

# Zentrale Auswertung von Betriebsstatistiken im Breitband-Wissenschaftsnetz mit dezentraler Vorverarbeitung

## Diplomarbeit im Fach Informatik

vorgelegt von

**Ulrich Tremel**

geb. am 04.11.1969 in Zwiesel

Angefertigt am

Institut für Mathematische Maschinen und Datenverarbeitung (IV)

Friedrich-Alexander-Universität Erlangen-Nürnberg

Betreuer: **Dipl.-Inf. Martin Heyer (RRZE)**  
**Dr. Peter Holleczek (RRZE)**  
**Prof. Dr. Fridolin Hofmann**

Beginn der Arbeit: 01.06.1996

Abgabe der Arbeit: 30.11.1996

Ich versichere, daß ich diese 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

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>5</b>
<b>2</b>	<b>Ausgewählte Aspekte der Datenkommunikation</b>	<b>7</b>
2.1	IP-Adreßerweiterungen . . . . .	7
2.1.1	Subnetting . . . . .	8
2.1.2	Supernetting . . . . .	8
2.2	Autonome Systeme . . . . .	10
2.3	Durchsatzberechnungen in Kommunikationsnetzen . . . . .	10
2.3.1	Theoretische Überlegungen . . . . .	11
2.3.2	Beispiel zur Berechnung eines Grenzdurchsatzes . . . . .	12
<b>3</b>	<b>Das Deutsche Forschungsnetz</b>	<b>14</b>
3.1	DFN-Verein . . . . .	14
3.2	Das Wissenschaftsnetz WiN . . . . .	16
3.3	Das Breitbandwissenschaftsnetz B-WiN . . . . .	17
3.3.1	Ursachen des gestiegenen Datenaufkommens . . . . .	17
3.3.2	Planung zum Ausbau des WiN . . . . .	18
3.3.3	Am B-WiN beteiligte Organisationen . . . . .	19
3.3.4	Struktur des B-WiN . . . . .	20
<b>4</b>	<b>Aufgabenstellung</b>	<b>24</b>
4.1	Definition einiger Begriffe . . . . .	24
4.2	Aufgaben im Überblick . . . . .	25
4.3	Konkretisierung der Verkehrsflußdaten . . . . .	25
4.4	Ausarbeitung von Vorschlägen . . . . .	26
4.5	Erzeugung der Accountingrohdaten . . . . .	27
4.5.1	Logisches Prinzip . . . . .	27
4.5.2	Technisches Prinzip . . . . .	28

<b>5</b>	<b>Beschreibung des Programmsystems</b>	<b>31</b>
5.1	Das Programmsystem im Überblick . . . . .	31
5.2	Datenbasis: Grundlage der Zuordnungen . . . . .	33
5.2.1	Datenbasis der Statistik . . . . .	33
5.2.2	Datenbasis des Außenpostens . . . . .	34
5.2.3	Berücksichtigung von Veränderungen . . . . .	35
5.3	Schnittstellen . . . . .	35
5.3.1	Eingabedatenschnittstelle: Rohdaten . . . . .	36
5.3.2	Steuerungsdatenschnittstelle: Datenbasis . . . . .	36
5.3.3	Ausgabedaten: Accountingenddaten . . . . .	38
5.4	Nachrichtensystem . . . . .	39
5.4.1	Beschreibung des Headers . . . . .	39
5.4.2	Beschreibung der Nachrichtentypen . . . . .	40
5.5	Beschreibung der Komponenten . . . . .	43
5.5.1	Zentrale . . . . .	43
5.5.2	Außenposten . . . . .	47
5.5.3	Statistik . . . . .	51
5.5.4	Manager . . . . .	52
<b>6</b>	<b>Implementierungsdetails</b>	<b>53</b>
6.1	Problematik der Signalbehandlung am Außenposten . . . . .	53
6.2	Zuordnung von IP-Adresse zu Subnetz/AS . . . . .	54
6.3	Datentyp der Zähler . . . . .	55
6.4	Rechengenauigkeit . . . . .	55
6.5	Maßnahmen zur Fehlertoleranz . . . . .	57
6.5.1	TCP als Kommunikationsmittel . . . . .	57
6.5.2	Ausfall der Zentrale . . . . .	57
6.5.3	Unterstützung der Fehlerdiagnose . . . . .	58
6.6	Protokollverifikation . . . . .	58
<b>7</b>	<b>Analyse der Daten</b>	<b>61</b>
<b>8</b>	<b>Zusammenfassung</b>	<b>65</b>

# Kapitel 1

## Einleitung

Von Zeit zu Zeit kann man in manchen Städten, vornehmlich zur Sommerszeit, wenn es warm ist, Grüppchen von Schülern beobachten, die an Straßenverkehrskreuzungen lagern. Ausgestattet mit Block und Bleistift machen sie für jedes vorbeikommende Fahrzeug einen Strich. Am Ende des Tages werden die Strichlisten an zentraler Stelle gesammelt und ausgewertet.

Als Ergebnis erhält man ein Tagesprofil der Verkehrsflüsse in der Stadt. Aufgrund dieser Daten kann man den Verlauf von Vorfahrtsstraßen, die Schaltung von Ampelphasen, also allgemein die Verkehrsführung besser an die Gegebenheiten anpassen. Dies entlastet die Innenstädte, schont Nerven und spart Zeit und Energie.

Das Thema dieser Arbeit ist ähnlich gelagert. Nicht die Verkehrsflüsse einer Stadt, sondern eines Datenkommunikationsnetzes sollen bestimmt und analysiert werden. An die Stelle der Straßenkreuzungen treten die Router des Datennetzes. Gezählt werden statt Automobilen und den aus Fahrschulbögen bekannten Handkarren Datenpakete und transportierte Informationseinheiten. Die Rolle der Schüler übernimmt eine Fähigkeit der Router, die es erlaubt, vorbeikommenden Verkehr zu protokollieren.

Bei dem zu analysierenden Netz handelt es sich um das Deutsche Wissenschaftsnetz (WiN), das vom Verein zur Förderung eines Deutschen Forschungsnetzes und der Telekom betrieben wird, und von der Wissenschaftsgemeinde in Deutschland zur Datenkommunikation genutzt wird.

Die Aufgabe dieser Arbeit besteht darin, die Verkehrsflußdaten, die die im Land verteilten Router erzeugen, dezentral zu bündeln und zu größeren organisatorischen und geographischen Einheiten zusammenzufassen und dann zentral abzulegen.

In Analogie zu dem eingangs geschilderten Beispiel sähe dies so aus, daß in jedem Stadtteil Büros eingerichtet werden, zu denen die Schüler ihre Listen tragen. In den Büros werden die Daten soweit ausgewertet, daß ein Bild über die Verkehrssituation des Stadtteils entsteht und anschließend ins Rathaus übermittelt. So muß nicht jeder Schüler bis zum Rathaus laufen und dort fällt nicht soviel Arbeit an.

In dem dieser Arbeit zugrundeliegenden Szenario hat dies ähnliche Vorteile. Durch die dezentrale Bündelung entsteht insgesamt nicht soviel Verkehr, als wenn alles gleich direkt zu einer zentralen Einheit geschickt wird, die zudem überlastet werden könnte.

Der Grund für die Zusammenfassung zu größeren organisatorischen Einheiten hat den Sinn, daß im allgemeinen niemand wissen möchte, wieviel Verkehr zwischen zwei speziellen Computern ausgetauscht worden ist. Die Auflösung ist viel zu fein, um damit weiterarbeiten zu können. Vielmehr ist es interessant zu erfahren, welches Verkehrsvolumen zwischen organisatorischen und geographischen Einheiten übertragen wird.

Interessant deshalb, weil es dadurch möglich wird, Engpässe in den Verbindungsleitungen festzustellen, die dann durch eine Umleitung bestimmter Ströme gemildert werden können. Oder falls diese Maßnahme nicht ausreicht, kann ermittelt werden, an welcher Stelle eine Erhöhung der Transportkapazität oder sogar eine neue Leitung am dringendsten benötigt wird.

Das Wissen um die Verkehrsströme kann also zur Verbesserung des Routings und zur Ausbauplanung genutzt werden. Kurz gesagt, realisiert diese Arbeit ein System, das die dafür benötigten Daten bereitstellt. Neben dieser Hauptaufgabe sollen Vorschläge zur Auswertung der Daten gemacht werden.

Die Arbeit ist folgendermaßen gegliedert:

Kapitel 2 befaßt sich mit allgemeinen Konzepten der Datenkommunikation, die im weiteren Verlauf wichtig sind. Kapitel 3 stellt das Wissenschaftsnetz selbst und die Organisationen in seinem Umfeld vor. Im Kapitel 4 wird die Aufgabenstellung präzisiert. Kapitel 5 beschreibt das erstellte Programmsystem. Kapitel 6 ist eine Ergänzung zur Beschreibung und fokussiert bestimmte Aspekte der Implementierung. Eine Präsentation von mit dem erstellten System erzeugten Daten gibt Kapitel 7. Das letzte Kapitel faßt die Arbeit zusammen.

# Kapitel 2

## Ausgewählte Aspekte der Datenkommunikation

Dieses Kapitel beschäftigt sich in loser Folge mit einigen Begriffen und Verfahren in der Datenkommunikation, die für den weiteren Text wichtig aber allgemeiner Natur sind, weshalb sie hier vorab stehen.

### 2.1 IP-Adreßerweiterungen

IP-Adressen bestehen aus einer 32 Bit breiten Zahl und setzen sich aus zwei Teilen zusammen, nämlich dem *Netzanteil* und dem *Rechneranteil*. Sie sind so konzipiert, daß beide Teile leicht extrahiert werden können. Dazu sind IP-Adressen in Klassen eingeteilt. Für jede Klasse ist festgelegt, wieviele Bits der Netzanteil umfaßt, und wieviele Bits den Rechner<sup>1</sup> identifizieren. Die Klasseneinteilung ist Tabelle 2.1 zu entnehmen.

Bit	0	1	2	3	8	16	24
Klasse A	0	Netzanteil			Rechneranteil		
Klasse B	1	0	Netzanteil			Rechneranteil	
Klasse C	1	1	0	Netzanteil			Rechneranteil

Tabelle 2.1: IP-Netzklassen (ohne Klassen C und D)

Durch dieses System wird das Routing in einem Internet erleichtert. Im Laufe der Zeit sind einige Probleme aufgetreten, die Erweiterungen des ursprünglichen Adressierungsschemas nötig machten. Die nächsten beiden Abschnitte beschreiben zwei davon.

---

<sup>1</sup>Anmerkung: Ist ein Rechner an mehrere Netze angeschlossen, besitzt er für jedes Netz eine eigene IP-Adresse. Hinter verschiedenen Adressen kann sich also derselbe Rechner verbergen.

### 2.1.1 Subnetting

Das erste hier behandelte Problem verursachte eine technologische Entwicklung. Als das Internet-Protokoll entwickelt wurde, bestand die Welt aus wenigen Großrechenanlagen. Heutzutage bevölkern viele Arbeitsplatzrechner und Personalcomputer, die über lokale Netze verbunden sind, die Landschaft.

Eine typische Organisation, die einen Internetanschluß realisieren möchte, verfügt über mehrere lokale Netze, an denen jeweils eine zweistellige Anzahl an Rechnern angeschlossen ist. Ein Klasse C Netz scheidet in diesem Fall aus, da nur 255 Rechner adressiert werden könnten. Also ist ein Klasse B Netz notwendig.

Da aber mehrere physikalische Netze vorhanden sind, von denen jedes eine eigene Netzadresse erhalten soll, bedient man sich des *Subnettings*. Dabei wird der Rechneranteil in zwei Teile aufgespaltet, von denen ein Teil das physikalische Netz bezeichnet und der andere Teil weiterhin den Rechner im physikalischen Netz<sup>2</sup>. Das Netz der Organisation besteht damit aus mehreren *Subnetzen*.

Außerhalb des Netzes bleibt diese Maßnahme verborgen. Innerhalb muß dafür Sorge getragen werden, daß von jeder Adresse der Subnetzanteil vom eigentlichen Rechneranteil unterscheidbar ist. Dies wird durch eine 32 Bit<sup>3</sup> breite Netzmaske realisiert, in der jene Bits gesetzt sind, die für den Netzanteil und dem Subnetzanteil reserviert sind. Da meistens nur zusammenhängende Netzmasken (es steht keine Null zwischen zwei Einsen) verwendet werden, genügt die Angabe der Länge der Maske.

### 2.1.2 Supernetting

Das zweite Problem besteht in der Verknappung von Klasse B Netzen. In den meisten Situationen ist ein Klasse C Netz, von denen es reichlich gäbe, zu klein. Deswegen bevorzugt man Klasse B Netze, deren Vorrat sich aber zu erschöpfen droht. Zwar befindet sich ein Nachfolger<sup>4</sup> des gegenwärtigen Internet-Protokolls Version 4 in Entwicklung, welcher mit 128 Bit breiten Adressen aufwartet; bis das Protokoll der nächsten Generation aber einsatzbereit ist, muß man sich anderweitig behelfen.

Eigentlich sind noch genug IP-Adressen vorhanden. Immerhin bietet der gewählte Raum Platz für  $2^{32} - 1$  unterschiedliche Werte. Allein die Klasseneinteilung ist für den heutigen Bedarf ungünstig gewählt: von den ca. 2 Millionen Klasse C Netzen ist nur ein kleiner Teil belegt.

Dieses Potential macht man sich nutzbar, indem man einer Organisation eine Gruppe von Klasse C Netzen zuweist, was als *Supernetting* bezeichnet wird. Nun bräuchte man aber für dieses *Supernet* mehrere Routingeinträge. Wünschenswert wäre allerdings nur ein einziger wie bei einem Klasse B Netz.

---

<sup>2</sup>Natürlich ist eine IP-Adresse keine physikalische Adresse. Es ist jedoch, zum Beispiel für Routingvorgänge, vorteilhaft, wenn ein IP-Netz dieselbe Ausdehnung hat wie ein physikalisches Netz.

<sup>3</sup>Subnetting ist nicht nur auf Klasse B Netze beschränkt; ansonsten wäre eine 16 Bit breite Maske ausreichend.

<sup>4</sup>IPv6



Deshalb trennt man sich vom klassenbasierten Adressierungs- und Routingschema und benutzt stattdessen das *Classless Inter-Domain Routing (CIDR)*. Es funktioniert so, als würde man den gesamten Adreßbereich mittels Subnetting aufteilen. Eine Netzadresse besteht nunmehr aus der ersten Adresse, die zu dem Netz gehört (im folgenden Basisadresse genannt), und aus einer Netzmaske.

Eine IP-Adresse behält ihre ursprüngliche Form bei. Will man aber, wie zum Beispiel während des Routings, ihre Netzzugehörigkeit bestimmen, muß sie mit allen Netzadressen in CIDR-Notation verglichen werden. Das bedeutet, daß sie durch ein Boolesches UND mit der Netzmaske der Netzadresse verknüpft wird. Wie beim Subnetting sind in der Netzmaske diejenigen Bits gesetzt, die zum Netzanteil gehören. Durch die logische Operation werden die Bits der zu identifizierenden IP-Adresse gelöscht, die den Rechneranteil ausmachen. Stimmt nun das Resultat der Operation mit der Basisadresse des Netzes überein, gehört die IP-Adresse zu diesem Netz.

Damit dies funktioniert, muß die Basisadresse ein Vielfaches von Zwei sein. Außerdem muß, wenn bei dem Vergleich mehrere Treffer erzielt worden sind, derjenige ausgewählt werden, bei dem die Maske mit den meisten gesetzten Bits verwendet wurde.

In der Praxis werden nur zusammenhängende Netzmasken benutzt, wodurch ein Netz aus aufeinanderfolgenden IP-Adressen besteht. Deshalb genügt es, nur die Netzmaskenlänge anzugeben. Es ist aber durchaus erlaubt, aus einem vergebenen Bereich einen Teil herauszulösen und ihn einem anderen Netz zuzuweisen. Wäre dies nicht der Fall, müßte man nicht mehr alle Einträge vergleichen, sondern könnte gleich beim ersten Treffer aufhören. Nichtsdestoweniger wird das Routing dennoch erleichtert, weil jetzt nur mit den längsten Masken<sup>5</sup> angefangen werden muß, um dann beim ersten Treffer stoppen zu können. Der Hauptzweck des Herauslösens besteht in der Ersparnis von Routingeinträgen.

Mit diesem System kann nun einer Organisation ein Block von aufeinanderfolgenden Klasse C Netzen zugewiesen werden, und es genügt ein Routingeintrag, wobei die Basisadresse der ersten Adresse des ersten Klasse C Netzes entspricht, und die Netzmaskenlänge passend zu der Länge des zugewiesenen Bereichs gewählt wird.

Es folgt zur Erläuterung ein erweitertes Beispiel aus [Come95]: Einer Organisation sollen 2048 aufeinanderfolgende IP-Adressen zugewiesen werden. Die erste Adresse sei 200.170.168.0. Tabelle 2.2 zeigt die erste und letzte Adresse des Bereichs in verschiedenen Formaten:

	Dezimal+Punkte	Dezimal	Dual mit Punkten
Erste	200.170.168.0	3366627328	11001000.10101010.10101000.00000000
Letzte	200.170.175.255	3366629375	11001000.10101010.10101111.11111111

Tabelle 2.2: Ausgewählter Bereich

Die erste und die letzte Adresse unterscheiden sich ab dem 22. Bit. Die Netzmaske hat also die ersten 21 Bits gesetzt. Das Netz würde damit in CIDR-Notation als 200.170.168.0/21 geschrieben werden. Nun könnte daraus der Bereich 200.170.170.0 - 234.170.171.255 einem anderen Netz zugeordnet werden.

<sup>5</sup>Im Englischen heißt dieses Paradigma *longest match*.

	Dezimal+Punkte	Dezimal	Dual mit Punkten
Erste	200.170.170.0	3366627840	11001000.10101010.10101010.00000000
Letzte	200.170.171.255	3366628351	11001000.10101010.10101011.11111111

Tabelle 2.3: Herausnahme von 512 Adressen

Die Länge der Netzmaske beträgt in diesem Beispiel 23. Damit wird das Netz 200.170.170.0/23 notiert. In einer Routingtabelle genügen die zwei angegebenen Einträge für ein korrektes Routing. Wäre die Herausnahme eines Bereichs nicht gestattet gewesen, hätte man drei Einträge benötigt. Nämlich einen für den ersten Bereich, einen für den herausgenommenen, einen dritten für den Rest, wobei der erste und der letzte Eintrag dasselbe Netz zum Inhalt gehabt hätten.

Es zeigt sich das hohe Maß an Flexibilität und Effizienz dieser Technik, was allerdings mit einer erhöhten Komplexität des Routings erkaufte wird. Einer IP-Adresse kann nun nicht mehr auf den ersten Blick angesehen werden, welchem Netz sie angehört. Außerdem müssen sämtliche Router auf einer Ebene diese spezielle Routingtechnik anwenden.

Wegen der Ähnlichkeit von Subnetting und Supernetting bezeichnet man oft ein in CIDR-Notation referenziertes Netz als Subnetz. Für den Rest der Arbeit gilt, daß “Subnetz” niemals im Bedeutungszusammenhang mit Subnetting gebraucht wird, sondern immer im Sinne von Supernetting.

## 2.2 Autonome Systeme

Das globale Internet besteht aus *Autonomen Systemen* (engl. *Autonomous System (AS)*). Ein Autonomes System ist ein Netzbereich der von einer Organisation selbständig verwaltet wird.

Die Einrichtung dieser Hierarchieebene erleichtert den Betrieb des Internets. Darunter ist vor allem die Organisation des Routings zu verstehen. Innerhalb eines autonomen Systems ist allein die zugehörige Organisation für das Routing verantwortlich. Um weltweite Erreichbarkeit zu realisieren, werden zwischen Autonomen Systemen Routinginformationen ausgetauscht. Zur Unterscheidung sind Autonome Systeme numeriert.

## 2.3 Durchsatzberechnungen in Kommunikationsnetzen

In vielen Bereichen ist der Durchsatz eines Systems von Interesse. Im folgenden wird ausgehend von der allgemeinen Definition die Bedeutung des Begriffes im Zusammenhang mit Kommunikationsnetzen herausgearbeitet. Die Definition von “Durchsatz” ist [JeVa] entnommen, deren Anwendung auf Kommunikationsnetze entspringt eigenen Überlegungen. Die allgemeine Definition von “Grenzdurchsatz” stammt wieder aus [JeVa]. Die Anwendung von “Grenzdurchsatz” auf Kommunikationsnetze findet sich implizit in [INT4].

### 2.3.1 Theoretische Überlegungen

Unter dem Durchsatz  $d$  einer Funktionseinheit versteht man den Quotienten aus der Anzahl erledigter Aufträge  $n$  und dem betrachteten Zeitraum  $t$ :

$$d(t) = \frac{n}{t} \quad (2.1)$$

Übertragen auf ein Kommunikationsnetz bedeutet dies, daß das Netz der Funktionseinheit gleichzusetzen ist, und die übertragenen Bits die Rolle der Aufträge spielen. Will man den Durchsatz eines Netzes bestimmen, genügt es zu beobachten, wieviele Bits das Netz im Meßintervall an ihren Bestimmungsort transportiert hat. Dabei ist es vollkommen gleichgültig, welchen Weg die Bits genommen haben. Selbst, wenn das Netz einmal eine optimale und bei einer zweiten Messung eine ungünstige Routingstrategie verfolgt, so wird in beiden Fällen bei gleicher Last derselbe Durchsatz festgestellt. Es erhöht sich lediglich die Verweilzeit der Aufträge. Dies gilt natürlich nur, falls die auftretende Last von der Funktionseinheit verarbeitet werden kann. Ist dies nicht Fall, ist der Durchsatz der Funktionseinheit gleich dem maximal möglichen Durchsatz und die Warteschlange der Aufträge wächst ins Unendliche<sup>6</sup>.

Der maximal mögliche Durchsatz heißt Grenzdurchsatz. In Gegensatz zum Durchsatz ist der Grenzdurchsatz abhängig von folgenden Faktoren:

- Topologie: Ein vollvermaschtes Netz weist einen höheren Grenzdurchsatz auf als ein sternförmiges.
- Leistungsfähigkeit der Netzkomponenten: Falls alle Komponenten durchlaufen werden müssen, wird der Grenzdurchsatz von der schwächsten bestimmt.
- Fluß der Daten: In einem Kommunikationsnetz müssen nicht immer alle Komponenten beansprucht werden. Zum Beispiel wirkt sich eine Leitung mit geringer Bandbreite nicht negativ auf den Grenzdurchsatz aus, wenn sie nur wenig belastet ist.
- Routing: Das Routing beeinflusst den Fluß der Daten.

Da der Fluß der Daten gewöhnlich dynamischer Natur ist, kann der Grenzdurchsatz nicht allgemein sondern immer nur in dessen Abhängigkeit bestimmt werden. Betrachtet man nun ein konkretes Netz mit fester Topologie, statischen Routingregeln und Komponenten mit bekannter Leistungscharakteristik, so ist der Grenzdurchsatz  $c$  des gesamten Netzes

$$c = \min\left(\frac{c_i}{A_i}\right) \quad (2.2)$$

mit  $c_i$  ist Grenzdurchsatz von Komponente  $i$ , und  $A_i$  ist Anteil am Gesamtfluß der Daten, der über Komponente  $i$  fließt. Das heißt, diejenige Komponente, deren Quotient aus Grenzdurchsatz und Anteil am kleinsten ist, bestimmt den Grenzdurchsatz des Systems.

Zur Veranschaulichung der angestellten theoretischen Überlegungen und um an späterer Stelle den Lesefluß nicht zu stören, folgt nun ein praktisches Beispiel. Es lehnt sich eng an eine ähnliche, in [INT4] durchgeführte Berechnung an.

---

<sup>6</sup>oder WWW- Benutzer deaktivieren "Load Images"!

### 2.3.2 Beispiel zur Berechnung eines Grenzdurchsatzes

Gegeben sei ein ringförmiges Kommunikationsnetz mit fünf Routern. An den Routern sind weitere Netze angeschlossen, die Verkehr erzeugen und verbrauchen. Das Ringnetz dient damit als Backbone für die gesamte Struktur. Sein Grenzdurchsatz soll berechnet werden.

Daten sollen nach der Strategie des kürzesten Weges geroutet werden. Bei fünf in einem Ring angeordneten Routern werden in einem Vorgang also höchstens drei berührt<sup>7</sup>. Die Router seien mit A, B, C, D und E bezeichnet und tragen gemäß Tabelle 2.4 zum Gesamtverkehr bei.

	A	B	C	D	E	$\Sigma$
A	10	5	5	3	2	25
B	3	10	3	2	2	20
C	1	4	7	2	1	15
D	2	3	5	20	5	35
E	1	1	1	1	1	5
						100

Tabelle 2.4: Verkehrsverteilung

Der in der Hauptdiagonale eingetragene Verkehr wird am dahinterliegenden Netz erzeugt und verbraucht, benutzt also nicht das Backbone. Es ist nun das am stärksten belastete Segment zwischen zwei Stationen zu bestimmen. In der Tabelle 2.5 sind oben spaltenweise die einzelnen Segmente aufgetragen (einmal im Uhrzeigersinn, einmal gegen den Uhrzeigersinn) und in den darunterliegenden Zeilen die Verkehrsströme, die zur Belastung der Teilstrecke beitragen.

Das Ergebnis der Berechnung ist in Abb. 2.1 dargestellt. Die Strecke A→B ist damit die am stärksten belastete. Sind alle Leitungen mit derselben Bandbreite ausgestattet, liegt an dieser Stelle der Engpaß im Backbone. Beträgt die Bandbreite zum Beispiel 2Mbps, so errechnet sich der Grenzdurchsatz des Ringnetzes nach Gleichung 2.2 zu

$$c_{\text{Ringnetz}} = \frac{2\text{Mbps}}{0.11} \approx 18.2\text{Mbps} \quad (2.3)$$

---

<sup>7</sup>Kommen Daten an vier Routern vorbei, hat jemand etwas falsch gemacht, denn im Kreis anders herum zu gehen, wäre kürzer gewesen.

	A→B	B→C	C→D	D→E	E→A	A→E	E→D	D→C	C→B	B→A
A→B	5									
A→C	5	5								
A→D						3	3			
A→E						2				
B→A										3
B→C		3								
B→D		2	2							
B→E						2				2
C→A									1	1
C→B									4	
C→D			2							
C→E			1	1						
D→A				2	2					
D→B								3	3	
D→C								5		
D→E				5						
E→A					1					
E→B	1				1					
E→C							1	1		
E→D							1			
$\Sigma$	11	10	5	8	4	7	5	9	8	6

Tabelle 2.5: Ermittlung der Segmentbelastung: In der ersten Zeile befinden sich alle Segmente. In den darunterliegenden Spalten finden sich die Verkehrsströme, die das Segment belasten. Sie sind in der Verkehrsverteilung zu finden. Die Summe in der letzten Zeile ist die Belastung des Segments.

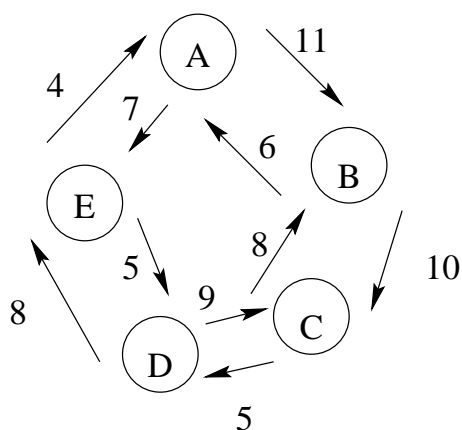


Abbildung 2.1: Segmentbelastung

# Kapitel 3

## Das Deutsche Forschungsnetz

Dieses Kapitel beschäftigt sich mit dem *Deutschen Forschungsnetz (DFN)*. Dazu wird zunächst der Betreiber des DFN, der *Verein zur Förderung eines Deutschen Forschungsnetzes (DFN-Verein)*, selbst vorgestellt und anschließend seine Aktivitäten in Vergangenheit und Gegenwart beschrieben.

### 3.1 DFN-Verein

In den USA und Großbritannien wurden bereits Ende der 60er Jahre Anstrengungen zum Aufbau flächendeckender Datenkommunikationsinfrastruktur unternommen (*ARPANET*, *SERCnet*). Im Jahre 1983 waren über das ARPANET bereits ca. 90% der amerikanischen Universitäten verbunden. Erste weltweite Verbindungen führten zur Prägung des Begriffs *Internet*. In Deutschland hingegen bestanden lediglich Datennetze mit regionalem Charakter (z.B der *Norddeutsche Rechnerverbund*). Diese Entwicklung veranlaßte 1982 das *Bundesministerium für Forschung und Technologie (BMFT)*, Schritte zur Schaffung einer landesweiten Kommunikationsstruktur für wissenschaftliche Einrichtungen zu unternehmen. Dazu wurde 1984 eine Organisation gegründet: der DFN-Verein.

Der DFN-Verein ist ein gemeinnütziger Verein und versteht sich als Interessenvertretung und Selbsthilfeorganisation der Wissenschaft an Hochschulen und an außeruniversitären, staatlichen und privatwirtschaftlichen Forschungseinrichtungen in Deutschland. Demzufolge hat er laut Satzung die Aufgabe,

*“... die Schaffung der wissenschaftlich-technischen Voraussetzungen für die Errichtung, den Betrieb und die Nutzung eines rechnergestützten Informations- und Kommunikationssystems für die öffentlich geförderte und die gemeinnützige Forschung in der Bundesrepublik Deutschland auf der Basis öffentlicher Übertragungsnetze unter Beachtung der entsprechenden internationalen Standards und Normen ...”*( §2, Abs. 1, zitiert nach [WWW1, “Gutachten des Wissenschaftsrates über den DFN-Verein”])

zu fördern.

Da an einer Kommunikation naturgemäß mehr als ein Objekt teilnimmt, und bei der Datenkommunikation sogar sehr viele Teilnehmer auftreten, ist es erforderlich, sich zueinander verträglicher Hilfsmittel zu bedienen. Daraus erklärt sich der Passus der Satzung, der zur Beachtung von internationalen Standards mahnt. Was heute im Zeitalter der offenen Kommunikation (Stichwort *Open System Interconnection (OSI)*) als selbstverständlich erscheint, war damals angesichts firmeneigener Netzlösungen (*SNA* von *IBM*, *DECNet* von *DEC*) ein wesentlicher Punkt.

In Bezug auf die Basis der Übertragungsnetze, die bislang aufgrund der Monopolstellung der *Deutschen Bundespost* (jetzt *Telekom*) gezwungenermaßen nur öffentlich sein konnte, zeigt sich ein Wandel. Durch die stufenweise Aufhebung des Monopols ist es zur Zeit schon privaten Gesellschaften möglich, eigene Infrastruktur durch Mietleitungen von der Telekom zu ergänzen und Kommunikationsdienste (außer Sprachübertragung) anzubieten.

Dem DFN-Verein ist es bereits jetzt<sup>1</sup> erlaubt, bundesweit eine Fernmeldeanlage zu errichten und zu betreiben. Dies erklärt sich aus der besonderen Situation des Vereins: er besteht aus einem geschlossenen Nutzerkreis und verfolgt keine kommerziellen Interessen.

Rund um den Aufbau und Betrieb eines Forschungsnetzes ergeben sich noch weitere Aufgaben:

- Unterstützung und Information der Benutzer des Netzes
- Überwachung der Dienstgüte und entsprechender Ausbau des Netzes
- Bereitstellung von Informationsdiensten
- Förderung der Entwicklung von neuen Produkten im Netzwerkbereich
- Modernisierung der angebotenen Dienste im Zuge der technischen Entwicklung und veränderter Anforderungen
- Veröffentlichung im Betrieb gewonnener Erfahrungen
- Rechtliche Vertretung der Mitglieder des Vereins
- Erprobung neuer Technologien

Die Organisationsstruktur des DFN-Vereins sieht folgendermaßen aus: Eine **Mitgliederversammlung** wählt die Zusammensetzung des **Verwaltungsrates**, bestimmt die Aufgaben, die sich der Verein stellt, und Mitgliedsbeiträge und entscheidet über Änderungen der Satzung. Der Verwaltungsrat koordiniert die Projekte und Aktivitäten des Vereins und wählt den **Vorstand**, der die Geschäfte des Vereins mit Hilfe der **Geschäftsstelle** führt.

Anfangs wurde der Verein aus Mitteln des BMFT finanziert. Nunmehr decken die Entgelte, die der Verein für seine Dienstleistungen fordert, die Kosten für deren Erbringung. Öffentliche Mittel werden für Entwicklung und Aufbau neuer Dienste verwendet.

---

<sup>1</sup>Genau seit dem 21.12.1995. Der Antrag an den Bundespostminister wurde bereits am 22.8.94 gestellt.

## 3.2 Das Wissenschaftsnetz WiN

Das Wissenschaftsnetz *WiN* wurde 1990 in Betrieb genommen und war bis zur Einführung des Breitbandwissenschaftsnetzes *B-WiN* das Rückgrat des Deutschen Forschungsnetzes. Es wird von der *DeTeSystem*, einer Tochter der Telekom, betrieben und basiert auf X.25 Technologie. Den Nutzern standen anfangs Übertragungsraten von 9.6Kbps und 64Kbps zur Verfügung. Im Jahre 1992 wurde das X.25 Netz modernisiert, so daß auch Anschlußraten von 2Mbps möglich waren.

Das WiN war das erste Netz mit überregionaler Bedeutung für die Wissenschaftsgemeinde und bildete mit seinen Auslandsanschlüssen und Gateways zu anderen Netzen einen wesentlichen Teil des Deutschen Internets. Die Wissenschaftsgemeinde verfügte seit diesem Zeitpunkt über ein leistungsfähiges Datennetz zur Kommunikation innerhalb Deutschlands, und über diverse Auslandszugänge war ein Zugriff auf Daten in aller Welt möglich.

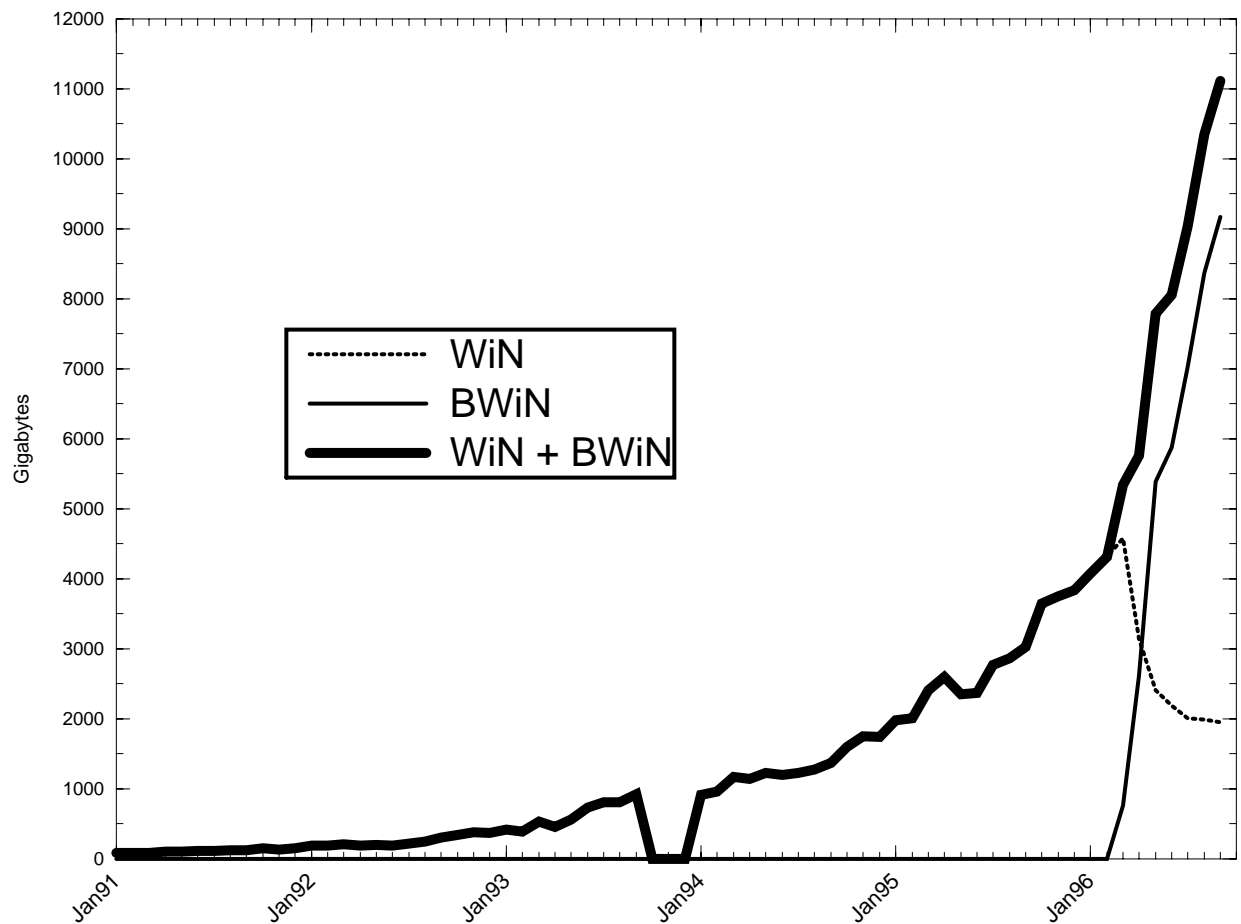


Abbildung 3.1: Transportiertes Datenvolumen im WiN. Für Oktober und November 1993 sind keine Daten verfügbar.



Diagramm 3.1 zeigt anhand des transportierten Datenvolumens den Erfolg des WiN. Wie unschwer zu erkennen ist, steigt das Datenaufkommen exponentiell an. War die Bandbreite des WiN anfangs noch mehr als ausreichend, traten infolge der gestiegenen Last immer mehr Engpässe auf. Der folgende Abschnitt stellt das B-WiN vor, das dieser Entwicklung Rechnung trägt. Ab jetzt wird für das auf X.25-Technologie basierende Wissenschaftsnetz der Ausdruck “X.25-WiN” und für das breitbandige Wissenschaftsnetz “B-WiN” verwendet. “WiN” steht nunmehr für beides.

### 3.3 Das Breitbandwissenschaftsnetz B-WiN

Das B-WiN ist die Antwort des DFN-Vereins auf gestiegenes Datenvolumen. Es wird nun zunächst der Hintergrund des Anstiegs beleuchtet und darauf aufbauend der Planungsprozeß geschildert. Nach einer kurzen Vorstellung der am B-WiN beteiligten Organisationen und deren Verhältnis zueinander folgt eine Beschreibung der technischen Realisierung.

#### 3.3.1 Ursachen des gestiegenen Datenaufkommens

Natürlich ist eine Ursache des steigenden Netzverkehrs die gewachsene Gemeinde der Einrichtungen, die sich am WiN beteiligen. Darüber hinaus haben sich aber auch die Anwendungen von Datennetzen verändert. Zu den traditionellen Anwendungen Elektronische Post, Dateitransfer, Netnews und Fernsitzung traten Anwendungen multimedialen Charakters. Ermöglicht hat diese Entwicklung der technologische Fortschritt bei den Datenendgeräten, deren Gestalt sich von einfachen Textterminals zu grafikfähigen und mit der Möglichkeit zur Tonwiedergabe versehenen Arbeitsplatzrechnern gewandelt hat. Daß sich das nicht nur auf die Datenendgeräte selbst auswirkt, sondern auch auf die Datenübertragungseinrichtungen, zeigen folgende Anwendungsbeispiele:

- Videokonferenz
- Videoübertragung von Vorlesungen
- Abruf visualisierter wissenschaftlicher Daten
- Verteilte Bilddatenbanken, Fachdatenbanken
- Rechnen an entfernten Hochleistungsrechnern
- Medizin: Ferndiagnose

Darüber hinaus sind grafische Anwendungen auch für Nutzer attraktiv, die sich bisher nicht mit Computern beschäftigt haben. Als Beispiel sei hier das *World Wide Web (WWW)* genannt, das sich im Gegensatz zu dem textbasierten Auskunftssystem *Gopher* großer Beliebtheit erfreut. So wuchs in jüngerer Vergangenheit zwar nicht so sehr die Anzahl ans WiN angeschlossener Institutionen, wohl aber die Zahl der Netznutzer<sup>2</sup>.

---

<sup>2</sup>Diese leicht nachvollziehbare Aussage beruht auf persönlicher Beobachtung in Rechnerräumen. Im übrigen weist der Vorsitzende des Verwaltungsrates, Prof. Dr. Dieter Maaß, schon frühzeitig im Editorial des Mitteilungsblattes des DFN im März 1994 [DFN, Heft 34] auf diesen Sachverhalt hin.

Nun sind Datenendgeräte leicht austauschbar; für Datenübertragungseinrichtungen gilt dies jedoch nicht. Deshalb ist ein sorgfältiger Planungsprozeß vonnöten. Er wird im folgenden gezeigt.

### 3.3.2 Planung zum Ausbau des WiN

Mit der Inbetriebnahme des WiN 1990 wurde der Forschungsgemeinde ein akzeptables Umfeld geschaffen. Wegen der im vorigen Abschnitt beschriebenen Entwicklung ist aber eine Erhöhung der Leistungsfähigkeit unumgänglich. Auch im Hinblick auf internationale Konkurrenzfähigkeit der deutschen Wissenschaft erscheint rasches Handeln angebracht. Die Vereinigten Staaten und das Vereinigte Königreich befinden sich immer noch in der Rolle der Vorreiter.

Bereits 1988, also noch vor Einführung des WiN, plante der DFN-Verein ein breitbandiges Kommunikationsnetz, stieß aber bei dem dafür zuständigen Monopolinhaber auf taube Ohren<sup>3</sup>. Aufgrund dieser Erfahrungen ist auch der schon erwähnte Antrag beim Bundespostminister auf eine eigene Lizenz zu verstehen. Ende 1994 startete der Verein das Projekt "Regionale Testbeds" mit dem Breitbandtechnologie und Breitbandanwendungen zumindest regional beschränkt getestet werden konnten. Auch das Interesse der Öffentlichkeit erweckte angesichts griffiger und medienwirksamer Begriffe wie Datenautobahn und Internet.

Aufgrund dieser Vorgänge und der fortschreitenden Liberalisierung des Telekommunikationsmarktes standen letztendlich drei technische Realisierungen zur Auswahl:

1. ein IP-Routernetz auf Basis von Monopolleitungen
2. der Aufbau eines virtuellen Netzes auf Basis von Datex-M der Telekom
3. ein ATM-Routernetz

Für ein IP-Routernetz spräche, daß kein zusätzlicher Aufwand zur Abwicklung von IP-Verkehr wie bei den anderen Lösungen entstünde. Im Hinblick auf isochronen Datenverkehr mit reservierter Bandbreite scheidet diese Möglichkeit jedoch aus. Gegen Datex-M sprach das Preis/Leistungsverhältnis. ATM erschien als zukunftssträchtige Technik, die alle Anforderungen an moderne, integrierende Datenkommunikation erfüllt. Außerdem hatte man schon in den Regionalen Testbeds Erfahrungen mit dieser Technologie gemacht. Nachdem die Entscheidung gefällt war, konnte noch Ende 1995 ein Vertrag mit der DeTeSystem unterzeichnet werden, die versprach, das Netz bis zum Frühjahr 1996 zu realisieren.

Der Vertrag behandelt auch die Dimensionierung des Netzes und schreibt einen Bedarf an 400Mbps Kernnetzdurchsatz fest. Diese Zahl orientiert sich an einer von Jessen für den DFN-Verein durchgeführten Studie [INT4]. Jessen projiziert dabei aufgrund von Daten aus den Jahren 1988 und 1994 die Entwicklung des traditionellen Datenverkehrs (*FTP*, *TELNET*, *SMTP*) auf das Jahr 1996 und trifft für neue Anwendungen (Audio, Video, Verteiltes

---

<sup>3</sup>Der Unmut im DFN-Verein darob geht deutlich aus dem Editorial zur Ausgabe März 1995 der DFN-Mitteilungen [DFN, Heft 37] vom stellvertretenden Vorsitzenden Prof. Dr. Helmut Pralle und dem in der selben Ausgabe abgedrucktem "Statement des DFN-Vereins zum Regulierungsrahmen Telekommunikation" hervor.

Rechnen, Fernsprechen) gewisse Annahmen, um so einen Überblick über das gesamte zu erwartende Datenvolumen zu erhalten. Natürlich sind diese Annahmen mit einer gewissen Unsicherheit behaftet, worauf Jessen in seiner Studie nachdrücklich hinweist. Genaueres Datenmaterial zu liefern, ist nicht zuletzt Aufgabe dieser Diplomarbeit.

Jessen untersucht auch die Leistungsfähigkeit der oben angeführten drei technischen Realisierungen mit Hilfe der in 2.3.2 vorgestellten Verfahren, da es sich im wesentlichen um Ringkonfigurationen handelt. Er kommt dabei zu dem Ergebnis, daß

- bei einer Anzahl von dreißig mit 34 Mbps versehenen Anschlüssen,
- die auf acht Aufpunkte zum Kernnetz verteilt sind,
- und deren Anteil und Richtung am Gesamtverkehr aufgrund von Daten aus dem Jahr 1994 geschätzt wird
- ein Ringnetz mit 34Mbps Verbindungen

einen Grenzdurchsatz von 222Mbps aufweist:

$$C_{HK N}^{34Mbps} = \frac{34Mbps}{0.153} = 222Mbps \quad (3.1)$$

Die exakte Berechnung der 15.3% für die anteilig am Gesamtdurchsatz am stärksten ausgelastete Leitung findet sich in der Studie.

Die gegenwärtige Konfiguration des B-WiN Backbones ist 3.4 zu entnehmen und weist ein teilweise vermaschtes Ringnetz mit nur 34Mbps Leitungen auf. Deshalb ist ein Ausbau auf 155Mbps im Gange, was eine Kapazität von

$$C_{HK N}^{155Mbps} = \frac{155Mbps}{0.153} = 1013Mbps \quad (3.2)$$

ermöglichen würde.

Bevor genauer auf die technische Realisierung eingegangen wird, werden nun die am B-WiN beteiligten Partner und ihre Relation zueinander dargestellt.

### 3.3.3 Am B-WiN beteiligte Organisationen

Der DFN-Verein stellt seinen Mitgliedern das WiN zur Verfügung. Dazu hat er Aufträge an folgende Organisationen vergeben:

**DeTeSystem:** Die DeTeSystem ist als Tochter der Telekom mit dem technischen Aufbau und Betrieb des B-WiN betraut.

**NOC:** Das *NOC (Network Operations Center)* ist für die Bereitstellung eines IP-Dienstes auf Basis des von der DeTeSystem aufgebauten Netzes zuständig. Das NOC besteht aus einer Projektgruppe des DFN-Vereins an der Universität Stuttgart.

**B-WiN-Labor:** Das B-WiN-Labor führt Maßnahmen zur Qualitätssicherung im B-WiN durch. Es begleitete schon den Betrieb des X.25 WiN durch Bestimmung von Größen wie Durchsatz und Delay, Nachstellung von Betriebssituationen unter Laborbedingungen, Fehlersuche und Beratung der Kunden. Zudem testet es im Zuge des Fortschritts neu entstandene Technologien. Die dabei im Vorfeld der Einführung des B-WiN gemachten Erfahrungen flossen in dessen Planung und Realisierung ein. Bei der Inbetriebnahme des B-WiN führte es für den DFN-Verein die Abnahme durch, wobei das Netz auf Funktionalität, Belastbarkeit und Dienstgüte getestet wurde.

Dies gehört nach wie vor zu den Aufgaben des Labors und in diesem Zusammenhang ist auch diese Diplomarbeit zu sehen. Das B-WiN Labor ist wie das NOC eine Projektgruppe des DFN-Vereins und hat seinen Sitz am *Regionalen Rechenzentrum Erlangen (RRZE)*.

Die Relationen der beteiligten Parteien zeigt Abb.3.2.

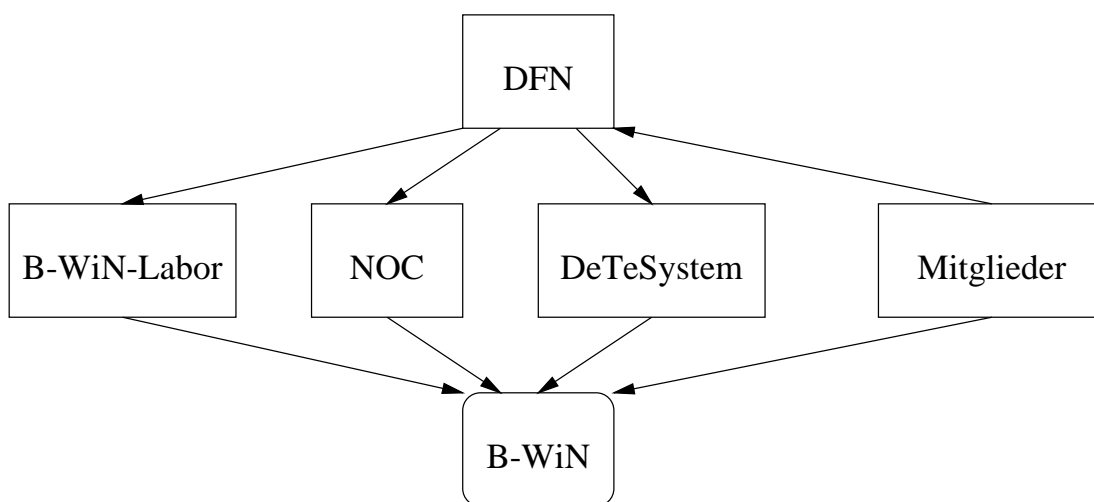


Abbildung 3.2: Am B-BWiN beteiligte Organisationen: Der DFN-Verein setzt sich aus Mitgliedern zusammen. WiN-Labor, NOC, DeTeSystem sind vom DFN-Verein mit dem Betrieb des B-WiN, das von den Mitgliedern genutzt wird, betraut.

### 3.3.4 Struktur des B-WiN

Dieser Abschnitt beschreibt die Struktur des B-WiN und seine Komponenten. Er bildet die Basis für die sich unmittelbar anschließende Erläuterung der Aufgabenstellung.

Die Erklärung erfolgt anhand von Abb. 3.3.

Für einen Kunden, dem der DFN-Verein einen Anschluß zum WiN einrichtet, stellt gegenwärtig der *WiN-Router (WR)* die Schnittstelle zum WiN dar. An diesen WiN-Router können der Kunde, oder im Fall von Gemeinschaftsanschlüssen auch mehrere Kunden ihre eigenen *Kundenrouter (KR)* anschließen. Der größere Kreis in der Abbildung markiert somit die äußere Grenze des B-WiN. Die WiN-Router sind an ATM-Switches angeschlossen, den *Kundenserviceswitches (KSS)*. Hier kann man eine logische Trennlinie innerhalb des WiN

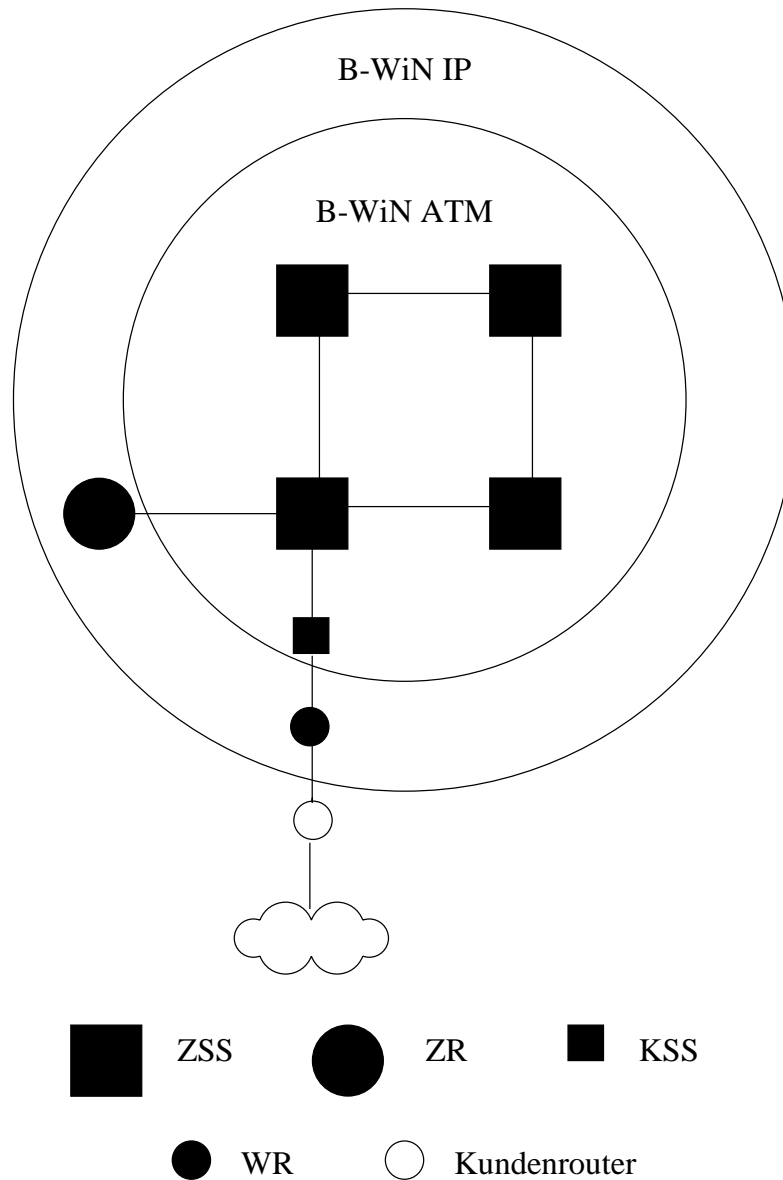


Abbildung 3.3: Struktur des Breitbandwissenschaftsnetzes

ziehen und einen IP-basierten Bereich und einen ATM-basierten Bereich unterscheiden, weil im WiN-Router IP-Pakete in ATM-Zellen verpackt werden. Dies versinnbildlicht der innere Kreis.

Die Kundenserviceswitches werden an *Zentralen Serviceswitches (ZSS)* zusammengefaßt. Für den IP-Dienst ist es notwendig, daß ATM-Zellen, die hier ankommen, wieder zu IP-Paketen zusammengesetzt werden, um die darin enthaltene Routinginformation zu erhalten. Dies geschieht am *Zentralen Router (ZR)*, der deswegen im IP-basierten Bereich angesiedelt ist. Die Zentralen Serviceswitches und ihre Verbindungswege bilden das Kernnetz. Es ist in Abbildung 3.4 dargestellt. Es handelt sich im wesentlichen um eine Ringstruktur. Die beiden Vermaschungen von Stuttgart und Berlin nach Köln erklären sich durch den Sitz des Auslandszuganges in Köln, weswegen dort mit erhöhtem Verkehrsaufkommen zu rechnen ist. Die Resultate dieser Arbeit können jedoch für eine Erarbeitung von alternativen Konfigurationen benutzt werden.

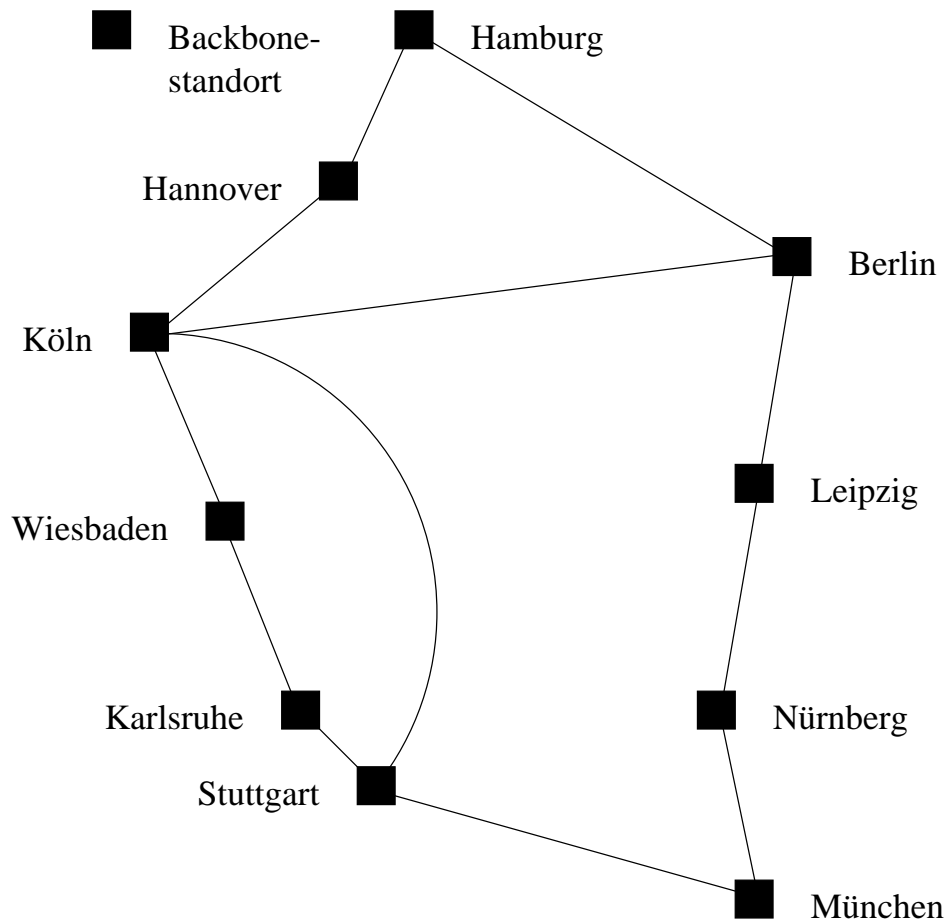


Abbildung 3.4: B-WiN Backbone

Die Konfiguration zu ändern, erleichtert der Umstand, daß die in Abb. 3.4 eingezeichneten Leitungen als Virtuelle Pfade<sup>4</sup> im ATM-Crossconnect<sup>5</sup> Netz der Telekom zu sehen sind. Im Bedarfsfall können zusätzliche freie Bandbreiten des Crossconnects für Verbindungen eingerichtet werden. Abbildung 3.5 zeigt diese Möglichkeit.

Soviel zum gegenwärtigen Stand des B-WiN. Für die Zukunft ist geplant, den Kunden auch einen direkten Zugang zu einem Kundenserviceswitch anzubieten. Bislang kann der Kunde sein privates Netz nur über einen WiN-Router anschließen. Mit dieser Erweiterung kann dann den Kunden nicht nur ein IP-Dienst angeboten werden, sondern zusätzlich auch ein ATM-Dienst.

<sup>4</sup>ATM arbeitet verbindungsorientiert. Eine Verbindung in einem ATM Netz wird durch einen Verbindungsbezeichner charakterisiert, welcher aus einem Bezeichner für den virtuellen Pfad (engl. *virtual path identifier (VPI)*) und einem Bezeichner für den Kanal (engl. *virtual circuit identifier (VCI)*) besteht. Mehrere Kanäle derselben Orientierung werden so unter einem Pfad zusammengefaßt, was die Organisation in den Vermittlungsstellen erleichtert. Im vorliegenden Fall werden vom Dienstleister virtuelle Pfade zwischen Vermittlungen (Switches) des Dienstnehmers eingerichtet.

<sup>5</sup>Dies ist die ATM-Backbone-Infrastruktur der Telekom.

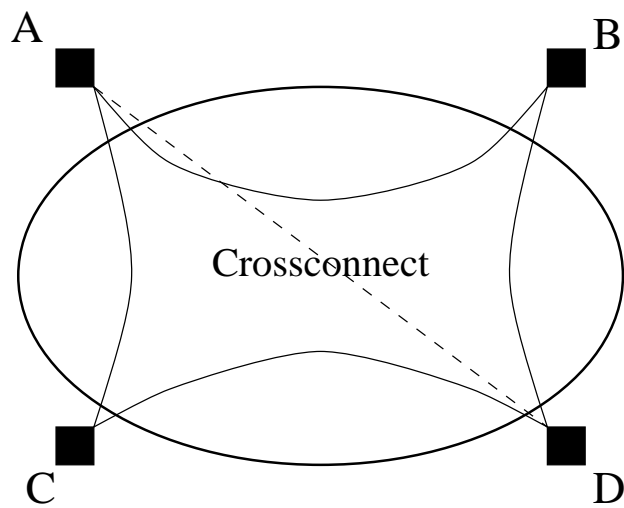


Abbildung 3.5: Die ZSS A, B, C und D sind jeweils mit ihren Nachbarn über einen Virtuellen Pfad durch das Crossconnect verbunden. Sollte Bedarf an zusätzlicher Bandbreite zwischen A und D bestehen, kann ein weiterer Pfad (gestrichelt eingezeichnet) eingerichtet werden.

# Kapitel 4

## Aufgabenstellung

Nachdem die letzten beiden Kapitel über Hintergründe und Umfeld informiert haben, beschreibt dieses Kapitel nun konkret die Aufgabenstellung. Dazu werden vorab einige Begriffe definiert, dann die einzelnen Aufgabengebiete im Überblick dargestellt und anschließend genau erläutert. Der letzte Teil des Kapitels (Abschnitt 4.5) vertieft die Kenntnisse über die Ausgangslage und führt direkt zum nächsten Kapitel, das die Lösung der Aufgabe beschreibt.

### 4.1 Definition einiger Begriffe

Die Bedeutung folgender Begriffe sei verbindlich für den Rest der Arbeit:

**Kernnetz:** Das Kernnetz (engl. Backbone) des B-WiN. Zu ihm gehören die Zentralen Service Switches und die sie verbindenden Leitungen des B-WiN. Es ist in Abb. 3.4 abgebildet.

**Kernnetzdurchsatz:** Der Durchsatz des Kernnetzes. Was “Durchsatz” für ein Kommunikationsnetz bedeutet, wurde in Abschnitt 2.3.1 erläutert.

**Standort:** Der geographische Standort eines ZSS. Gegenwärtig gibt es 10 Standorte, die ebenfalls Abb. 3.4 entnommen werden können.

**Anschluß:** Der Anschluß eines Kunden des DFN-Vereins an das B-WiN über einen WR.

**Subnetz:** Das Subnetz eines Kunden. Nach der Aussage am Schluß von Abschnitt 2.1.2 ist damit immer ein in CIDR-Notation dargestelltes Netz gemeint.

**Autonomes System:** Ein Autonomes System (AS) im Sinne von Abschnitt 2.2. Das WiN ist zum Beispiel das Autonome System mit der Nummer 1275.

**Viertelstunde:** Das feinste Intervall, nach dem Verkehrsflußdaten aufgelöst werden. Dieses liegt momentan tatsächlich bei 15 Minuten.



## 4.2 Aufgaben im Überblick

Grob gesagt sind zwei Aufgaben gestellt:

1. Entwurf und Implementierung von Software zur Ermittlung von Verkehrsflußdaten im B-WiN
2. Ausarbeitung von Vorschlägen zur Interpretation der Daten

Die Formulierung des Themas setzt noch gewisse Rahmenbedingungen Punkt 1 betreffend: Zur Ermittlung von brauchbaren Verkehrsflußdaten ist es im Verlauf der Verarbeitung notwendig, die Daten zu größeren Einheiten zusammenzufassen. Dies bedeutet zum Beispiel, den von zwei Rechnern erzeugten Verkehr zwei Anschlüssen zuzuweisen. Derartige Zuordnungen sollen bereits am Ort der Erfassung der Daten erfolgen. Als Folge davon ist die Software als verteiltes System zu realisieren. Das System soll sich Netzwerkfehlern gegenüber tolerant verhalten.

## 4.3 Konkretisierung der Verkehrsflußdaten

Die unter Punkt 1 des gerade gegebenen Überblicks angeführten Verkehrsflußdaten werden nun genauer spezifiziert.

Jeglicher IP-Verkehr im B-WiN ist zu berücksichtigen. Dabei sollen Quelle und Ziel, Anzahl transferierter IP-Pakete, Anzahl übertragener Bytes und Zeitpunkt der Übertragung protokolliert werden. Derartige Vorgänge bezeichnet man mit dem Begriff *Accounting*, die aufgezeichneten Größen heißen *Accountingdaten*.

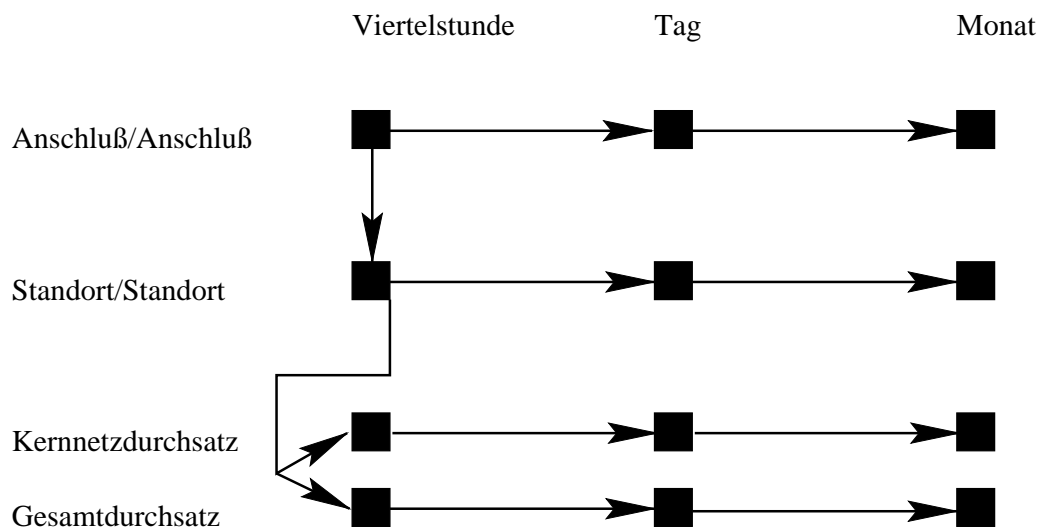


Abbildung 4.1: Übersicht über alle zu erzeugenden Daten. Die ausgefüllten Kästchen repräsentieren die einzelnen Sammlungen von Daten. Die Pfeile geben die Erarbeitungsreihenfolge wieder.

Quelle und Ziel der Accountingdaten sollen jeweils nach Anschluß und Standort aufgeschlüsselt werden. Aus den Standortdaten sollen Gesamt- und Kernnetzdurchsatz errechnet werden. Die Viertelstunde umfaßt 15 Minuten<sup>1</sup>, soll aber konfigurierbar sein. Ferner sollen die Daten aus einer Viertelstunde zu Tages- und Monatsstatistiken aggregiert werden. Abbildung 4.1 gibt eine tabellarische Übersicht aller zu erzeugenden Daten.

Der Vollständigkeit halber sei hier auch das Format der Eingabedaten genannt. Sie bestehen aus Quadrupeln der Form

Quell-IP-Adresse  $\times$  Ziel-IP-Adresse  $\times$  Anzahl Pakete  $\times$  Anzahl Bytes

und werden im folgenden auch *Accountingrohdaten* genannt. Wie sie erzeugt werden, wird am Schluß des Kapitels geschildert.

Im teilweisen Vorgriff auf die Lösung sei hier die Wandlung der Eingabedaten zu den Ausgabedaten beschrieben. Die Adreßinformationen von Quelle und Ziel werden einem Subnetz bzw. einem AS zugeordnet, die wiederum auf einen Anschluß abgebildet werden, welcher seinerseits einem Standort zugeschrieben ist. Tabelle 4.1 gibt diesen Vorgang wieder.

Quelle	Ziel	Pakete	Bytes
IP-Adr.	IP-Adr.	Zahl	Zahl
↓			
Subnetz/AS	Subnetz/AS	Zahl	Zahl
↓			
B-WIN Anschluß	B-WIN Anschluß	Zahl	Zahl
↓			
Standort	Standort	Zahl	Zahl

Tabelle 4.1: Wandel der Daten

## 4.4 Ausarbeitung von Vorschlägen

Die in 4.3 beschriebenen Daten geben Auskunft über Umfang und Richtung der Datenflüsse im B-WiN in verschiedenen Zeitrastern. Sie werden dazu benutzt, Engpässe festzustellen,

<sup>1</sup>Erscheint diese Aussage trivial, ist die zu Beginn des Kapitels getroffene Definition zu konsultieren.

die dann durch Schaltung neuer Verbindungen oder Änderung des Routings beseitigt werden können. Außerdem bilden sie die Grundlage für Vorhersagen zukünftiger Volumenentwicklungen. Nicht zuletzt dient die Feststellung des Kernnetzdurchsatzes der Einhaltung der im Vertrag zwischen DFN-Verein und DeTeSystem vereinbarten Größe<sup>2</sup>.

Speziell in dieser Arbeit sollen Vorschläge erarbeitet werden, wie aus den gewonnenen Daten verbesserte Routingregeln abgeleitet werden können.

## 4.5 Erzeugung der Accountingrohdaten

Es bleibt noch zu erklären, wie die Accountingrohdaten erzeugt werden. Dazu ist es vonnöten, die in Kapitel 3 gegebene Darstellung des B-WiN vertiefen und zwar in logischer und technischer Hinsicht. Der Inhalt dieses Abschnitts führt unmittelbar zur im nächsten Kapitel vorgestellten Lösung, denn die Accountingrohdaten sind Eingabedaten des zu erstellenden Systems. Außerdem erleichtern die folgenden Erläuterungen die Einordnung des zu erstellenden Systems in das Gesamtsystem.

### 4.5.1 Logisches Prinzip

Mithin das wichtigste Ziel, das mit den Ergebnissen der Arbeit verfolgt wird, ist die Bestimmung des Kernnetzdurchsatzes des B-WiN. Davon, daß dies nicht so einfach ist, zeugen folgende Überlegungen.

Abb.4.2 vermittelt ein logisches Abbild des B-WiN. Zur Bestimmung des Kernnetzdurchsatzes sollte es eigentlich genügen, die *Linkstatistiken* der ZSS auszuwerten. Sie geben Auskunft über jeglichen Datenverkehr zwischen zwei benachbarten Switches. Leider werden dabei viele Daten mehrfach gezählt. Zum Beispiel taucht Verkehr von A nach C sowohl in der Linkstatistik von A nach B als auch in der von B nach C auf, was nach der Definition des Begriffes “Kernnetzdurchsatz” nicht erlaubt ist!

Da die Statistiken keine Information über Quelle und Senke der Daten enthalten, sind derartige Mehrfachzählungen nicht zu erkennen und zu vermeiden. Abhilfe schafft ein nochmaliger Blick auf Abb. 4.2. Für ein Kommunikationsnetz genügt es, den Verkehr zu zählen, der das Netz verläßt, wenn man seinen Durchsatz bestimmen möchte. Alternativ kann man auch den Verkehr an der Stelle zählen, an der er in das Netz eintritt, wenn man davon ausgeht, daß das Netz jeden Auftrag ordnungsgemäß erledigt und keinen vergißt.

In der Abbildung ist zu sehen, daß die einzigen Zugangspunkte zum Backbone die an den jeweiligen Standorten angeschlossenen Zentralen Router sind. Außerdem gibt es zur Zeit an einem Standort einen Internationalen Router, der logisch wie ein ZR zu sehen ist. Aus Lastgründen ist es aber — ohne den Normalbetrieb empfindlich zu stören — nicht möglich, an dieser Stelle das Accounting zu aktivieren.

---

<sup>2</sup>Gegenwärtig 400Mbps; siehe dazu auch 3.3.2

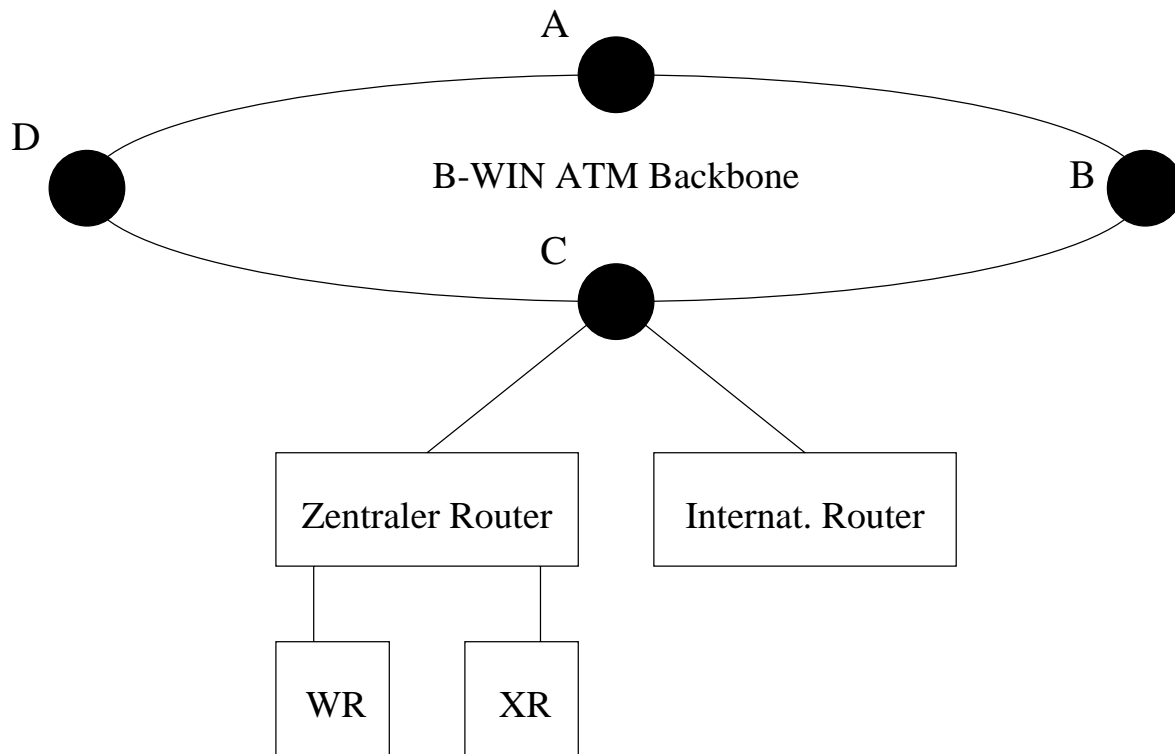


Abbildung 4.2: Breitband-Wissenschaftsnetz: Logische Struktur

Es ist erforderlich, noch eine Stufe in der Hierarchie nach unten zu wandern. Hier finden sich die WiN-Router und die X.25-Router. Wenn man das Accounting an den Interfaces<sup>3</sup> dieser Router, die zum ZR führen, durchführt, erfasst man allen Verkehr, der über die WiN- und X.25-Router zum Kernnetz strömt. Nicht erfasst wird mit dieser Methode derjenige Verkehr, der vom IR aus ins Kernnetz gelangt. Deshalb wird Accounting auch an den Interfaces betrieben, die von den WR respektive XR zu den darunterliegenden Netzen führen.

Damit wird allerdings Verkehr von WR/XR zu WR/XR an zwei Stellen gezählt. Deswegen wird bei der Weiterverarbeitung der Daten dieser Fall mittels eines Vergleiches von Quelle und Ziel erkannt und dann nur einmal berücksichtigt.

Eine Darstellung der beteiligten Hardwarekomponenten gibt nun der nächste Abschnitt.

## 4.5.2 Technisches Prinzip

Bislang wurde die Struktur des B-WiN nur logisch und sehr vergrößert dargestellt. Um aber genau nachvollziehen zu können, an welcher Stelle die Accountingrohdaten erzeugt werden und wo sie weiterverarbeitet werden, ist eine genauere Betrachtung der physikalischen Strukturen notwendig.

Abb. 4.3 zeigt in der oberen Bildhälfte einen Ausschnitt des B-WiN. Der in der Mitte liegende ZSS ist weiter ausgeführt. Er ist mit zwei KSS verbunden, von denen wieder einer verfeinert

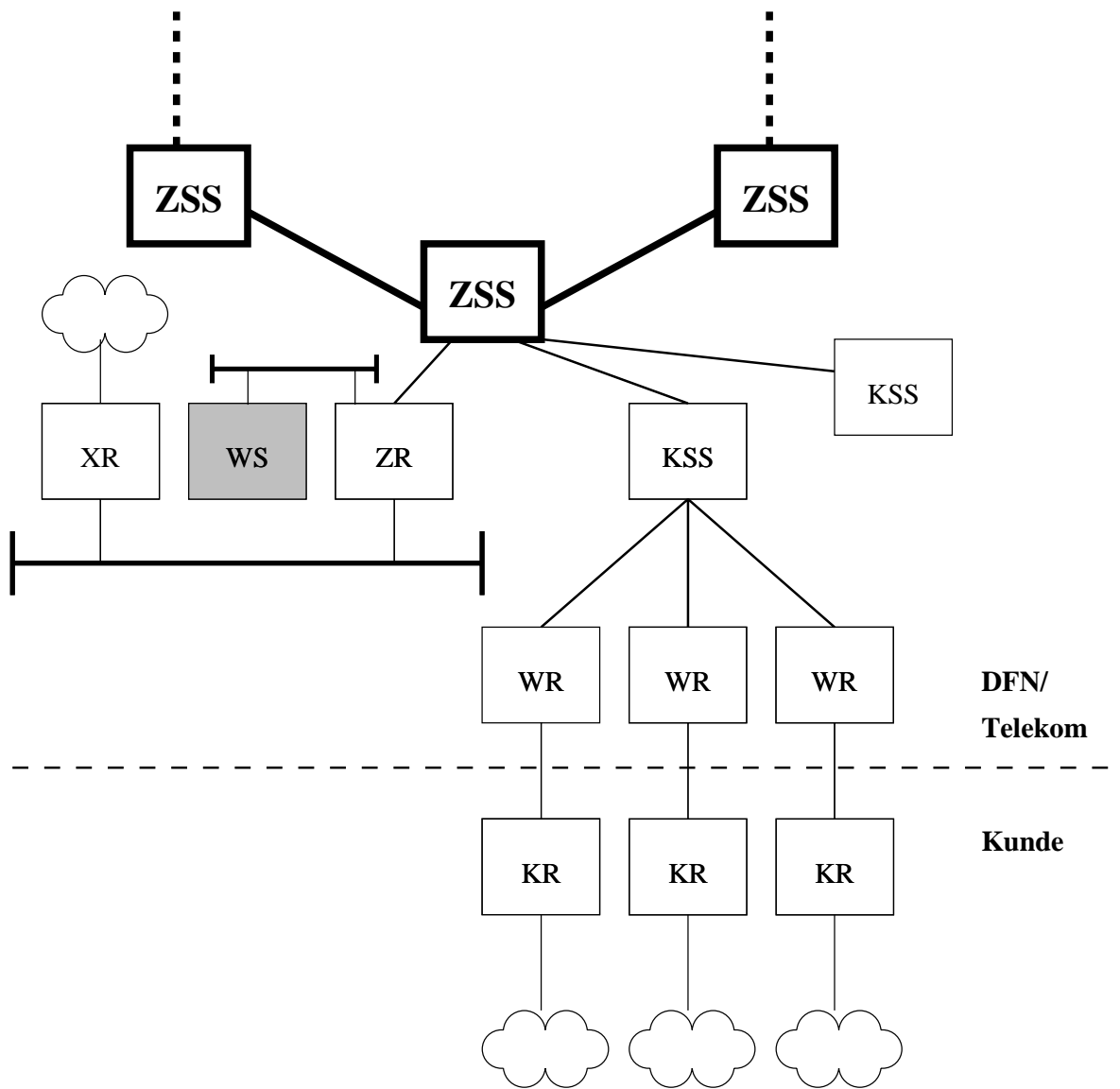
<sup>3</sup>Das Accounting an einem Router erfolgt auf der Basis der Interfaces, die ihn mit angeschlossenen Netzen verbinden. Für jedes Interface kann angegeben werden, ob der in ausgehende Richtung fließende Verkehr erfasst werden soll.

ist: Drei WR sind an ihn angeschlossen. Weiterhin ist an dem ZSS ein ZR eingezeichnet. Diese Komponenten waren auch schon in Abb. 3.3 zu sehen.

Neu in der Grafik sind mit dem ZR jeweils über ein lokales Netz verbunden, die grauschraffierte Workstation WS und der X.25 Router XR mit der Wolke, die das dahinterliegende X.25 Netz repräsentieren soll.

Wie im vorigen Abschnitt beschrieben, wird das Accounting an allen Interfaces der XR und WR eingeschaltet. In periodischen Abständen werden die dort gesammelten Daten von besagter Workstation abgeholt und zusammengefaßt zu Subnetzen bzw. Autonomen Systemen auf Festplatte zwischengespeichert. Diese Phase wird vom DFN-NOC organisiert. Weil ab da, eine IP-Adresse immer entweder einem Subnetz oder einem Autonomen System zugeordnet ist, wird ab jetzt auch der Begriff Subnetz/AS benutzt, was als “Subnetz beziehungsweise Autonomes System” zu lesen ist.

Alles weitere wird vom in dieser Arbeit erstellten Programmsystem erledigt. Es wird im folgenden Kapitel beschrieben.



ZSS Zentraler Service Switch  
 KSS Kunden Service Switch

WR WIN Router  
 KR Kunden Router

XR X.25 Router  
 WS Workstation  
 ZR Zentraler Router

Abbildung 4.3: Verfeinerte Darstellung eines ZSS

# Kapitel 5

## Beschreibung des Programmsystems

Dieses Kapitel widmet sich ausführlich dem erstellten Programmsystem. Es wird zunächst ein grober Überblick über die Bestandteile des Systems, ihre Aufgaben und ihre Relation zueinander gegeben. Es folgt die Beschreibung einer wichtigen Datenstruktur, der Datenbasis. Sie taucht in jeder Komponente in anderer Gestalt auf und wird an einer Stelle konzentriert beschrieben. Die sich anschließende Beschreibung der Schnittstellen erlaubt eine Abgrenzung und Einordnung des Systems von bzw. in seine Umwelt. Abschnitt 5.4 stellt das Nachrichtensystem vor, das die Komponenten zur Kommunikation benutzen. Den Abschluß bildet eine präzise Beschreibung der einzelnen Komponenten.

### 5.1 Das Programmsystem im Überblick

Das Programmsystem besteht aus vier Klassen von Aktivitätsträgern. An jedem Knotenstandort verarbeitet ein Prozeß (im folgenden *Außenposten (AP)* genannt) die angelieferten Daten vor und schickt sie an einen Prozeß an einem zentralen Standort (im folgenden *Zentrale* genannt). Eine dritte Komponente ist für die Aufbereitung und Aggregation der Daten zuständig. Sie wird *Statistik* genannt. Ein *Manager* kontrolliert das System<sup>1</sup>. Die weitere Erklärung erfolgt anhand des Schaubildes Abb. 5.1.

Am linken Rand der Abbildung sind einige Router aufgereiht. Sie stellen die WR und XR aus Abb. 4.3 dar. Die von ihnen ausgehenden Pfeile führen zu Accountingworkstations, die verfeinerte Darstellungen der in Abb. 4.3 grau eingezeichneten Workstation sind. Das Kästchen “Abfrage” im Inneren der WS repräsentiert den Abfragemechanismus des NOC.

Das Ergebnis der Abfrage gelangt über die zylindrische Form, die einen Plattenspeicher versinnbildlichen soll, an einen Außenposten. Der AP erhält die Daten der einzelnen Router in der Form Subnetz/AS (siehe dazu Tab. 4.1), faßt sie zusammen und ordnet sie einem Anschluß zu. Übersteigt die Menge der im AP gespeicherten Daten einen bestimmten Wert, werden die ältesten Daten an die Zentrale geschickt.

Die Zentrale wiederum sammelt die Daten der einzelnen AP und schreibt sie auf Platte. Dort werden sie einmal am Tag von der Statistikkomponente bearbeitet und die in Abb. 4.1 gezeigten Schritte durchgeführt.

---

<sup>1</sup>Ähnliche Architekturen werden in [IETF1], [IETF2] und [IETF3] vorgeschlagen.

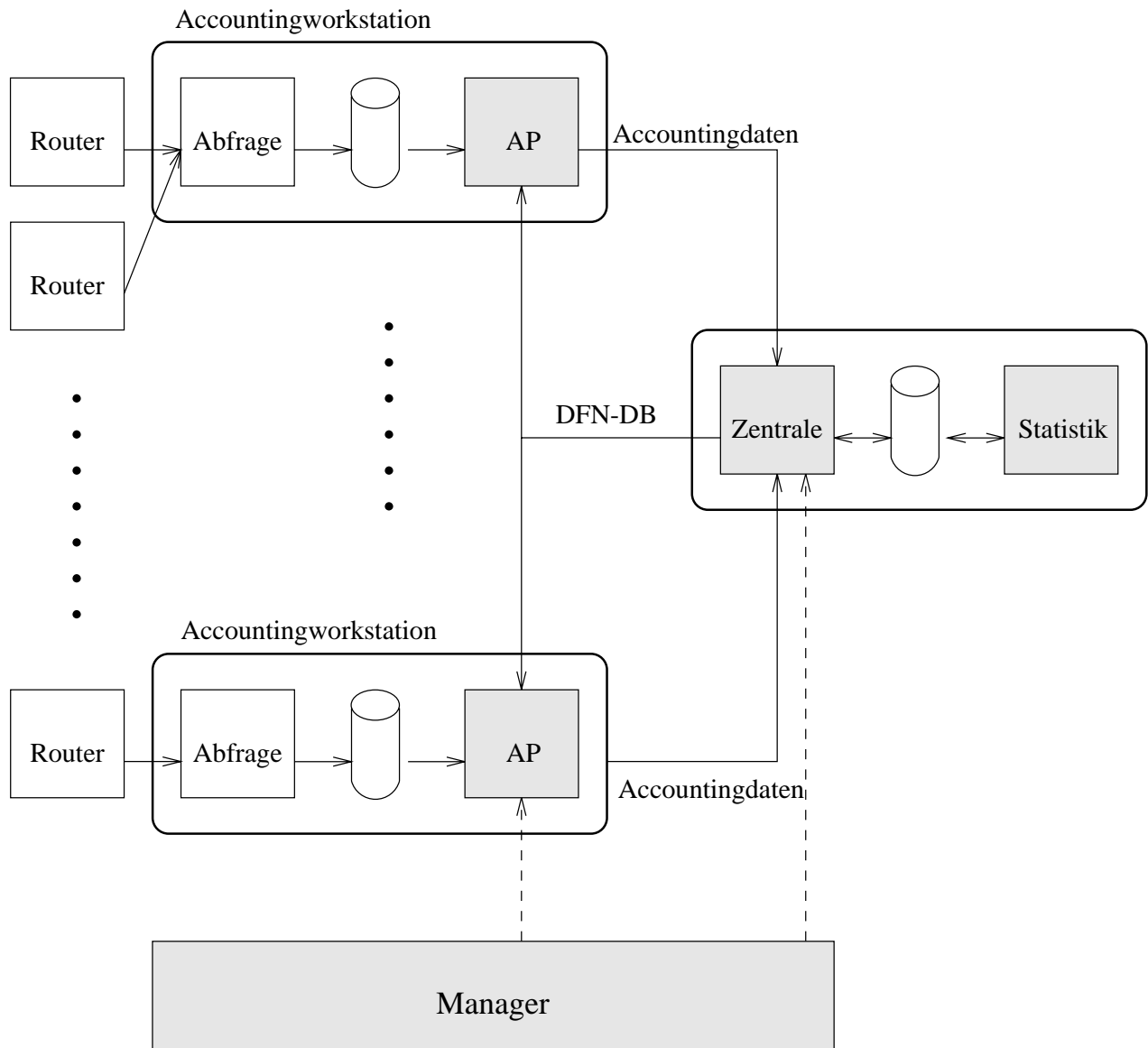


Abbildung 5.1: Architektur des Programmsystems. Grau hinterlegte Kästchen symbolisieren zum Programmsystem gehörende Komponenten.



Mit Hilfe des Managers können die Prozesse erstmalig gestartet werden und ihr Zustand abgefragt werden. Außerdem versorgt er die AP und die Statistik mit einer *Datenbasis*. Diese wird von den AP benötigt, um Subnetze/AS Anschlüssen zuordnen zu können. Die Statistik braucht sie, um den Standort der Anschlüsse bestimmen zu können.

Dazu kontaktiert der Manager die AP nicht direkt, sondern schickt stattdessen der Zentrale eine Datenbasis, die sie dann sofort an die AP weiterleitet. Sollte ein AP aufgrund einer Störung eine Datenbasis verlieren, kann er sie von der Zentrale auch anfordern. Der Umweg über die Zentrale wird beschritten, weil der Manager als Schnittstelle zum Menschen konzipiert ist und somit nicht ständig aktiv ist.

## 5.2 Datenbasis: Grundlage der Zuordnungen

In der Aufgabenstellung und im gerade gezeigten Überblick ist bereits angeklungen, daß das Programmsystem im Zuge der Verarbeitung gewisse Zuordnungen vornimmt. Zuerst wird ein Subnetz/AS einem Anschluß zugeordnet und dieser dann einem Standort. Das dafür notwendige Wissen ist in einer Datenbasis organisiert. Sie enthält folgende Informationen:

- Welche Subnetze/AS, Anschlüsse, Standorte gibt es?
- Welche Subnetze/AS gehören zu einem Anschluß?
- Welche Anschlüsse umfaßt ein Standort?

Aus welcher Quelle dieses Wissen geschöpft wird, enthüllt der Abschnitt 5.3.2. Im Programmsystem wird es an zwei Stellen angewendet: durch den Außenposten und durch die Statistik. Dabei wird jeweils nicht der komplette Inhalt benötigt, sondern nur ein Teil. Den Umfang dieser Teilmengen, und wie das Wissen repräsentiert wird, erläutern die beiden folgenden Abschnitte. Der sich daran anschließende Abschnitt macht noch auf eine besondere Problematik aufmerksam.

### 5.2.1 Datenbasis der Statistik

Die Statistikkomponente aggregiert Accountingdaten der feinsten zeitlichen Auflösungsstufe (= Viertelstunde) zu größeren Intervallen und errechnet aus Anschluß/Anschluß-Matrizen Standort/Standort-Matrizen (siehe dazu Abb.4.1). Die Datenbasis, die sie dazu benötigt, besteht deshalb aus zwei Listen. Die erste Liste enthält alle Standorte, die zweite Liste alle Anschlüsse mit der Angabe, zu welchem Standort ein Anschluß gehört.

Um die Listen rechnergerecht gebrauchen zu können, ist jedem Standort eine Zahl zwischen einschließlich Null und nicht einschließlich der Anzahl der Standorte zugeordnet. Analog ist jedem Anschluß eine Zahl zugeordnet. Diese Zahl heiße ab jetzt *Identität* des Standorts bzw. des Anschlusses. Die Identitäten werden beim Einlesen der Datenbasis (in Abschnitt 5.3.2 beschrieben) vergeben und konsistent in allen Programmbereichen verwendet. Sie dienen gleichzeitig als Indizes für zu erstellende Matrizen.

Die Statistikdatenbasis wird beim Einlesen als Datei abgespeichert. Hier ein Ausschnitt als Beispiel:

```
N_KNOTS 14 N_CUSTS 76
STANDORTE
0 Berlin
1 FEHLER
2 Hamburg
3 Hannover
4 Karlsruhe
5 Koeln
6 Leipzig
7 Muenchen
8 Unknown
9 Non-WiN
10 Nuernberg
11 Stuttgart
12 Wiesbaden
13 X.25-WiN
ANSCHLUESSE
0 AWI1 2
1 DESY1 2
2 DKRZ1 2
3 DLR-Koeln1 5
4 FEHLER 1
5 FH-Amberg1 10
6 FH-Aschaffenburg1 10
7 FH-Coburg1 10
8 FH-Deggendorf1 7
9 FH-Hof1 10
10 FH-Ingolstadt1 10
[abgeschnitten]
```

Vier Schlüsselworte erleichtern dem Rechner das Einlesen der Datei. Hinter *N\_KNOTS* bzw. *N\_CUSTS* steht jeweils die Anzahl der Standorte bzw. Anschlüsse. *STANDORTE* leitet die Liste der Standorte ein. Auf jeder Zeile befindet sich ein Standort, wobei zuerst die Identität und dann der Name des Standortes angegeben ist. Ab *ANSCHLUESSE* folgt die Liste der Anschlüsse. Für diese Liste gilt das gleiche Format wie bei der Standortliste mit der Ausnahme, daß ein dritter Eintrag in der Zeile mit der Identität des Standortes, zu dem der Anschluß gehört, aufgeführt wird.

Liest die Statistikkomponente die Datei ein, werden die beiden Listen in Vektoren gespeichert. Da die Listen sortiert sind, kann mit binärer Suche auf die Elemente zugegriffen werden.

### 5.2.2 Datenbasis des Außenpostens

Der Außenposten ordnet Subnetze und Autonome Systeme Anschlüssen zu. Er unterhält deswegen eine Datenbasis, die für jedes Subnetz und jedes Autonome System den zugehörigen Anschluß ausweist, wobei der Anschluß durch seine Identität vertreten wird.

Die Repräsentation der Datenbasis erfolgt durch zwei getrennte Hashtabellen. Die erste Hash-tabelle enthält alle bekannten Subnetze und den jeweils zugehörigen Anschluß. Der Schlüssel zu einem Eintrag ist die arithmetische Summe der Netzadresse des Subnetzes und der Netz-maskenlänge. Kollisionen werden durch fortgesetzte Addition eines konstanten Summanden aufgelöst. Ähnliches gilt für die zweite Hashtabelle, die alle verwendeten Autonomen Systeme enthält. Der Zugriff erfolgt hier über die Nummer des AS.

Am Außenposten ist keine Information über Standorte vorhanden. Auch die konkreten Na-men der Anschlüsse sind ihm nicht bekannt, da er nur die Identitäten benutzt.

### 5.2.3 Berücksichtigung von Veränderungen

Der Inhalt der Datenbasis ist einem steten Wandel unterworfen. Es können Standorte, An-schlüsse, Subnetze oder Autonome Systeme hinzukommen, wegfallen, oder es könnte sich eine Zugehörigkeit ändern. Dementsprechend ist mit häufigen Aktualisierungen zu rechnen. Als Konsequenz ist jede konkrete Datenbasis nur für einen bestimmten Zeitraum gültig. Deshalb müssen Accountingdaten immer einen Verweis auf die Datenbasis enthalten, auf Grundlage derer sie erzeugt wurden. Werden die Accountingdaten dann zu einer größeren Einheit zusammengefaßt, müssen die beteiligten Datenbasen zu einer einzigen verschmelzt werden. Abb. 5.2 illustriert diesen Umstand. Wie die Verschmelzung bewerkstelligt wird, erklärt Abschnitt 5.5.3.

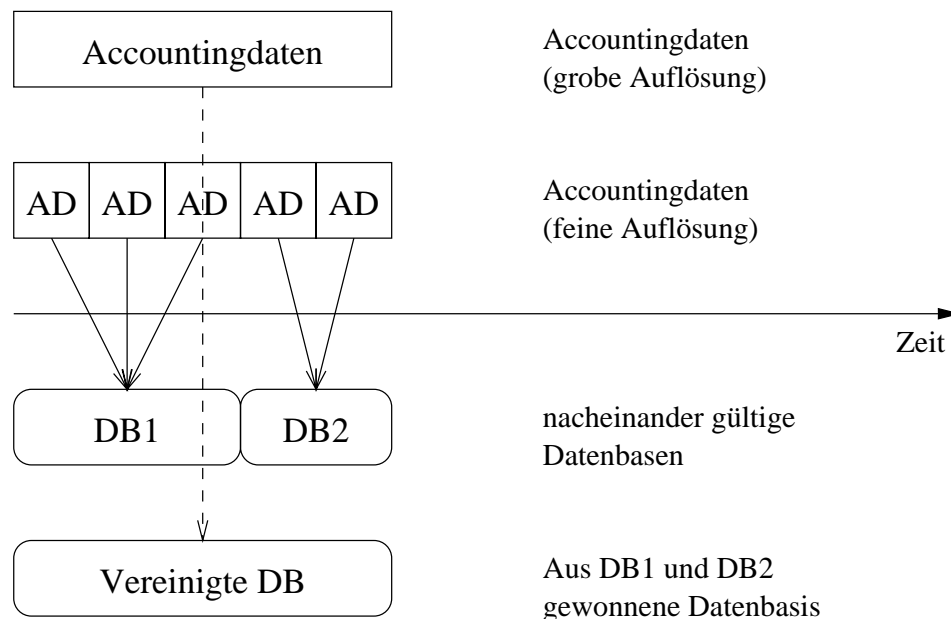


Abbildung 5.2: Vereinigte Datenbasis

## 5.3 Schnittstellen

Abb.5.3 zeigt, daß das Programmsystem drei Datenschnittstellen aufweist: Die Eingabedaten werden unter Einfluß von Steuerungsdaten zu Ausgabedaten verarbeitet. Im folgenden wird

jede dieser Schnittstellen beleuchtet.

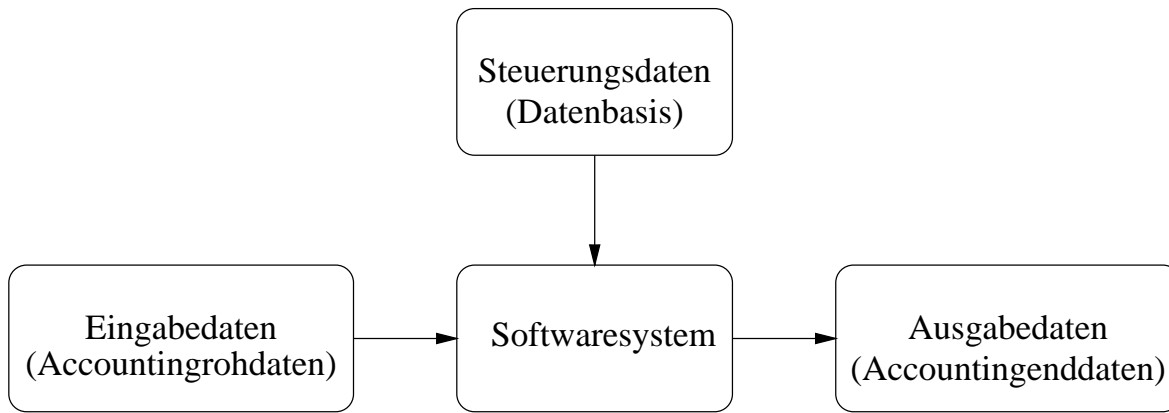


Abbildung 5.3: Schnittstellen des Programmsystems

### 5.3.1 Eingabedatenschnittstelle: Rohdaten

Die Eingabedatenschnittstelle ist an der Accountingworkstation anzusiedeln. Auf der einen Seite befindet sich das Abfragesystem des NOC, auf der anderen der Außenposten. Bis sechs Wochen vor dem Abgabetermin der Arbeit stand diese Schnittstelle nicht genau fest. Es wurde nur soviel vereinbart: Die Übergabe geschieht mittels einer Datei, welche folgende Information enthält:

- Die IP-Adresse des Routers, von dem die Accountingdaten stammen.
- Der Zeitraum, in dem die Accountingdaten ermittelt wurden, im Format Startzeitpunkt (in Sekunden nach dem 01.01.1970) + Offset (in Sekunden).
- Die Accountingdaten selbst (Quelle, Ziel, Anzahl Pakete, Anzahl Bytes). Quelle und Ziel sind vom NOC bereits einem Subnetz/AS zugeordnet.

Letztendlich einigte man sich dann auf dieses Übergabesystem: Die IP-Adresse des Routers und der Startzeitpunkt sind im Dateinamen enthalten. Die Länge des Intervalls ist implizit durch den Startzeitpunkt der letzten Übergabedatei gegeben. Ein periodisch laufender Prozeß überprüft, welche Dateien seit seinem letzten Aufruf hinzugekommen sind. Dazu unterhält er eine Datei, in der für jeden Router der letzte Startzeitpunkt vermerkt ist. Aufgrund dieser Information, kann er neue Dateien identifizieren und gleichzeitig das Accountingintervall bestimmen. Nachdem er nun Startzeitpunkt und Offset der neuen Dateien ermittelt hat, schreibt er diese Informationen zusammen mit dem jeweiligen Dateinamen in eine gesonderte Datei und löst am Außenposten ein Signal aus. Dies veranlaßt den AP, die gesonderte Datei zu lesen und die darin enthaltenen Einträge abzuarbeiten und schließlich zu entfernen.

### 5.3.2 Steuerungsdatenschnittstelle: Datenbasis

Der Inhalt der Datenbasis ist hinlänglich in Abschnitt 5.2 beschrieben worden. Die Datenbasis ist nicht selbst Gegenstand der durch das Programmsystem zu vollziehenden Daten-

verarbeitung. Vielmehr findet der Vollzug unter ihrem Einfluß statt. Deshalb taucht sie hier unter dem Begriff “Steuerungsdaten” auf.

Neue Steuerungsdaten in das System einzubringen, ist Aufgabe des Managers. Er erwartet sie als Textdatei in folgender Form:

```

STANDORTE
    Standort1
    Standort2
    .
    .
    .
ANSCHLUESSE
    Anschluss1
    Anschluss2
    .
    .
    .
NTAB
    a . b . c . d / m ; <Anschluss> ; <Standort>
    .
    .
    .
ATAB
    ASxxxxx ; <Anschluss> ; <Standort>
    asxxxxx ; <Anschluss> ; <Standort>

```

Die ausschließlich in Großbuchstaben gedruckten Wörter sind Schlüsselwörter, die für ein korrektes Einlesen sorgen. Die Wörter “STANDORTE” und “ANSCHLUESSE” sprechen für sich selbst. “NTAB” leitet die Liste der Subnetze ein. “a.b.c.d/m” bezeichnet ein Netz in CIDR-Notation. Dahinter folgen der Anschluß, dem das Netz zugeordnet ist und der Standort. “ATAB” kennzeichnet den Beginn der Liste mit den Autonomen Systemen. Die Syntax ist ähnlich wie in “NTAB”-Sektion.

Diese feste Form der Schnittstelle wurde gewählt, weil noch nicht völlig klar ist, auf welchem Weg die Steuerungsdaten beschafft werden. Ursprünglich war vorgesehen, die Daten mittels der *DFN-Datenbasis* zu gewinnen, welche Angaben über die im B-WiN existierenden Netze und Router enthält und vom DFN-Verein verwaltet wird.

Die benötigte Information ist in der DFN-Datenbasis implizit gegeben. Die eingetragenen Netze beinhalten die Angabe des WiN-Routers über den sie zu erreichen sind, womit der Anschluß identifiziert worden ist. Der WiN-Router wiederum weist einen Eintrag aus, über welchen ZR er zu erreichen ist, was den Standort liefert.

Zur Zeit stellt das NOC eine Verzeichnisstruktur zur Verfügung, in der die geforderte Information expliziter gegeben ist. Zur Anpassung an das gerade benutzte System wird eine separate Komponente benutzt, die nicht Bestandteil des in dieser Arbeit erstellten Programmsystems ist.

### 5.3.3 Ausgabedaten: Accountingenddaten

Im Kapitel 4 (Aufgabenstellung) wurde bereits aufgeführt, welche Daten zu erzeugen sind. Es sind dies Verkehrsflußdaten in verschiedenen Zuordnungen und zeitlichen Auflösungen. Sie werden in Form von Textdateien gespeichert. Zur Illustration sei hier ein Auszug aus einer Datei mit Verkehrsflußdaten zwischen Anschlüssen:

```

VERSION 846072000
AP 3249425322 1
AP 3249425326 1
AP 2210155045 1
DATA
18;49;6426;3113982;69156;
19;49;801;54729;1674;
21;49;2322;977526;21906;
23;49;126;26829;639;
24;49;3762;1327689;30168;
26;49;9;630;18;
39;49;576;35073;1107;
41;49;27;1881;54;
45;49;603;167643;3888;
49;23;135;9171;279;
49;24;5652;710604;18567;
65;49;2241;130293;4203;

```

Die Dateien werden in einer Verzeichnishierarchie abgelegt. In der obersten Ebene sind Verzeichnisse angesiedelt, die Daten für jeweils einen Monat im Jahr enthalten. Innerhalb dieser Monatsverzeichnisse sind Verzeichnisse für die Tage des Monats. Im Tagesverzeichnis befinden sich die Dateien der Viertelstunden<sup>2</sup>. Sie enthalten im Dateinamen die Startzeit der Viertelstunde.

Das Accountingintervall einer Datei ergibt sich somit aus der Lage der Datei in der Hierarchie und aus ihrem Dateinamen.

Die erste Zeile einer Datei enthält die Version der Datenbasis mit der die Datei erstellt worden ist (Schlüsselwort *VERSION*). Ohne die Angabe der Version ist die Datei nutzlos, da nur die Identitäten abgespeichert werden und nicht die Klartextnamen der Anschlüsse oder Standorte. Durch Verwendung der Identitäten wird zum einen Platz gespart und zum anderen wird die maschinelle Weiterverarbeitung erleichtert, da die Daten schon in kodierter Form vorliegen.

Die beiden Zahlen hinter dem Schlüsselwort *AP* teilen mit, welche Außenposten Daten für diese Datei beigesteuert haben. Die erste Zahl ist die IP-Adresse des Außenpostens, die zweite gibt an, wie oft der AP Daten angeliefert hat. Durch diese Angabe können Fehlfunktionen aufgespürt werden.

Nach dem Schlüsselwort *DATA* folgen Zeilen mit fünf durch Strichpunkt getrennten Feldern. Die ersten beiden Einträge geben Quell- und Zielidentität an. Es schließen sich die Anzahl übertragener Pakete, Bytes und ATM-Zellen an.

---

<sup>2</sup>Neben der in 4.1 definierten allgemeinen Bedeutung, kann ab jetzt mit "Viertelstunde" auch eine konkrete Viertelstunde gemeint sein, zum Beispiel von 14:00 bis 14:15.

## 5.4 Nachrichtensystem

Ein verteiltes System, das gemeinsam eine Aufgabe bewältigt, benötigt ein Nachrichtensystem. Das in dieser Arbeit implementierte Nachrichtensystem setzt auf dem verbindungsorientierten, zuverlässigen Transportprotokoll TCP auf. Die darin enthaltenen fehlertoleranten Maßnahmen kommen somit auch dem Gesamtsystem zugute. Im folgenden werden erst die Nachrichtentypen, die zur Kommunikation verwendet werden, vorgestellt, bevor dargestellt wird, welche Nachrichtentypen zwischen den einzelnen Teilnehmern des verteilten Systems zur Anwendung kommen.

### 5.4.1 Beschreibung des Headers

Jede Nachricht besteht aus zwei Teilen, nämlich einem Header und einem Body. Der Header ist von fester Länge und für alle Typen gleich. Er beinhaltet folgende Felder:

Name	Länge (in Bytes)
Typ der Nachricht	1
Länge des Bodies	4
Zeitstempel	4
Option	4
Schlüssel	16
<hr/>	
$\Sigma$ 29	

Tabelle 5.1: Felder des Headers

Anhand des Typs kann ein Empfänger feststellen, welche Nachricht er bekommen hat und wie er sie behandeln muß.

Das zweite Feld gibt die Länge des Bodies an. Jede Nachricht wird in zwei Schritten gelesen. Zuerst der Header und dann der Body. Die Angabe der Bodylänge ermöglicht einem Empfänger, genau die Menge an Speicher zu reservieren, die für die Aufnahme des Bodies nötig ist. Die angegebene Länge kann dann mit der tatsächlich gelesenen verglichen werden, was eine Fehlererkennung ermöglicht. Selbige Fehlererkennung erfolgt auch mit dem Header, da dessen Länge bekannt ist.

Der Zeitstempel enthält die Absendezeit der Nachricht und dient zu Kontrollzwecken. Das Optionsfeld bietet Platz, welcher von jedem Nachrichtentyp unterschiedlich genutzt wird. Es teilt Information mit, die noch vor dem Lesen des Bodies von Interesse sind.

Die restlichen 16 Bytes sind momentan nicht genutzt. Es ist vorgesehen, sie für Verschlüsselungsoperationen oder Paßwortmechanismen zu verwenden.

## 5.4.2 Beschreibung der Nachrichtentypen

### Typ: Datenbasis Transfer

Der Nachrichtentyp “Datenbasis Transfer” wird zum Verschicken einer Datenbasis verwendet. Das Option-Feld des Headers enthält den Zeitpunkt, ab dem die Datenbasis gültig wird. Der Body ist in drei Teile gegliedert. Der erste Teil enthält die Anzahl der Anschlußidentitäten und die Anzahl der Subnetze und Autonomen Systeme.

Sodann folgt eine Liste der Subnetze, wobei jedes Listenelement aus der Netzadresse, der Länge der Netzmaske und der zugewiesenen Anschlußidentität besteht. Den Abschluß bildet die Liste der Autonomen Systeme unter Angabe der Nummer des AS und wiederum der Anschlußidentität.

### Typ: Datenbasisanforderung

Der Nachrichtentyp “Datenbasisanforderung” besteht nur aus einem Header. Das Optionsfeld gibt die Zeit an, für die die Datenbasis benötigt wird.

### Typ: Keine Datenbasis verfügbar

Dieser Nachrichtentyp transportiert nur eine Information, nämlich, daß eine vorausgehende Anforderung nicht erfüllt werden kann, weil für den gewünschten Zeitpunkt keine Datenbasis vorhanden ist. Dementsprechend hat diese Nachricht keinen Body und das Optionsfeld ist unbenutzt.

### Typ: Schicke Daten

Eine komplette Anschluß/Anschluß-Accountingdatenmatrix wird übertragen. Die Viertelstunde, für die die Matrix gilt, steht im Optionsfeld. Die ersten vier Bytes des Bodies enthalten die Versionsnummer der Datenbasis anhand derer die Daten verarbeitet wurden. Den Rest des Bodies füllt die Matrix.

### Typen: Hallo und Hallo OK

Der Typ “Hallo” überprüft den Zustand einer Komponente. Wenn die Komponente aktiv ist, schickt sie ein “Hallo OK” zurück. Beide Typen bedürfen momentan keines Bodies. Das Optionsfeld ist ebenso unbenutzt. Dies kann sich aber in Zukunft ändern, wenn mehr Zustandsinformation abgefragt werden soll.

### Typ: Kill

Eine Komponente, die diesen Nachrichtentyp erhält, wird aufgefordert sich zu beenden. Die Nachricht hat keinen Body und das Optionsfeld ist leer.

Tabelle 5.2 faßt die Typen zusammen.



Nachrichtentyp	Optionsfeld	Body
Datenbasis Transfer	Gültigkeitszeitpunkt	Anzahl Identitäten, Subnetze, AS, Liste mit Subnetzen/AS
Datenbasis Anforderung	Gültigkeitszeitpunkt	Keiner
Keine Datenbasis verfügbar	Unbenutzt	Keiner
Schicke Daten	Accountingzeitpunkt	Datenbasis Version; Matrix
Hallo	Unbenutzt	Keiner
Hallo OK	Unbenutzt	Keiner
Kill	Unbenutzt	Keiner

Tabelle 5.2: Nachrichtentypen

## Kommunikationsstruktur

Zu Beginn dieses Kapitels wurde die Architektur des Programmsystems im Überblick vorgestellt. Dabei wurde angedeutet, welche Kommunikation zwischen den Komponenten stattfindet. Zum Beispiel enthält Abb. 5.1 mit “Accountingdaten” beschriftete Pfeile, die von den Außenposten zur Zentrale führen. Jetzt, da die Nachrichtentypen bekannt gemacht wurden, wird klar, daß dies mittels des Typs “Schicke Daten” geschieht. Die Abbildung ist aber hinsichtlich des Nachrichtensystems unvollständig und ungenau, da es an dieser Stelle auf die Einordnung des Programmsystems in sein Umfeld und auf Lage und Aufgabe der Komponenten ankam.

Abbildung 5.4 zeigt nun, welche Nachrichten zwischen den einzelnen Komponenten ausgetauscht werden. Der Inhalt der Zeichnung orientiert sich an der Spezifikationsmethode *PASS*, die als Teil einer Programmierumgebung für verteilte Echtzeitanwendungen in [HoAn89] vorgestellt wird.

Nachrichten werden nur zwischen den Komponenten des Programmsystems (Manager, Zentrale, Außenposten) ausgetauscht. Natürlich sind in der Realität mehrere Instanzierungen der Komponente “Außenposten” vorhanden, stellvertretend ist aber hier nur eine eingezeichnet.

Um an dieser Stelle ein vollständiges Bild aller auftretender Kommunikation zu zeichnen, sind auch der Schnittstellenprozeß, der die Abfrage des NOC mit dem Programmsystem verbindet, und die Abfrage des NOC selbst in der Graphik enthalten. Mit der Abfrage des NOC erfolgt keine direkte Kommunikation, sondern es werden vom AP und vom Schnittstellenprozeß die vom NOC abgelegten Accountingrohdaten gelesen. Der Schnittstellenprozeß ist nicht permanent vorhanden, sondern wird periodisch gestartet, was durch das Uhrsymbol verdeutlicht werden soll, erledigt seine Aufgabe und terminiert.

Welche Aktionen durch die kommunikativen Handlungen ausgelöst werden, wird bei der Ablaufbeschreibung der einzelnen Komponenten gezeigt.

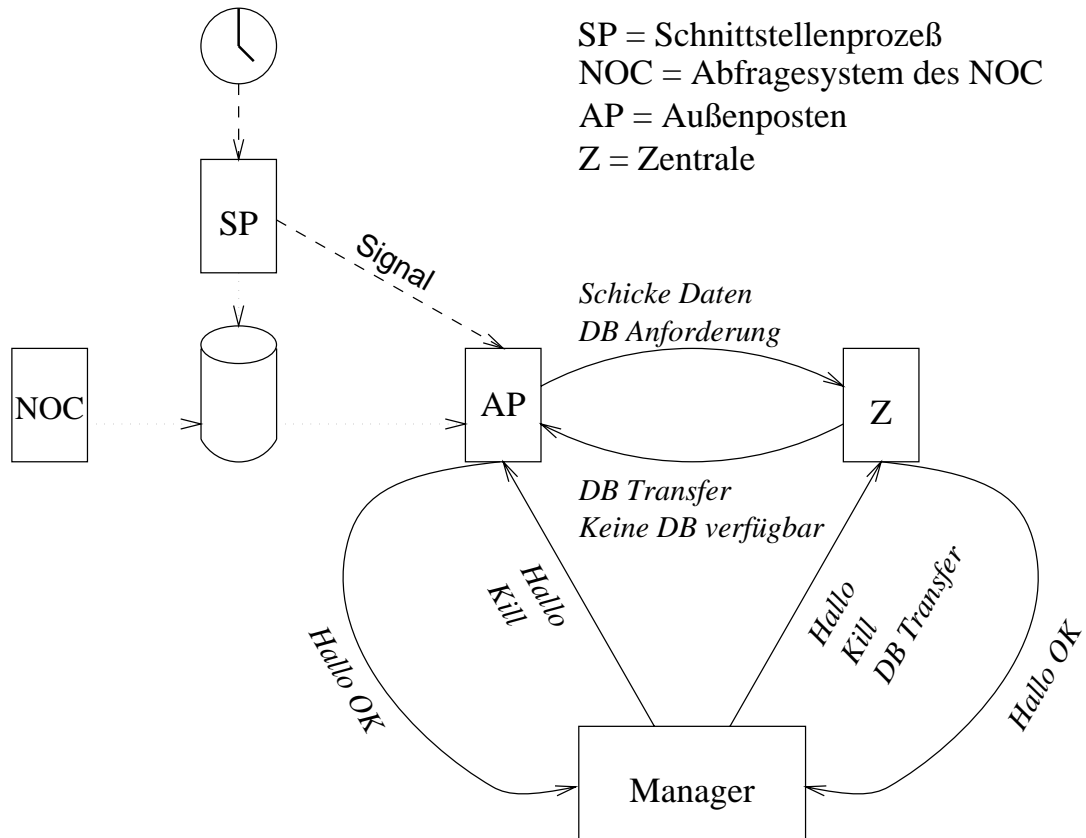


Abbildung 5.4: Kommunikationsstruktur des Programmsystems

## 5.5 Beschreibung der Komponenten

### 5.5.1 Zentrale

Die Zentrale nimmt die Viertelstundenaccountingdaten der Außenposten entgegen, addiert sie und schreibt sie auf Festplatte. Außerdem verteilt sie vom Manager geschickte Datenbasen an die Außenposten und kann im Bedarfsfall Datenbasen auf Anfrage verschicken. Im Sinne des Client/Server-Konzeptes ist sie ein Server, der auf Anfragen wartet. Gelegentlich wird sie auch zum Client und zwar dann, wenn vom Manager eine neue Datenbasis eintrifft, die sie dann an alle Außenposten sendet.

#### Serverart

Die Serverfunktion ist sequentieller Natur (siehe dazu Abb. 5.5); das heißt, sobald ein Auftrag eintrifft, wird er komplett bearbeitet und dann der nächste Auftrag entgegengenommen. Während der Bearbeitungszeit ankommende Aufträge werden von der Netzwerksoftware in eine Warteschlange eingereiht.

Eine andere Möglichkeit wäre gewesen, für jeden Auftrag einen Prozeß zu generieren, der den Auftrag bearbeitet, während der ursprüngliche sofort wieder für neue Aufträge bereitsteht (Paralleler Server, siehe dazu Abb. 5.6). Auf einem Monoprozessor bedeutet dieses Konzept jedoch eine Verringerung des Durchsatzes, da zusätzlicher Aufwand durch Prozeßgenerierung, -verwaltung und -umschaltung verursacht wird. Weiterhin beansprucht diese Alternative mehr Speicherplatz.

Speziell für die vorliegende Anwendung ist zu beachten, daß bei der parallelen Variante die einzelnen Prozesse auf einem gemeinsamen Datenbestand arbeiten, so daß synchronisierende Maßnahmen getroffen werden müssen, wodurch sich die Komplexität erhöht.

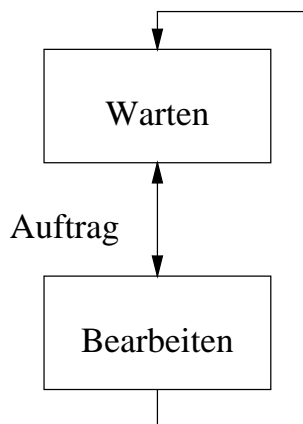


Abbildung 5.5: Sequentieller Server

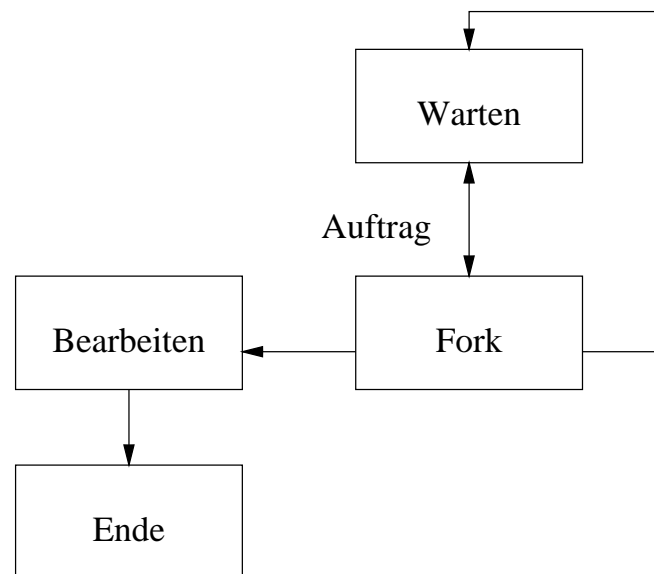


Abbildung 5.6: Paralleler Server

Bei einem Multiprozessorsystem ließe sich sicherlich ein erhöhter Durchsatz an Serveraufträgen erzielen. Der Gesamtdurchsatz des Rechensystems hänge aber von der übrigen Last ab. Bei geringer Belastung stiege er, bei einer permanenten Auslastung aller Funktionseinheiten wäre der Grenzdurchsatz kleiner als bei der gewählten sequentiellen Lösung.

Da die von der Zentrale zu erfüllende Aufgabe nicht sehr rechenaufwendig ist und im geforderten Zeitrahmen<sup>3</sup> ohne weiteres erledigt werden kann, wurde die insgesamt ressourcenschonendere, effizientere Variante gewählt.

## Datenmodell

Bevor der funktionelle Ablauf der Zentrale erläutert wird, werden zwei zentrale Datenstrukturen vorgestellt, nämlich die Viertelstundenliste und die Liste der Datenbasen.

Die Zentrale behält immer eine gewisse Anzahl an aktuellen Viertelstundenaccountingdaten im Programmspeicher, wodurch ein ständiges Speichern und Laden auf den Hintergrundspeicher vermieden wird. Bei Erreichen der (einstellbaren) Grenze werden die ältesten Daten auf Platte geschrieben, von wo sie bei Bedarf von der Zentrale aber auch wieder eingelesen werden können.

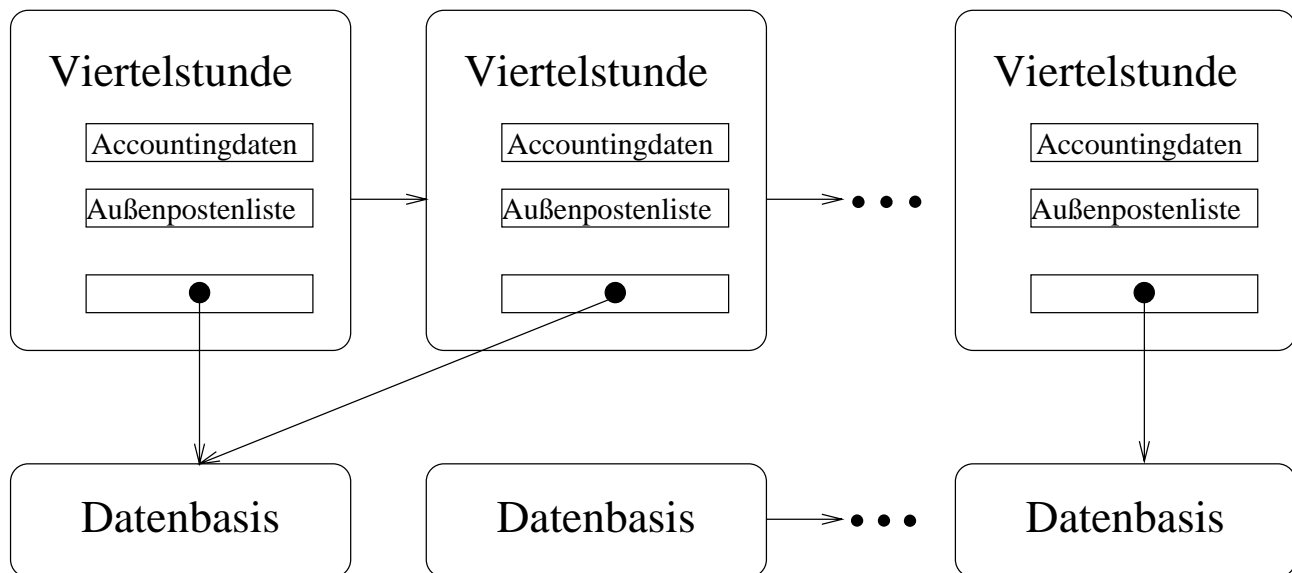


Abbildung 5.7: Zentrale Datenstrukturen der Zentrale

Neben den eigentlichen Accountingdaten enthält ein Viertelstundeneintrag auch eine Liste mit den derzeit bekannten Außenposten. Für jeden der Außenposten ist in der Liste vermerkt, wie oft er für die Viertelstunde Daten geliefert hat. Im Normalfall sollte jeder AP genau einmal Daten besteuern. Deshalb warnt die Zentrale, wenn dieser Wert überschritten wird. Die Außenpostenliste findet sich auch in der Viertelstundendatei wieder.

Eng verzahnt mit der Viertelstundenliste ist die Liste der Datenbasen, denn jeder Eintrag der Viertelstundenliste referenziert einen Eintrag in der Datenbasenliste. Sobald eine neue Datenbasis an der Zentrale eintrifft, wird sie in die Liste aufgenommen. Gleichzeitig wird

<sup>3</sup>Alle 15 Minuten ist mit einem Schwung an Aufträgen zu rechnen.

versucht, nicht mehr benötigte Datenbasen zu entfernen. Eine Datenbasis wird als nicht mehr benötigt angesehen, falls jedes der folgenden drei Kriterien zutrifft:

- Es gibt eine Datenbasis neueren Datums.
- Kein Eintrag der Viertelstundenliste enthält eine Referenz auf die fragliche Datenbasis.
- Die Datenbasis hat ein bestimmtes Alter überschritten.

Beide Listen und ihre Abhängigkeit zeigt Abb. 5.7.

## Ablaufbeschreibung

Sobald die Zentrale gestartet wurde, belegt sie den für das Softwaresystem vorgesehenen Port<sup>4</sup> und wartet an diesem auf das Eintreffen von Nachrichten. Die Zentrale versteht fünf der in Abschnitt 5.4.2 vorgestellten Nachrichtentypen. Alle anderen an dem Port ankommenden Daten werden abgelehnt.

Nur infolge einer akzeptierten Nachricht führt die Zentrale weitere Aktionen aus. Welche Nachrichten das sind, und was sie für Aktionen nach sich ziehen, zeigt Abb. 5.8 im Überblick. Die folgenden Zeilen geben eine detaillierte Beschreibung.

Trifft eine neue Datenbasis ein, wird sie in die Datenbasisliste eingefügt. Bei der bislang gültigen Datenbasis wird der Gültigkeitszeitpunkt der neuen Datenbasis als Endezeitpunkt eingetragen. Anschließend wird versucht, mittels des weiter oben geschilderten Verfahrens nicht mehr benötigte Datenbasen zu löschen.

Anzumerken ist, daß die Zentrale die in der Datenbasis vorhandenen Subnetz/AS-Listen gar nicht extrahiert und Hashtabellen anlegt, wie dies ein AP tun würde, sondern lediglich den empfangenen Bytestrom abspeichert, um ihn dann wieder verwenden zu können, wenn sie die Datenbasis an die Außenposten weiterleitet. Die genannten Listen werden nämlich von der Zentrale gar nicht benötigt, da die Zuordnung schon an den AP vorgenommen wurde. Wirklich gebraucht wird an der Zentrale nur, wann die Datenbasis gilt und wieviele Anschlüsse sie enthält, und nur soviel wird extrahiert.

Als letzte Aktion der für diese Nachricht vorgesehenen Aktionenfolge, wird die neue Datenbasis an alle Außenposten geschickt. Sollte ein Außenposten nicht erreichbar sein, versucht die Zentrale bis zu dreimal, ihn doch noch zu versorgen. Ist die Zustellung dann immer noch nicht erfolgt, gibt die Zentrale auf und gibt eine Warnung von sich, da anzunehmen ist, daß der AP nicht aktiv ist. Wird der AP daraufhin neugestartet, wird er sehr bald merken, daß er über keine Datenbasis verfügt und von sich aus die Zentrale kontaktieren.

Damit sind wir beim Nachrichtentyp “Anforderung einer Datenbasis”, der von der Zentrale verstanden wird. Die Zentrale durchsucht dabei die Datenbasenliste. Ist die Suche erfolgreich, wird dem Auftraggeber die gewünschte Ware geliefert. Bei erfolgloser Nachschau, wird stattdessen eine Nachricht vom Typ “Keine Datenbasis verfügbar” zurückgeschickt.

Ein einfaches “Hallo” beantwortet die Zentrale mit einem ebenso einfachen “Hallo OK”.

---

<sup>4</sup>Momentan der Geburtstag des Autors.

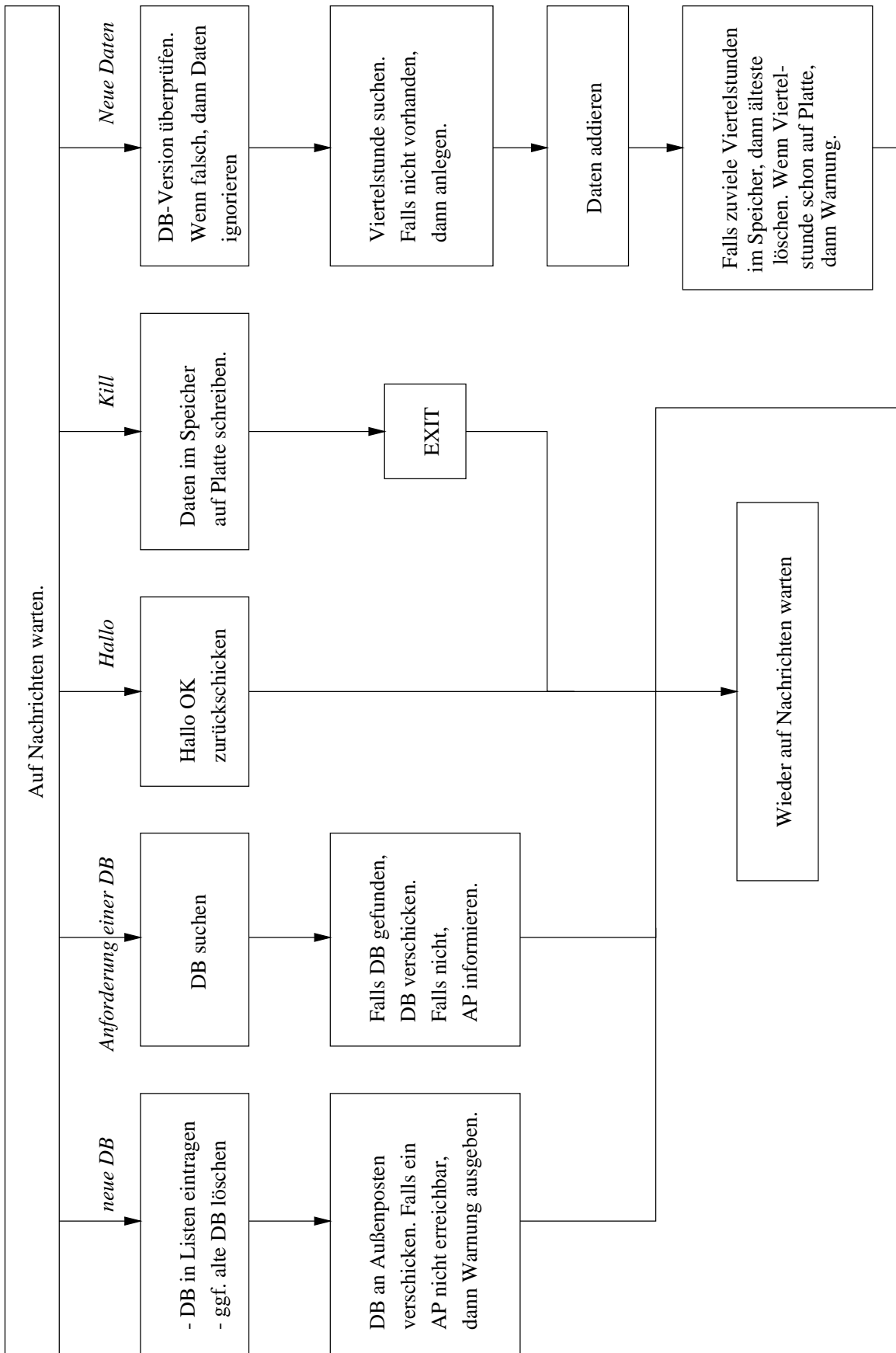


Abbildung 5.8: Funktionales Modell der Zentrale

Bei der Nachricht “Kill” werden noch sämtliche sich im Speicher befindende Accountingdaten auf Platte gerettet, bevor sich die Zentrale terminiert.

Die letzte zu besprechende Nachricht ist der Typ “Neue Daten”. Als erstes vergleicht die Zentrale, ob die Datenbasisversion, mit der der AP die Daten erstellt hat, mit der Version, die sie für diesen Zeitabschnitt erwartet, übereinstimmt. Ist dies nicht der Fall, ist ein schwerwiegender Fehler aufgetreten, weil nicht zu entscheiden ist, welche Version die richtige ist. Deshalb erzeugt die Zentrale eine Warnung und behandelt die Daten nicht weiter.

Geht man davon aus, daß die Zentrale die richtige Version verwendet, was der wahrscheinlichere Fall ist, da der Weg zur Zentrale kürzer ist als zum AP, also mit weniger Fehlerquellen behaftet ist, könnte man zusätzlich zur Ausgabe eine Warnung auch folgendes Verfahren anwenden: Die Zentrale informiert den AP, daß seine Versionsnummer falsch ist. Der AP fordert daraufhin die Datenbasis der Zentrale an und rekalkuliert die Accountingdaten. Sodann schickt er sie erneut an die Zentrale. Aufgrund des gestellten Zeitrahmens ist diese Vorgehensweise zugunsten anderer Arbeiten zurückgestellt worden, zumal bei massiven Tests die Situation nicht aufgetreten ist, und konnte letztendlich nicht mehr verwirklicht werden.

Stimmen die Versionen jedoch überein, überprüft die Zentrale, ob für die angekommenen Daten bereits ein Eintrag in der Viertelstundenliste enthalten ist, zu dem die Daten addiert werden könnten. Ist dies nicht der Fall, wird ein neuer Eintrag in der Liste erzeugt. Dazu muß dynamisch Speicher für die Datenmatrix angefordert werden. Um die Größe des Speichers zu bestimmen, wird zu dem geschaffenen Viertelstundeneintrag die dafür zuständige Datenbasis assoziiert, aus der diese Information hervorgeht, nämlich, wieviele Anschlüsse vorhanden sind.

Egal, ob der Viertelstundeneintrag schon bestand oder erst erzeugt wurde, ist er jetzt auf alle Fälle vorhanden. Die neu angekommenen Daten werden addiert.

Falls die Viertelstundenliste nun zu lang geworden sein sollte, wird der älteste Eintrag in eine Datei transferiert. Sollte die Datei schon vorhanden sein, wird eine Warnung erzeugt, und die Daten aus dem Speicher zu denen in der Datei addiert. Dieser Fall würde zum Beispiel ständig auftreten, wenn die Maximallänge der Viertelstundenliste auf Eins gesetzt wäre.

### 5.5.2 Außenposten

Der Außenposten bearbeitet die vom NOC erzeugten Daten, führt die Zuordnung von Subnetz/AS zu Anschluß durch und schickt das Ergebnis an die Zentrale.

Wie die Zentrale ist er als sequentieller Server angelegt und unterhält als zentrale Datenstrukturen je eine Liste mit Datenbasen und Viertelstundenaccountingdaten. Abweichend von der Konstruktion der Zentrale, wird ein AP nicht nur durch Nachrichten zu Aktionen veranlaßt, sondern auch durch ein bestimmtes Signal<sup>5</sup>. Die sich daraus ergebenden Komplikationen werden noch ausführlich behandelt.

---

<sup>5</sup>SIGUSR1

## Ablaufbeschreibung

Zunächst sollen aber die Aktionenfolgen, die aufgrund von eintreffenden Nachrichten zur Ausführung gelangen, behandelt werden. Abb. 5.9 enthält einen Überblick des funktionalen Modells eines Außenpostens.

Wie die Zentrale akzeptiert auch ein AP eine Nachricht vom Typ "neue DB". Im Unterschied zur Zentrale werden die in der Nachricht enthaltenen Listen ausgepackt und wie in Abschnitt 5.2.2 dargestellt in Hashtabellen umgewandelt. Die Anpassung der Größe der Hashtabellen erfolgt automatisch, und berechnet sich aus der Menge an zu speichernden Elementen zuzüglich eines definierbaren Überschusses. Um eine gute Streuung zu erreichen, wird von dem ermittelten Wert die nächste Primzahl genommen.

Da anhand der Hashtabellen für Subnetze bzw. Autonome Systeme nachgeschlagen werden soll, welchem Anschluß sie angehören, ist der Hashschlüssel bei einem AS die AS-Nummer, bei einem Subnetz die Summe aus Netzadresse und Netzmaskenlänge. Die Hashfunktion besteht aus der Modulofunktion mit der Größe der Tabelle als Basis.

Im Vorfeld der Implementierung wurde auch ein Test mit sortierten Listen und binärer Suche durchgeführt. Im Vergleich zu dem hashbasierten System war jedoch eine Geschwindigkeitseinbuße von ca. 10% zu verzeichnen<sup>6</sup>.

Die Reaktionen auf die Nachrichten "Hallo" und "Kill" sind genau wie bei der Zentrale, außer daß bei "Kill" der AP die Daten nicht auf Platte schreibt, sondern versucht, sie an die Zentrale zu schicken.

## Durch ein Signal hervorgerufene Aktionen

Nachdem alle Aktionen, die durch Nachrichten ausgelöst werden, beschrieben wurden, wird nun erläutert, was nach Eintreffen des Signals *SIGUSR1* passiert. Dieses Ereignis stößt den wichtigsten Arbeitsgang eines AP an, nämlich die Bearbeitung der Accountingrohdaten. Die spezielle Problematik, die durch die Berücksichtigung eines asynchronen Ereignisses gegeben ist, wird in Abschnitt 6.1 besprochen.

Wie im Abschnitt über die Eingabedatenschnittstelle dargestellt, wird das Signal von einem speziellen Prozeß, der das Abfragesystem des NOC von dem hier beschriebenen System entkoppelt, erzeugt, wenn er eine neue Datei für den AP erstellt hat. Die Datei enthält die Dateinamen mit den Accountingrohdaten der vom NOC abgefragten Router. Jede Zeile der Datei enthält einen Eintrag mit Dateinamen, IP-Adresse des Routers und dem Abfrageintervall.

Für jede dieser Zeilen führt der AP folgende Aktionen aus: Das Abfrageintervall muß zunächst in Viertelstunden aufgeteilt werden. Zwar sind die Abfrageintervalle im Augenblick meistens genau eine Viertelstunde lang, doch kann davon abgewichen werden, wenn zum Beispiel die Abfrage gestört worden ist. Der AP ist deshalb nicht auf ein festes Abfrageintervall angewiesen, sondern akzeptiert alle Längen.

---

<sup>6</sup>Auf eine detaillierte Darstellung des Tests wird hier verzichtet.



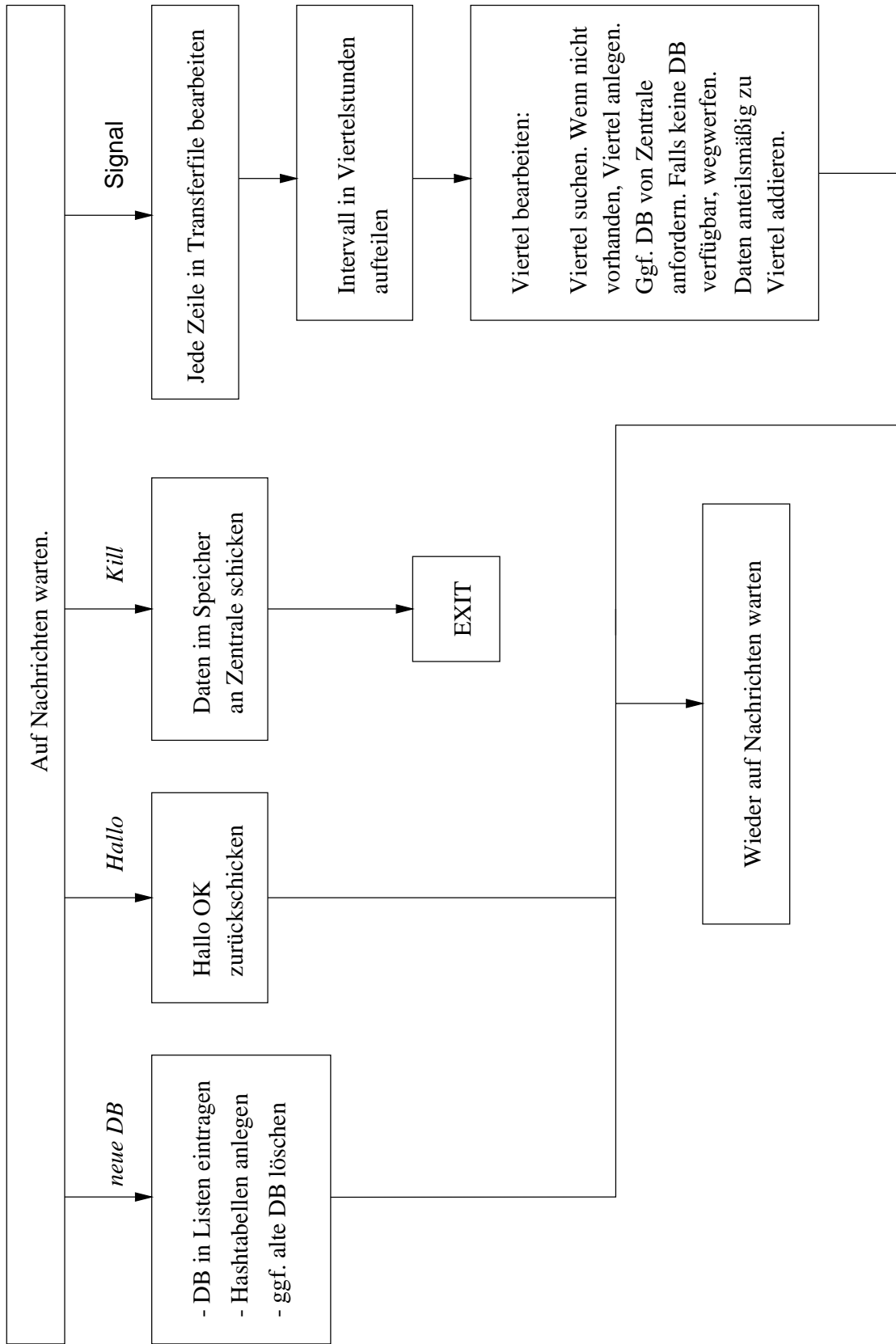


Abbildung 5.9: Funktionales Modell des Außenpostens

Je nachdem, wie lange das Intervall ausgefallen ist, steht jetzt eine mehr oder weniger lange Liste an Viertelstunden<sup>7</sup> zur Disposition, auf die die Accountingrohdaten verteilt werden müssen. Gemäß ihrem Anteil am Gesamtintervall werden nun die Accountingrohdaten den Viertelstunden zugeschrieben. Die Zählung der Viertelstunden beginnt immer mit der vollen Stunde. Wobei als Grundlage die Uhrzeit des Rechners, an dem der AP läuft, herangezogen wird. Die Bezugsuhrzeit, die in Accountingrohdaten angegeben ist, kann sich eventuell von der des Außenpostens unterscheiden, je nachdem wie genau die beteiligten Uhren synchronisiert sind.

Da sich aber der Unterschied im Bereich weniger Sekunden bewegt, wird an dieser Stelle auf spezielle Uhrensynchronisationsmaßnahmen verzichtet. Auch ist dies nicht ein spezielles Problem der hier beschriebenen Anwendung, sondern ein allgemeines in verteilten Plattformen. Es existieren daher Lösungen auf Systemebene.

Zur Veranschaulichung der Aufteilung des Gesamtintervalles in Viertelstunden diene ein Beispiel: Die Accountingrohdaten gelten für den Bereich von 14.00 - 14.35 Uhr. Der AP nimmt eine Einteilung in die drei Viertelstunden 14.00 - 14.15, 14.15 - 14.30, 14.30 - 14.35 vor. Der prozentuale Anteil an den Gesamtdaten der den beiden ersten Viertelstunden zugerechnet wird ist jeweils  $\frac{15}{35} = \frac{3}{7} \approx 43\%$ . Der Anteil für die letzte Viertelstunde ist  $\frac{5}{35} = \frac{1}{7} \approx 14\%$ .

Sollen nun 100 Bytes auf die drei Viertelstunden verteilt werden, erhalten die beiden ersten je 43 Bytes und das dritte 14 Bytes.

Bevor die Accountingdaten nun tatsächlich gemäß ihrem Anteil zu einer Viertelstunde addiert werden, muß festgestellt werden, ob die bewußte Viertelstunde sich in der Viertelstundenliste befindet. Ist dies nicht der Fall, wird sie angelegt. Wie bei der Zentrale ist es dabei notwendig, die Viertelstunde mit einer Datenbasis zu verbinden.

Sollte die Datenbasis nicht in der Datenbasenliste vorhanden sein, wird bei der Zentrale um die Datenbasis nachgesucht. Dieser Fall kann natürlich nur nach einer vorausgegangenen Fehlfunktion aufgetreten sein, weswegen dieser Vorfall mit einer Warnung kundgetan wird. Sollte die Zentrale nicht erreichbar sein, oder auch nicht über die gewünschte Datenbasis verfügen, müssen die Daten unberücksichtigt bleiben, was wiederum eine Warnung zur Folge hat.

Nachdem nun die Viertelstunde ausgemacht worden ist, werden die Accountingrohdaten anhand der zuständigen Datenbasis vorverarbeitet. Wie schon früher erwähnt besteht die Vorverarbeitung darin, ein Subnetz/AS einem Anschluß zuzuordnen, was durch eine Suche in der Subnetz- bzw. AS-Hashtabelle geschieht.

Ist die Zuordnung vollendet, werden die vorverarbeiteten Accountingdaten zu der Viertelstunde addiert. Dabei werden auch aus den Werten für übertragene Pakete und Bytes die übertragenen ATM-Zellen anhand der vertraglich festgelegten Formel<sup>8</sup> ermittelt. Die Berechnung der ATM-Zellen ist notwendig, da der Kernnetzdurchsatz nicht in Bytes, sondern in Zellen erfolgt. Genauso wird mit jeder Viertelstunde des aufgeteilten Accountingintervalls verfahren. Beziehen sich aufeinanderfolgende Viertelstunden auf dieselbe Datenbasis,

---

<sup>7</sup>Nicht zu verwechseln mit der sich im Hauptspeicher des AP befindlichen Viertelstundenliste der aktuellsten Viertelstundeneinträge!

<sup>8</sup>Die Formel kann nicht exakt sein, weil dazu sämtliche Paketlängen bekannt sein müßten. Die Herleitung einer geeigneten Berechnungsvorschrift findet sich zum Beispiel in [INT2].

braucht die Vorverarbeitung nur bei der ersten Viertelstunde durchgeführt werden. Für alle weiteren Viertelstunden mit derselben Datenbasis entfällt dieser Schritt.

Bevor nun die nächste Zeile in der durch den gesonderten Prozeß erzeugten Datei bearbeitet wird, testet der AP die Länge der Viertelstundenliste. Überschreitet sie einen Maximalwert, wird die älteste Viertelstunde an die Zentrale geschickt.

### 5.5.3 Statistik

Die Statistik führt die in Abb. 4.1 durch Pfeile dargestellten Arbeitsschritte durch. Ausgehend von dem Kästchen in der linken oberen Ecke, werden alle anderen durch Kästchen repräsentierte Dateien erzeugt. Das besagte Kästchen in der Ecke enthält eine Anschluß/Anschluß-Matrix in feinsten zeitlicher Auflösung und wird von der Zentrale erzeugt. Da Zentrale und Statistik zwei getrennte Komponenten sind, ist diese Datei als eine systeminterne Schnittstelle anzusehen. Da die feinste zeitliche Auflösung 15 Minuten beträgt, sollten für jeden Tag 96 Dateien von der Zentrale erzeugt worden sein.

#### Betriebsmodi

Die Weiterverarbeitung durch die Statistik kann auf drei verschiedene Weisen geschehen, wobei der jeweilige Betriebsmodus neben den dafür notwendigen Parametern der Statistik über Kommandozeilenargumente mitgeteilt wird. Es folgt eine Beschreibung der drei Betriebsmodi:

**Cronmodus:** Diese Betriebsart ist dafür vorgesehen, einmal täglich als Cron-Job aktiv zu werden und die Accountingdaten des vergangenen Tages auszuwerten. Für jede Anschluß/Anschluß-Datei wird die dazugehörige Standort/Standort-Datei erstellt und jeweils eine Datei für den Kernnetzdurchsatz und den Gesamtdurchsatz. Anschließend werden vier Dateien mit einer Zusammenfassung des Tages erzeugt. In Abb. 4.1 sind dies alle Pfeile außer den vier horizontalen Pfeilen zwischen den Spalten Tag und Monat. Der gesamte Vorgang wird ab jetzt mit *Tagesaccounting* bezeichnet.

**Tagesmodus:** Für die durch Kommandozeilenparameter angegebenen Tage wird das Tagesaccounting durchgeführt. Dies ist zum Beispiel sinnvoll, wenn bestimmte Tage nicht mit dem Cronmodus bearbeitet wurden.

**Monatsmodus:** Für das angegebene Monat wird das *Monatsaccounting* durchgeführt. Es besteht aus den vier Pfeilen zwischen Tag und Monat in Abb. 4.1.

#### Beschreibung der Arbeitsschritte

Der erste Teil des Tagesaccountings besteht darin, alle Dateien in Viertelstundenauflösung zu berechnen, dargestellt durch die vertikalen Pfeile in der Abbildung.

Um aus einer Anschluß/Anschluß-Matrix die Standort/Standort-Matrix zu gewinnen, wird zunächst die zuständige Statistikdatenbasis geladen. Sie geht, wie in Abschnitt 5.3.3 beschrieben, aus der ersten Zeile der Anschluß/Anschluß-Datei hervor. Sodann werden für

jedes Matrixelement die Identitäten der Anschlüsse auf die Identitäten der Standorte abgebildet. Die Abbildungsvorschrift ist in der *ANSCHLUESSE*-Sektion der Statistikdatenbasis enthalten. Das Ergebnis ist die neue Standort/Standort-Matrix.

Die Summe über alle Matrixelemente der Standort/Standort-Matrix ergeben den Gesamtdurchsatz. Zieht man davon die Elemente der Hauptdiagonale ab, erhält man den Kernnetzdurchsatz.

### Arbeitsschritte mit Vereinigter Datenbasis

Schwieriger gestaltet sich der zweite Teil des Tagesaccountings oder das Monatsaccounting. An und für sich handelt es sich um eine einfache Addition. In Abschnitt 5.2.3 wurde aber bereits darauf hingewiesen, daß dabei eine Vereinigte Datenbasis zu benutzen ist.

Um zwei Datenbasen zu verschmelzen, werden erst einmal alle kennzeichnenden Identitäten entfernt, dann die Elemente in *STANDORTE*- und *ANSCHLUESSE*-Sektion vereinigt<sup>9</sup> und zum Schluß wieder eindeutige Identitäten vergeben. Es erfolgt also eine Vereinigung der Elemente unter Neuvergabe der Identitäten. Damit ist gewährleistet, daß alle Elemente, die in einer der beiden beteiligten Datenbasen, vertreten waren, auch in der Vereinigten Datenbasis auftauchen und jedes Element der Vereinigten Datenbasis eine eindeutige Identität besitzt. Dieser Vorgang wird solange wiederholt, bis alle am Verschmelzungsprozeß beteiligten Datenbasen in die Vereinigte Datenbasen eingegangen sind.

Anhand der Vereinigten Datenbasis kann nun die Aggregation der Teile des kürzeren Accountingzeitraums zu dem längeren vorgenommen werden. Dabei muß jede Identität aus dem kürzeren Zeitraum auf die Identität der Vereinigten Datenbasis umgerechnet werden.

#### 5.5.4 Manager

Der Manager dient zum Starten und Beenden von der Zentrale und den Außenposten, der Zustandsabfrage und der Einspeisung von Datenbasen, wobei auch die Identitäten für Standorte und Anschlüsse festgelegt werden. Die Funktionen werden jeweils durch separate, ausführbare Programme realisiert. Der Aufruf der Programme mit passenden Parametern erfolgt aus einer in *Tcl/Tk* erstellten, graphischen Oberfläche.

Die Zustandsabfrage und -anzeige beschränkt sich momentan auf die Information, ob eine Komponente aktiv ist oder nicht. Für die Zukunft wäre hier denkbar, noch andere Parameter (gegenwärtig im Speicher befindliche Viertelstunden, für einen Zeitraum gültige Datenbasis, Art und Zeitpunkt der letzten ausgeführten Aktionen) abzufragen.

Die Statistik ist zur Zeit überhaupt nicht in den Manager integriert, sondern muß von Hand aufgerufen werden. Hier wäre es schön, wenn man aus dem Manager heraus, die von der Statistik erzeugten Daten verwalten, sichten und archivieren könnte.

Auch Datenbasen können vom Manager nur verschickt werden. Eine Erleichterung wäre es, wenn man sich einen Überblick über alle vorhandenen Datenbasen verschaffen und sich diese dann anzeigen lassen könnte.

---

<sup>9</sup>“vereinigt” ist im mengentheoretischen Sinn zu verstehen. Algorithmisch wird die Vereinigung dadurch vollzogen, daß die Elemente beider Datenbasen alphabetisch sortiert und anschließend unifiziert werden.

# Kapitel 6

## Implementierungsdetails

Diese Kapitel stellt gewisse Aspekte der Implementierung in den Brennpunkt, die im vorangegangenen Kapitel nur sehr kurz oder gar nicht behandelt wurden, um nicht vom Wesentlichen abzulenken.

### 6.1 Problematik der Signalbehandlung am Außenposten

Wie in der Beschreibung des Außenpostens in Abschnitt 5.5.2 dargestellt, wird der AP über ein Signal informiert, daß neue Accountingrohdaten bereitstehen. Das Eintreffen des Signals ist als asynchrones Ereignis zu betrachten und stört den normalen Programmablauf.

Zwar suggeriert Abbildung 5.9, daß die Wirkungsweise des Signals genau der einer Nachricht entspricht. Und in der Tat ist dieser Sachverhalt bei den meisten in der Realität beobachtbaren Abläufen gegeben, da zwischen zwei Signalen genügend Zeit bleibt, die anfallende Arbeit zu erledigen. Damit wird sich der AP immer im Wartezustand befinden, wenn das Signal eintrifft.

Was ist aber, wenn der AP gerade eine neue Datenbasis empfängt und das Signal eintrifft? Oder, wenn zwei Signale zu dicht aufeinanderfolgen? Mit einem Wort, wenn sich der AP nicht im Wartezustand befindet? Die Wahrscheinlichkeit dafür ist zwar gering, aber dennoch im Bereich des Möglichen.

Deswegen wird bei Zustellung des Signals nur ein Zähler erhöht und an der gestörten Stelle mit der Ausführung fortgefahren, wodurch keine kritischen Operationen wie dynamische Speicheranforderungen beeinträchtigt werden. Falls sich der AP im Wartezustand befunden hat, wird unmittelbar festgestellt, daß der blockierende Betriebssystemaufruf, der zum Wartezustand führt, unterbrochen worden ist, worauf die eigentliche Signalbehandlung in Form der Behandlung der neuen Accountingrohdaten angestoßen wird.

War der AP gerade an anderer Stelle tätig, ist dies durch eine Nachricht veranlaßt worden. Der AP wird irgendwann mit der Bearbeitung der Nachricht fertig werden und dann die Signalbehandlung durchführen. In diesem Fall könnte aber auch ein Systemaufruf unterbrochen worden sein, wie zum Beispiel der TCP-Verbindungsaufbau oder das Lesen und

Schreiben vom TCP-Stream. Ein automatisches Wiederaufsetzen aller Systemaufrufe verbietet sich, weil das Signal dann nur nach Eintreffen einer Nachricht berücksichtigt würde. Entweder trifft man für jeden Systemaufruf von Hand Vorkehrungen für ein Wiederaufsetzen oder man blockiert die Zustellung des Signals an den Prozeß während der Abarbeitung einer Nachricht.

Eine Implementierung einer der beiden letztgenannten Auflösungsöglichkeiten wurde aus Zeitmangel zugunsten anderer Tätigkeiten zurückgestellt. Die Verwendung von Signalen war in der Konzeption des Programmsystems ohnehin nicht vorgesehen und wird nur kurzfristig als Übergangslösung eingesetzt. Ursprünglich war geplant, die Funktionalität des Prozesses, der das Signal erzeugt, im Außenposten anzusiedeln. Wie in der Beschreibung der Eingabedatenschnittstelle schon dargelegt, war diese Schnittstelle bis kurz vor Ende der Bearbeitungszeit ständigem Wandel unterworfen. Deshalb wurde zur Entkoppelung besagter Prozeß unter Gebrauch eines Signals zur Interprozeßkommunikation eingesetzt. Er ist mit der Skriptsprache *Perl* implementiert, was schnelle Anpassungen erlaubt.

Für die Zukunft ist vorgesehen, den Prozeß in *C* zu implementieren und statt eines Signals, das systemeigene Nachrichtensystem zu verwenden, womit eine saubere Lösung der Probleme, die sich durch den Einsatz des Signals ergeben, erreicht wird.

## 6.2 Zuordnung von IP-Adresse zu Subnetz/AS

Wie im vorigen Abschnitt bewegt sich auch in diesem die Thematik im Umfeld der Eingabedatenschnittstelle. Wie in den Abschnitten 4.5.2 und 5.3.1 schon erwähnt, verarbeitet das NOC die Accountingdaten der Router vor, indem es Quelle und Ziel, die die Router als IP-Adresse angeben, in Subnetze bzw. Autonome Systeme umwandelt.

Ganz zu Beginn der Arbeit ist davon ausgegangen worden, daß der Umfang der Arbeit auch die Abfrage der Accountingdaten der Router einschließt. Für diese Aufgabenstellung ist eine etwas andere Konstruktion erarbeitet worden, bei der die Außenposten in etwa die Funktionalität besessen hätten, die jetzt das NOC-System und der Prozeß an der Eingabedatenschnittstelle haben. Auch die Zuordnung von Subnetz/AS zu Anschluß hätte noch der damalige AP durchgeführt. Er hätte jedoch keine Viertelstundenliste geführt. Die Aufteilung in Viertelstunden wäre an der Zentrale vorgenommen worden. Die Implementierung befand sich auch schon in einem fortgeschrittenen Stadium, als das NOC den Teil der Aufgabe übernahm, den es jetzt erledigt.

Interessant am ehemaligen Konzept war die effiziente Umwandlung von IP-Adressen in Subnetze/AS. In Abschnitt 2.1.2 wurde gefordert, daß der Vergleich einer Adresse mit den Routingeinträgen in CIDR-Notation zum Zwecke der Leitwegebestimmung<sup>1</sup> in einer bestimmten Reihenfolge, die die Länge der Netzmaske berücksichtigt, erfolgen muß.

Die Leistung, die der Router dabei erbringen muß, ist äquivalent zu der, die für die Umwandlung einer IP-Adresse in ein Subnetz oder AS erforderlich ist, denn bei beiden Vorgängen wird dieselbe Identifikation vorgenommen.

---

<sup>1</sup>Deutsch für Routing

Der einfachste Lösungsalgorithmus ist die sequentielle Suche in einer Liste mit den Netzeinträgen in CIDR-Notation, sortiert nach Netzmaskenlänge. Ein wesentliches effizienteres Verfahren ist möglich, wenn man annimmt, daß die Netzmaskenlänge eine bestimmte Länge nicht unterschreitet.

Für die damalige Implementierung wurde dieser Wert auf acht festgesetzt. Damit weiß man, daß die ersten acht Bits einer zu identifizierenden IP-Adresse auf jeden Fall zum Netzanteil gehören. Aufgrund dessen ist es nun möglich, die Netzadressen in CIDR-Notation in einer kombinierten Hashtabelle/Liste zu repräsentieren. Die ersten acht Bits sind der Schlüssel zu einem Hascheintrag. Der Hascheintrag besteht wieder aus einer verketteten Liste in entsprechender Sortierung. Der Vorteil besteht in einer verkürzten Listenlänge, der Mehraufwand für die Hashsuche fällt nicht ins Gewicht.

### 6.3 Datentyp der Zähler

Als Zähler für Pakete, Bytes und ATM-Zellen wird ein 64 Bit breiter, vorzeichenloser, ganzzahliger Datentyp verwendet. Die größte darstellbare Zahl ist damit  $2^{64} - 1 = 18446744073709551615 \approx 1,8 \cdot 10^{19}$ . Wie aus Abbildung 3.1 ersichtlich ist, bewegen sich die Zähler im Bereich  $10^{12}$ , was durch den gewählten Typ abgedeckt ist.

Ein nur 32 Bit breiter Datentyp ist nicht ausreichend, da  $2^{32} - 1 = 4294967295 \approx 4,3 \cdot 10^9$  zu gering ausfällt.

### 6.4 Rechengenauigkeit

Bei der Beschreibung des Außenpostens in Kapitel 5 wurde erklärt, daß Daten “gemäß ihrem Anteil zu einer Viertelstunde addiert werden”. Das Wort “Anteil” wirkt sich in einer ganzzahligen Division aus, bei der Bruchteile abgeschnitten werden. In der Praxis schlägt sich dies in einer gewissen Ungenauigkeit nieder.

Es folgt ein Rechenbeispiel zur Illustration: Drei Pakete mit zusammen 100 Bytes sind zwischen 8:55 Uhr und 9:50 Uhr eingetroffen. Da eine Viertelstunde momentan 900 Sekunden lang ist, werden die Daten auf fünf Viertelstunden aufgeteilt. Das Gesamtintervall beträgt  $55 \cdot 60 = 3300$ . In Tabelle 6.1 sind die Anteile, die jeder Viertelstunde zugeschlagen werden, zusammengefaßt.

Durch die Division sind also ein Byte und alle drei Pakete verloren gegangen. Eine bessere Vorschrift für die Berechnung der Zuschläge ergibt sich, wenn man statt des alleinigen Anteils des in eine Viertelstunde fallenden Intervalles am Gesamtintervall, alle Vorgängeranteile hinzunimmt und von dem dann berechneten Zuschlag, die Vorgängerzuschläge abzieht. Man stellt also sicher, daß bei jedem Schritt, die Zahl der zugewiesenen Pakete oder Bytes, in der Summe, dem bis jetzt vergebenen Anteil an der Gesamtheit entsprechen. Tabelle 6.2 zeigt die Auswirkungen auf das Beispiel.

Wie zu erwarten stimmen am Schluß die Zuschläge in der Summe. Man könnte noch anmerken, daß das Paket, das der fünften Viertelstunde zugeschlagen worden ist, vielleicht besser der zweiten, längeren Viertelstunde gegeben hätte werden sollen.

Viert.	Uhrzeit	Anteil	Zuschlag Pakete	Zuschlag Bytes
1	8:55 - 9:00	$\frac{300}{3300}$	$\frac{300 \cdot 3}{3300} = 0$	$\frac{300 \cdot 100}{3300} = 9$
2	9:00 - 9:15	$\frac{900}{3300}$	$\frac{900 \cdot 3}{3300} = 0$	$\frac{900 \cdot 100}{3300} = 27$
3	9:15 - 9:30	$\frac{900}{3300}$	$\frac{900 \cdot 3}{3300} = 0$	$\frac{900 \cdot 100}{3300} = 27$
4	9:30 - 9:45	$\frac{900}{3300}$	$\frac{900 \cdot 3}{3300} = 0$	$\frac{900 \cdot 100}{3300} = 27$
5	9:45 - 9:50	$\frac{300}{3300}$	$\frac{300 \cdot 3}{3300} = 0$	$\frac{300 \cdot 100}{3300} = 9$
		$\sum \frac{3300}{3300}$	$\sum 0$	$\sum 99$

Tabelle 6.1: Aufteilung auf Viertelstunden

Viert.	Uhrzeit	Gesamtanteil	Gesamt/Zuschlag Pakete	Gesamt/Zuschlag Bytes
1	8:55 - 9:00	$\frac{300}{3300}$	$\frac{300 \cdot 3}{3300} = 0/0$	$\frac{300 \cdot 100}{3300} = 9/9$
2	9:00 - 9:15	$\frac{1200}{3300}$	$\frac{1200 \cdot 3}{3300} = 1/1$	$\frac{1200 \cdot 100}{3300} = 3627$
3	9:15 - 9:30	$\frac{2100}{3300}$	$\frac{2100 \cdot 3}{3300} = 1/0$	$\frac{2100 \cdot 100}{3300} = 63/27$
4	9:30 - 9:45	$\frac{3000}{3300}$	$\frac{3000 \cdot 3}{3300} = 2/1$	$\frac{3000 \cdot 100}{3300} = 90/27$
5	9:45 - 9:50	$\frac{3300}{3300}$	$\frac{3300 \cdot 3}{3300} = 3/1$	$\frac{3300 \cdot 100}{3300} = 100/10$
		$\sum \frac{3300}{3300}$	$\sum 3$	$\sum 100$

Tabelle 6.2: Alternative Aufteilung auf Viertelstunden

Im Programmsystem wird bis jetzt die erste, ungenauere Berechnungsvorschrift verwendet. Die aufgezeigten Fehler treten nur bei kleinen Zahlen auf und sind dann dementsprechend auch gering. Bei Grössenordnungen, die sich in den Billionenbereich erstrecken, können einige zehn Pakete vernachlässigt werden, ohne die Genauigkeit merklich zu beeinflussen. Dafür gestaltet sich die Berechnung einfacher und schneller. Doch für zukünftige Erweiterungen, kann auch das komplexere Verfahren herangezogen werden, oder man stellt von ganzzahligen Zählern auf Fließkommagrößen um, was aber auch zu Problemen bei der Rechengenauigkeit führen kann, vor allem, wenn Operanden stark unterschiedlicher Größe beteiligt sind.

Eine letzte Möglichkeit ist, am Außenposten — also an der Stelle, an der die Daten das System betreten und noch am verstreutesten vorliegen — erst einmal alle Größen mit einer Zehnerpotenz zu multiplizieren, und an der Statistik — also dann, wenn die Daten das System verlassen und am verdichtesten sind — durch dieselbe Zehnerpotenz zu teilen. Dadurch wird der ganzzahlige Wertebereich besser ausgereizt und die Genauigkeit erhöht.

Eine Bewertung, welcher der vorgeschlagenen Erweiterungen die genaueste bei vertretbaren Kosten ist, sprengt den Rahmen dieser Arbeit.



## 6.5 Maßnahmen zur Fehlertoleranz

Laut [Wald95, Kapitel 8] ist Fehlertoleranz die *“Die Eigenschaft eines Systems trotz Fehler das korrekte (erwartete, normale, rechtzeitige) Verhalten zu zeigen, z. B. die geforderte Dienstleistung zu erbringen”*. Gerade in einem verteilten System, bei dem mehrere Instanzen kooperativ zur Erfüllung einer Aufgabe beitragen, ist dies von besonderer Bedeutung. Von Zeit zu Zeit müssen nämlich die Partner in Kontakt treten, um im Zuge der Lösung Informationen auszutauschen. Daß für jeden Zustand einer Instanz beim Eintreffen jeder Nachricht eine gewünschte Aktion definiert ist, ist eine Frage der Vollständigkeit und Korrektheit der Protokollspezifikation. Treten aber Fehler bei der Übertragung auf, ist ein Partner nicht erreichbar oder ausgefallen, sind fehlertolerante Maßnahmen gefragt. Außerdem könnte eine Instanz aufgrund interner Fehler mit von der Partnerinstanz abweichenden Daten aufwarten.

Vollständigkeit und Korrektheit des Protokolls, das in dem in dieser Arbeit erstellten System verwendet wird, wird am Schluß des Kapitels nachgewiesen.

Welche Maßnahmen zur Fehlertoleranz im erstellten System getroffen worden sind, erläutern folgende Unterabschnitte. Da die Maßnahmen in konkreten Fehlersituationen schon weitgehend in der Beschreibung des Programmsystems dargelegt wurden, werden hier Überlegungen grundsätzlicher Natur, Ergänzungen und Erweiterungsvorschläge aufgeführt.

### 6.5.1 TCP als Kommunikationsmittel

In 5.4 wurde schon darauf hingewiesen, daß TCP zur Kommunikation benutzt wird. TCP beinhaltet *Timeout* und *Retransmit* - Mechanismen, so daß temporäre Störungen des Verbindungsweges an der Anwendung transparent sind. Das unzuverlässige, verbindungslose *UDP* garantiert bei der Übertragung gar nichts, ist daher ungeeignet.

Um auch zeitlich größeren Störungen zu begegnen, wird beim Versenden wichtiger Nachrichten (Datenbasis Transfer, Datenbasisanforderung, Schicke Daten) mehrmals versucht, den Partner zu erreichen. Die Versuche werden momentan direkt hintereinander getätigt. Es böte sich an, zwischen den Versuchen mit steigender Versuchszahl, eine wachsende Verzögerung vorzusehen. Soll sich aber der Prozeß, der sich in einer Verzögerung befindet nicht blockieren, sondern für andere Aufgaben bereitstehen, ist eine Art Scheduler nötig. Dies erhöhte die Komplexität und war in der Bearbeitungszeit nicht zu realisieren.

### 6.5.2 Ausfall der Zentrale

Sollte die Zentrale über das realisierte Toleranzintervall hinaus von einem AP nicht erreichbar sein, führt dies zu Verlust von Accountingdaten. Als Alternative wäre denkbar, daß der AP in einen auf sich gestellten Betrieb umschaltet, indem er statt die Accountingdaten abzuschicken, sie lokal auf Platte abspeichert und von Zeit zu Zeit überprüft, ob die Zentrale wieder erreichbar ist. Sobald dies der Fall ist, liest er die abgespeicherten Daten wieder ein und fällt in den Normalbetrieb zurück.

Problematisch dabei ist, daß der AP eventuell mit einer falschen Datenbasis gearbeitet hat, was zu einem Konflikt mit der Zentrale führt, wenn er dort die Daten abliefern will. Auf die Behebung dieses Konflikts ist schon bei der Beschreibung der Zentrale hingewiesen worden.

### 6.5.3 Unterstützung der Fehlerdiagnose

Bei der Behandlung von Fehlersituation in Kapitel 5 ist öfters geschrieben worden, daß das System eine Warnung ausgibt. Dazu wird der *Log-Mechanismus* des Systems benutzt. Der Log-Mechanismus erlaubt Außenposten oder Zentrale, Informationen über die durchgeführten Arbeitsschritte mittels einer Datei dem Menschen mitzuteilen.

Von diesem kann die Datei dann zum Zwecke der Fehlerdiagnose gebraucht werden. Die Einträge der Datei bestehen aus der Uhrzeit der Meldung, dem Namen und Host des Prozesses, einer kurzen Erklärung des Vorfalls (eventuell mit Parametern versehen) und einer Kennzeichnung der Art des Eintrags. Vier verschiedene Arten sind möglich.

**Routinemeldung:** Sie erlauben, den Verlauf der Verarbeitung nachzuvollziehen.

**Warnung:** Ein Situation, die nicht hätte vorkommen sollen, ist eingetreten.

**Systemaufrufwarnung:** Ein Betriebssystemaufruf ist fehlgeschlagen. Das Programm kann aber weiterarbeiten.

**Fataler Systemaufruffehler:** Ein Betriebssystemaufruf ist fehlgeschlagen. Das Programm kann nicht weiterarbeiten und terminiert.

Bei den letzten zwei Arten wird auch noch die Fehlerbeschreibung, die das Betriebssystem gibt, mit angegeben. Als Vorschlag für zukünftige Erweiterungen, könnte man in bestimmten Fällen automatisch eine Mail an einen Menschen verschicken, um auf besonders kritische Fälle, die ein sofortiges Eingreifen erforderlich machen, hinzuweisen.

## 6.6 Protokollverifikation

Außenposten und Zentrale können sich in protokollarischer Sicht nur in einem Zustand befinden, dem Warten auf Nachrichten. Wird eine Nachricht bearbeitet, werden weitere Nachrichten in eine Warteschlange eingereiht und von der Instanz erst entgegengenommen, wenn die mit der aktuellen Nachricht verbundenen Aktionen beendet sind. Ob diese Aktionen sinnvoll für die Erfüllung der Gesamtaufgabe sind, ist in Hinsicht auf die Korrektheit des Protokolls unerheblich, denn die Aktionen bewirken keinen Wechsel in einen anderen Zustand.

Da zum Austauschen von Nachrichten aber eine Verbindung aufgebaut werden muß, ergibt sich ein Problem. Und zwar dann, wenn die Zentrale eine Nachricht an einen bestimmten AP sendet, der seinerseits eine Nachricht an die Zentrale sendet. Wie aus Abb. 5.4 ersichtlich ist, verschickt die Zentrale nur zwei Nachrichten (“Datenbasis Transfer” und “Keine Datenbasis verfügbar”). Die Situation tritt also auf, wenn die Zentrale vom Manager eine neue Datenbasis erhalten hat und sie die Datenbasis an einen Außenposten weiterleitet, der gerade eine Nachricht an die Zentrale schickt.

Nach den eingangs gemachten Aussagen sollte dies kein Problem darstellen, da beide Nachrichten in eine Warteschlange eingeordnet werden. Weil aber sowohl Zentrale als auch Außenposten darauf warten, daß ihre Nachricht vom jeweils anderen Partner entgegengenommen wird, kann keiner in den Zustand zurückkehren, in dem dies möglich ist.

Um dies formal erfassen zu können, genügt also ein Zustand nicht. Abb. 6.1 gibt die Protokollautomaten für Zentrale und AP an.

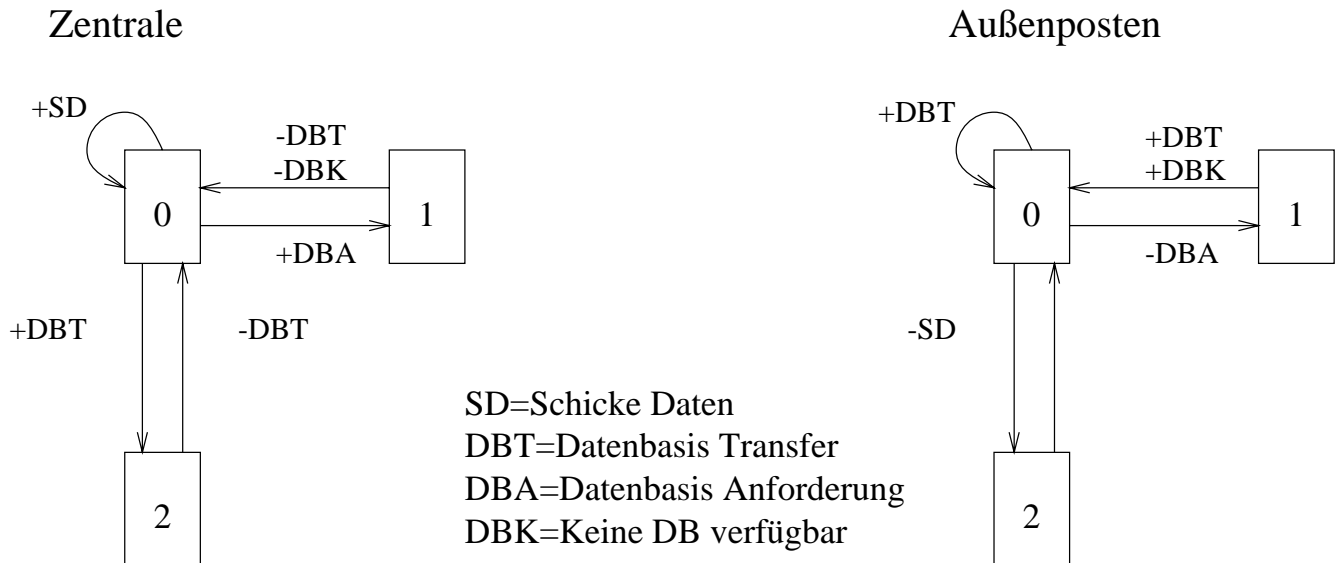


Abbildung 6.1: Protokollautomaten für Zentrale und AP. Ein “+” bedeutet, daß eine Nachricht gelesen wird. Ein “-”, daß eine Nachricht gesendet wird.

Der Manager wird nicht in die Betrachtung mit einbezogen, da hier nur das Protokoll zwischen Zentrale und AP betrachtet werden soll. Er tritt aber dennoch in Erscheinung indem bei der Zentrale die Nachricht “+DBT” behandelt wird, die nur vom Manager stammen kann. Zentrale und AP umfassen jeweils drei Zustände. Das Empfangen oder Versenden von Nachrichten bewirkt die dargestellten Zustandswechsel. Ähnliche Darstellungen finden sich z.B. in [Tane92].

Abb. 6.2 enthält den erweiterten Produktautomaten unter Berücksichtigung des Vollduplex-Kanals.

Ohne die gestrichelt eingezeichneten Zustandsübergänge, wären die beiden grau hinterlegten Zustände Deadlocksituationen. Sie kennzeichnen den Fall, daß die Zentrale eine Datenbasis verschicken möchte, während der AP Daten verschickt oder eine Datenbasis anfordert.

Die Verklemmung wird jedoch durch TCP-Timeoutmechanismen aufgelöst. Dies wird durch die gestrichelten Pfeile dargestellt. Damit ist die Verklemmungsfreiheit des Protokolls nachgewiesen. Ein Vergleich der mit “0” bezeichneten Zustände in Abb. 6.1 mit der Kommunikationsstruktur in Abb. 5.4 liefert die Vollständigkeit des Protokolls, da nur von diesem Zustand aus Nachrichten der Warteschlange gelesen werden.

Eine Möglichkeit, die durch Timeout aufgelöste Verklemmung gänzlich zu vermeiden, wäre zum Beispiel für das Versenden der Datenbasis seitens der Zentrale einen eigenen Prozeß von der Zentrale abzuspalten und ihn die Aktion ausführen zu lassen.

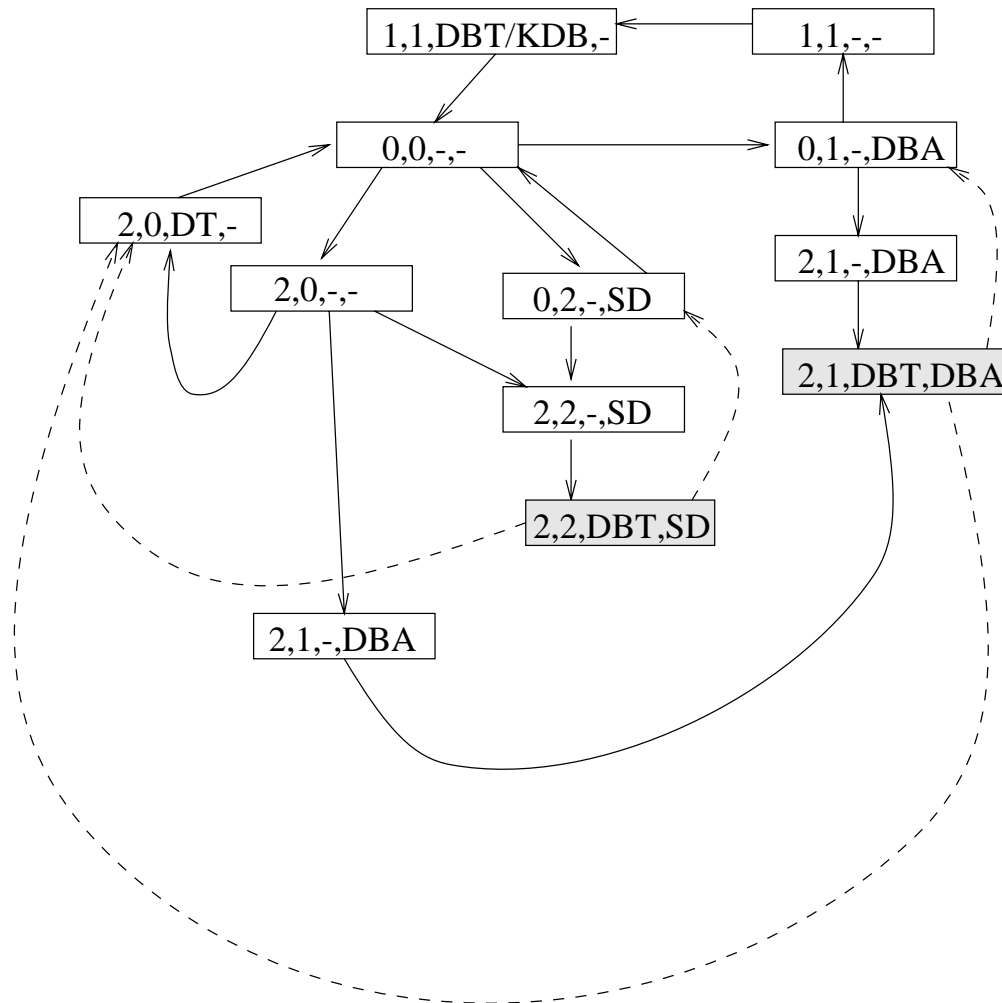


Abbildung 6.2: Erweiterter Produktautomat. Die Positionen im Quadrupel bezeichnen von links nach rechts den Zustand der Zentrale, des Außenpostens, des Kanals von Zentrale nach AP, des Kanals von AP nach Zentrale. Ein “-” kennzeichnet einen leeren Kanal. Gestrichelte Zustandsübergänge werden durch Timeouts bewirkt.

# Kapitel 7

## Analyse der Daten

In den beiden letzten Kapiteln wurde das System zur Gewinnung von Verkehrsflußdaten im B-WiN beschrieben. Da glücklicherweise bis auf einen Standort alle Standorte Accountingdaten liefern, können mit dem System erzeugte, realistische Daten hier präsentiert und analysiert werden. Es wird gezeigt, wie man die Analyse zum Zwecke der Optimierung des Routings einsetzen kann. Allerdings sind die Daten mit Vorsicht zu genießen, da die zugrundeliegende Datenbasis noch nicht in einer befriedigenden Form vorliegt.

Ausgangspunkt der Untersuchungen in diesem Kapitel ist eine Standort/Standort-Matrix, die in der Woche vom 12.11.96 bis zum 18.11.96 mit dem System erstellt worden ist. Sie ist in Tabelle 7.1 zu finden, bei der die ursprünglichen Identitäten der Standorte durch deren Klartextnamen ersetzt wurde.

GByte	Ber	Ham	Han	Kar	Köl	Lei	Mün	Nür	Stu	Wie	Non	Unk	B-W	X.25	$\Sigma$
Berlin	3	8	9	3	6	30	4	3	6	0	52	36	28	37	225
Hamburg	38	16	5	2	7	1	14	0	1	0	18	5	9	6	122
Hannover	7	41	52	7	21	13	10	7	8	0	43	13	29	33	284
Karlsruhe	6	2	6	65	9	7	5	5	18	0	45	14	23	18	223
Köln	6	7	15	5	57	7	6	5	8	0	70	14	46	32	278
Leipzig	8	3	5	4	5	41	2	6	3	0	39	11	22	25	174
München	3	4	7	6	4	3	13	14	7	0	32	11	13	6	123
Nürnberg	4	1	2	3	4	12	12	81	4	0	23	7	21	8	182
Stuttgart	9	5	48	120	61	7	16	7	36	0	50	13	31	16	419
Wiesbaden	4	2	5	9	4	4	7	10	4	0	0	0	0	2	51
Non-WiN	60	37	56	68	108	61	44	43	100	0	0	0	0	30	607
Unknown	18	8	18	20	30	19	13	13	21	0	0	226	0	9	395
B-WiN	21	8	21	16	33	19	15	19	16	0	0	0	8	18	194
X.25-WiN	17	8	18	11	26	17	12	12	21	0	20	6	5	7	180
Summe	204	150	267	339	375	241	173	225	253	0	392	356	235	247	3457

Tabelle 7.1: Standort/Standort Matrix vom 12.11.96 bis zum 18.11.96. Für einen konkreten Eintrag in der Tabelle ist am Anfang der Zeile die Quelle angegeben und an oberster Stelle der Spalte das Ziel. Von Berlin nach Hamburg sind also 8 GByte übertragen worden.

Die letzten vier Standorte sind virtuell. “X.25-WiN” steht für das sich immer noch im Betrieb befindliche X.25-WiN. “B-WiN” bezeichnet im WiN geroutete Autonome Systeme, die nicht weiter aufgelöst werden. “Unkown” wird Adressen zugewiesen, die nicht erkannt werden konnten. “Non-Win” steht für Bereiche außerhalb des WiN. Der Standort “Wiesbaden” ist ein richtiger Standort, liefert aber noch keine Daten.

Etwas aufschlußreicher ist Tabelle 7.2, die statt der absoluten Werte, deren Anteil an der Gesamtsumme angibt.

Prozent	Ber	Ham	Han	Kar	Köl	Lei	Mün	Nür	Stu	Wie	Non	Unk	B-W	X.25	$\Sigma$
Berlin	0,1	0,2	0,3	0,1	0,2	0,9	0,1	0,1	0,2	0,0	1,5	1,0	0,8	1,1	6,5
Hamburg	1,1	0,5	0,1	0,1	0,2	0,0	0,4	0,0	0,0	0,0	0,5	0,1	0,3	0,2	3,5
Hannover	0,2	1,2	1,5	0,2	0,6	0,4	0,3	0,2	0,2	0,0	1,2	0,4	0,8	1,0	8,2
Karlsruhe	0,2	0,1	0,2	1,9	0,3	0,2	0,1	0,1	0,5	0,0	1,3	0,4	0,7	0,5	6,5
Köln	0,2	0,2	0,4	0,1	1,6	0,2	0,2	0,1	0,2	0,0	2,0	0,4	1,3	0,9	8,0
Leipzig	0,2	0,1	0,1	0,1	0,1	1,2	0,1	0,2	0,1	0,0	1,1	0,3	0,6	0,7	5,0
München	0,1	0,1	0,2	0,2	0,1	0,1	0,4	0,4	0,2	0,0	0,9	0,3	0,4	0,2	3,6
Nürnberg	0,1	0,0	0,1	0,1	0,1	0,3	0,3	2,3	0,1	0,0	0,7	0,2	0,6	0,2	5,3
Stuttgart	0,3	0,1	1,4	3,5	1,8	0,2	0,5	0,2	1,0	0,0	1,4	0,4	0,9	0,5	12,1
Wiesbaden	0,1	0,1	0,1	0,3	0,1	0,1	0,2	0,3	0,1	0,0	0,0	0,0	0,0	0,1	1,5
Non-WiN	1,7	1,1	1,6	2,0	3,1	1,8	1,3	1,2	2,9	0,0	0,0	0,0	0,0	0,9	17,6
Unknown	0,5	0,2	0,5	0,6	0,9	0,5	0,4	0,4	0,6	0,0	0,0	6,5	0,0	0,3	11,4
B-WiN	0,6	0,2	0,6	0,5	1,0	0,5	0,4	0,5	0,5	0,0	0,0	0,0	0,2	0,5	5,6
X.25-WiN	0,5	0,2	0,5	0,3	0,8	0,5	0,3	0,3	0,6	0,0	0,6	0,2	0,1	0,2	5,2
Summe	5,9	4,3	7,7	9,8	10,8	7,0	5,0	6,5	7,3	0,0	11,3	10,3	6,8	7,1	100,0

Tabelle 7.2: Standortmatrix (relativ)

Für den menschlichen Betrachter ist das immer noch eine Zahlenwüste. Deshalb wird in Abbildung 7.1 der prozentuale Anteil als analoge Größe und zwar als Durchmesser eines Kreises dargestellt. Eine andere Möglichkeit wäre, mit Farbwerten zu arbeiten. Sehr schön zu sehen ist jetzt der starke Verkehr in der Hauptdiagonale, der den regionalen Verkehr darstellt.

Auffällig ist der starke Verkehr zwischen Stuttgart und Karlsruhe, zumal in der Gegenrichtung nicht soviel Verkehr stattfindet. Um dieses Phänomen zu klären, könnte man jetzt in den Anschluß/Anschluß Matrizen nach den Anschlüssen in Stuttgart und Karlsruhe suchen und überprüfen, ob bestimmte Anschlüsse dafür verantwortlich sind.

Visualisierung stellt also ein wichtiges Hilfsmittel der Analyse dar. Die beiden Tabellen und die Grafik sind von Hand erstellt worden. Dieser Prozeß ließe sich auch automatisieren.

Für eine Verbesserung des Routings ist aber die Kenntnis der Segmentbelastung wichtiger als reine Verkehrsflußmatrizen. Die Segmentbelastung kann nach dem in Abschnitt 2.3.2 gezeigten Verfahren geschehen. Die dazu notwendige Verkehrsverteilung ist in Tabelle 7.2 zu finden. Kennt man auch die verwendete Routingstrategie, steht der Anwendung des Verfahrens nichts mehr im Wege.

Allerdings ist die Komplexität im B-WiN wegen der Vermaschungen etwas höher, sodaß sich die Berechnung schwieriger gestaltet. Eventuell lohnt es sich, spezielle Software zu

entwickeln<sup>1</sup>, die als Eingabeparameter die Topologie des Netzes, Verkehrsverteilung und die Routingregeln erhält und die Segmentbelastung ausgibt. Damit wäre es auch möglich, die Wirkung einer anderen Routingstrategie zunächst zu simulieren, bevor sie in der Praxis angewandt wird.

Egal ob die Segmentbelastung von Hand oder maschinell erstellt worden ist, erlaubt sie es, Engpässe zu erkennen. Nun ermittelt man die Verkehrsströme, die zur Belastung des Segments beitragen und überprüft, ob man nicht den einen oder anderen auf ein weniger belastetes umleiten kann.

---

<sup>1</sup>Falls nicht schon Werkzeuge existieren, die das Gewünschte leisten.

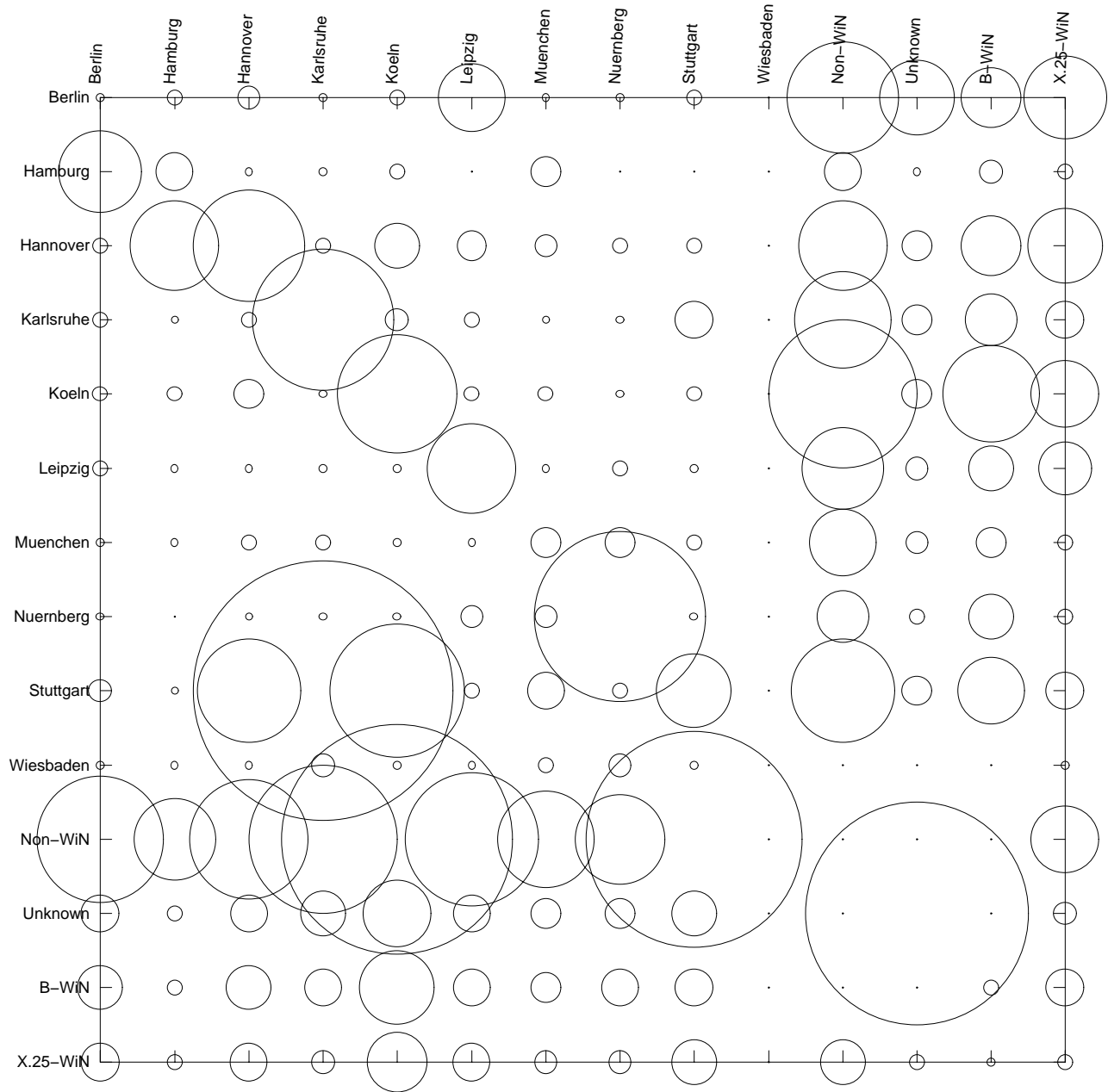


Abbildung 7.1: Analoge Darstellung des prozentualen Anteils. Der Kreis von “Unknown/Unknown” ist auf die Hälfte verkleinert.



# Kapitel 8

## Zusammenfassung

Die Kenntnis von Umfang und Richtung der Verkehrsströme in einem Datenkommunikationsnetz ist von fundamentaler Bedeutung. Sie ist die Voraussetzung für Ausbauplanung, Optimierung des Routings und Abrechnung zwischen Betreiber und Kunde.

In dieser Arbeit wurde ein Programmsystem realisiert, das die Verkehrsflüsse im B-WiN ermittelt. Auf Erweiterungsmöglichkeiten ist an verschiedenen Stellen der Arbeit hingewiesen worden. Jedoch ist die gewünschte Funktionalität vollständig implementiert, sodaß für jede Erweiterung zu prüfen ist, ob sich der Aufwand dafür lohnt.

Zum Zeitpunkt, da diese Zeilen geschrieben werden, befindet sich das System seit zwei Wochen in stabilem Betrieb. Die hinsichtlich der Zuverlässigkeit und Fehlertoleranz getroffenen Maßnahmen scheinen ausreichend zu sein. Lediglich die Umstellung von der Benutzung von Signalen auf das Nachrichtensystem zur Kommunikation zwischen dem Schnittstellenprozeß an der Eingabedatenschnittstelle und dem Außenposten ist ratsam. Wie in Abschnitt 6.1 dargelegt, handelt es sich dabei nur um eine Übergangslösung, die nicht allen denkbaren, wenn auch unwahrscheinlichen Situationen gewachsen ist.

Eine Ausbaumöglichkeit, die bis jetzt nur am Rande in Abschnitt 5.4.1 erwähnt worden ist, besteht in der Verwendung von Verschlüsselungs- und Paßwortmechanismen im Nachrichtensystem. Eine Verschlüsselung der transportierten Information ist nicht wichtig, weil keine Klartextnamen sondern nur Identitäten verschickt werden. Ohne Kenntnis der zugrundeliegenden Datenbasis sind diese für einen Datendieb wertlos.

Dringlicher wäre die Verwendung von Paßwörtern. Jeder, der die Portnummer des Nachrichtensystems kennt, könnte falsche Information einschleusen. Zwar ist es möglich, bei jedem Verbindungswunsch die Ursprungsadresse auf Zugehörigkeit zum System zu testen, doch ist es problemlos, sich eine IP-Adresse anzueignen. Eine gewisse Sicherheit böte ein bei jeder Nachricht angegebenes Paßwort.

In Kapitel 7 konnten mit dem System erzeugte Daten präsentiert werden. Es wurde gezeigt, wie sie zu einer Optimierung des Routings benutzt werden können. Eine tatsächliche Optimierung gehört nicht zum Umfang der Aufgabenstellung. Die Aufbereitung der Daten und die Unterstützung des Optimierungsprozesses durch Automatisierung stellen jedoch lohnende Betätigungsfelder dar.

# Literaturverzeichnis

- [Come95] Douglas E. Comer  
**Internetworking with TCP/IP** Volume I: Principles, Protocols, and Architecture  
3. Auflage  
Prentice-Hall, New Jersey, 1995
- [DFN] Verein zur Förderung eines Deutschen Forschungsnetzes e.V.  
**DFN-Mitteilungen**  
Hefte 34 bis 40  
DFN-Verein, Pariser Str. 44, 10707 Berlin
- [Geih95] Kurt Geihls  
**Client/Server-Systeme: Grundlage und Architekturen**  
Internat. Thomson Publ., Bonn, 1995
- [HoAn89] P. Holleczeck, C. Andres  
**A Programming Environment for Distributed Realtime Applications**  
in "Proceedings of the Twenty-Second Annual Hawaii International Conference on System Sciences", Vol II  
IEEE Computer Society Press, 1989
- [IETF1] Internet Engineering Task Force  
**Traffic Flow Measurement: Architecture**
- [IETF2] Internet Engineering Task Force  
**The Opstat Client-Server Model for Statistics Retrieval**
- [IETF3] Internet Engineering Task Force  
**A Model for Common Operational Statistics**
- [INT1] Martin Heyer  
**(Technisches) Konzept für das Accountingsystem für das B-WiN**  
Internes Papier
- [INT2] Martin Heyer  
**Verfahren zur Überwachung der Einhaltung der Dienstgüteparameter**  
Internes Papier
- [INT3] H.-M. Adler, Y. Clemens, M. Heyer, G. Hoffmann, K.-E. Maass, J. Pattloch S. Schweizer, K. Ullmann

- Das Breitband-Wissenschaftsnetz des Deutschen Forschungsnetzes**  
Zur Spezifikation und Beschaffung eines ATM-Netzes für die deutsche Wissenschaft  
Internes Papier
- [INT4] E. Jessen  
**Grobes Last- und Leistungsmodell für ein Hochgeschwindigkeitsnetz für den Verein zur Förderung eines Deutschen Forschungsnetzes e. V.**  
Erste Fassung am 30.03.94, Ergänzung am 06.01.95  
Internes Papier
- [JeVa] E. Jessen, R. Valk  
**Rechensysteme: Grundlagen der Modellbildung**  
Springer-Verlag, Berlin, Heidelberg, 1987
- [KeRi90] Brian Kernighan, Dennis Ritchie  
**Programmieren in C**  
Zweite Ausgabe  
Carl Hanser Verlag, München, Wien, 1990
- [KePi87] Brian Kernighan, Rob Pike  
**Der UNIX-Werkzeugkasten**  
Programmieren mit UNIX  
Carl Hanser Verlag, München, Wien, 1987
- [Kla89] Rainer Klar  
**Digitale Rechenautomaten**  
4. Auflage  
Walter de Gruyter, Berlin, New York, 1989
- [Knut73] Donald E. Knuth  
**The Art of Computer Programming**  
Volume 3/Sorting and Searching  
Addison-Wesley, 1973
- [Oust94] John K. Ousterhout  
**Tcl and the Tk toolkit**  
Addison-Wesley, Massachusetts, 1994
- [Pryc93] Martin de Prycker  
**Asynchronous Transfer Mode**  
Solution for Broadband ISDN  
Second Edition  
Ellis Horwood Limited, West Sussex, 1993
- [Sedg91] Robert Sedgewick  
**Algorithmen**  
Addison-Wesley, Bonn, 1991

- [Stev1] W. Richard Stevens  
**Advanced Programming in the UNIX Environment**  
Addison-Wesley, 1992
- [Stev2] W. Richard Stevens  
**Programmieren von UNIX-Netzen**  
Carl Hanser Verlag, München, Wien, 1992
- [Tane92] Andrew Tanenbaum  
**Computer-Netzwerke**  
Wolfram's Fachverlag, Attenkirchen, 1992
- [VORT] P. Holleczeck, Y. Clemens, I. Heller, M. Heyer, M. Slopianka  
**Das Breitband Wissenschaftsnetz (B-WiN) und das Erlanger B-WiN-Labor**  
Vortrag im Rahmen des RRZE-Kolloquiums
- [Wald95] Klaus Waldschmidt (Hrsg.)  
**Parallelrechner**  
Architekturen - Systeme - Werkzeuge  
Teubner, Stuttgart, 1995
- [WWW1] WWW-Administration des DFN-Vereins  
**Home Page des DFN-Vereins**  
<http://www.dfn.de/dfn/home.html>  
Benutzte Links:  
"Deutschen Forschungsnetzes" (dfn.html)  
"Porträt des DFN-Vereins" (dfn-portraet.html)  
"Gutachten des Wissenschaftsrates über den DFN-Verein" (wiss-rat/home.html)

# Abbildungsverzeichnis

2.1	Segmentbelastung . . . . .	13
3.1	Transportiertes Datenvolumen im WiN . . . . .	16
3.2	Beteiligte Organisationen . . . . .	20
3.3	Struktur des Breitbandwissenschaftsnetzes . . . . .	21
3.4	B-WiN Backbone . . . . .	22
3.5	Crossconnect der DeTeSystem . . . . .	23
4.1	Zu erzeugende Daten . . . . .	25
4.2	Breitband-Wissenschaftsnetz: Logische Struktur . . . . .	28
4.3	Verfeinerte Darstellung eines ZSS . . . . .	30
5.1	Architektur des Programmsystems . . . . .	32
5.2	Vereinigte Datenbasis . . . . .	35
5.3	Schnittstellen des Programmsystems . . . . .	36
5.4	Kommunikationsstruktur . . . . .	42
5.5	Sequentieller Server . . . . .	43
5.6	Paralleler Server . . . . .	43
5.7	Zentrale Datenstrukturen der Zentrale . . . . .	44
5.8	Funktionales Modell der Zentrale . . . . .	46
5.9	Funktionales Modell des Außenpostens . . . . .	49
6.1	Protokollautomaten für Zentrale und AP . . . . .	59
6.2	Erweiterter Produktautomat . . . . .	60
7.1	Analoge Darstellung . . . . .	64

# Tabellenverzeichnis

2.1	IP-Netzklassen . . . . .	7
2.2	Ausgewählter Bereich . . . . .	9
2.3	Herausnahme von 512 Adressen . . . . .	10
2.4	Verkehrsverteilung . . . . .	12
2.5	Ermittlung der Segmentbelastung . . . . .	13
4.1	Wandel der Daten . . . . .	26
5.1	Felder des Headers . . . . .	39
5.2	Nachrichtentypen . . . . .	41
6.1	Aufteilung auf Viertelstunden . . . . .	56
6.2	Alternative Aufteilung auf Viertelstunden . . . . .	56
7.1	Standortmatrix (absolut) . . . . .	61
7.2	Standortmatrix (relativ) . . . . .	62