistik.....	37
4.2.4 Die Online-Darstellung der Daten.....	38
4.3 PROGRAMMIERTECHNISCHE UMSETZUNG.....	39
4.4 TESTPHASEN.....	39
4.5 ORGANISATORISCHE UND ADMINISTRATIVE UMSETZUNG.....	40
5. VORSTELLUNG DER WEBSEITE UND FALLBEISPIELE.....	42
5.1 EINLEITUNG.....	42
5.2 VORSTELLUNG DER WEBSEITE.....	43
5.2.1 Die Startseite.....	43
5.2.2 Die aktuelle Messung.....	44
5.2.3 Die Tages-, Monats- und Jahresübersicht.....	45
5.2.4 Die Host-Liste.....	46
5.2.5 Die Suche nach Hosts.....	48
5.2.6 Die Liste der Stationen.....	50
5.2.7 Sonstiges.....	50
5.3 FALLBEISPIEL 1: AOL IM MONAT MAL.....	51
5.4 FALLBEISPIEL 2: COMPU SERVE DEUTSCHLAND UND AOL.....	52
5.5 FALLBEISPIEL 3: METRONET.....	53
5.6 FALLBEISPIEL 4: DIE 5 GROSSEN PROVIDER IM TAGESVERGLEICH.....	54
5.7 FALLBEISPIEL 5: DIE 5 GROSSEN PROVIDER IM MONATSVERGLEICH.....	55
6. ZUSAMMENFASSUNG UND AUSBLICK.....	56
ANHANG A	58
I. PRESSEMITTEILUNG VOM 23. MÄRZ 1998.....	58
II. FALTBLATT ZUM VORTRAG „PROVIDER IM VERGLEICH“, 19. UND 20. JUNI 1998.....	59
III. NETZAUSBAU VOM NETZPROVIDER XLINK 1998.....	61
LITERATUR- UND ADRESSENVERZEICHNIS.....	62
I. LITERATUR.....	62
II. INTERNETADRESSEN.....	62
III. RFC-LISTE.....	63

Abbildungs- und Tabellenverzeichnis

TABELLE 1. GEGENÜBERSTELLUNG OSI- UND TCP/IP-SCHICHTEN.....	13
TABELLE 2. TCP/IP-PROTOKOLLE	13
BILD 1. PING.....	15
BILD 2. TRACEROUTE.....	16
BILD 3. TELNET-SITZUNG.....	17
BILD 4. FTP-SITZUNG.....	18
BILD 5. FINGER.....	19
BILD 6. HTTP-VERBINDUNG MITTELS TELNET.....	20
TABELLE 3. HTTP-REQUESTMETHODEN.....	21
BILD 7. CRONTAB.....	25
BILD 8. ANYSPEED MESSUNG.....	28
BILD 9. INTERNET WEATHER.....	30
BILD 10. VERGLEICH "PING"- UND HTTP-REQUESTS.....	33
TABELLE 4. VERGLEICH "PING"- UND HTTP-REQUESTS.....	34
BILD 11. SCHEMA „LOKALE MESSSTATION“.....	36
BILD 12. STARTSEITE.....	43
BILD 13. STATISTIK DER JÜNGSTEN DATENERFASSUNG.....	44
BILD 14. TAGESÜBERSICHT.....	45
BILD 15. HOST-LISTE.....	46
BILD 16. DATEN EINES HOSTS.....	47
BILD 17. SUCHE.....	48
BILD 18. DATEN MEHRERER HOSTS.....	49
BILD 19. STATIONEN.....	50
BILD 20. AOL IM MONAT MAL.....	51
BILD 21. COMPU SERVE UND AOL IM TAGESVERGLEICH.....	52
BILD 22. TAGESÜBERSICHT METRONET.....	53
BILD 23. DIE 5 GROSSEN PROVIDER IM TAGESVERGLEICH.....	54
BILD 24. DIE 5 GROSSEN PROVIDER ANFANG JUNI.....	55
TABELLE 5. RFC-LISTE	63

1. Einleitung

1.1 Motivation und Zielsetzung

Die breite Nutzung des Internets, und dabei insbesondere der immer intensiver werdende Gebrauch des World Wide Webs (kurz: WWW), führte in den vergangenen Monaten und Jahren zu einer spürbaren Zunahme des Verkehrs auf den Datennetzen.

Zuerst nur für eine Kommunikation zwischen Universitäten und staatlichen Einrichtungen gedacht, hat sich das Internet zu einem weltweiten Informationsnetz entwickelt, welches für Anwendungen aus allen Bereichen des menschlichen Lebens genutzt wird. Der Ausbau der Datenverbindungen ist deswegen ein Gebiet mit großen und noch immer spürbaren Auswirkungen. So sorgte zum Beispiel die Fertigstellung eines neuen Transatlantikkabels im vergangenen Jahr kurzfristig zu einer Halbierung der Wartezeiten beim Zugriff auf Internet-Seiten in Amerika.

Der Ausbau dieser Datenverbindungen wird zur Zeit hauptsächlich unter kommerziellen Gesichtspunkten vorangetrieben. Die Bedeutung der Uni-Netze, welche in Deutschland durch das DFN administriert werden, und von denen noch vor wenigen Jahren noch der Netzausbau ausging, nimmt in Vergleich mit den kommerziellen Netzen ab. So werden neue Datennetze zur Zeit nur noch von Firmen oder Organisationen wie Netsurf (IS), Xlink, Nacamar oder der Telekom verlegt, während das DFN sich naturgemäß nur auf die Verbindungen zwischen Universitäten und Forschungseinrichtungen beschränkt¹.

Obwohl durch die neuen kommerziellen Netze viele Leistungsengpässe überwunden wurden, bedeutete dies nicht zugleich eine Entlastung des Internets: Durch die kommerziellen Netzbetreiber gelangten große Bevölkerungsteile in den Genuß des nunmehr günstigen Internets, so daß die Anzahl der Benutzer und damit der Datenverkehr weiter stieg und steigt. Ungeklärt bleibt dabei oft das Verhältnis zwischen vorhandenen Datennetzen, benötigten Datenraten und den Betriebskosten: Steigt die Zahl der Nutzer und damit das Datenaufkommen, während zur selben Zeit die Übertragungskapazitäten gleich bleiben, kommt es zu Wartezeiten; Besitzt ein Netzdienst jedoch zu wenig Kunden wird dieser unrentabel. Neben dieser rein wirtschaftlichen Überlegung sind jedoch auch technische Hintergründe zu bedenken, wie zum Beispiel die enge Verflechtungen der unterschiedlichen Netze untereinander und die tageszeitlich unterschiedliche Auslastung der jeweiligen Dienste. Große Bedeutung spielen auch die vielen verschiedenen Sub-Provider, die auf den Datennetzen der großen Netzbetreiber aufsitzen und eigene Einwahlpunkte anbieten.

¹ Es darf hier nicht der Eindruck entstehen, daß das DFN alleine selbst Leitungen verlegt. Der Großteil aller Leitungen des DFNs sind von der Telekom angemietet. Jedoch wurden diese Leitungen erst nach Auftrag der DFN verlegt.

Bei all diesen unterschiedlichen Einflüssen auf das Internet stellt sich daher die Frage, inwieweit sich dies insgesamt auf den einzelnen Benutzer auswirkt. Ebenso wichtig sind die Fragen danach, wo die Flaschenhälse eines Netzes sind, wie auch danach, wie sich ein Netz im Verlauf der Zeit in seinem Antwortverhalten ändert.

1.2 Aufgabenstellung und Überblick

Das Ziel dieser Arbeit ist es, die Schwachstellen in den Datennetzen zu finden und ein Maß über die gemittelte Geschwindigkeit des Zugriffes auf allgemeine Online-Angebote zu geben. Dabei sollen einzelne Rechner im Netz anhand ihres Antwortverhaltens miteinander vergleichbar gemacht und diskrete Werte über deren „Quality of Service“ (kurz: QoS) über unterschiedliche Zeiträume hinweg, gesammelt und ausgewertet werden.

Hierbei wird insbesondere Werte auf eine möglichst neutrale und objektive Messung gelegt, welche weitgehend automatisiert wird. Die Ergebnisse der Messungen werden online abrufbar sein unterhalb der Adresse <http://speed.xbaer.com>.

Die Arbeit wird auch nach dem offiziellen Ende der Studienarbeit in Eigenregie von Wolfgang Wiese als Projekt weiterentwickelt.

Im ersten Teil der Arbeit wird auf die theoretischen Hintergründe eingegangen, auf welche Weise eine solche Messung gemacht werden kann. Es wird dabei auch auf mögliche alternative Konzepte eingegangen, sowie eine Diskussion über die Vor- und Nachteile der hier verwendeten Verfahren gemacht. Da ein wichtiger Teil dieser Arbeit programmiertechnischer Natur ist, werden hier auch Grundlagen in den verwendeten Tools und der verwendeten Programmiersprache Perl besprochen. Weitere Schwerpunkte dieses Kapitels werden die nötigen mathematischen Formeln, sowie ein detaillierterer Blick auf die Netzgeschichte und die Providerdienste sein.

Im dritten Kapitel wird der Ansatz vorgestellt und diskutiert, welcher in dieser Arbeit benutzt wurde. Daneben wird der „State of the Art“ bei ähnlichen Projekten oder Arbeiten wie diesem angegeben.

Gegenstand von Kapitel 4 ist die konkrete Umsetzung des vorher diskutierten Verfahrens. Hierbei wird auch auf die nötigen Vorarbeiten eingegangen, wie die Administration des gesamten Projektes, sowie der nicht unwichtigen Öffentlichkeitsarbeit.

In Kapitel 5 wird die Webseite der Arbeit vorgestellt und besonders markante Fallbeispiele angegeben. So werden wir auf typische Verläufe in den Reaktionszeiten hinweisen.

Kapitel 6 wird einen kurzen Ausblick auf den möglichen weiteren Ausbau und der Entwicklung der Arbeit, sowie eine Zusammenfassung über die Arbeit geben.

Anhand A enthält die Abdrucke einige Faltblätter und Informationen, mit denen bereits Öffentlichkeitsarbeit um diese Arbeit gemacht wurde und noch wird.

Der Anhang B enthält den kommentierten Source-Code der erstellten Programme. Dieser wird als Extraband ausgegeben, da ansonsten das Format dieser Niederschrift gesprengt werden würde.

2. Theoretischer Überblick

2.1 Geschichte des Internets

Die Entwicklung des Internets begann Ende der fünfziger Jahre, als sich amerikanische Experten der Rand Corporation vor der Frage gestellt sahen, wie man ein militärisches Kommando- und Überwachungsnetzwerk entwickeln könnte, welches auch nach einem Atomkrieg wenigstens zum Teil funktionsfähig bleiben würde, so daß amerikanische Regierungsstellen weiterhin sicher und effektiv miteinander kommunizieren könnten.

Da eine zentrale Steuerung im Kriegsfall getroffen werden könnte, mußte das Netz ohne diese auskommen.

1964 veröffentlichte die Rand Corporation eine Lösung, welche auf den folgenden Prinzipien aufgebaut war:

- Alle Knoten des Netzes bekommen den gleichen Status,
- Alle Knoten haben eigene Autorität, um Nachrichten zu senden, weiterzuleiten oder zu empfangen,
- Nachrichten werden in einzelne Pakete gepackt, welche, mit einer eigenen Adresse versehen, ihren eigenen Weg durch das Netz gehen können.

Diese Technik verbreitete sich bis 1968 als allein amerikanisches Projekt bei einigen Universitäten (insbes. bei der MIT und der UCLA), um dann auch in Großbritannien an der NPL („National Physical Laboratory“) getestet zu werden.

1969 wurde das ARPANET („Advanced Research Project Agency Network“) ins Leben gerufen, welches zu diesem Zeitpunkt erst aus 4 Knoten bestand. Da sich das Militär aber mehr und mehr anderen Technologien zuwendete, wurde bald der Zugriff auf dieses Netz freigegeben auf Personen und Einrichtungen, welche sich nicht nur mit der militärischen Forschung befaßten. So wuchs das ARPANET schnell weiter: 1971 waren 15 Knoten, 1972 bereits 37 Knoten im ARPANET. Mit dem Ausbau des Netzes und der Verbesserung der Rechner wurde das alte Kommunikationsprotokoll NCP („Network Control Protocol“) bald vom besseren TCP/IP ersetzt.

Schnell wurde das ARPANET auch mit anderen Netzen, wie das ALOHA-Funknetz, UUCP, Usenet und vielen campusweiten Universitätsnetzen verbunden.

1983 wurde das Netz in einem militärischen Teil (MILNET) und einem forschungsorientierten Teil untergliedert (ARPANET). Die Verbindungen blieben jedoch bestehen und erlaubten eine Fortsetzung der Kommunikation. Sie wurden später „DARPA Internet“ genannt, woraus sich später der Name „Internet“ entwickelte, den man für die Verbindungsnetzwerke zwischen

Netzen benutzte. Ebenfalls 1983 wurde das Ethernet von DEC, INTEL und XEROX vorgestellt, welches ab der Schicht 3 des ISO/OSI-Referenzmodelles, die gleichen Protokolle wie das ARPANET verwenden konnte. Dies führte nochmals zu einem Entwicklungsschub, da damit Tausende Ethernet-LANs über das ARPANET verbunden werden konnten. Aufgrund dieser Entwicklungen wird das Jahr 1983 oft als der eigentliche Entstehungszeitpunkt des Internets angesehen.

1986 begann mit der Gründung des NSFNET („National Science Foundation Network“) ein neuer Abschnitt in der Entwicklung des Netzes: Es wurde ein domainorientiertes Adressierungskonzept eingeführt, welches das Netz in sechs Internet-Domains mit den Kennungen GOV, MIL, EDU, NET, ORG und COM aufspaltete.

Parallel hierzu wurden die (Haupt-)Leitungen zwischen den Netzen von anfangs 56kbps, über 1,5 Mbps, bis 1987 auf 45Mbps gebracht, was ebenfalls die Entwicklung von grafischen Oberflächen (z.B. X-Windows) unterstützte. 1993 begannen Experimente mit ATM welche Leitungsgeschwindigkeiten von 155 Mbps bzw. 622 Mbps ermöglicht. Das ARPANET endete formal 1989, als ein Großteil der Netze in das NSFNET übergegangen war.

1995 zog sich das NFS aus dem Betrieb des Internets zurück: Zu diesem Zeitpunkt war aus dem ehemaligen Forschungsnetz ein Netz geworden, welches nur noch zu einem kleinen Teil für Forschungen benutzt wurde. Das Internet ging in die Hand von meist kommerziellen Anbietern wie AOL, MCI oder US-Sprint. Der Backbone des Internets wird durch ein Netz von NAPs („Network Access Point“) gebildet, welche von den kommerziellen Anbietern betrieben werden. Dies spiegelt sich auch an den Nutzern des Internets wieder: Noch 1993/94 waren mehr als die Hälfte der Nutzer Studenten. Begünstigt durch die Einführung des WWWs sank diese Zahl in den folgenden Jahren erheblich, so daß in Amerika bereits über 90% der Benutzer Privatleute oder Firmen sind.

Es wird geschätzt, daß zur Zeit bereits jeder 10. Haushalt in den USA an das Internet angeschlossen ist. Seit 1992 verdoppelt sich jedes Jahr die Zahl der Hosts.

Der Erfolg des Internets liegt aber nicht nur in der Kommerzialisierung und der großen Zahl an Dienstleistungen, sondern insbesondere an seiner freiheitlichen Struktur. Kauffels bringt dies auf zwei Punkte:

„1. Das Internet ist kein Netz im eigentlichen Sinne, sondern ein Verbund zehntausender Einzelnetze mit –weitgehend- gemeinsamen Protokollen der Schichten 3 – 7.

2. Das Internet hat keinen eigentlichen Besitzer oder Betreiber, vielmehr ist es ein soziales Instrument zur Kommunikation, das deshalb auf die vielfältigsten Arten und Weisen genutzt werden kann.“

Damit ist die Entwicklung des Internets aber noch lange nicht abgeschlossen. In den USA wird zur Zeit das sogenannte „*Internet₂*“ aufgebaut, welches größere Bandbreiten und höhere Geschwindigkeiten verspricht. Auch geht der Ausbau der Bandbreiten bei bestehenden Verbindungen weiter voran.

Daneben wird über die Einführung eines neuen Namensraumes diskutiert, bei dem neue Top-Level-Domains eingeführt werden sollen. Hintergrund dieser Diskussion ist die Tatsache, daß der 1986 definierte Adressenbereich zu Neige geht und die bisherigen Domainnamen nicht mehr streng geführt werden. Eine Einführung der neuen Domainnamen (voraussichtlich WEB, ART, SHOP, HOME und FIRM) wird jedoch nicht vor 1999 erwartet.

Europa hinkt der Entwicklung in den USA leider etwas hinterher. So waren traditionell die wichtigen Netze allesamt zur Unterstützung von Wissenschaft und Forschung vorgesehen, während erst ab 1995 der Ausbau von kommerziellen Netzen durch Onlinedienste, wie MAZ, ECRC, IBM oder Xlink², begann. Die Wartezeiten und Übertragungsraten, welche diese Netze zur Verfügung stellen, führen aber immer noch zu herben Enttäuschungen. Ebenso sind die Kosten für die professionelle Internet-Nutzung in Europa noch sehr hoch, verglichen mit den USA.

Es ist noch nicht absehbar, wohin sich das Internet in Zukunft weiterentwickelt. Man kann aber jetzt schon feststellen, daß das Internet zu vielen neuen Entwicklungen geführt hat und durch die Möglichkeit der Kommunikation über alle Grenzen hinweg einen großen Einfluß auf die soziale Entwicklung der Menschen nimmt. Hier sei insbesondere das Schlagwort des „Globalen Dorfes“ genannt.

2.2 Internet-Protokolle und -Tools

2.2.1 Einleitung

Das Internet wurde hauptsächlich entwickelt, um Remote-Control, File-Transfer und Electronic-Mail zu ermöglichen. Aus diesem Grund bilden TELNET, FTP („File Transfer Protocol“) und SMTP („Simple Mail Transfer Protocol“) die ältesten und bekanntesten TCP/IP-Anwendungen. Aufbauend auf diesen Anwendungen wurden Tools wie FINGER oder HTTP („Hypertext Transfer Protocol“) entwickelt. Andere Services sind PING und TRACEROUTE. Diese benutzen jedoch nicht TCP/IP-Pakete, sondern ICMP-Pakete

² Ein Übersicht des Leitungsausbaus von Xlink befindet sich Anhang.

(„Internet Control Message Protocol“), welche auf Schicht 3 des ISO/OSI-Referenzmodells ansetzen und Hilfsmittel zur Überwachung und Fehlerbehandlung zur Verfügung stellen. In den folgenden Kapiteln wird auf diese Tools und Protokolle genauer eingegangen. Da das TCP/IP etwa zehn Jahre älter ist, als das OSI-Referenzmodell, werden bei TCP/IP andere Schichten definiert. Da wir im folgendem jedoch vermehrt das OSI-Referenzmodell benutzen, ist es sinnvoll eine Gegenüberstellung beider Modelle zu machen:

<i>OSI</i>		<i>TCP/IP</i>	

Anwendungsschicht	Application-Level
Darstellungsschicht	
Kommunikationssteuerschicht	
Transportschicht	Transmission-Level
Vermittlungsschicht	Internet-Level
Sicherungsschicht	Network-Level
Bitübertragungsschicht	

Tabelle 1. Gegenüberstellung OSI- und TCP/IP-Schichten

Die folgende Tabelle zeigt auf, unter welchen Schichten welche TCP/IP-Protokolle zu finden sind. Dabei zeigt die rechte Spalte die Dienste, welche vom NFS zusätzlich implementiert wurden und sich bereits stark an das OSI-Referenzmodell orientieren.

<i>OSI</i>	<i>TCP/IP</i>	

Tabelle 2. TCP/IP-Protokolle

Anwendungsschicht	TELNET,		NFS
Darstellungsschicht	FTP,		XDR
Kommunikationssteuerschicht	SMTP		RPC
Transportschicht	TCP, UDP		UDP
Vermittlungsschicht	IP (ARP, RARP), ICMP		
Sicherungsschicht	(CSMA/CD)	Token	Token
Bitübertragungsschicht	Ethernet (Bus)	Bus	Ring

Im Folgendem werden wir uns aber nicht mit den Details von TCP/IP befassen; Diese werden bereits in verschiedenen Vorlesungen über Kommunikationssysteme besprochen und finden sich in diverse Bücher über das Thema. Besonders Empfehlenswert ist dabei das Buch „Computernetzwerke“ von Tanenbaum.

Statt dessen wird in den folgenden Kapiteln auf die Tools oberhalb der vierten Schicht (TELNET, FTP,SMTP, HTTP, FINGER), und den Tools PING und TRACEROUTE aus der dritten Schicht eingegangen.

Kapitel 2.2.2 beschreibt sodann auch PING und TRACEROUTE, wobei der Schwerpunkt auf PING gelegt wird. Danach verlassen wir die Schicht 3 und befassen uns in Kapitel 2.2.3 mit TELNET, FTP, SMTP und FINGER. Als Teil der Anwendungsschicht besprechen wir zuletzt in Kapitel 2.2.4 kurz das HTTP.

2.2.2 PING und TRACEROUTE

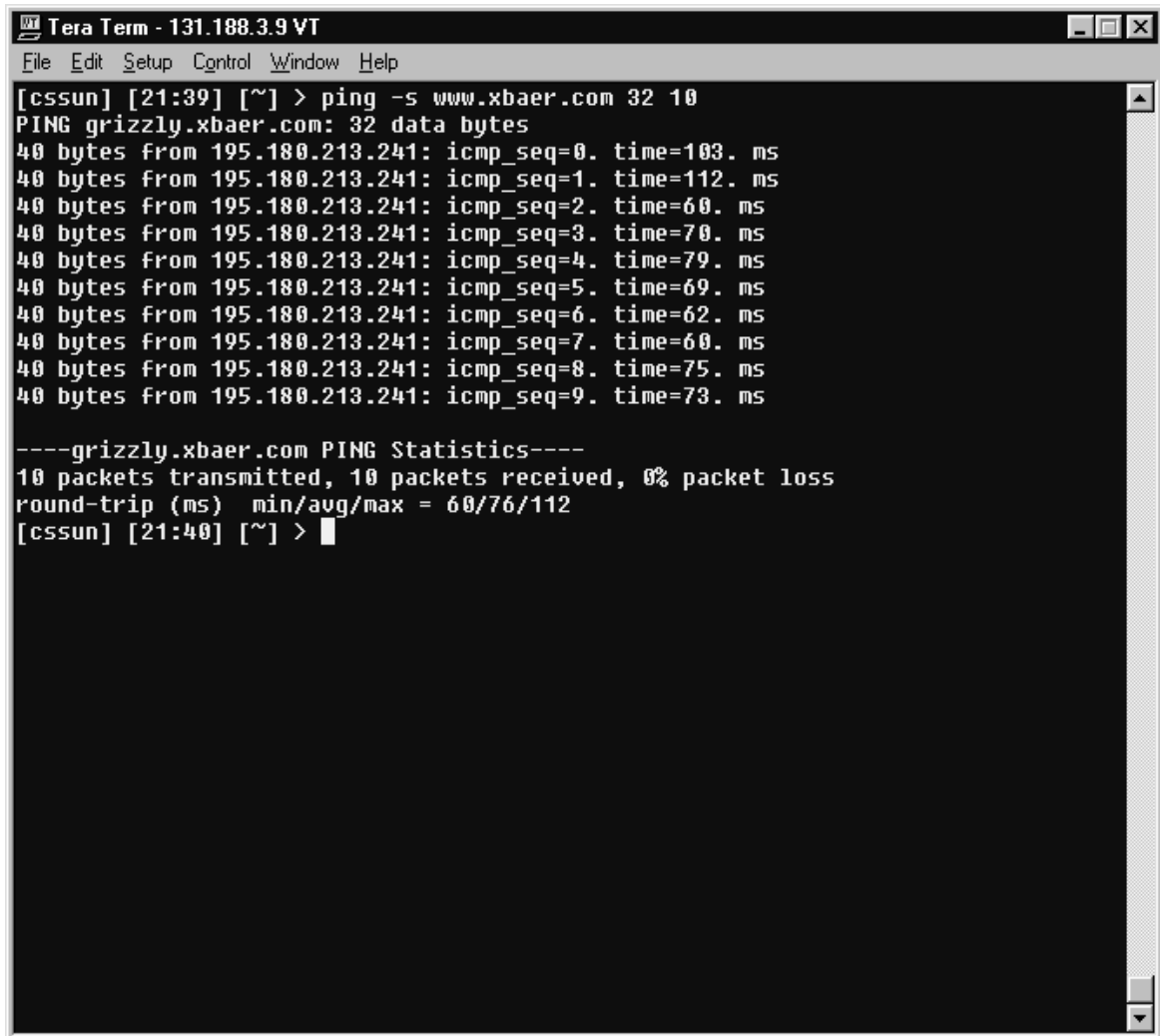
Wie bereits erwähnt, garantiert IP nicht dafür, daß Pakete ihr Ziel erreichen. Aus diesem Grund wurde das ICMP geschaffen, welches Teil jeder IP-Implementierung ist. Im Gegensatz zu IP-Paketen werden ICMP-Pakete direkt von der IP-Software gesendet und empfangen, und nicht wie bei IP-Paketen von den Anwendungsprozessen.

Normalerweise werden durch ICMP-Nachrichten nicht erreichbare Rechner gemeldet oder Anforderungen an Sender übermittelt, die Senderate zu ändern. Somit werden ICMP-Nachrichten hauptsächlich von der IP-Software verwendet und gestatten damit eine gewisse Flußkontrolle.

Das Anwendungsprogramm ping sendet gezielt ICMP-Pakete als Echoauftrag. Ist die TCP/IP-Software auf dem „angepingten“ Host aktiv und die Datenübertragung war erfolgreich, kann dieser antworten indem er die angekommenen Pakete unverändert zurücksendet. Dabei gestattet das Programm, die Zahl und die Größe der zu sendenden ICMP-Pakete festzulegen. Mit dem Absenden der Pakete wird die Zeit gestoppt, so daß das Programm in der Lage ist,

die Laufdauer der Pakete zu messen. Außerdem werden die zurückkommenden Pakete auf Fehler geprüft.

Der folgende Screenshot zeigt ein Beispielaufwurf:



```
Tera Term - 131.188.3.9 VT
File Edit Setup Control Window Help
[cssun] [21:39] [~] > ping -s www.xbaer.com 32 10
PING grizzly.xbaer.com: 32 data bytes
40 bytes from 195.180.213.241: icmp_seq=0. time=103. ms
40 bytes from 195.180.213.241: icmp_seq=1. time=112. ms
40 bytes from 195.180.213.241: icmp_seq=2. time=60. ms
40 bytes from 195.180.213.241: icmp_seq=3. time=70. ms
40 bytes from 195.180.213.241: icmp_seq=4. time=79. ms
40 bytes from 195.180.213.241: icmp_seq=5. time=69. ms
40 bytes from 195.180.213.241: icmp_seq=6. time=62. ms
40 bytes from 195.180.213.241: icmp_seq=7. time=60. ms
40 bytes from 195.180.213.241: icmp_seq=8. time=75. ms
40 bytes from 195.180.213.241: icmp_seq=9. time=73. ms

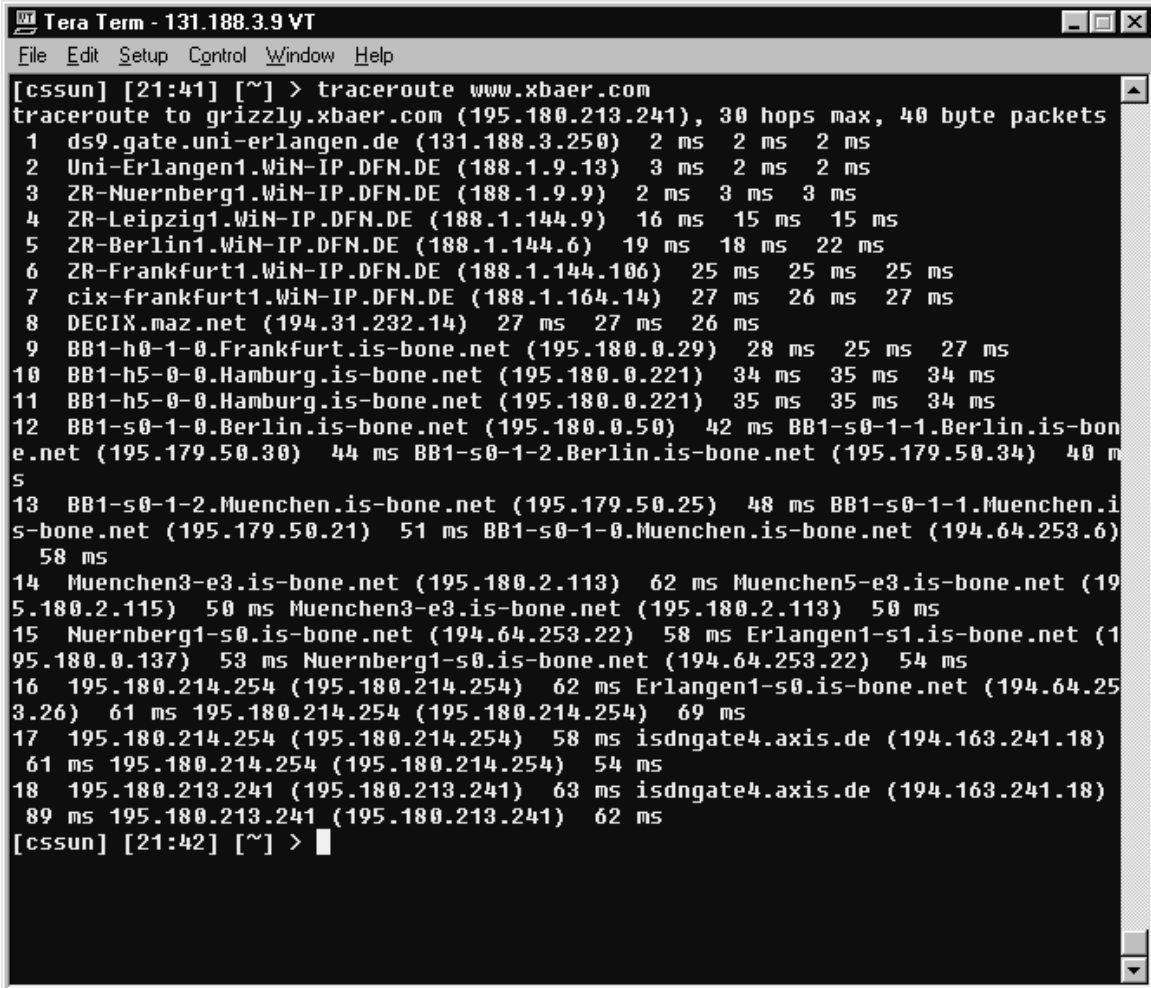
----grizzly.xbaer.com PING Statistics----
10 packets transmitted, 10 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 60/76/112
[cssun] [21:40] [~] > █
```

Bild 1. Ping

In diesem Fall wurde der Host `www.xbaer.com` mit 10 Paketen zu je 32 Bytes angepingt. Nachdem alle Pakete gesandt wurden, erstellt das Programm eine kleine Statistik, in der minimale, maximale und mittlere Antwortzeit aufgeführt werden. Das schnellste Paket war 60 ms unterwegs, das langsamste 112 ms. Im Mittel waren alle Pakete 76 ms unterwegs. Ebenfalls angegeben ist die Prozentzahl der verlorenen Pakete, nämlich 6 %.

Das Programm `traceroute` baut auf dem Programm `ping` auf, macht aber ungleich mehr: Es wird ein ping ausgeführt, wobei allerdings jeder beteiligte Host, der die Nachricht erhält und weiterzuleiten hat, aufgefordert wird bei Erhalt des Paketes ein Echo zu senden.

Somit erhält man eine detaillierte Angabe darüber, welchen Weg das Paket bis zum Zielrechner gegangen ist. Der folgende Screenshot zeigt dies am Beispiel des Hosts ww.xbaer.com.



```
Tera Term - 131.188.3.9 VT
File Edit Setup Control Window Help
[cssun] [21:41] [~] > traceroute www.xbaer.com
traceroute to grizzly.xbaer.com (195.180.213.241), 30 hops max, 40 byte packets
 1 ds9.gate.uni-erlangen.de (131.188.3.250)  2 ms  2 ms  2 ms
 2 Uni-Erlangen1.WiN-IP.DFN.DE (188.1.9.13)  3 ms  2 ms  2 ms
 3 ZR-Nuernberg1.WiN-IP.DFN.DE (188.1.9.9)   2 ms  3 ms  3 ms
 4 ZR-Leipzig1.WiN-IP.DFN.DE (188.1.144.9)  16 ms 15 ms 15 ms
 5 ZR-Berlin1.WiN-IP.DFN.DE (188.1.144.6)   19 ms 18 ms 22 ms
 6 ZR-Frankfurt1.WiN-IP.DFN.DE (188.1.144.106) 25 ms 25 ms 25 ms
 7 cix-frankfurt1.WiN-IP.DFN.DE (188.1.164.14) 27 ms 26 ms 27 ms
 8 DECIX.maz.net (194.31.232.14) 27 ms 27 ms 26 ms
 9 BB1-h0-1-0.Frankfurt.is-bone.net (195.180.0.29) 28 ms 25 ms 27 ms
10 BB1-h5-0-0.Hamburg.is-bone.net (195.180.0.221) 34 ms 35 ms 34 ms
11 BB1-h5-0-0.Hamburg.is-bone.net (195.180.0.221) 35 ms 35 ms 34 ms
12 BB1-s0-1-0.Berlin.is-bone.net (195.180.0.50) 42 ms BB1-s0-1-1.Berlin.is-bone.net (195.179.50.30) 44 ms BB1-s0-1-2.Berlin.is-bone.net (195.179.50.34) 40 ms
13 BB1-s0-1-2.Muenchen.is-bone.net (195.179.50.25) 48 ms BB1-s0-1-1.Muenchen.is-bone.net (195.179.50.21) 51 ms BB1-s0-1-0.Muenchen.is-bone.net (194.64.253.6) 58 ms
14 Muenchen3-e3.is-bone.net (195.180.2.113) 62 ms Muenchen5-e3.is-bone.net (195.180.2.115) 50 ms Muenchen3-e3.is-bone.net (195.180.2.113) 50 ms
15 Nuernberg1-s0.is-bone.net (194.64.253.22) 58 ms Erlangen1-s1.is-bone.net (195.180.0.137) 53 ms Nuernberg1-s0.is-bone.net (194.64.253.22) 54 ms
16 195.180.214.254 (195.180.214.254) 62 ms Erlangen1-s0.is-bone.net (194.64.253.26) 61 ms 195.180.214.254 (195.180.214.254) 69 ms
17 195.180.214.254 (195.180.214.254) 58 ms isdngate4.axis.de (194.163.241.18) 61 ms 195.180.214.254 (195.180.214.254) 54 ms
18 195.180.213.241 (195.180.213.241) 63 ms isdngate4.axis.de (194.163.241.18) 89 ms 195.180.213.241 (195.180.213.241) 62 ms
[cssun] [21:42] [~] >
```

Bild 2. Traceroute

2.2.3 TELNET, FTP, SMTP und FINGER

Telnet ist ein Internet-Protokoll, mit dessen Hilfe eine Dialogsitzung auf einem entfernten Rechner möglich ist. Meist ist mit Telnet aber nicht das Protokoll, sondern der gleichnamige Client gemeint. Dieser erlaubt die bidirektionale byte-orientierte Kommunikation zwischen Terminals, Anwendungen und Prozessen auf einem Rechner, und Terminals, Anwendungen und Prozessen auf einem anderen Rechner.

Beim Aufbau einer Telnet-Sitzung wird zunächst eine TCP-Verbindung zum Server-Host (Host, auf welchem das Terminal angeschlossen ist) hergestellt. Eingaben von der einen Seite werden in ein Standardformat umgewandelt und zum Server geschickt. Es werden dabei

Protokolle ähnlich dem CCITT-X.28 Protokoll verwendet. Antworten vom Server werden auf derselben Weise zurückgewandelt.

Das folgende Bild zeigt eine typische Telnet-Sitzung. Es wird deutlich, daß die Rechengrenzen in den Hintergrund treten und ein Arbeiten auf einem entfernten Rechner kaum mehr einen Unterschied macht zum Arbeiten an ein lokales Terminal.

```
Tera Term - 131.188.3.9 VT
File Edit Setup Control Window Help
[cssun] [15:00] [~] > telnet www.xwolf.com
Trying 151.196.75.24...
Connected to www.xwolf.com.
Escape character is '^]'.

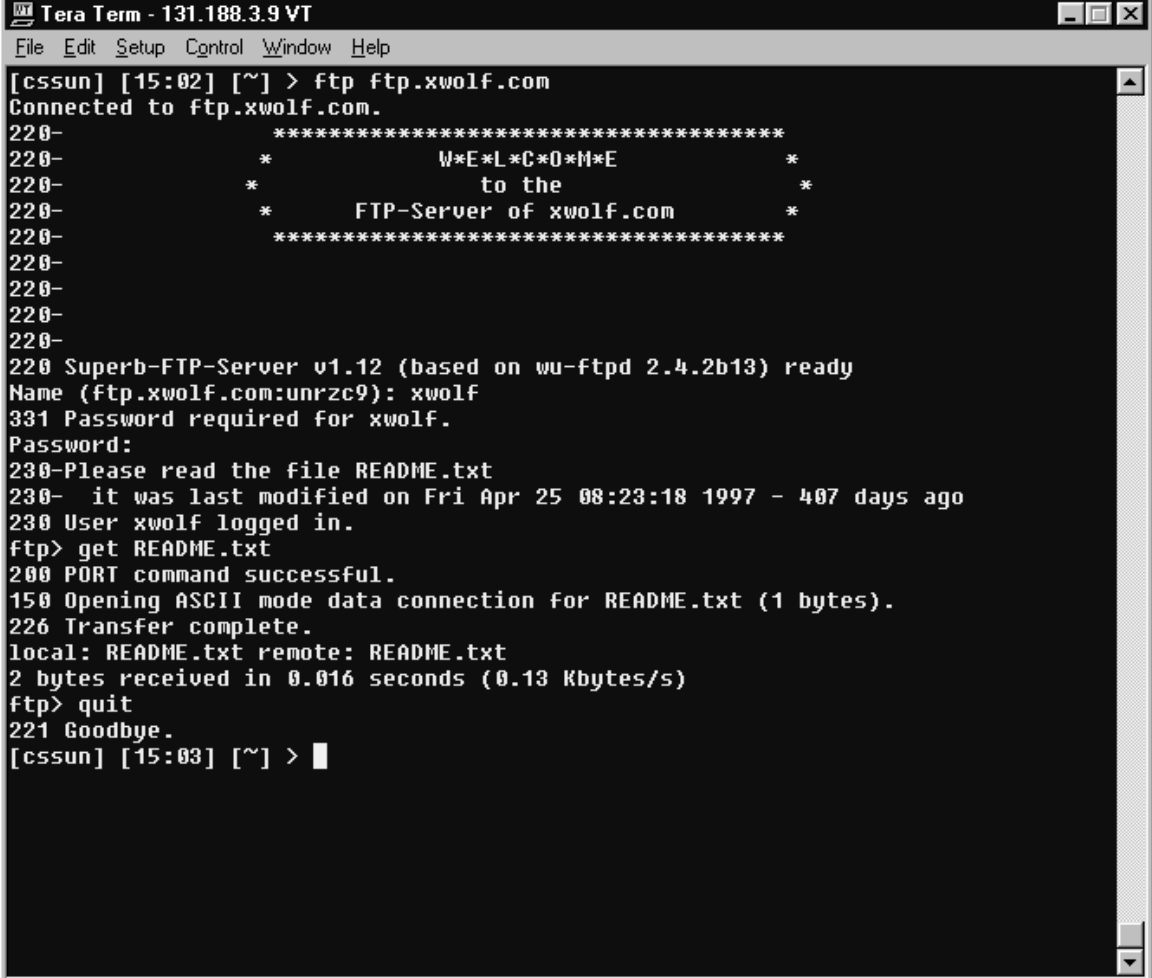
UNIX(r) System U Release 4.0 (wolf)

login: xwolf
Password:
Last login: Thu Jun 4 16:14:01 from info1.rrze.uni-e
wolf:/opt/home2/xwolf>ls
aas                data                local.profile       public ftp
autoresponders    data.tar            mail                 public_html
bin                email.fwd           pgp-key.asc         README.txt
board32.pl         email.fwd~          pgp-key.pgp         sample.procmairc
board32.pl~       inbox.old           PGP-out             speed
cgi-xbin           index.html          PGP-Test.txt        webservice
cgilocal           local.cshrc         pictures            welcome
cgilocal.tar      local.login         provider.txt
wolf:/opt/home2/xwolf>
```

Bild 3. Telnet-Sitzung

Das File Transfer Protocol gestattet den Austausch von Dateien zwischen unterschiedlichen Rechnern. FTP benutzt Telnet zum Aufbau der Verbindung und zum Senden der Befehle. Der eigentliche Datentransfer erfolgt aber über eine eigene Datenverbindung, welche zwischen dem User-DTP („User Data Transfer Process“) und dem Server-DTP erfolgt. Dabei können gleichzeitig mehrere Verbindungen gleichzeitig aufgebaut werden.

Alle Daten werden bei FTP in ein Netz-Datenformat umgewandelt und dann an den Empfänger gesandt. Dieser bildet dann die empfangenen Daten auf seine eigene Datenstruktur ab. Optional ist aber auch eine transparente Übertragung, d.h. ohne Konvertierung, möglich. Das folgende Bild zeigt eine typische FTP-Sitzung.



```
Tera Term - 131.188.3.9 VT
File Edit Setup Control Window Help
[cssun] [15:02] [~] > ftp ftp.xwolf.com
Connected to ftp.xwolf.com.
220- *****
220- *           W*E*L*C*O*M*E           *
220- *           to the                 *
220- *           FTP-Server of xwolf.com *
220- *****
220-
220-
220-
220-
220- Superb-FTP-Server v1.12 (based on wu-ftpd 2.4.2b13) ready
Name (ftp.xwolf.com:unrzc9): xwolf
331 Password required for xwolf.
Password:
230-Please read the file README.txt
230- it was last modified on Fri Apr 25 08:23:18 1997 - 407 days ago
230 User xwolf logged in.
ftp> get README.txt
200 PORT command successful.
150 Opening ASCII mode data connection for README.txt (1 bytes).
226 Transfer complete.
local: README.txt remote: README.txt
2 bytes received in 0.016 seconds (0.13 Kbytes/s)
ftp> quit
221 Goodbye.
[cssun] [15:03] [~] > █
```

Bild 4. FTP-Sitzung

SMTP erlaubt das Übertragen von Mitteilungen zu Hosts, welche am selben LAN („Lokal Area Net“) angeschlossen sind, oder über Gateways erreichbar sind. Eine Nachricht wird dadurch versandt, daß der Benutzer den Mail-Client beauftragt diese zu versenden. Dann baut der Sender-SMTP eine Verbindung zum Receiver-SMTP auf, welcher sich auf den vorher angegebenen Zielrechner befindet. Die Nachricht wird übertragen und der Receiver-SMTP ordnet die Nachricht den oder die lokalen Empfänger auf dem Host zu.

Finger schließlich ist ein Anwendungsprogramm welches auf Telnet aufbaut. Es ermöglicht festzustellen, wie der richtige Name eines Benutzers auf einem Rechner ist, wann und wo dieser eingeloggt war und ob weitere Informationen verfügbar sind.

Finger ist somit ein einfaches Nachrichten- und Informationssystem. Allerdings ist es nicht auf jedem Rechner im Internet installiert, bzw. kann nicht immer von außen abgerufen werden, so daß dieses System nicht sehr zuverlässig ist. Hinzu kommt, daß die Informationen, welche Finger über einen Benutzer bekommt, von Host zu Host verschiedenen sein können.

Das folgende Bild zeigt eine typische Finger-Abfrage auf dem Rechner *cssun.rrze.uni-erlangen.de*.



```
Tera Term - 131.188.3.9 VT
File Edit Setup Control Window Help
[cssun] [22:41] [~] > finger unrzc9
Login name: unrzc9                In real life: Wolfgang Wiese
Directory: /home/rzsunhome/unrz/unrzc9 Shell: /bin/csh
On since Jun  5 21:31:22 on pts/4 from unrzc9.dialin.rrze.uni-erlangen.de
Mail last read Tue Mar 31 22:51:59 1998
No Plan.

Login name: unrzc9                In real life: Wolfgang Wiese
Directory: /home/rzsunhome/unrz/unrzc9 Shell: /bin/csh
On since Jun  5 21:46:29 on pts/29 from unrzc9.dialin.rrze.uni-erlangen.de
[cssun] [22:41] [~] > █
```

Bild 5. Finger

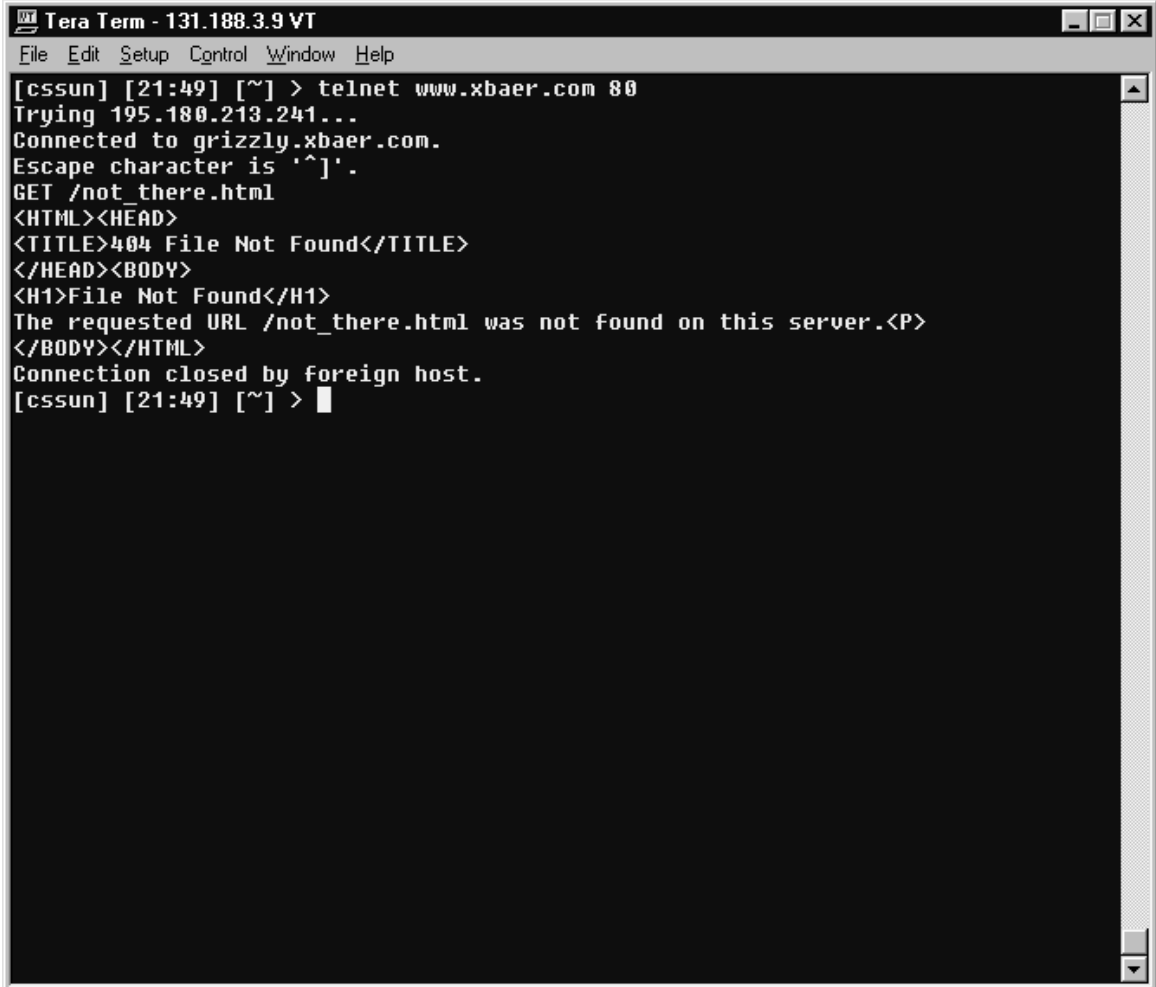
2.2.4 HTTP

Das Hypertext Transfer Protocol ist das Standardtransferprotokoll im Web. Es ist nicht verbindungsorientiert und arbeitet über TCP/IP-Verbindungen. Wenn ein Client (in der Regel

ein Webbrowser wie der Netscape Communicator) auf durch eine URL („Uniform Resource Locator“) bezeichnete Daten zugreifen will, baut er eine Verbindung zum Webserver auf über die er einen HTTP-Request sendet. Dieser Request ist in der Regel eine ASCII-Anfrage auf dem Port 80. Der Server antwortet mit einer MIME-ähnlichen Antwort und baut dann selbständig die TCP/IP-Verbindung wieder ab.

Der HTTP-Request ist ein einfacher Befehl, welcher aus den Teilen Zugriffsmethode, Identifier und HTTP-Version besteht. Auf allen neueren Servern kann man aber auch einfachere Anfragen senden, wo lediglich die Zugriffsmethode und der Identifier angegeben sind. In diesem Fall kommt die Antwort ohne Header und ohne MIME-Codierung.

Folgendes Bild zeigt ein Beispiel, wie eine solche Übertragung aussehen kann:



```
Tera Term - 131.188.3.9 VT
File Edit Setup Control Window Help
[cssun] [21:49] [~] > telnet www.xbaer.com 80
Trying 195.180.213.241...
Connected to grizzly.xbaer.com.
Escape character is '^]'.
GET /not_there.html
<HTML><HEAD>
<TITLE>404 File Not Found</TITLE>
</HEAD><BODY>
<H1>File Not Found</H1>
The requested URL /not_there.html was not found on this server.<P>
</BODY></HTML>
Connection closed by foreign host.
[cssun] [21:49] [~] > █
```

Bild 6. HTTP-Verbindung mittels Telnet

Die Arten der Zugriffsmethoden sind streng standardisiert. Es werden jedoch nicht alle Methoden von allen HTTP-Versionen unterstützt. Üblicherweise werden nur GET-, POST-

und HEAD-Requests benötigt. Die folgende Tabelle gibt eine Übersicht über die Zugriffsmethode:

<i> Methode </i>	<i> Beschreibung </i>
GET	Anfrage zum Lesen einer Webpage
POST	Anhängen einer benannten Resource. (Meist benutzt bei CGI-Programmen. Daten werden für den Benutzer unsichtbar gesandt.)
HEAD	Anfrage zum Lesen des Headers einer Webpage
PUT	Anfrage zum Speichern einer Webpage (Meist Browserintern und bei Datenübertragung über das Web.)
DELETE	Entfernen einer Webpage
LINK	Verbindet zwei Ressourcen. (In der Regel durch Module ersetzt, welche ein Redirect ausführen.)
UNLINK	Löst die Verbindung zwischen zwei Ressourcen

Tabelle 3. HTTP-Requestmethoden

Das HTTP wird ständig weiterentwickelt. Alle Versionen bauen aber auf dieselben Konzepte auf, so daß sie „abwärtskompatibel“ sind. Neben dem normalen HTTP wird außerdem das S-HTTP („Secure HTTP“) benutzt, welches die Schwächen vom normalen HTTP bei der Sicherheit von Datenübertragungen minimiert. Im Gegensatz zum normalen HTTP, wo die Informationen im Klartext übertragen werden –und somit von dritter Seite aus gelesen werden könnten-, wird bei S-HTTP die Übertragung verschlüsselt. Als Konkurrenzprodukt zu S-HTTP fungiert SSL („Secure Socket Layer“), bei dem zwischen der TCP/IP-Schicht und den höheren Schichten eine weitere Sicherungsschicht eingefügt wird.

2.3 Mathematische Grundlagen

2.3.1 Einleitung

Ein nicht unwesentlicher Teil der Arbeit besteht darin, die gesammelten Daten auszuwerten und aufzubereiten. Dabei müssen aus einer großen Anzahl an diskreten Werten statistische Informationen extrahiert werden, welche die Messungen überschaubar und verstehbar machen. In diesem Stadium der Arbeit sind komplexe Methoden zur Berechnung von Abschätzungen und Prognosen noch nicht nötig. Aus diesem Grund können wir uns auf die Grundlagen der

Statistik beschränken, welche Formeln für Mittelwert, Standardabweichung und den Fehler angeben. Dies erfolgt in den Kapitel 2.3.2 bis 2.3.4.

Lediglich über die Fehlerfortpflanzung wird eine genauere Diskussion nötig, da wir zeitlich gebundene Mittelwerte z.B. eines Tages weiterverwenden um den Mittelwert eines Monats zu berechnen. Dies geschieht im Kapitel 2.3.5.

Von der Herleitung der Formeln wird abgesehen, da diese zu den Grundlagen gezählt werden und als Bekannt vorausgesetzt werden können.

2.3.2 Mittelwert

Die Charakterisierung einer statistischen Auswertung kann durch die Maßzahlen von Mittelwert und Varianz erfolgen. Der Mittelwert –auch arithmetisches Mittel genannt- ist gegeben durch die Formel:

$$\hat{\mu} = \hat{\mu}(x_1, \dots, x_n) = \frac{1}{n} \sum_{i=1}^n x_i = \bar{x}.$$

x_i ($i = 1, 2, \dots, n$) sind dabei die einzelnen gemessenen Merkmalswerte.

2.3.3 Varianz (Standardabweichung)

Zur Berechnung der Abweichung der gemessenen Werte vom arithmetischen Mittel dient die Varianz. Sie ist gegeben durch die Formel:

$$\hat{\sigma}^2 = \frac{q}{n} = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2$$

wobei q die Summe der quadratischen Abweichungen ist:

$$q = \sum_{i=1}^n (x_i - \mu)^2$$

Der Wert σ^2 wird dabei als Standardabweichung oder mittlere quadratische Abweichung bezeichnet.

2.3.4 Mittlerer quadratischer Fehler

Neben der Charakterisierung von Meßwerten durch Mittelwert und Varianz besteht auch die Möglichkeit anstelle der Varianz einen Wert anzugeben, mit dessen Hilfe eine Aussage über die Güte des Meßwertes möglich ist.

Dabei wird der Meßwert nach seiner Abweichung vom Mittelwert beurteilt. Ein Meßwert sollte dabei innerhalb einer gewissen Schranke um den Mittelwert zu finden sein. Diese Schranke wird als Fehler bezeichnet und berechnet sich aus der Varianz geteilt durch die Zahl der Meßwerte:

$$F = \sigma^2 / n$$

2.3.5 Fehlerfortpflanzung

In unserem Experiment werden wir mehrmals täglich Werte messen, welche jeweils mit einem Fehler behaftet sind³. Alle diese fehlerbehafteten Werte werden zusammen genutzt um neue Werte zu berechnen. Zum Beispiel wird aus den fehlerbehafteten Stundenwerten ein Tageswert berechnet. Und aus den Tageswerten wird später wieder ein Monatswert berechnet. Aus den Monatswerten wiederum werden Jahreswerte berechnet.

Da die einzelnen Meßwerte einen Fehler haben, pflanzt sich dieser auf alle folgenden Werte fort. Mit Hilfe des *Gaußschen Fehlerfortpflanzungsgesetzes* läßt sich der mittlere Fehler aus den Resultaten von mittleren Fehlern von Eingangsdaten berechnen. Da wir zudem von ungewichteten Einzelmessungen⁴ ausgehen können, gilt der folgende Satz:

Der mittlere Fehler des Mittelwertes aus n mit gleicher Präzision durchgeführten Messungen ist gleich dem durch \sqrt{n} geteilten mittleren Fehler der Einzelmessung.

Die Formel für den mittleren Fehler von fehlerbehafteten Einzelmessungen lautet also:

$$F_A = \sigma_A^2 / n$$

Wobei:

$$\sigma_A^2 = (\sigma_1^2 + \sigma_2^2 + \dots + \sigma_n^2) / n^2$$

Hiermit haben wir alle benötigten Formeln zur Erstellung der Statistiken. Erst bei einem weiteren Ausbau der Arbeit, wie in Kapitel 6 vorgeschlagen wird, werden weitere Formeln notwendig.

³ Hier von Fehlern zu sprechen ist nicht ganz korrekt. In Wirklichkeit ist jede Einzelmessung für sich gesehen nicht Fehlerbehaftet. Statt dessen ist der Wert abhängig von Ort und Zeitpunkt der Messung. Da aber die Meßwerte aufgrund äußerer Einflüsse zu jedem Meßzeit- und Ortspunkt anders ausfallen, entsteht bei jeder Messung ein relativer Fehler, welcher berücksichtigt werden muß. Siehe hierzu auch das Fehlerverteilungsgesetz.

⁴ Ungewichtete Einzelmessungen entsprechen hier Messungen, welche alle mit der gleichen Präzision ausgeführt wurden.

2.4 Programmiertechnische Grundlagen

2.4.1 Einleitung

Der größte Teil der Arbeit bestand aus der Programmierung aller benötigten Skripten und Programme. Dabei wurde die Programmiersprache Perl verwendet, welche sich in den letzten Jahren besonders auf Unix-Rechnern, sowie bei CGI-Programmen durchgesetzt hat.

Da Perl seine Wurzeln in der Sprache C hat, finden sich alle Konstrukte und Funktionen von C in Perl wieder. Lediglich das Speichermanagement wurde gegenüber C stark vereinfacht.

Zudem ist Perl eine Interpretersprache. In Kapitel 2.4.4 werden wir genauer auf die Vorteile von Perl eingehen und einige Grundlagen dazu erklären.

Neben Perl werden auch Shell-Skripte und CRONTAB-Einträge benötigt. Diese sind in der Regel kurz und werden nur zur Steuerung von Abläufen verwendet. Wir besprechen Shell-Skripte in Kapitel 2.4.2 und gehen dann in Kapitel 2.4.3 auf CRONTAB-Einträge ein.

2.4.2 Shell-Skripte

Shell-Skripte sind in der Regel kurze Programme⁵, welche nicht länger als 100 Zeilen sind.

Diese werden direkt auf dem Terminal ausgeführt, auf dem sie aufgerufen werden. Dabei wird die Shell benutzt, eine Benutzerumgebung auf dem Terminal, welche einfache Befehle, wie „cd“, „pwd“, „time“, „alias“, usw. ermöglicht.

Ein Shell-Skript ruft nun diese Befehle in einer festgelegten Reihenfolge auf. Dabei besteht kein Unterschied zwischen dem Aufruf von Befehlen über ein Skript, oder dem Aufruf vom Benutzer direkt.

Ein einfaches Beispiel für ein Shell-Skript findet sich im Anhang B, Kapitel 2.3. Der Befehlsumfang jedes Shell-Skriptes ist abhängig von der Shell, welche man benutzt. Auf Unix-Rechner sind in der Regel mehrere Shells installiert, wie SH, TCSH und CSH. Auch wenn die Shells auch längere und komplexe Funktionen ermöglichen, geht der Gebrauch von Shell-Skripten kaum über die von einfachen Makros hinaus.

2.4.3 CRONTAB-Einträge

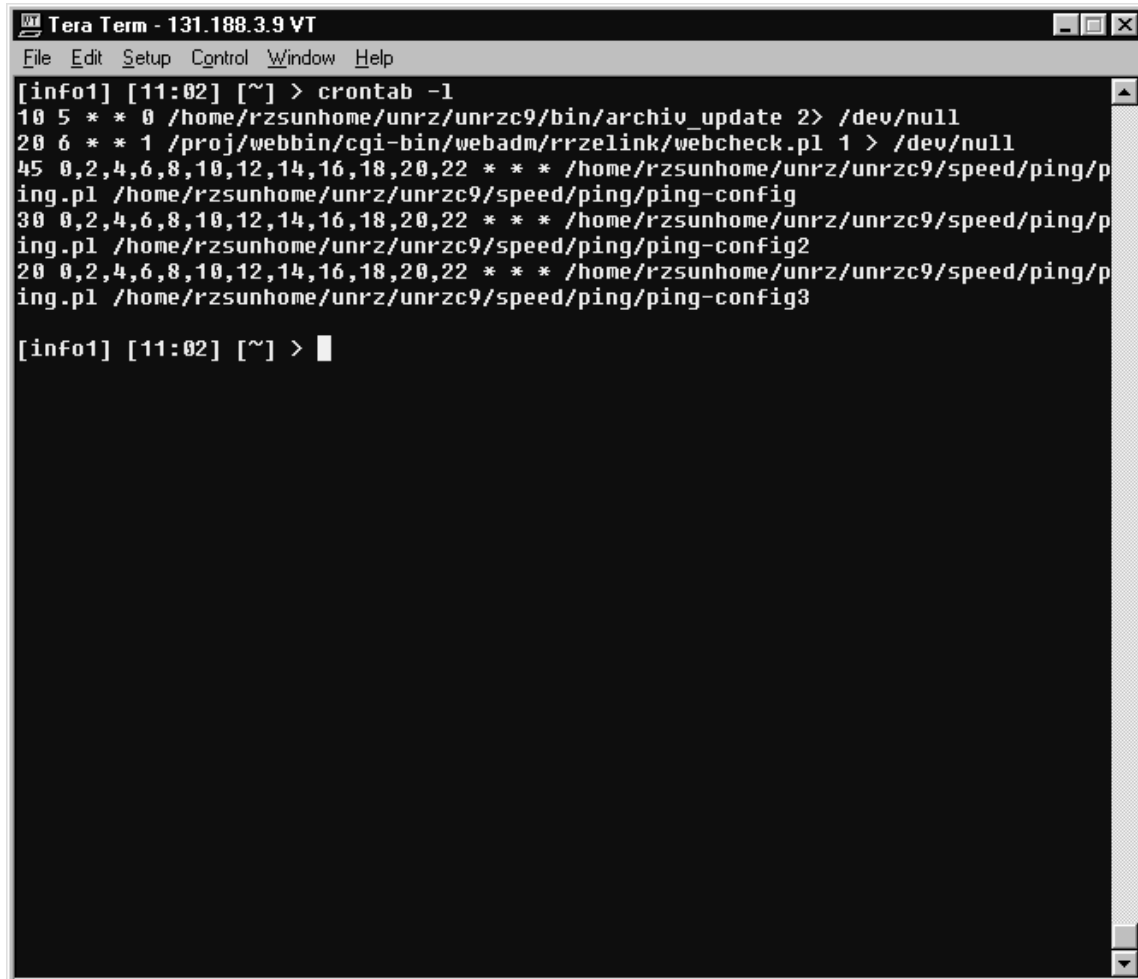
Crontab ist ein Unix-Programm, welches es erlaubt, Programme und Skripten zu einem vorher festgelegtem Zeitpunkt automatisch ausführen zu lassen. Dabei hat man die Wahl zwischen

⁵ Es gibt keinen fest definierten Unterschied zwischen den Begriffen „Skript“ und „Programm“. In der Anwendung spricht man von Programmen, wenn der Source-Code sehr lang ist, und von Skripten, wenn der Source-Code nicht länger als 100 Zeilen ist. Es hat sich ebenfalls durchgesetzt, bei CGI-Anwendungen von Skripten zu sprechen, während alle sonstigen umfangreichen Anwendungen als Programme bezeichnet werden. Hier spielt auch der amerikanische Einfluß eine große Rolle.

mehreren Einstellungen, so daß man Programme und Skripten mehrmals am Tag, in der Woche, im Monat oder im Jahr ausführen lassen kann.

Der Vorteil daran ist, daß man dabei nicht selbst online sein muß.

Folgendes Bild zeigt eine typische Cron-Tabelle:



```
Tera Term - 131.188.3.9 VT
File Edit Setup Control Window Help
[info1] [11:02] [~] > crontab -l
10 5 * * 0 /home/rzsunhome/unrz/unrzc9/bin/archiv_update 2> /dev/null
20 6 * * 1 /proj/webbin/cgi-bin/webadm/rrzelink/webcheck.pl 1 > /dev/null
45 0,2,4,6,8,10,12,14,16,18,20,22 * * * /home/rzsunhome/unrz/unrzc9/speed/ping/ping.pl /home/rzsunhome/unrz/unrzc9/speed/ping/ping-config
30 0,2,4,6,8,10,12,14,16,18,20,22 * * * /home/rzsunhome/unrz/unrzc9/speed/ping/ping.pl /home/rzsunhome/unrz/unrzc9/speed/ping/ping-config2
20 0,2,4,6,8,10,12,14,16,18,20,22 * * * /home/rzsunhome/unrz/unrzc9/speed/ping/ping.pl /home/rzsunhome/unrz/unrzc9/speed/ping/ping-config3
[info1] [11:02] [~] > █
```

Bild 7. Crontab

Im Bild sieht man, daß das Programm webcheck.pl an jedem 1. Wochentag um 6 Uhr 20 ausgeführt wird, den Parameter „1“ übergeben bekommt und etwaige Ausgaben auf „/dev/null“ umleitet.

Diese Crontab steuert übrigens auch die lokalen Messungen, welche vom RRZ Erlangen ausgehen. Die letzten 3 Einträge, welche jeweils das Programm ping.pl aufrufen, werden alle 2 Stunden gestartet. Das Programm ping.pl bekommt dabei je nach Aufruf einen anderen Parameter übergeben, welcher in diesem Fall eine Datei ist, in der Einstellungen stehen. Siehe hierzu auch den Anhang B.

2.4.4 Perl

Perl ist eine Interpretersprache, welche zur 4. Sprachengeneration gehört. Es enthält daher Möglichkeiten zu komplexen Strukturen, wie assoziative Arrays ebenso, wie rekursive Funktionen und Pointer. Im Gegensatz zu anderen Programmiersprachen übernimmt Perl für den Programmierer viele kleine Teilaufgaben, wie das Speicherhandling und tiefere Systemfunktionen. Dies heißt aber nicht, daß der Programmierer dies nicht selbst auch machen kann. Tom Christiansen, einer der Gründer von Perl sagt dazu:

„With Perl you're free to do The Right Thing, however you care to define it.“

Eine der großen Vorteile von Perl gegenüber anderen Sprachen liegt in dem Funktionsumfang zur Behandlung von Strings mittels regulärer Ausdrücke.

Ein Perl-Programm beginnt –wie ein Shell-Skript – immer mit der Angabe des Pfades zum Perl-Interpreter. Danach folgt der eigentliche Programmcode. Dieser hat große Ähnlichkeit mit einer Mischung aus Pascal und C, wobei aus beiden Sprachen das jeweils Beste genommen wurde.

Da Perl seine Wurzeln in C hat, sieht man auch daran, daß es einfach möglich ist Source-Code von C in Perl-Programmen einzufügen, bzw. Perl in C-Programmen. Auch ist es möglich, Perl-Programme, die ja eigentlich Interpreter-Code sind, zu compilieren⁶.

Wie in C, ist es bei Perl möglich, Libraries und Module, welche zusätzliche Funktionen beinhalten, einzufügen. Auf der Perl-Homepage <http://www.perl.com> findet sich eine große Sammlung an fertigen Funktionen zu nahezu jeden Zweck. Zudem wurde ein Projekt Namens CPAN ins Leben gerufen, bei dem professionelle Programmierer Code und Libraries hinterlegen können. Die CPAN-Bibliothek gehört inzwischen zu einer der größten Sammlungen an Programmier-Libraries. Perl ist Standardmäßig auf allen UNIX- und LINUX-Rechner vorinstalliert.

Verzichtet man auf tiefere Systemfunktionen, ist Perl plattformunabhängig. Versionen für Windows, Mac und Amiga sind vorhanden.

⁶ Zur Compilation von Perl-Programmen bedarf es allerdings einiger zusätzlicher Perl-Module.

3. Lösungsansätze

3.1 Einleitung

In diesem Kapitel werden die Lösungsansätze vorgestellt, mit denen man sich der Aufgabenstellung nähern kann. Alle Ansätze bauen auf zwei grundlegende Verfahren auf:

- Bei der ersten Methode werden Übertragungsmessungen, von einem Home-Rechner, welcher per Modem oder ISDN (oder xDSL) an dem Dialin-Rechner eines Providers angeschlossen ist, gemacht.
- Bei der zweiten Methode werden von einem Rechner im Internet Messungen zu mehreren entfernten Rechnern ausgeführt.

Diese Methoden setzen auf unterschiedlichen Zielvorstellungen auf. Bei der ersten Methode wird keine Abschätzung über die Zugriffsgeschwindigkeit gemacht, mit der man auf Seiten entfernter Rechner, d.h. außerhalb des Dialin-Providers, zugreifen kann. Bei der zweiten Methode wird dagegen nur die Zugriffsgeschwindigkeit von einem stationären Host zu anderen vorher definierten Rechner gemessen.

Die Lösungsansätze bestehen zum Teil aus einer der beiden Methoden oder aus deren Erweiterungen. Anhand des Beispiels der Computerzeitschrift c't werden wir in Kapitel 3.2 die erste Methode besprechen.

Die zweite Methode ist im Vergleich zu der ersten weniger umfangreich zu organisieren. Sie findet sich daher häufiger in Netz. In Kapitel 3.3 zeigen wir anhand des Beispiels der Site vom „Internet Weather“ die Vor- und Nachteile auf.

Im Kapitel 3.4 wird der Ansatz dieser Arbeit vorgestellt, welcher eine Variante der zweiten Methode ist. Wir werden an dieser Stelle auch erörtern, ob die verwendete Methode der „ping“-Requests überhaupt benutzt werden darf.

3.2 Ansatz der „c't“

Wie oben bereits angedeutet liegt der Ansatz der c't darin, von einem Home-PC aus die Zugriffsgeschwindigkeit auf dem Dialin-Server eines Providers zu messen.

Dabei wird die Übertragungsraten mittels verschiedener Modems und ISDN gemessen. Die Messung selbst erfolgt durch ein ping-ähnliches Programm, welches während einer Sitzung kontinuierlich die verfügbaren Datenraten anzeigt.

Solche Programme sind im Netz frei erhältlich. So zum Beispiel das Programm „AnySpeed“ von PY Software. Das folgende Bild zeigt es bei der Messung der Datenraten des Rechners 131.188.3.9 (cssun.rrze.uni-erlangen.de) mit einem handelsüblichen Modem mit 28.8kbps:

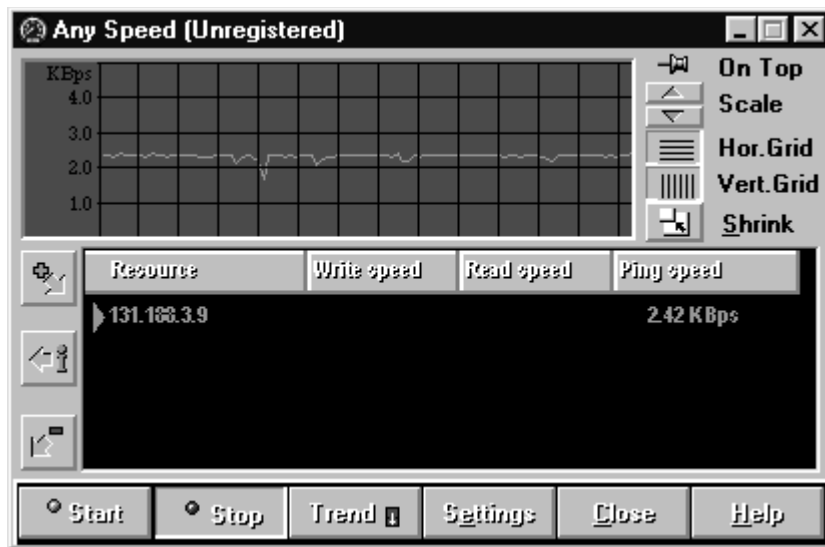


Bild 8. AnySpeed Messung

Ein ähnliches Programm wird auch von der c't benutzt. Im Gegensatz zu der obigen Beispielmessung setzt die c't aber mehrere unterschiedliche Modems und ISDN-Anlagen ein, um die Qualität der Dialin-Anlagen des Providers zu testen. Dabei werden nicht nur die Datenraten zu den jeweiligen Dialin-Rechner gemessen, sondern auch die von entfernten Rechner.

Damit erzielt die c't Messungen relativ genaue Messungen. Allerdings gibt es 4 Punkte, welche die Ergebnisse wieder in Frage stellen:

- Die Messungen der c't sind aus Kosten- und Organisationsgründen bei jeden Dialin-Provider zeitlich begrenzt. Es können auch nicht alle Dialin-Provider berücksichtigt werden. Statt dessen werden in jedem Monat Provider aus einer anderen bundesdeutschen Stadt gemessen. Lediglich deutschlandweite Provider, wie AOL, Metronet, T-Online und CompuServe wurden im Rahmen des Magazins „Netnite“ im letzten Jahr kontinuierlich über mehrere Monate hinweg gemessen⁷.
- „Böswillige“ Provider haben die Möglichkeit, die Messungen zu ihrem Gunsten zu beeinflussen, indem sie kurzfristig, für die Dauer der Messung, Kapazitäten für die Dialin-Zugänge bereitstellen oder bessere Leitungen anmieten, welche

⁷ Das Magazin „Netnite“ lief im Jahr 1997 monatlich im ZDF. Dort wurden jeweils Online-Statistiken über die 4 großen Onlinedienste AOL, CompuServe, T-Online und Metronet vorgestellt, welche in Zusammenarbeit mit der c't entstanden. Seitdem der Vertrag von „Netnite“ aber vom ZDF nicht verlängert wurde, wurde dieser Dienst im Juli 1997 eingestellt.

KAPITEL 3. LÖSUNGSANSÄTZE

später wieder abgebaut werden könnten. Ein Beweis für ein solches Verhalten gibt es zwar noch nicht, aber es steht im Rahmen der Möglichkeiten. Man bedenke, daß es zur Zeit zwischen den Providern einen harten Verdrängungswettbewerb gibt. Eine Kennzeichnung als schneller Dialin-Provider bedeutet Vorteile, die sich finanziell stark auswirken können.

- Die Messung ist leider abhängig von den Telefonleitungen der Telekom: Es kann nicht dafür garantiert werden, daß die Qualität dieser Leitungen überall gleich ist. Die Übertragungsfehler, die durch die Telekom-Schaltungen und – Leitungen entstehen gehen in die Meßwerte des jeweiligen Providers ein.
- Die Messung der $c't$ berücksichtigt nicht Onlinedienste, welche keine Dialin-Eingänge anbieten. Auch ist der Nutzen der Messungen für Institutionen oder Personen, welche einen direkten Zugang zum Netz haben mehr als eingeschränkt.
- Die Ergebnisse der Messung sagen nur wenig darüber aus, wie schnell ein Kunde eines Dialin-Providers auf die Seiten einer anderen Site zugreifen kann.

Trotz dieser Nachteile bietet diese Art der Messung auch große Vorteile aufgrund ihrer Genauigkeit bei der Unterteilung in verschiedene Modemgeschwindigkeiten und ISDN. Auch ist die Messung stark ausbaufähig. Lediglich die hohen Kosten für die Einzelmessungen verhindern, daß man kontinuierliche Messungen über längere Zeiträume machen kann.

3.3 Ansatz vom „Internet Weather“

Der Ansatz der Web-Site „Internet Weather“ ist es, „ping“-Requests an mehrere Rechner aus einer vorgegebenen Liste zu senden und die erhaltenen Werte online darzustellen. Dabei erfolgen diese Requests bis zu 10 mal in der Stunde pro angepingten Host. Das folgende Bild zeigt die Darstellung der Ergebnisse:

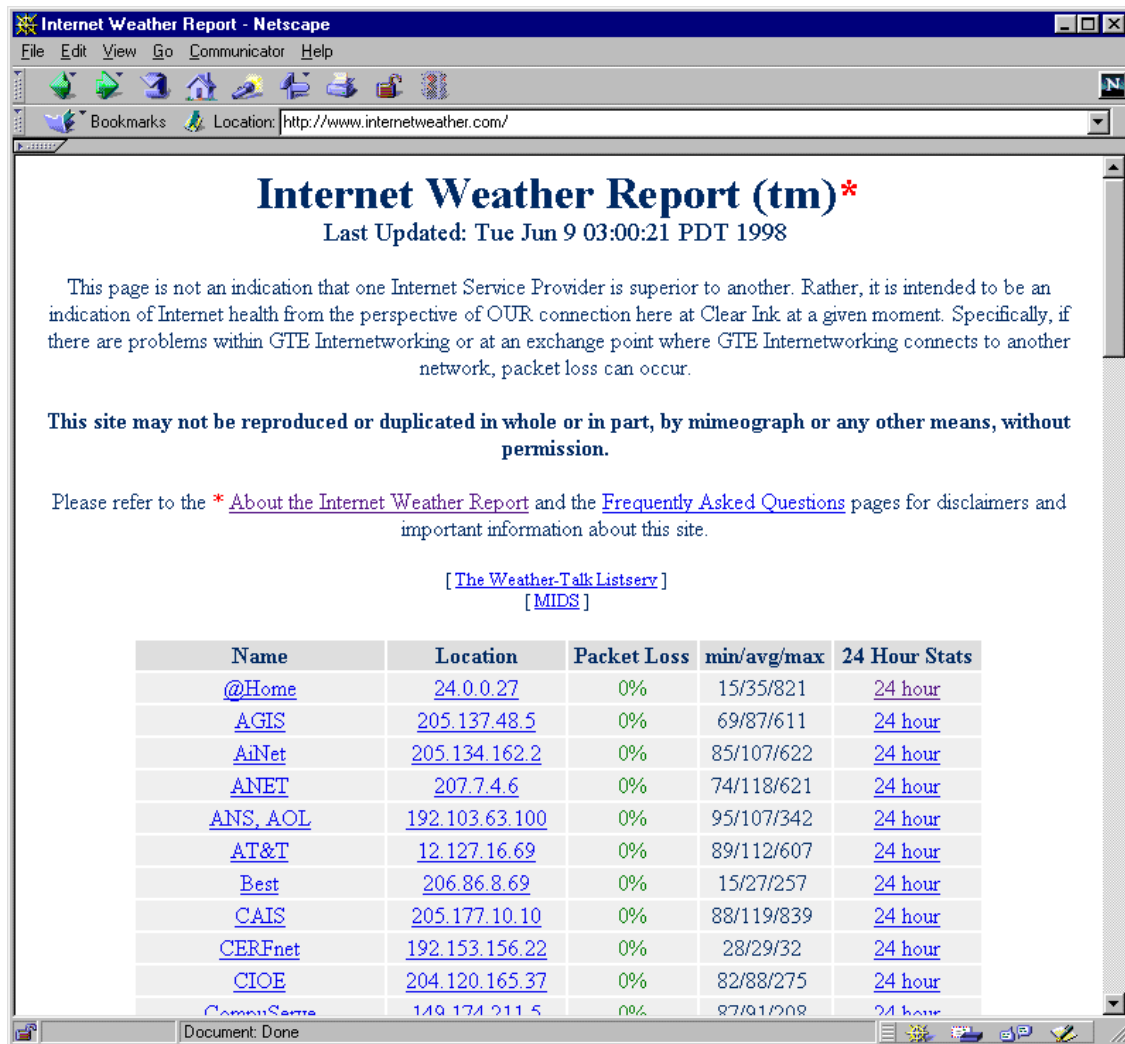


Bild 9. Internet Weather

Bei dieser Messung wird nur die Zeitdauer der „ping“-Requests von dem Rechner des „Internet Weather’s“ aus gemessen. Die Ergebnisse sind also nur lokal richtig und erlauben keinen Vergleich mit anderen Rechnern.

Hinzu kommt, daß die „ping“-Requests nur die Zeitspanne angeben, wie lange es dauert, bis ein entfernter Rechner mittels seiner TCP/IP-Software ein Lebenszeichen von sich gibt. Der Zugriff auf die eigentlichen Webseiten der beteiligten Rechner, oder gar deren Dialin-Zugänge wird nicht gemessen. Ein weiterer Nachteil der „ping“-Messung gegenüber der Dialin-Messung von c’t ist der, daß entfernte Rechner, welche nicht in der Messung aufgenommen werden wollen, diese es dadurch verhindern können, indem sie „ping“-Requests von ihrem Rechner nicht beantworten lassen⁸. Dies sind aber Ausnahmen.

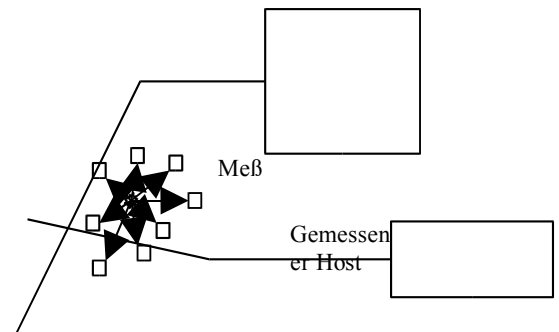
⁸ Natürlich können Provider eine Messung durch die c’t auch verhindern, indem sie dieser kein Testaccount zur Verfügung stellen. Dies kann –und wird– dadurch umgangen, indem die c’t erst nach der Messung dem Provider von dieser unterrichtet.

KAPITEL 3. LÖSUNGSANSÄTZE

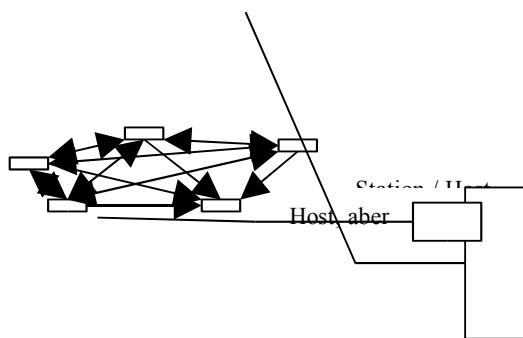
Dieser Ansatz der Messung zeichnet sich dadurch aus, daß er sehr einfach zu realisieren ist und keine Kosten durch Telefonanbieter auftreten, da die Messung autonom arbeiten kann. Somit stehen kontinuierlichen Dauerzeitmessungen kaum Hindernisse im Wege.

3.4 Ansatz dieser Arbeit

Der Ansatz dieser Arbeit ist eine Variation der zweiten Methode. Beim Internet Weather und dessen Klones gehen alle Messungen sternförmig von einem einzigen Rechner aus. Aus diesem Grund sind die Ergebnisse allesamt nur lokal verwertbar.



Die Idee hinter dieser Arbeit ist deswegen eine logische Weiterführung: Anstelle von einem einzigen Rechner werden n unterschiedliche Rechner (im folgendem als **Stationen** bezeichnet) genommen, welche unabhängig voneinander Hosts messen. Außerdem können die Stationen ebenfalls Webseiten anbieten, also auch Hosts sein, welche selbst gemessen werden.



Im nächsten Schritt werden die jeweils lokalen Ergebnisse aller Stationen zentral gesammelt und ausgewertet. Die Zentrale kann dabei auch wieder als Station und Host auftreten.

Folgenden Funktionen werden von der Zentrale ausgeführt:

Alle Stationsdaten werden gelesen, gespeichert und ausgewertet. Hierzu gehört auch die Erstellung von Statistiken, welche sich aus der Gesamtzahl aller Stationsdaten ergeben.

Fehler innerhalb der Stationsdaten werden erkannt und ggf. dem Administrator gemeldet. Die Statistikerstellung wird im Falle eines Stationsfehlers entsprechende Ausnahmebehandlungen aktivieren.

Es werden Hostlisten verwaltet, welche alle Stationen zur Verfügung gestellt werden. Die Verwaltung arbeitet dabei ähnlich wie Routingtables: Nimmt eine einzelne Station einen neuen, bisher unbekannt Host in seine lokale Liste auf, werden diese Hostdaten an die Zentrale weitergeleitet und dann unter Umständen in die zentrale Liste übernommen. Einzelne Stationen lesen wieder diese Liste und übernehmen so neue Hosts.

Es wird eine dauerhafte Liste aller Stationen geführt, welche genau Auskunft über den Status einer jeden Station gibt.

KAPITEL 3. LÖSUNGSANSÄTZE

Online-Statistiken werden erstellt, die Einblick in die Messungen geben. Aufgrund dem Umfang der Daten muß die Zentrale ausreichende Such-Funktionen zur Verfügung stellen.

Zur Messung der Zugriffsgeschwindigkeit werden "ping"-Requests verwendet. Von der Verwendung von direkten HTTP-Verbindungen wird in diesem Stadium der Arbeit abgesehen. **Zusätzliche** Messungen über HTTP-Requests werden aber Gegenstand zukünftiger Erweiterungen sein. Siehe hierzu auch Kapitel 6.

Da "ping"- und HTTP-Requests nicht gleichgesetzt werden können, ist eine Diskussion darüber nötig, ob es erlaubt sein kann, die Meßwerte alleine aus "ping"-Requests zu beziehen.

Obwohl "ping"-Requests nur bis zur Schicht 3 des OSI/ISO-Referenzmodelles messen, und die höheren Schichten nicht erreichen, führt dies nicht unbedingt zu falschen Ergebnissen. Bei "ping"-Requests werden ICMP-Pakete an Hosts versandt und die Zeit bis zum Eintreffen des Echos gemessen. Das ICMP-Paket wird dabei als Datenteil eines IP-Paketes geführt. Zu jedem ICMP-Paket kommt daher noch ein Header von 8 Byte.

Bei einem HTTP-Request dagegen wird eine TCP/IP-Verbindung aufgebaut. Dabei wird mit der *Sliding-Window* genannten Technik gearbeitet, wobei der Sender eine ganze Reihe an Bytes senden kann, ohne auf eine Quittung warten zu müssen. Über eintreffende Quittungen wird Protokol geführt, so daß die Verbindung erst abbricht, wenn alle Quittungen angekommen sind, wobei die Reihenfolge unwichtig ist.

Dies bedeutet Unterschiede bei den Zugriffsgeschwindigkeiten, welche gemessen werden können. Folgendes Bild zeigt einen Vergleich der Zugriffszeiten über "ping"- und HTTP-Requests⁹ mit 64 Bytes:

⁹ Diese Vergleichsmessung wurde mit dem Skript „pingcheck.pl“ ausgeführt. Das Skript hierzu findet sich im Anhang B.


```

Tera Term - 131.188.3.9 VT
File Edit Setup Control Window Help
-----
[cssun] [18:32] [~/speed] > pingcheck.pl http://www.uni-bremen.de 20
Messung Ping HTTP Differenz
1 31 66 35
2 54 68 14
3 39 44 5
4 63 42 -21
5 103 39 -64
6 47 264 217
7 70 45 -25
8 40 112 72
9 33 69 36
10 60 48 -12
11 56 49 -7
12 57 51 -6
13 109 43 -66
14 93 39 -54
15 105 48 -57
16 51 42 -9
17 50 29 -21
18 31 46 15
19 83 39 -44
20 70 42 -28
-----
62.2 61.2 -1
[cssun] [18:37] [~/speed] >
[cssun] [18:38] [~/speed] > pingcheck.pl http://www.xbaer.com 5
Messung Ping HTTP Differenz
1 88 87 -1
2 82 24 -58
3 80 27 -53
4 103 25 -78
5 86 46 -40
-----
87.8 41.8 -46
[cssun] [18:47] [~/speed] >

```

Bild 10. Vergleich “ping”- und HTTP-Requests

Da die Größe der übertragenen Daten bei “ping”-Request fest vorgegeben, bei HTTP-Requests aber abhängig von der jeweiligen Seite ist, wurden die Zugriffszeiten entsprechend der Bytezahl umgerechnet. Die Tabelle zeigt somit für beide Requests den Zugriff auf 72 Byte (64 + 8 Bytes) an.

Mehrere unabhängige Messungen lassen die mittlere Zeit angeben, mit der sich “ping”- und HTTP-Requests unterscheiden. In der folgenden Tabelle werden die mittleren Requests auf verschiedene, zufällig ermittelte Hosts aufgelistet. Das Ergebnis gibt den mittleren Unterschied zwischen beiden Request-Methoden an.

KAPITEL 3. LÖSUNGSANSÄTZE

Host	Ping (ms)	HTTP (ms)	Differenz (ms)
www.germany.net	112.8	830.4	717.6
www.happyhippo.com	179.1	313.6	134.5
www.heise.de	43.1	348.5	305.4
www.kiss.de	139.1	1169.8	1030.7
www.megabit.de	328.5	661.7	333.2
www.metronet.de	38.7	197.0	158.2
www.nacamar.de	33.1	62.1	28.9
www.netscape.com	112.3	3.5	-108.7
www.ngate.de	36.2	411.1	374.9
www.peri.net	158.7	60.5	-98.2
www.perinet.de	69.6	47.6	-22.0
www.providerliste.ch	495.4	508.1	12.6
www.suchen.de	55.5	27.5	-28
www.uni-berlin.de	25.3	624.0	598.7
www.uni-hamburg.de	26.6	3.6	-23
www.uni-hannover.de	32.3	724.6	692.3
www.uni-kassel.de	169.5	30.4	-139.1
www.uni-sb.de	20.7	475.9	455.2
www.webring.de	128.6	340.3	211.7
www.winfiles.com	267.3	305.0	37.7
www.xlink.de	429.0	28.6	-400.4
www.zdnet.de	179.6	426.7	247.0
Mittel und Fehler	140 (+/- 28) (20 % Fehler)	345 (+/- 67) (19 % Fehler)	205 (+/- 72) (35 % Fehler)

Tabelle 4. Vergleich “ping”- und HTTP-Requests

Die Hosts wurden dabei jeweils 50 mal gemessen, wobei die Zeiten für den Zugriff auf 72 Byte berechnet sind.

Die Tabelle zeigt, daß “ping”-Requests im Durchschnitt bessere Zugriffszeiten auf Hosts erhalten, als HTTP-Requests. Aufgrund des hohen Fehlers (35%) kann man aber nicht auf einen festen oder relativen Wert geschlossen werden, der den Unterschied festlegt. Dieser muß ggf. bei den Einzelmessungen erbracht werden und kann nicht auf andere Hosts aufgerechnet werden. (Dies ist auch logisch, da alle Host unterschiedliche Hard- und Softwarekonfigurationen besitzen, welche nicht vergleichbar sein müssen. Der Host www.netscape.com macht dies besonders deutlich: Diese Seite muß Millionen Hits pro Woche verarbeiten können. Deswegen ist der Server so konfiguriert, daß er HTTP-Requests schneller bearbeiten kann, als andere Requests¹⁰.)

Es bleibt somit festzuhalten, daß die “ping”-Messungen zwar die HTTP-Messungen nicht ersetzen können, aber dennoch als untere Grenze für den HTTP-Zugriff genommen werden können. In anderen Worten: Die Ergebnisse der “ping”-Messungen geben die Mindestzeit an, die ein Zugriff auf einen ausgesuchten Host benötigt.

¹⁰ Bei Netscape erfolgt dies u.a. dadurch, daß ein Großteil der Seiten in einem schnellen Cache liegen und nicht mehr vom Webserver von einer Festplatte gelesen werden müssen.

4. Umsetzung

4.1 Einleitung

Wie im Kapitel 3.4 deutlich wurde, besteht der Ansatz aus mehreren Teilen. So fordert die dezentrale Messung nicht nur eine programmiertechnisch effektive Umsetzung, sondern auch einen großen administrativen und organisatorischen Aufwand. Damit genug Stationen errichtet werden können, muß das Projekt durch Werbung und Presseinformationen bekannt gemacht werden.

Die programmiertechnische Umsetzung muß die Dezentralität und die Unabhängigkeit der Stationen berücksichtigen. Damit ist ein großer Aufwand bei den Algorithmen verbunden, die die Stationen verwalten und die Daten der Stationen auf Fehler oder Manipulationen prüfen. Dennoch muß dies flexibel genug sein, damit neue Stationen jederzeit ohne Probleme aufgenommen werden können.

Somit sind schon im Vorfeld Überlegungen nötig, bei denen diese Probleme berücksichtigt werden. Schließlich müssen Testphasen die aktuelle Version der Umsetzung auf Fehler testen, ohne gleichzeitig den Normalbetrieb zu stören.

In den folgenden Kapiteln werden die Probleme, welche bei der Umsetzung der Arbeit auftraten, besprochen. Dabei wird immer die Zielsetzung des Projektes, welche in Kapitel 1.2 definiert und in Kapitel 3.4 konkretisiert wurde, im Auge behalten.

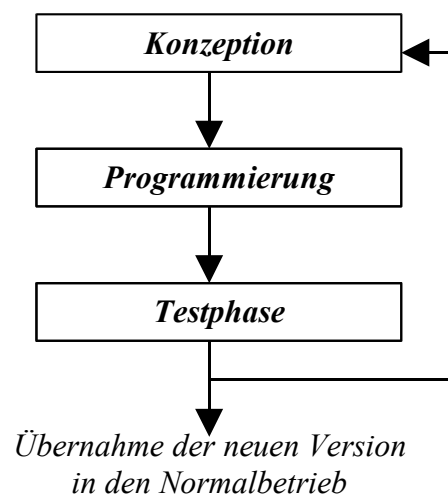
Kapitel 4.2 behandelt die grundsätzliche Planung und Konzeption für die Arbeit. Anhand von Ablaufdiagrammen wird der Weg der Daten und der Messung verfolgt. Kapitel 4.3 baut auf dieser Konzeption aus und bespricht die Programmierung.

Die Programmierung und das Konzept wird durch Tests geprüft. Wie diese Tests aussahen und welche Folgen sie hatten, wird in Kapitel 4.4 besprochen.

Hier wird deutlich, daß Konzeption, programmiertechnische Umsetzung und Testphasen wiederkehrende Arbeitsschritte sind.

Der Übersicht halber werden wir aber nur die bisher erfolgreichste Umsetzung vorstellen, ohne tiefer auf dessen Entwicklung einzugehen.

Zuallerletzt wird in Kapitel 4.5 auf die Öffentlichkeitsarbeit eingegangen, die nötig war, um das Projekt bekannt zu machen.



KAPITEL 4. UMSETZUNG

4.2 Konzeptionelle Umsetzung und Planung

Der dezentrale Charakter der Messung und die Komplexität der Datenerfassung erfordert die Trennung des Projektes in mehrere Teile:

- Die Lokale Meßstation
- Die Zentrale Sammelstelle
- Die Verarbeitung der Daten und Berechnung der Statistik
- Die Online-Darstellung der Daten

Die Trennung des ersten Teils von den letzten drei Teilen wird durch die räumliche Trennung erzwungen. Die Aufteilung der letzten drei Teile dagegen ergibt sich aus der Aufteilung in drei Verarbeitungsphasen innerhalb der Zentrale.

(Eine alte Version der des Projektes hatte noch alle drei Phasen innerhalb eines einzigen Programmes. Hier hatten aber die Testphasen gezeigt, daß das Programm schlecht auf Änderungen im Source-Code reagierte und unübersichtlich wurde.)

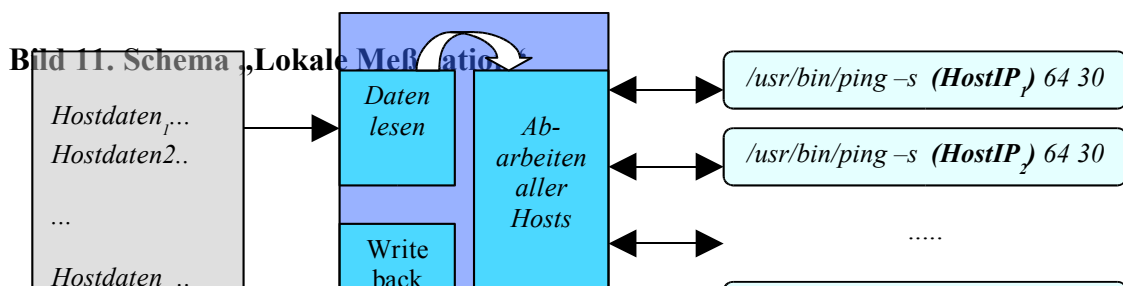
Alle drei Phasen sind unabhängig voneinander. Da sie allerdings auf gemeinsame Datenbestände zugreifen, bzw. auf die Ergebnisse einer vorherigen Phase warten müssen, ist es nötig, daß die Phasen zeitlich nacheinander ausgeführt werden. Im Falle von Tests hat man aber die Möglichkeit die Phasen alleine auszuführen.

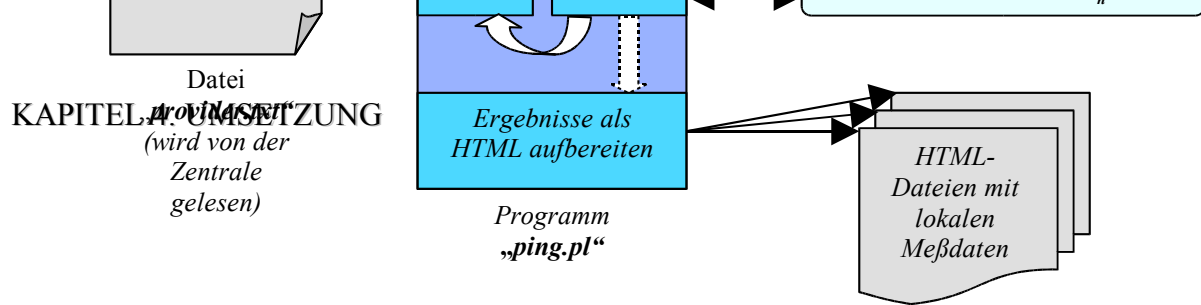
4.2.1 Die Lokale Meßstation

Wie bereits vorher erwähnt, ist die Hauptaufgabe der lokalen Station, verschiedene Hosts zu messen und die Ergebnisse dieser Messungen für die Zentrale zugreifbar zu machen. Als Unteraufgabe muß das Programm der Station optional anbieten, eine lokale Statistik anfertigen zu können. Ablaufschritte sind:

- (1) Auslesen der Datenbank (Datei „provider.txt“)
- (2) Generieren der IP- oder WWW-Adresse aus den Daten und ausführen von „ping“ auf diesen Adressen.
- (3) Zurückspeichern der Daten in die Datenbank.
- (4) Optionales erstellen von HTML-Dateien mit den lokalen Ergebnissen

Die folgende Grafik schematisiert den Programmablauf.





Die Erstellung der Datenbank erfolgt beim erstmaligen Einrichten der Station. Neue Hosts können zur Zeit manuell oder halbautomatisch¹¹ zugefügt werden.

4.2.2 Die Zentrale Sammelstelle

Das Skript, welches die Daten der einzelnen Stationen sammelt, muß sicherstellen, daß die Daten weitgehend fehlerfrei und nicht veraltet sind. Aus naheliegenden Gründen ist die Sammlung der Daten die erste Verarbeitungsphase der Zentrale.

Im einzelnen führt die Sammelstelle, welche im Programm den Namen „getstationdata.pl“ trägt, folgende Schritte aus:

- (1) Liste der Stationen lesen (Enthält auch Fehlerzähler)
- (2) Abarbeiten der einzelnen Stationen:
 - (a) HTTP-Verbindung zur Station aufbauen und Lesen der dortigen Daten
 - (b) Daten und Verbindungsstatus prüfen
 - (c) Wenn keine Fehler aufgetreten sind, werden die Daten in einer stationsspezifischen Datenbank gespeichert und etwaige Fehlerzähler zurückgesetzt.
 - (d) Wenn Fehler aufgetreten sind, wird der Fehlerzähler geprüft. Bei Überschreitung einer Grenze der Administrator benachrichtigt und die Station aus der Liste der Stationen genommen und in einer Fehlerliste vermerkt. Ist der Fehlerzähler noch unterhalb der Grenze, wird er erhöht.
- (3) Zurückschreiben der Stationsliste.

4.2.3 Die Verarbeitung der Daten und Berechnung der Statistik

Die Verarbeitung der Daten und damit die Berechnung der Statistik macht in der Konzeption nur einen geringen Umfang aus. In der späteren programmiertechnischen Umsetzung jedoch wird sich dieser Teil sich als der umfangreichste herausstellen. Dies liegt jedoch weniger an der Komplexität dieser Abarbeitungsphase als vielmehr darin, daß die Zahl der verarbeiteten

¹¹ Halbautomatisch derart, daß die Stationen durch das Anmeldeskript bei der Zentrale jederzeit eine neue Version der Host-Liste zugeschickt bekommen können. Damit dies geschieht, muß der jeweilige Stationsadministrator dies über das Anmeldeskript manuell auslösen.

KAPITEL 4. UMSETZUNG

Datensätze sehr groß ist und die in Kapitel 2 besprochenen mathematischen Formeln auf diese Datensätze angewendet werden müssen.

Folgende Phasen werden durchlaufen:

- (1) Lesen der gesammelten Stationsdaten
- (2) Extrahieren der Hostdaten aus den Stationsdaten
- (3) Speichern der Hostdaten, getrennt nach Hosts
- (4) Berechnung der aktuellen Statistik für einzelne Hosts und Speicherung der Aktualldaten
- (5) Berechnen und Speichern der Tagesdaten aus den Aktualldaten und den vorhandenen Tagesdaten
- (6) Berechnen und Speichern der Monatsdaten aus den Tagesdaten und den vorhandenen Monatsdaten
- (7) Berechnen und Speichern der Jahresdaten aus den Monatsdaten und den vorhandenen Jahresdaten.

4.2.4 Die Online-Darstellung der Daten

Die Online-Darstellung liest die in der vorherigen Phase gespeicherten Daten aus und erstellt daraus HTML-Dateien, welche Online abgerufen werden können.

Folgenden Schritte werden dabei ausgeführt:

- (1) Auslesen der Datei mit den Standardhosts
- (2) Auslesen der Aktualldaten und speichern der Daten der Standardhosts in HTML-Datei über die aktuelle Statistik
- (3) Auslesen der Tagesdaten und speichern der Daten der Standardhosts in HTML-Datei der Tagesstatistik
- (4) Auslesen der Monatsdaten und speichern der Daten der Standardhosts in HTML-Datei der Monatsstatistik
- (5) Auslesen der Jahresdaten und speichern der Daten der Standardhosts in HTML-Datei der Jahresstatistik

Während der Speicherung in HTML-Dateien wird zusätzlich eine zweite Datei erstellt, bei der die Ergebnisse nach der Zugriffsgeschwindigkeit sortiert sind. Bei der ersten Datei sind sie alphabetisch sortiert.

Es werden nur die Daten einer kleinen Liste an Standardhosts ausgegeben, während der Rest der Daten nur mittels eines CGI-Skriptes abgerufen werden können, welches die jeweils geforderten Daten liest und mittels einer HTML-Ausgabe zurückgibt. Die Datei der

KAPITEL 4. UMSETZUNG

Standardhosts enthält eine Liste von Provider-Hosts, welche allgemein bekannt sind und von größerem Interesse sind.

4.3 Programmiertechnische Umsetzung

Die programmiertechnische Umsetzung baut auf die konzeptionellen Planungen und Schemata auf. Wie bereits oben vermerkt, wurde die Programmiersprache Perl verwendet. Dies geschah nicht nur aus dem Grund, daß Perl unter Unix-System weitverbreitet ist, sondern auch wegen dessen besonderen Fähigkeiten bei der Manipulation von Variablen. So machten die Transformationen von textuellen Variablen, welche man von den lokalen Stationen erhielt, in diskrete Werte, aus denen man die Statistik berechnete, keine Probleme mehr.

Im Anhang B ist der Programmcode genau angegeben. Die obige konzeptionelle Planung findet sich innerhalb der einzelnen Programme und Skripten wieder in Funktionsdeklarationen, wie z.B. in dem Skript „speed2html.pl“:

```
...  
&GetZeit;  
&Get_Stations;  
&MakeHostActs;  
&MakeDayStat;  
&MakeMonthStat;  
...
```

In der Funktion *&Get_Stations* erfolgt Schritt (1) der Konzeption, in der Funktion *&MakeHostActs* Schritt (2), usw..

Neben der reinen Umsetzung der Konzeption war das Debugging, sowie die Ermöglichung von Tests ein wesentlicher Anteil an der Programmierung. Hierzu wurden während der Tests Debugging-Ausgaben eingebaut, die den Status eines jeden Schrittes innerhalb des Programmes meldeten.

4.4 Testphasen

Die Tests teilen sich in zwei Arten:

- Test des Programmcodes auf Fehler, und
- Test der Konzeption in Bezug auf die Statistik

KAPITEL 4. UMSETZUNG

Die erste Testart wurde immer dann gemacht, wenn Teile des Programmcodes geändert wurden. So mußte z.B. durch Tests sichergestellt werden, daß das Programm selbständig auf Fehler bei den Stationen reagieren kann oder das es nicht abbricht, falls bei einer Gleichung durch 0 geteilt werden würde. Diese Tests waren in allen Teilen des Codes nötig, konnten aber noch während der Programmierung gemacht werden.

Die zweite Art von Tests dagegen bezog sich auf die Statistik selbst. Es wurden Fragen gestellt, wie „*Wie verändert sich die Jahresstatistik eines Hosts, falls eine neue Station hinzukommt, und ist dies dann richtig?*“, „*Wie reagiert die Statistik, wenn für eine Auswertung keine neuen Daten eines Hosts vorliegen, dieser aber vorher bereits in der Statistik auftrat?*“ oder „*Wie verhält sich die Statistik im Dauerbetrieb?*“.

Solche Fragen ließen sich nicht nur durch einen Blick in die Konzeption bzw. dem Programmcode lösen. So waren längerfristige Tests nötig, bei denen das Skript autonom, wie im Normalbetrieb, ablief, aber die Daten später manuell auf ihre Richtigkeit überprüft wurden. Der längste Dauertest lief 6 Wochen, von Anfang Dezember bis Mitte Februar. Seit Mitte Februar 98 blieb die Struktur der Datensätze gleich. (Siehe hierzu auch Anhang B, Kapitel 3.4).

Neben der Auswertung der längerfristigen Ergebnisse, wie der Auswertung der Jahresstatistik, werden außerdem öfters Stichproben gemacht, bei denen geprüft wird, ob die berechneten Werte stimmen. Unterstützt wird dies auch durch das Feedback von Benutzern der Webseite. Viele Ideen und Vorschläge wurden bereits gemacht wovon einige verwirklicht wurden, wie z.B. die Sortierung der Standard-Hostliste sowohl nach dem Alphabet, als auch nach der Zugriffszeit.

4.5 Organisatorische und Administrative Umsetzung

Da das Projekt auf die Mithilfe von anderen Webmastern angewiesen ist, die sich bereit erklären als Station an der Messung teilzunehmen, ist eine gute Öffentlichkeitsarbeit notwendig. Dies geschah insbesondere durch folgende Aktionen:

1. Eintragung der Webseite in nationale und internationale Suchmaschinen
2. Setzen eines Links von einer gut besuchten Seite (Realm of CGI) auf die Internet Speed-Seite.
3. Verfassung und Versendung einer Pressemitteilung an Nachrichtenorganisationen, Magazine und E-Zines.

KAPITEL 4. UMSETZUNG

4. Verfassung und Versendung von Informationen über das Projekt an Provider.
(Bisher nur für deutschsprachige Provider geschehen)
5. Bekanntgabe der in der Statistik besten Hosts und dem besonderem Gewicht der Statistik auf Provider.
6. Vorstellung des Projektes im Rahmen von Vorträgen beim „Tag der offenen Tür“ des RRZE (am 19. und 20. Juni).

Besonders der Link auf der Seite des „Realms of CGI“¹² machte das Projekt in der Anfangsphase schnell bekannt.

Zur weiteren Verbreitung führten auch einige Artikel in Computer- und Internetzeitschriften, welche zwar nicht immer direkt über das Projekt selbst berichteten, aber einen Link hierauf anführten. Mit dem kommenden Erscheinen der halbjährlich herausgegebenen Zeitschrift „Online Professional“ kann eine weitere Verbreitung erwartet werden, da in dieser ein Interview mit mir zu finden sein wird, wo es unter anderem um das Projekt geht.

Neben der Bekanntgabe des Projektes durch Werbung besteht die administrative Arbeit aber auch in der Überwachung der laufenden Stationen und der Beantwortung von Anfragen jeglicher Art. So kommen wöchentliche mehrmals Anfragen, diesen oder jenen Host mit in die Messung aufzunehmen, Auskunft darüber zu geben, welche besonderen Dienstleistungen ein spezieller Host noch bietet, usw..

Die ganze Vielfalt der Aufgaben eines Webmasters müssen bewältigt werden. Natürlich lassen sich insbesondere Mailanfragen ebenfalls weitgehend automatisieren, aber davon wird in diesem Stadium des Projektes noch abgesehen.

¹² Das Realm of CGI ist eine Site, welche sich mit der Programmierung von CGI-Programmen beschäftigt. Es ist zur Zeit die umfangreichste deutschsprachige Webseite auf diesem Gebiet. Die Site verzeichnet bis zu 15000 Besucher täglich.

5. Vorstellung der Webseite und Fallbeispiele

5.1 Einleitung

Der Abruf der Daten erfolgt über eine Webseite, welche unterhalb der URL

<http://speed.xbaer.com> zu finden ist. Die Webseite gibt dem Benutzer zum einen eine

Schnellübersicht über eine Liste von 25 ausgewählten Hosts, als auch diverse Suchmasken, bei der einer oder mehrere Hosts in der gesamten Datenbank gesucht werden können.

Bevor wir spezielle Fallbeispiele angeben, wird in Kapitel 5.2 die derzeitige Oberfläche vorgestellt¹³.

In den Kapiteln 5.3 bis 5.6 werden anhand der bekannten Onlinedienste AOL, CompuServe, Metronet und T-Online Fallbeispiele aufgeführt.

¹³ Die Anpassung dieser Oberfläche an die neuesten Standards bei Webseiten ist bereits in Planung. Sie wird Ende Juli bereits ein neues Outfit besitzen, welches aber nichts an der Funktionalität ändert. Lediglich das Design und die Benutzerführung wird verbessert. Dies ist aber nicht mehr Teil der Studienarbeit, sondern wird zu dem Projekt „Internet Speed“ gehören, welches ich in Eigenregie weiterführe.

5.2 Vorstellung der Webseite

5.2.1 Die Startseite



Bild 12. Startseite

Das Bild zeigt die Einstiegsseite, wenn man die URL <http://speed.xbaer.com> gewählt hat. Die einzelnen Unterseiten werden über die Navigationsleiste auf der linken Bildschirmhälfte ausgewählt.

5.2.2 Die aktuelle Messung

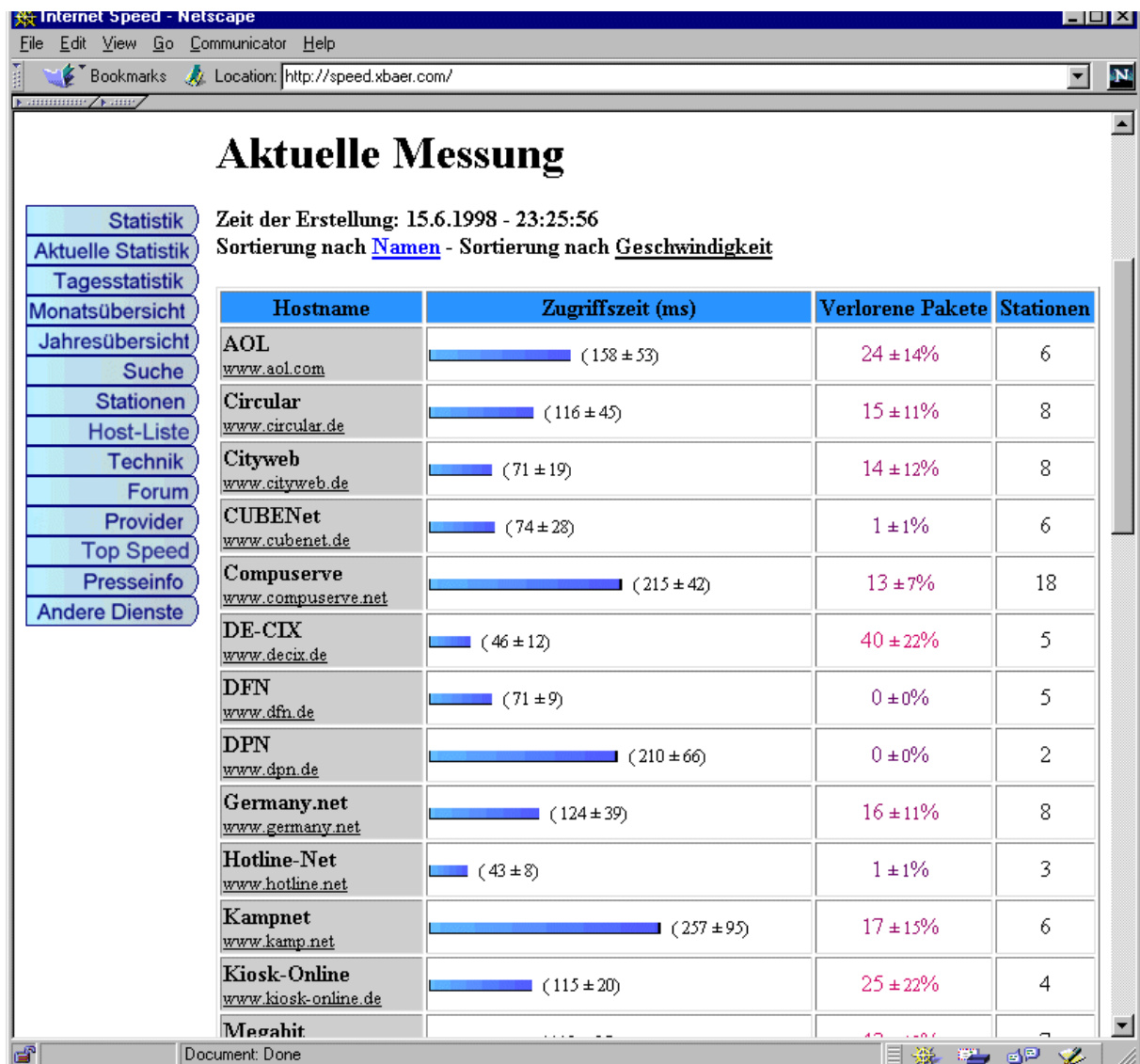


Bild 13. Statistik der jüngsten Datenerfassung

Dieses Bild zeigt die Statistik, welche sich alleine aus den Daten der letzten Erfassung ergeben. Es wird an dieser Stelle nur eine kleine Teilliste an Hosts ausgegeben.

5.2.3 Die Tages-, Monats- und Jahresübersicht

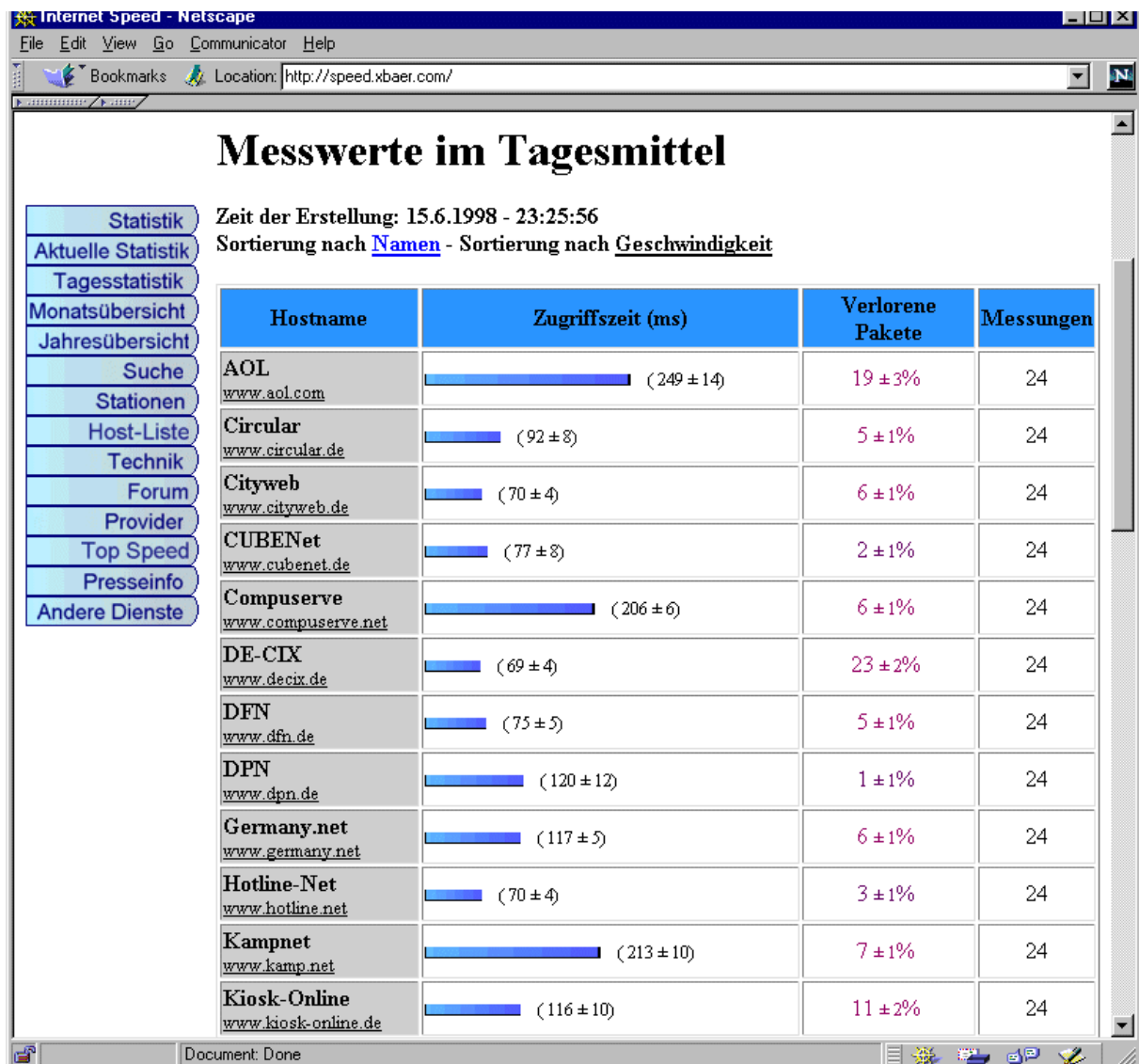


Bild 14. Tagesübersicht

Hier werden die Hosts, die auch in der Aktuellen Messung gezeigt worden in der Statistik über den gesamten Tag gezeigt. (Bzw. über die Daten, die man vom Tag bereits hat.)

Bei der Monats- und der Jahresübersicht sieht die Ausgabe gleich aus, nur die Werte sind entsprechend anders.

5.2.4 Die Host-Liste

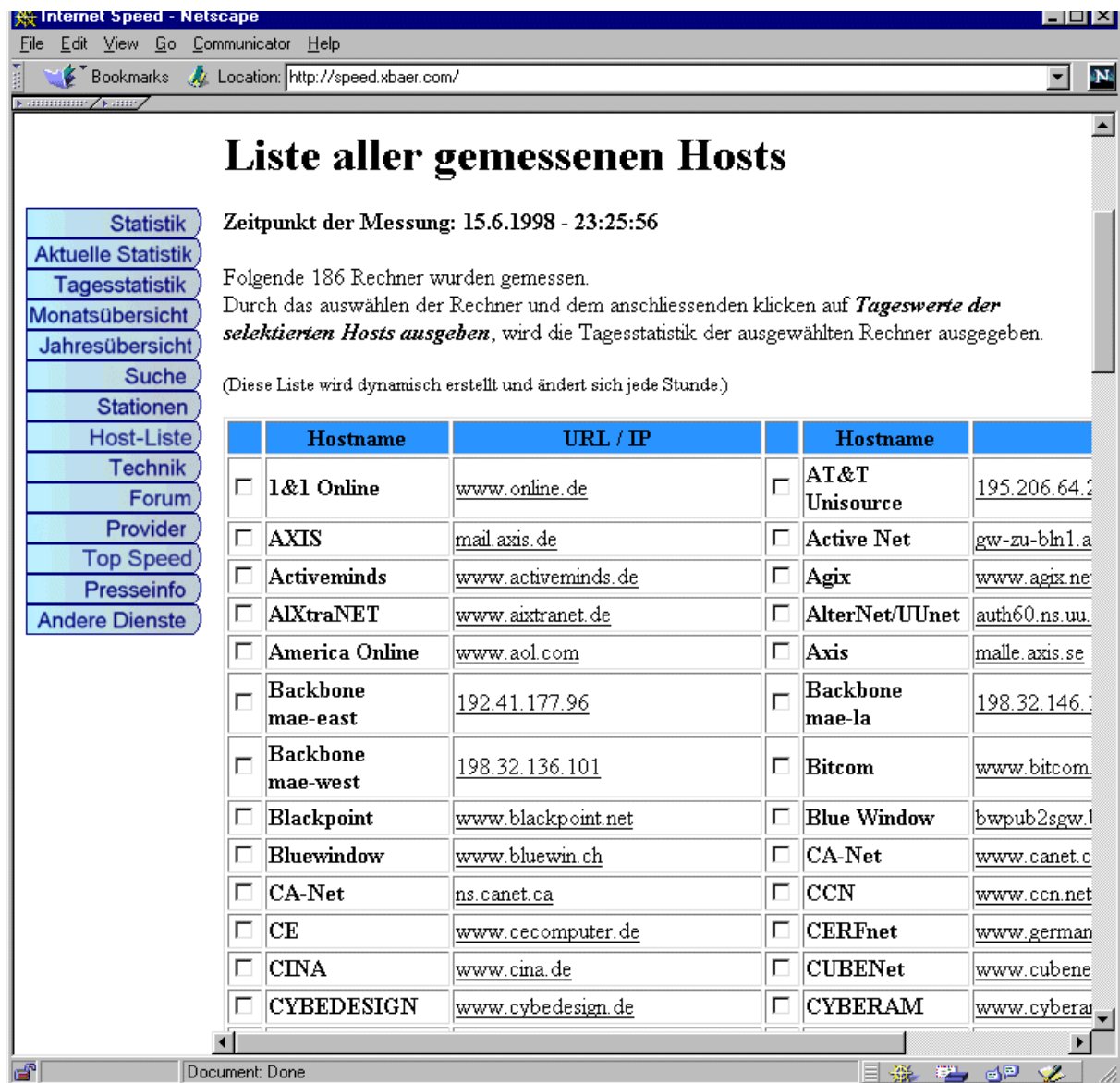


Bild 15. Host-Liste

Diese Liste zeigt alle Hosts an, welche bei der letzten Auswertung erfaßt wurden. Man kann sich anhand dieser Liste detaillierte Ausgaben geben lassen, indem man die einzelnen Hosts selektiert und dann durch das „klicken“ auf den Einschaltknopf die Wiedergabe startet.

KAPITEL 4. UMSETZUNG

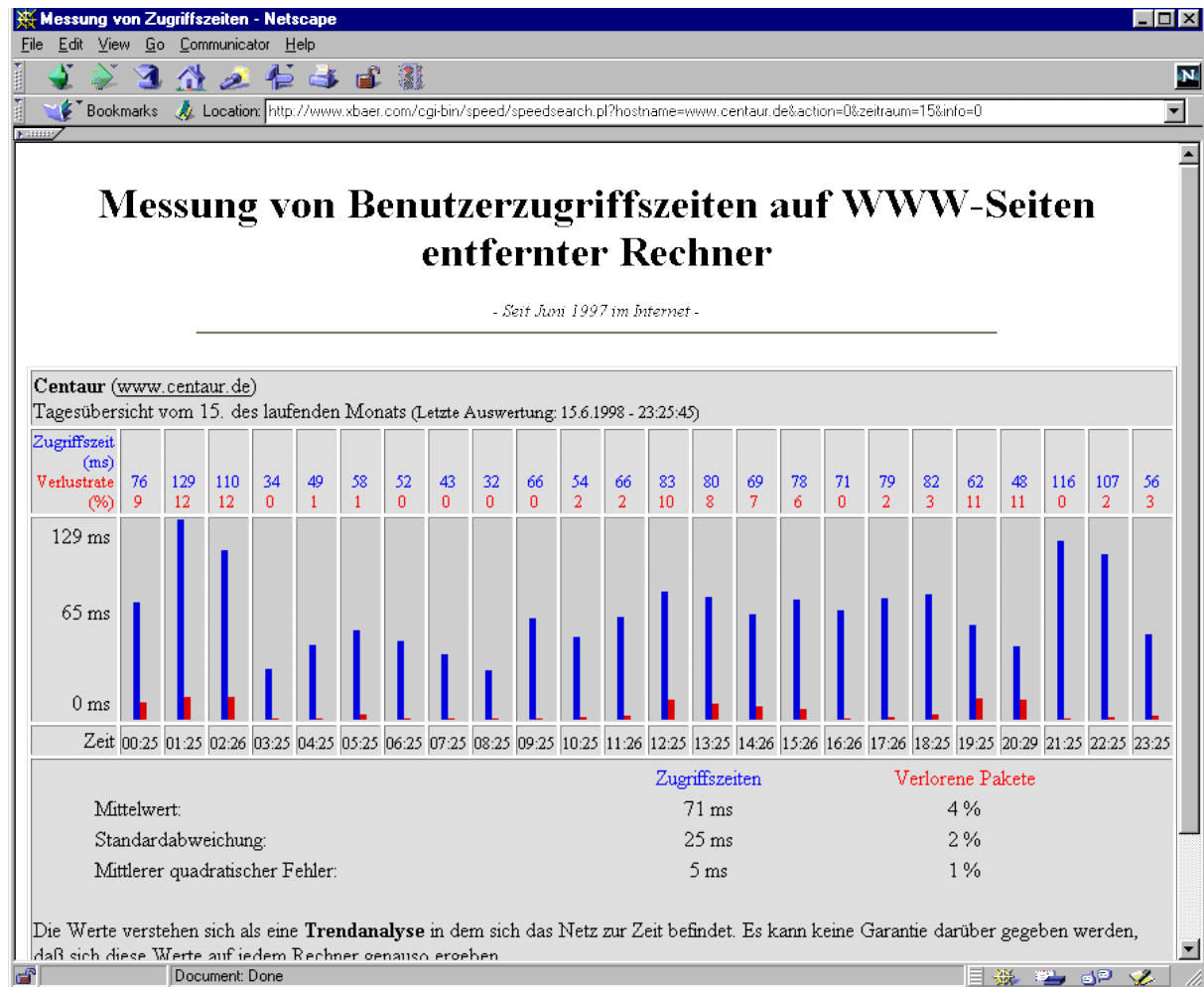


Bild 16. Daten eines Hosts

Nach der Auswahl eines Hosts werden die Daten abgerufen und on-the-fly in eine gut lesbare Form gebracht.

5.2.5 Die Suche nach Hosts



Bild 17. Suche

Auf dieser Seite lassen sich die Daten einer oder mehrerer Stationen ausgeben und vergleichen. Das folgende Bild zeigt ein typisches Ergebnis:

KAPITEL 4. UMSETZUNG

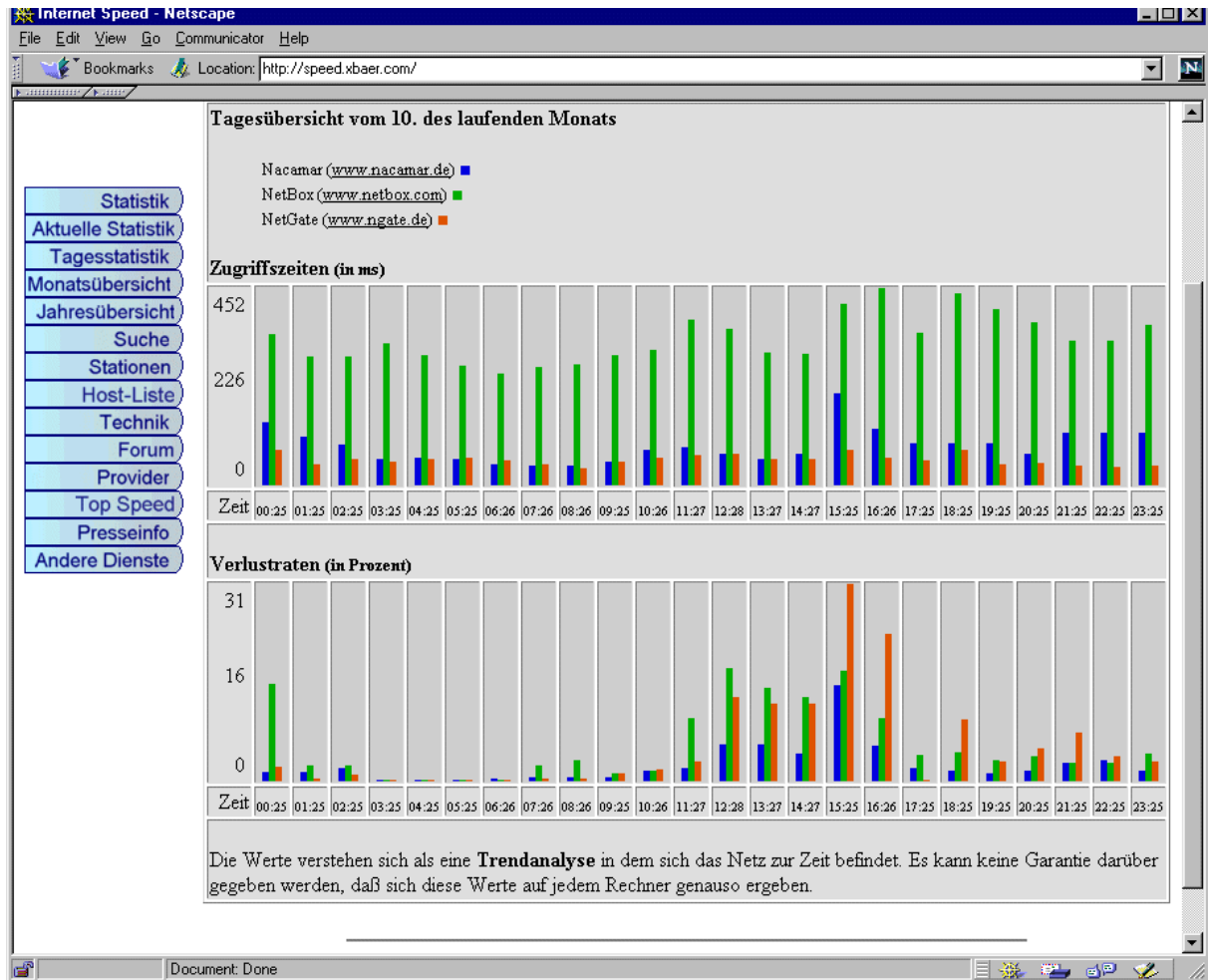


Bild 18. Daten mehrerer Hosts

5.2.6 Die Liste der Stationen

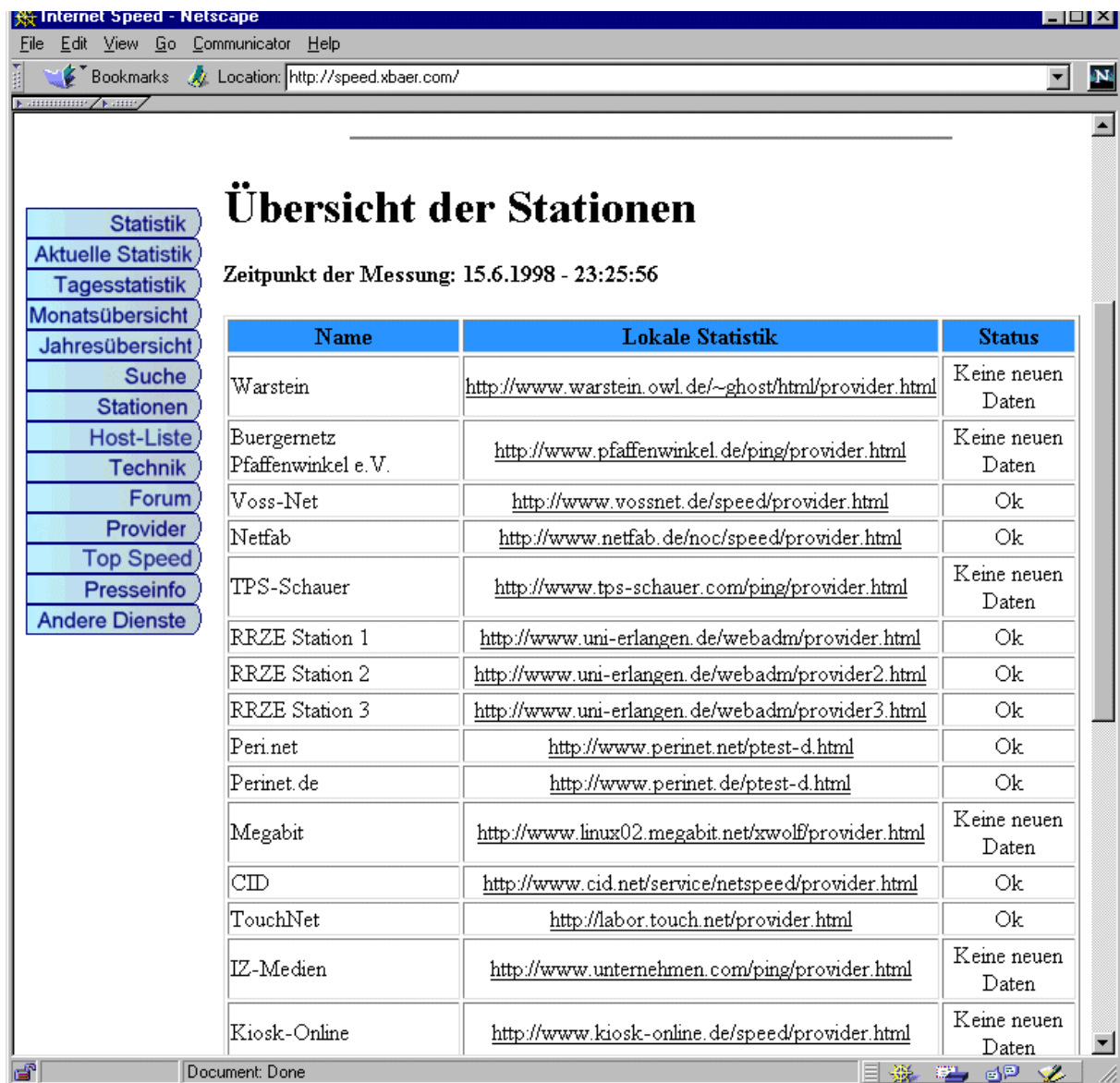


Bild 19. Stationen

Das Bild zeigt die Übersicht über die Stationen und deren Status. Zusätzlich sind Links auf die stationseigenen, lokalen Statistiken vorhanden.

5.2.7 Sonstiges

Die sonstigen Menüpunkte der Oberfläche dienen lediglich der Information. Unter dem Punkt „Provider“ findet sich eine Suchmaske, in der sich die textuellen Informationen finden lassen, welche einzelne Provider von sich gaben.

Der Punkt „TopSpeed“ zeigt eine Liste der Hosts an, welche in den vergangenen Monaten die Statistik angeführt hatten, d.h. welche Hosts den schnellsten Zugriff hatten.

KAPITEL 4. UMSETZUNG

Unter „Technik“ und „Presseinfo“ finden sich erklärende Texte zur Arbeit, die aber nicht an diesen Text heranreichen. Der Text dieser Ausarbeitung wird die bisherigen Informationen ersetzen.

5.3 Fallbeispiel 1: AOL im Monat Mai

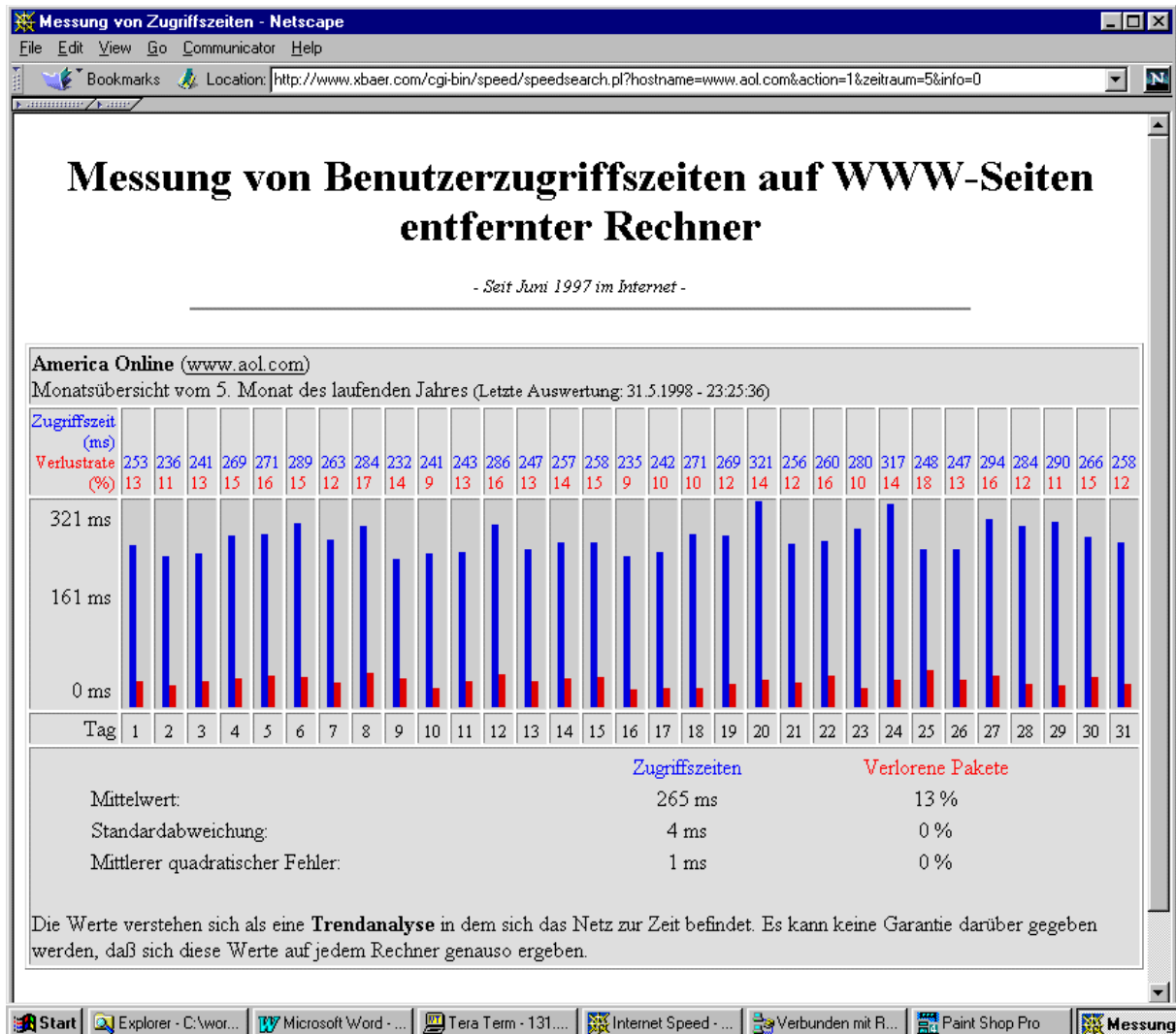


Bild 20. AOL im Monat Mai

AOL zeigt bei der Monatsübersicht ein typisches Verhalten: An den Wochenenden und Feiertagen (1.-3., 9.-10., 16.-17., 21.-23., 30.-31.) sind die Zugriffszeiten besser, während sich die Zugriffszeit in der Woche verschlechtert. Bemerkenswert dabei ist, daß Sonntags bereits ein Anstieg zu erkennen ist.

5.4 Fallbeispiel 2: CompuServe Deutschland und AOL

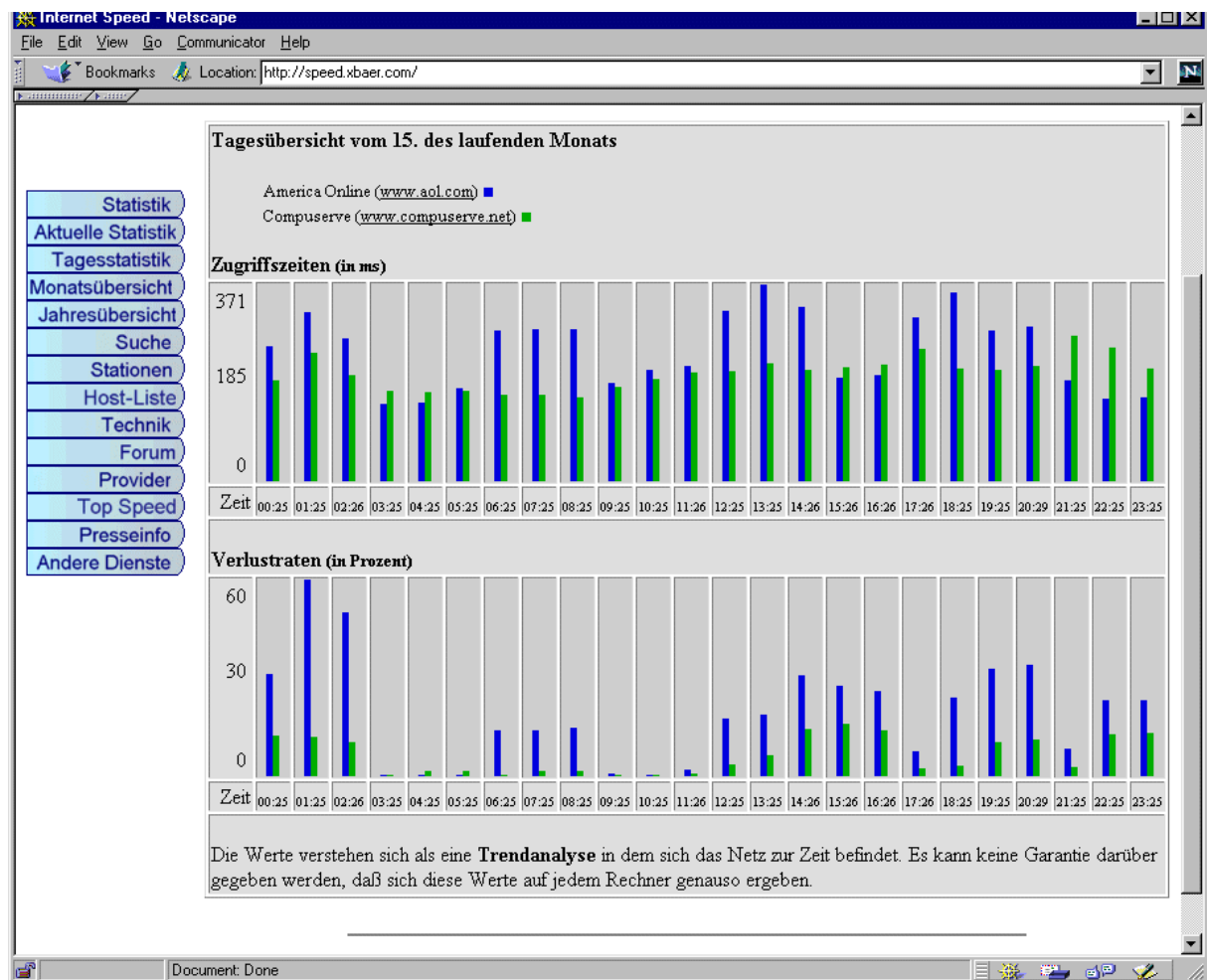


Bild 21. CompuServe und AOL im Tagesvergleich

Aus dem Tagesvergleich von AOL und CompuServe erkennt man, daß CompuServe über den Tag hinweg viel stabiler auf Anforderungen reagiert, während bei AOL Probleme zu bestimmten bevorzugten Zeiten vorherrschen.

5.5 Fallbeispiel 3: Metronet

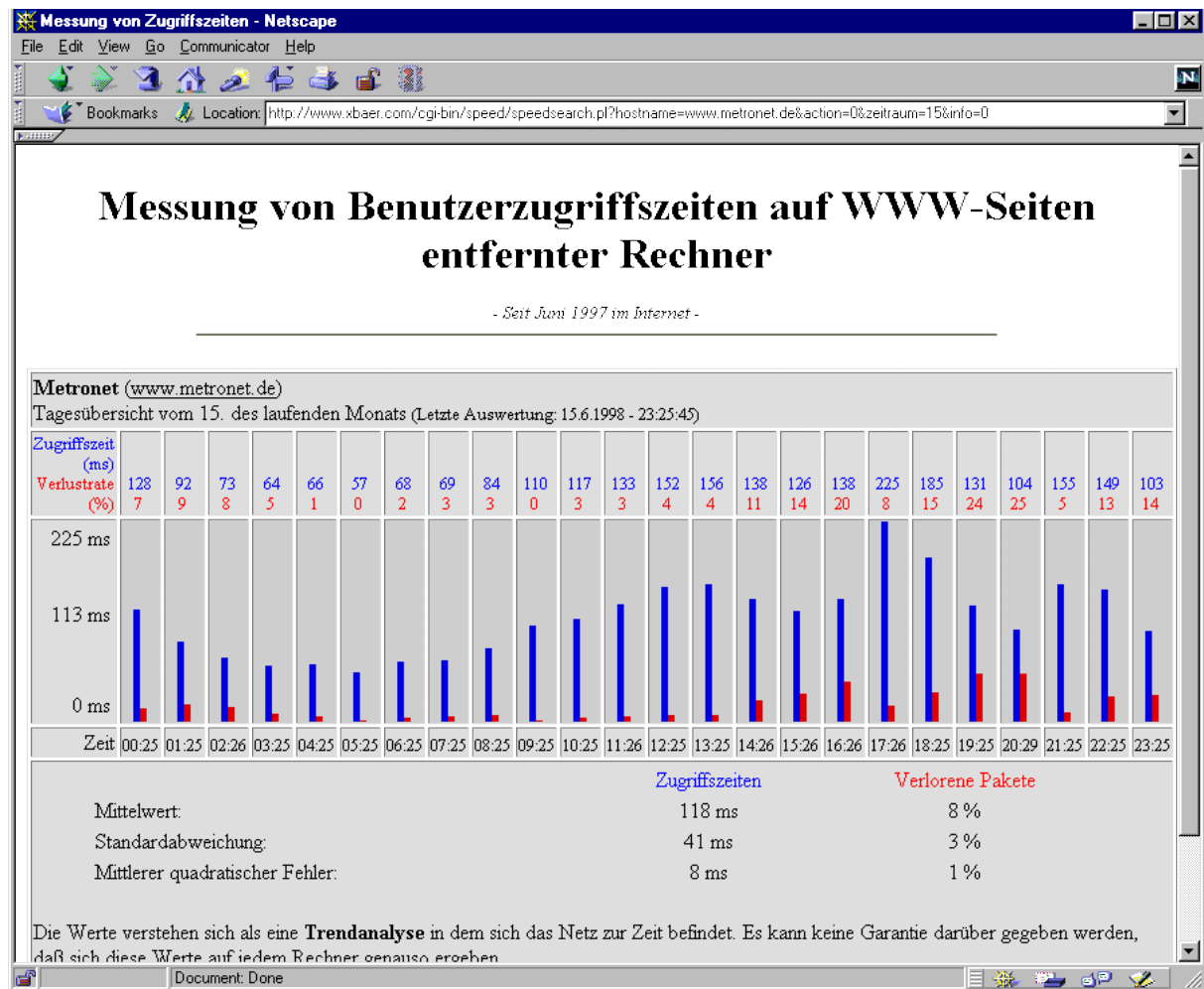


Bild 22. Tagesübersicht Metronet

Metronet zeigt den typischen Verlauf eines nationalen Providers: Er wird nicht, wie z.B. AOL und CompuServe durch Zeitgrenzen im internationalen Raum beeinflusst. Besonders deutlich wird dies auch bei dem Anstieg der Zugriffe am Abend: So findet sich in der Zeit zwischen 19.00 und 20.00 Uhr ein Tal, der sich dadurch erklären läßt, daß viele Leute lieber bis nach 21.00 Uhr warten, um dann zum günstigeren Telekom-Tarif die Telefonleitungen zu benutzen. Der Anstieg um 18.00 läßt sich durch die Leute erklären, die nicht bis 21.00 Uhr warten können, um ihre EMail's downzuladen. Die Steigerung über den Tag, von morgens nach 8.00 bis um 15.26 lassen sich durch den Datenzugriff aus Büros und Organisationen erklären.

5.6 Fallbeispiel 4: Die 5 großen Provider im Tagesvergleich

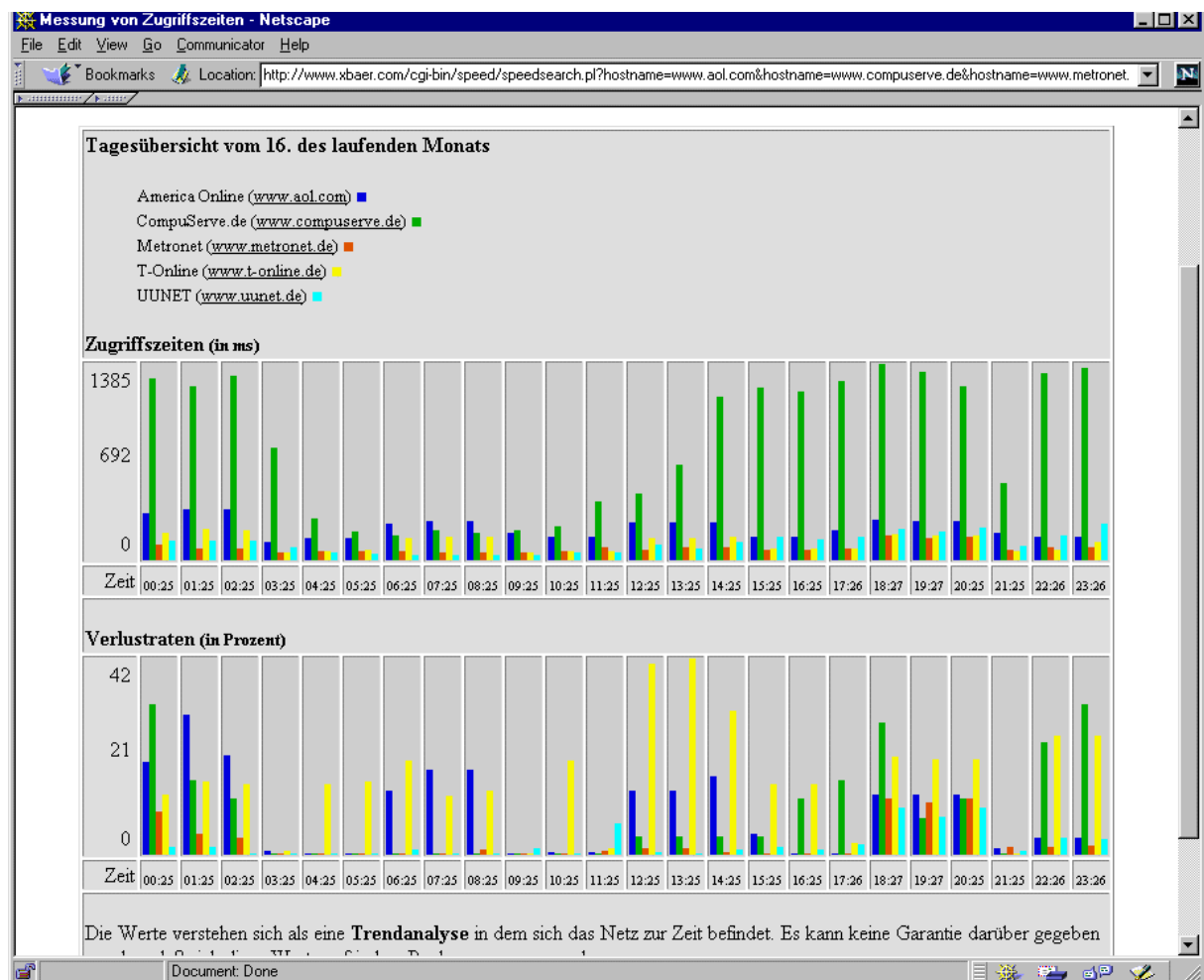


Bild 23. Die 5 großen Provider im Tagesvergleich

Das Bild zeigt 5 der größten Provider in Deutschland. Man erkennt, daß alle über den Tag hinweg vergleichbare Abläufe haben. Lediglich bei der Zugriffszeit von CompuServe zeigt sich, daß dieser zu den Hauptverkehrsaufkommen große Probleme hat.

Bei den Verlustraten dagegen fällt auf, daß T-Online Probleme bei der Qualität der Datenübertragungen hat.

5.7 Fallbeispiel 5: Die 5 großen Provider im Monatsvergleich

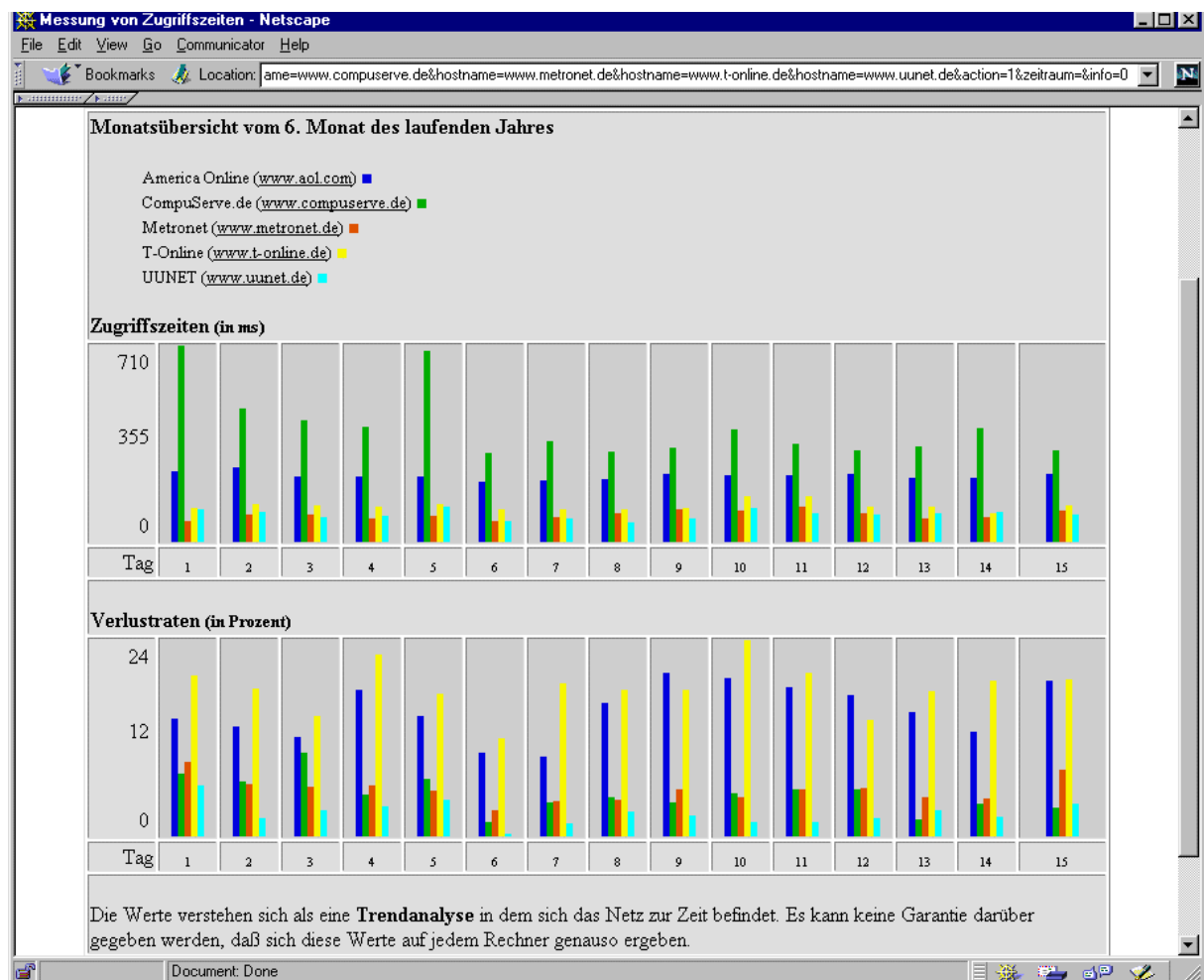


Bild 24. Die 5 großen Provider Anfang Juni

Wie bereits in der Tagesübersicht deutlich wurde, kann CompuServe nicht mit den anderen Providern vom Zugriff her mithalten. Schlechte Zugriffszeiten zeigt auch AOL.

Bei beiden kann dies dadurch erklärt werden, daß diese Firmen ihre Hauptrechner in Amerika haben und deswegen die Informationen weitere Wege nehmen müssen.

Die Verlustraten zeigen, wie auch bei der Tagesübersicht, daß T-Online und AOL auch hier Probleme haben.

Alle Verläufe sind außerdem beeinflusst durch die Wochentage: Die schlechtesten Zugriffszeiten bei allen Hosts sind die Werktage.

6. Zusammenfassung und Ausblick

Gegenstand der Arbeit war die Messung der Zugriffszeiten auf Rechner im Internet. Zu diesem Zweck wurde ein Netz von Stationen entwickelt, welche dezentral und unabhängig voneinander Messungen ausführen und die Ergebnisse einer zentralen Auswertung zur Verfügung stellen.

Im zweiten Teil der Arbeit wurden mathematische Formeln vorgestellt, welche zur Berechnung der Statistiken herangezogen worden. Auch wurde gezeigt, daß das verwendete Verfahren, mittels „ping“-Requests die Zugriffszeiten zu messen, ausreichend und zulässig ist.

Die Umsetzung der Arbeit war im starken Maße durch die Struktur des Internets beeinflusst. So war es unabdingbar, daß die Messungen dezentral durch viele unabhängige Rechner durchgeführt wurden. Die Verwendung eines einzigen Rechners als zentrale Meßstation, so wie es bei vielen anderen Seiten im Netz vorgeschlagen wird, hätte keine ausreichend nützlichen Ergebnisse erbracht.

Dadurch wurde die Organisation und die Bekanntmachung bereits vor der endgültigen Fertigstellung zu einem sehr wichtigen Bestandteil der Arbeit. Soweit möglich, mußten die Anmeldung neuer Stationen, die Überprüfung von Meßdaten, sowie die Erstellung von aktuellen Informationsseiten, automatisiert werden. Es wurde deutlich, daß eben durch den hohen Grad an Automatisierung, diese Arbeit sich selbst durch die Aufnahme neuer Stationen und gemessener Hosts verselbstständigt hat. Während der Erstellung dieser Niederschrift wurden halbautomatisch 2 neue Stationen und vollautomatisch 14 neue Hosts zur Messung aufgenommen.

Durch die zunehmende Zahl der Stationen wurde dabei nicht nur die Genauigkeit der Messung erhöht, sondern auch die Messung als solche bekannter. Bei den Providern in der Schweiz wurde dies besonders bemerkbar: Waren noch vor 3 Monaten nur 3 schweizer Provider in der Messung, stieg die Zahl in den letzten Wochen durch viele Anträge auf Aufnahme nahezu lawinenartig auf zur Zeit 17 Provider an.

Obwohl die Aufgaben der Arbeit erfüllt wurden und sie bereits ein Entwicklungsstand erreicht hat, der es erlaubt, sie an dieser Stelle eigenständig weiterlaufen zu lassen, ist geplant, aus ihr ein eigenes Webprojekt zu machen. So gibt es noch viele Erweiterungsmöglichkeiten, das zukünftige Projekt „Internet Speed“ zu verbessern:

- Messung über HTTP-Requests. Neben den bisherigen „ping“-Requests wird mit Hilfe von zusätzlichen HTTP-Requests eine noch genauere Zugriffszeit bestimmt.

KAPITEL 4. UMSETZUNG

- Genauere Aufschlüsselung der Statistik. Hier sollte die Statistik die Möglichkeit bekommen, auch ganz bestimmte Zeitintervalle, sogenannte Zeitfenster, detailliert darzustellen und auszuwerten.
- Berechnung von Prognosen und Abschätzungen aus den vorhandenen Datensätzen um auf zukünftige Zugriffszeitverhalten zu schließen.
- Verbesserung der Oberfläche, um ein noch ergonomischeres und intuitiveres Abrufen der Daten zu ermöglichen.
- Ausdehnung der Messung auf andere Bereiche des Internets.
- Berücksichtigung von Dialin-Zugriffszeiten nach dem Vorbild der c't-Messung
- Berechnung eines Internet-Indices, welches die tatsächliche Auslastung der erfaßten Teils des Internets angibt.

Das Projekt „Internet Speed“, welches mit dem Ende dieser Studienarbeit offiziell ins Leben gerufen wird, wird einige dieser Punkte umsetzen.

War das Anfangsziel der Arbeit nur die Möglichkeit, die Zugriffsgeschwindigkeit zu messen, mit der ein Benutzer auf die Webseiten entfernter Rechner zugreifen kann, hat sich daraus die Möglichkeit entwickelt, die Geschwindigkeit und die Auslastung von Teilen des Internets darzustellen.

Hierauf aufbauend bietet sich ein weites Feld für weitere Arbeiten. Ganz besonders auf dem Gebiet der Statistik: Waren vor dieser Arbeit nur Statistiken über die Zahl der ans Internet angeschlossenen Rechner und dem Durchsatz von einzelnen Datenleitungen möglich, haben wir nun die Chance, die Entwicklung des Internets auch durch Zahlen über Zugriffszeiten zu bewerten.

Die Frage „Ist das Internet schneller geworden?“ wird in Zukunft beantwortbar werden. Aber wird uns die Antwort gefallen?

Anhang A

I. Pressemitteilung vom 23. März 1998

PRESSEINFORMATION

betreffend:

"Messung von Benutzerzugriffszeiten auf WWW-Seiten entfernter Rechner"
<http://speed.xbaer.com>

Das Projekt "Messung von Benutzerzugriffszeiten auf WWW-Seiten entfernter Rechner" ist ein Dienst, bei dem die minimalen Benutzerzugriffszeiten von Providern und anderen Rechnern im Internet (wie Backbones und Router) gemessen werden.

Der Dienst ist kostenfrei und seit mehr als einem halben Jahr im Netz.

Es werden zur Zeit über 120 verschiedene Provider und andere Rechner gemessen, mehr als bei irgendeinem anderen ähnlichen Dienst dieser Art.

Das Projekt "Messung von Benutzerzugriffszeiten auf WWW-Seiten entfernter Rechner" entstand aus einem einfachen Ping-Skript, das nur von einem einzelnen Provider ausgehend ermittelte, wie schnell dieser ist. Auch heute wird dieses Verfahren noch häufig angewendet, z.B. bei Internet Weather Report (<http://www.internetweather.com/>), Internet Health Report (<http://www.digitaldaze.com/support/nethealth/>) und bei vielen mehr. Bei all diesen Angeboten besteht der Nachteil nun darin, daß jeweils nur lokale Messungen durchgeführt werden, die zwar für die Kunden des jeweiligen Providers interessant sein mögen, aber keinen Vergleich zwischen verschiedenen Providern erlauben.

Im Mai 1997 entstand deshalb die Idee, diese Messungen zusammenzuführen und eine gemeinsame Statistik aufzubauen, die für jeden Internetbenutzer von Aussagekraft ist.

Die Version 1.0 des Meßprogramms ging im Juni ans Netz; Zunächst beschränkt auf deutsche Provider und Hosts. Damit war damals die erste deutschsprachige Seite ins Leben gerufen, die einen solchen Service anbot. Anderslautende Meldungen, die aussagen, daß die Seite des "Internet-Wetters" als erste solch einen Dienst in Deutschland aufmachte, sind falsch.

In der Zeit von Juni bis Oktober 1997 wurde das Programm mehrfach verbessert und das Netz der Stationen ausgebaut. Ab September 1997 bis Anfang 1998 fand dann der erste große Dauerbetrieb statt, bei dem unter Realbedingungen Messungen durchgeführt wurden und Fehler und Probleme verifiziert und beseitigt werden konnten.

Seit Januar 1998 läuft die Messung von Benutzerzugriffszeiten nunmehr dauerhaft, daß heißt ohne Unterbrechung durch interne Änderungen. Zur Zeit verfügt der Dienst über 17 unabhängige Stationen in Europa, welche u.a. von namhaften Providern wie Germany-Net oder TouchNet betrieben werden.

KAPITEL 4. UMSETZUNG

Doch damit ist noch nicht das Ende der Entwicklung erreicht. In Zukunft soll eine zweite Meßmethode integriert werden, die sich nicht alleine auf das "pingen" von Rechnern verläßt, sondern auch zusätzliche Verfahren berücksichtigt. (Sobald diese neuen Teile fertiggestellt sind, wird darüber berichtet werden. Bitte haben Sie Verständnis dafür, daß wir uns in Hinblick auf einige neue Angebote im Netz, die offensichtlich Teile dieses Services kopierten und als "Neuheit" selbst anboten, gezwungen sehen, die Details zunächst geheimzuhalten).

Weitere Informationen und Details über die Technik finden sich unterhalb der oben genannten Seite <http://speed.xbaer.com/> .

Presseinformation vom 23. März 1998 von Wolfgang Wiese, XBaer Comm.
EMAIL: xwolf@xwolf.com

II. Faltblatt zum Vortrag „Provider im Vergleich“, 19. und 20. Juni 1998

Bewegen Sie den Mauszeiger auf diesen Punkt und drücken Sie die linke Taste der Maus.



Ein Messungsbildschirm zeigt eine Auswahl-Liste mit gemessenen Providern angegeben ist. Suchen Sie aus dieser Liste Ihren Provider aus, "klicken" diesen an und bestätigen Sie dann die Eingabe.



Provider im Vergleich

Wie schnell ist mein Provider? Die Antwort auf diese Frage erhalten die Besucher in stündlich durchgeführten Tests, die aktuell am Tag der offenen Tür im RRZE gezeigt werden. Dem Provider-Ranking werden die großen kommerziellen Internetdiensteanbieter unterzogen, aber auch das RRZE als Dienstleister der Wissenschaft stellt sich dem kritischen Vergleich.

Vorfürhungen, *Fr/Sa*: 14.00 und 16.00 Uhr
SUN-CIP-Raum 1.153
Vortrag: W. Wiese

Impressum:
Herausgeber:
Regionales Rechenzentrum Erlangen
Friedrich-Alexander-Universität Erlangen-Nürnberg
Martensstraße 1, 91058 Erlangen
Juni 1998
Redaktion:
Thomas Deckert
thom@ies.rrze.uni-erlangen.de

Regionales Rechenzentrum Erlangen
Friedrich-Alexander-Universität Erlangen-Nürnberg



RRZE - informiert

Tage der offenen Tür
19. + 20. Juni 1998

Provider im Vergleich



KAPITEL 4. UMSETZUNG

Provider im Vergleich

oder
 Frust statt Lust im Internet
 - wenn die Datenautobahn zum
 Trampelpfad wird ...

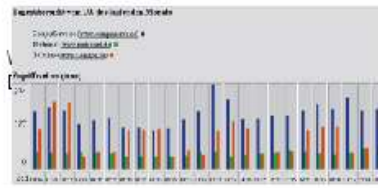


Jeder der im Internet surft, hat es schon erlebt: Lange Wartezeiten, bis die gewünschte Seite endlich vollends auf dem Bildschirm erscheint.
 Warum dies so ist, und ob Sie etwas dagegen machen können, erfahren Sie im Vortrag "Provider im Vergleich".

Hier zeigen wir Ihnen, daß die Wartezeiten nicht immer nur an lang-samen Modems, ISDN/ADSL-Karten oder schlechten Software-Konfigurationen liegen, sondern zumeist an dem "Service Ihres Vertrauens", dem Internet-Provider.

Wir werden live Messungen mitverfolgen, bei denen verglichen wird, wer schneller ist: T-Online, AOL, CompuServe, Metronet oder jemand anderes. Das Ergebnis könnte Sie überraschen (?).

Wer ist der schnellste Provider ?



Provider	Response Time (ms)	Score	Rating
CompuServe	1140	3405	1.5
Metronet	760	4405	1.5
Time	2140	3405	1.5
CompuServe	760	3405	1.5
CompuServe	610	3405	1.5
CompuServe	1300	3405	1.5
CompuServe	1400	4405	1.5
CompuServe	1400	3405	1.5
CompuServe	1200	5405	1.5
CompuServe	760	4405	1.5
CompuServe	2140	4405	1.5
CompuServe	760	7405	1.5

Angebote für eigene Homepages.

AOL <http://www.aol.com>

CompuServe <http://www.compuserve.com>
 Metronet <http://www.metronet.de>
 MSN <http://www.msn.de> (eingestellt)
 T-Online <http://www.t-online.de>

Presence- oder WebSpace-Provider
 Diese Provider stellen keine oder nur eingeschränkt Einwahlmöglichkeiten zur Verfügung und bieten hauptsächlich die professionelle Vermietung und Pflege von Homepages oder Unternehmensseiten an.

Easynet <http://www.easynet.de>
 Geocities <http://www.geocities.com>
 Online Dienste Lümburg <http://www.odh.de>
 Peinet <http://www.peinet.de>
 XBaer Communications <http://www.xbaer.com>

Wie schnell ist mein eigener Provider?

Auf der Seite <http://speedtest.com> finden Sie den dauerhaften Vergleichstest von Providern und Dienstleistern im Internet. Auf der linken Menüliste sehen Sie den Punkt "Suche".



KAPITEL 4. UMSETZUNG

III. Netzausbau vom Netzprovider Xlink 1998



Quelle: Informationsseiten Xlink: <http://www.xlink.de>

Literatur- und Adressenverzeichnis

I. Literatur

- „Moderne Datenkommunikation – Eine strukturierte Einführung“, 2. Aufl.; Internat. Thomson Publ., 1997, ISBN 3-8266-4016-0, Franz-Joachim Kauffels
- „Verteilte Systeme unter UNIX: Eine praxisorientierte Einführung“, Vieweg, 1992, ISBN 3-528-05194-9, Werner Brecht
- „Das Internet Handbuch für Windows“, dpunkt, 1995, ISBN 3-920993-10-1, Karl Obermayer
- „Web Security – A Step-by-Step Reference Guide“, Addison Wesley, 1998, ISBN 0-201-63489-9, Lincoln Stein
- „Computernetzwerke“, 3. Aufl.; Prentice Hall, 1997, ISBN 3-8272-9536-X, Adrew Tanenbaum
- „Das Internet: Weltweit vernetzt“, IWT Verlag, ISBN 3-88322-467-7, Peter Klau
- „Stochastik“, Vieweg, ISBN 3-528-05443-3, Gerhard Hübner
- „Handbuch der Mathematik“, BZ, ISBN 3-8166-0015-8, Hsg. Gottwald, Küstner, Hellwich, Kästner
- „Programming Perl“, 2nd Edition, O’Reilly, ISBN 1-56592-149-6, Wall, Christiansen & Schwartz

II. Internetadressen

- Internet Speed Homepage: <http://speed.xbaer.com>
- Meßstation RRZE 1: <http://www.uni-erlangen.de/webadm/provider.html>
- RFCs („Request for Comments“): <ftp://nic.ddn.mil> bzw. <ftp://ftp.uu.net>
- Internet Weather Report: <http://www.internetweather.com>
- Internet Statistics: <http://www.internetstats.com>
- Digital Daze™ - Internet Health Report:
<http://www.digitaldaze.com/support/nethealth/>
- Xlink Netztopologie: <http://www.warstein.owl.de/about/topologie.html>
- FOCUS Online – Der günstige Internet Zugang:
<http://www.focus.de/DD/DDO/ddo.htm>
- PY Software: <http://www.kagi.com/yellow/default.html>

III. RFC-Liste

Folgende Tabelle gibt einen Überblick über die RFCs, welche in Bezug auf diese Arbeit interessant sind:

RFC-Nummer	Titel der RFC
RFC-768	User Datagram Protocol (UDP)
RFC-791	Internet Protocol (IP)
RFC-792	Internet Control Message Protocol (ICMP)
RFC-793	Transmission Control Protocol (TCP)
RFC-822	Standard for the Format of ARPA Internet Text Messages
RFC-854	Telnet Protocol
RFC-882	Domain Names – Concepts and Facilities
RFC-883	Domain Names – Implementations
RFC-917	Internet Subnets
RFC-919	Broadcasting Internet datagrams
RFC-922	Broadcasting Internet datagrams in the Presence of Subnets
RFC-940	Toward an Internet Standard Scheme for Subnetting
RFC-947	Multi-Network Broadcasting within the Internet
RFC-959	File Transfer Protocol
RFC-966	Host groups: A Multicast Extension for the Internet Protocol
RFC-973	Domain System Changes and Observations
RFC-974	Mail Routing and the Domain System
RFC-988	Host Extensions for IP Multicasting
RFC-997	Internet Numbers
RFC-1009	Requirements for the Internet Gateways
RFC-1010	Assigned Numbers
RFC-1011	Official ARPA-Internet Protocols

Tabelle 5. RFC-Liste