INSTITUT FÜR INFORMATIK
der Technischen Universität München
Lehrstuhl VIII – Rechnerkommunikation – Prof. Dr. E. Jessen

ATM
Die Technik des B-ISDN

Volker Gerd Fischer und Helmut Gogl

März 1998

Hauptseminar Wintersemester 1997/1998

Version 1.1

Vorwort

Das vorliegende Manuskript ist das Ergebnis des am Lehrstuhl für Rechnerkommunikation durchgeführten Hauptseminars “ATM - Die Technik des Breitband-ISDN”.

Die einzelnen Beiträge wurden von dem jeweiligen Referenten nach seiem Vortrag ausgearbeitet und von uns in diesem Dokument zusammengestellt. In den einzelnen Kapiteln sind deshalb Sachverhalte mehrfach aufgeführt, bzw. werden von verschiedenen Sichtweisen aus dargestellt. Aber gerade die Vermittlung und Vertiefung dieser wichtigen Sachverhalte, welche im Kontext von ATM eine Rolle spielen war ein wesentliches Ziel dieser Vortragsreihe.

ATM ist eine neue Technik deren Standardisierung sich noch im Aufbau befindet. Spezifikationen sind beinahe halbjährlich rigidem Wandel unterworfen und erst ein Teil dieser Spezifikationen hat einen gewissen “steady-state” erreicht. Zentrum der Entwicklung ist die Institution des ATM-Forums[1] in der sich alle Hersteller zusammengeschlossen haben, um eine einheitliche Spezifikation zu ermöglichen. Grundlage des Seminars waren folgerichtig alle dort veröffentlichten “white-papers”. Dennoch haben sich sogar im Laufe des Seminars Änderungen der Spezifikationen ergeben, so daß zum Zeitpunkt der Erstellung dieses Manuskripts (März 1998) einige kleinere Punkte nicht mehr auf dem neuesten Stand sind.

Dennoch denken wir, daß dieses Skript eine wertvolle Einführung in die wichtigsten Konzepte von ATM darstellt, auch deshalb, weil auf die sonst übliche Strenge in der Darstellung der Sachverhalte verzichtet wurde.

Wir bitten alle Leser die vielen kleinen und großen Fehler, welche sich noch im Dokument befinden zu entschuldigen und uns falls möglich Korrekturvorschläge zukommen zu lassen[2].

München im März 1998

Volker Fischer und Helmut Gogl

Ankündigung des Hauptseminars

Einführung

ATM wird als **die** Netzwerktechnik der Zukunft gepriesen. Praktisch alle großen Telekommunikations-Dienstleister haben sich auf ATM als zukünftige Technologie geeinigt. Als herausragende Eigenschaften von ATM gelten hohe Bandbreite, Skalierbarkeit und Multimediafähigkeit.
Bereits heute wird ein großer Teil des WAN/MAN-Verkehrs über ATM abgewickelt (z.B. im Breitband Wissenschaftsnetz B-WiN) und auch im lokalen Bereich geht die Tendenz mit der Verfügbarkeit und Erschwinglichkeit der Komponenten in Richtung von ATM.

Ziel des Hauptseminars ist es, einen umfassenden Überblick über die Technik, den Stand der Entwicklung und die zukünftigen Möglichkeiten von ATM zu geben.
Außerdem werden Möglichkeiten der sanften Migration von bestehenden Netzstrukturen hin zu ATM besprochen.

[1] `www.atm-forum.org`

[2] email: `fischerv@informatik.tu-muenchen.de` bzw. `gogl@informatik.tu-muenchen.de`

Themen

1. ATM - Grundlagen und Einordnung in die Netzwerklandschaft
2. Physikalische Schicht
3. ATM Schicht
4. AAL Schicht
5. Signalisierung und ATM-Routing (1)
6. Signalisierung und ATM-Routing (2)
7. Protokolle der Anwendungsschicht
8. ATM Switches
9. Verkehrs- und Überlastkontrolle
10. ATM Meßtechnik
11. IP Über ATM / LANE / MPOA
12. IPnG über ATM / RSVP und Aufbau von Virtual Private Networks

Organisatorisches

Das Seminar richtete sich vorwiegend an die Studenten im Hauptstudium des Diplomstudiengangs Informatik an der TU und der LMU München. Nach erfolgreicher Teilnahme wurde ein Schein ausgestellt, der an beiden Universitäten als Hauptseminarschein anerkannt wird.

Teilnehmer

Vortrag	Kapitel	Vortragender	Vortragsdatum	Betreuer
1	1	Vincenzo Michel Peccia	13.11.97	Volker Fischer
3	2	Dierk Wiedemann	20.11.97	Volker Fischer
4	3	Norbert Menzl	27.11.97	Volker Fischer
5	4	Michael Riß	11.12.97	Volker Fischer
6	5	Reinhard Dietrich	18.12.97	Volker Fischer
7	6	Götz Bock	15.01.98	Volker Fischer
8	7	Christian Schwingenschlögl	22.01.98	Volker Fischer
10	8	Andreas Gräf	05.02.98	Helmut Gogl
11	9	Peter Vrzal	12.02.98	Volker Fischer
12	10	Ben Auch	19.02.98	Volker Fischer

Inhaltsverzeichnis

1 ATM Grundlagen und Einordnung in die Netzwerklandschaft

1.1 Überblick

ATM wird als *die* Netzwerktechnik der Zukunft gepriesen. Praktisch alle großen Telekommunikations-Dienstleister haben sich auf **ATM** als zukünftige Technologie geeinigt. Als herausragende Eigenschaften von **ATM** gelten hohe Bandbreite, Skalierbarkeit und Multimediafähigkeit. Bereits heute wird ein großer Teil des WAN/MAN-Verkehrs über **ATM** abgewickelt (z.B. im Breitband Wissenschaftsnetz B-WIN) und auch im lokalen Bereich geht die Tendenz mit der Verfügbarkeit und Erschwinglichkeit der Komponenten in Richtung von **ATM** .

Ziel dieser Ausarbeitung ist es, einerseits einen grundlegenden Überblick über die Netzwerklandschaft zu geben, und andererseits den Leser in **ATM** - die Technik des Breitband-ISDN einzuführen.

1.2 Anforderungen an die moderne Datenkommunikation

Betrachtet man die Entwicklung der letzten Jahre im Bereich der Computersysteme (Prozessorleistung) mit der Weiterentwicklung im Bereich Datenkommunikation (Übertragungsgeschwindigkeiten) so wird deutlich, daß die Datenkommunikation den heutigen Anforderungen nicht mehr genügt. Die Prozessorleistung und Speicherausrüstung hat sich in den letzten 10 Jahren verhundertfacht, aber die Übertragungsgeschwindigkeiten haben in WANs nur um das zehnfache zugenommen, bzw. sind im LAN-Bereich sogar gleich geblieben - und das bei enorm ansteigendem Bedarf ! Abbildung 1 zeigt die Entwicklung der letzten Jahre im Bereich Prozessorgeschwindigkeit und Speicherausstattung von Personalcomputern, Abbildung 2 die Entwicklung der Übertragungsgeschwindigkeiten im Netzwerkbereich.

Jahrzehntelang wurden Informationen in der Datenverarbeitung hauptsächlich in alphanumerischer Form gespeichert, bis zu Beginn der neunziger Jahre durch den rapiden Preisverfall für Massenspeicher und Prozessoren der Einsatz von Multimedia-Anwendungen auch auf EDV-Systemen möglich wurde. Informationen werden heute zunehmend in Form von Bild-, Text-, Video-, und Ton/Sprachdokumenten gespeichert (Multimedia-Datenbanken) und auch zur interaktiven Kommunikation benutzt (Videokonferenz).

Nun einige Beispiele in Bezug auf die QoS-Parametern (Quality of Service) unterschiedlicher Datenübertragungen wie Sprache, Telnet, HDTV und Video.

Beschränkt man sich bei der Übertragung von Sprache auf eine Frequenzbandbreite von 4 kHz, so ist für die digitale Übertragung eine Kapazität von 64 Kbit/s nötig.

Bei Telnet-Anwendungen besteht meist ein Bildschirminhalt aus 40 Zeilen zu je 80 Zeichen. Jedes Zeichen wird mittels 8 Bit kodiert, so daß ein Bildschirminhalt einer Datenmenge von 25,6 Kbit/s entspricht. Eine Übertragung eines gesamten Bildschirminhaltes ist bei vielen Anwendungen jedoch gar nicht nötig, da meisten nur Ein- bzw. Ausgaben auf dem Bildschirm aktualisiert werden. So sinkt die durchschnittliche Datenübertragungsrate deutlich ab, und es zeigt sich, daß Übertragungsgeschwindigkeiten von 2,4 bis 4,8 Kbit/s ausreichend sind.

Der Bandbreitenbedarf für heutige Multimedia-Anwendungen liegt zwischen 10 Mbit/s (Video) und 900 Mbit/s (HDTV). Eine wesentliche Rolle dabei spielen die verwendeten Optimierungs- und Kompressionsverfahren. So benötigt ein Video im MPEG-1 Format mit einer Auflösung von 352 x 288 eine Übertragungsbandbreite von 1,15 Mbit/s und ein Video in HDTV-Qualität

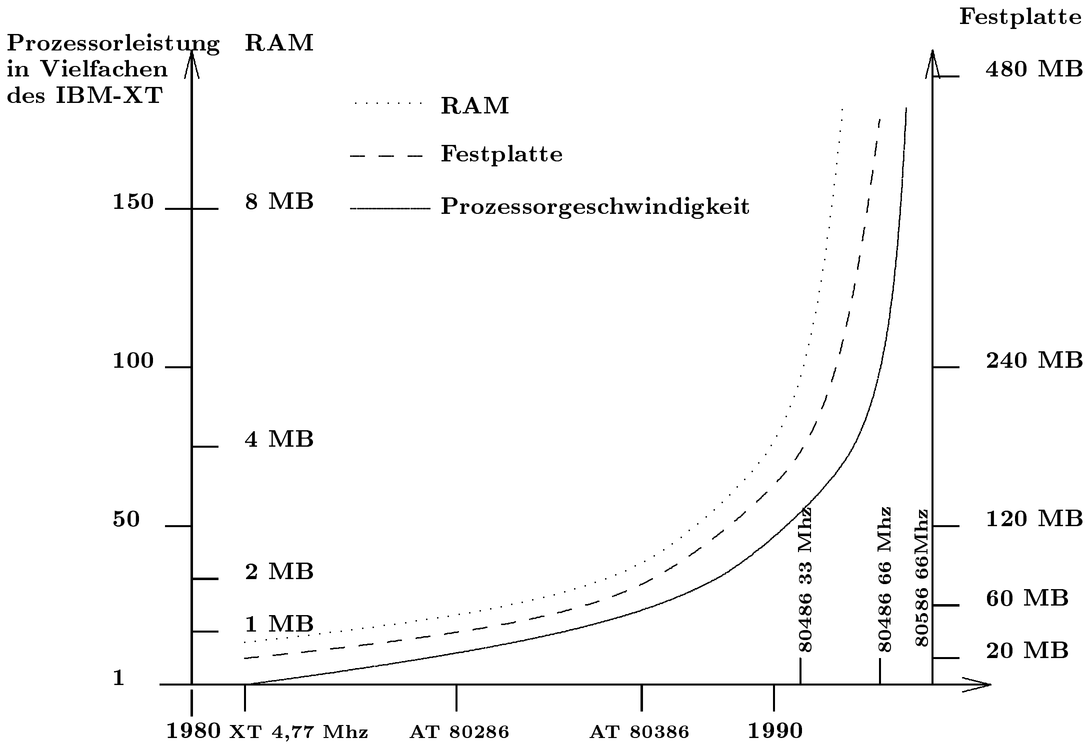


Abbildung 1: Entwicklung Prozessorgeschwindigkeit und Speicherausstattung von PCs

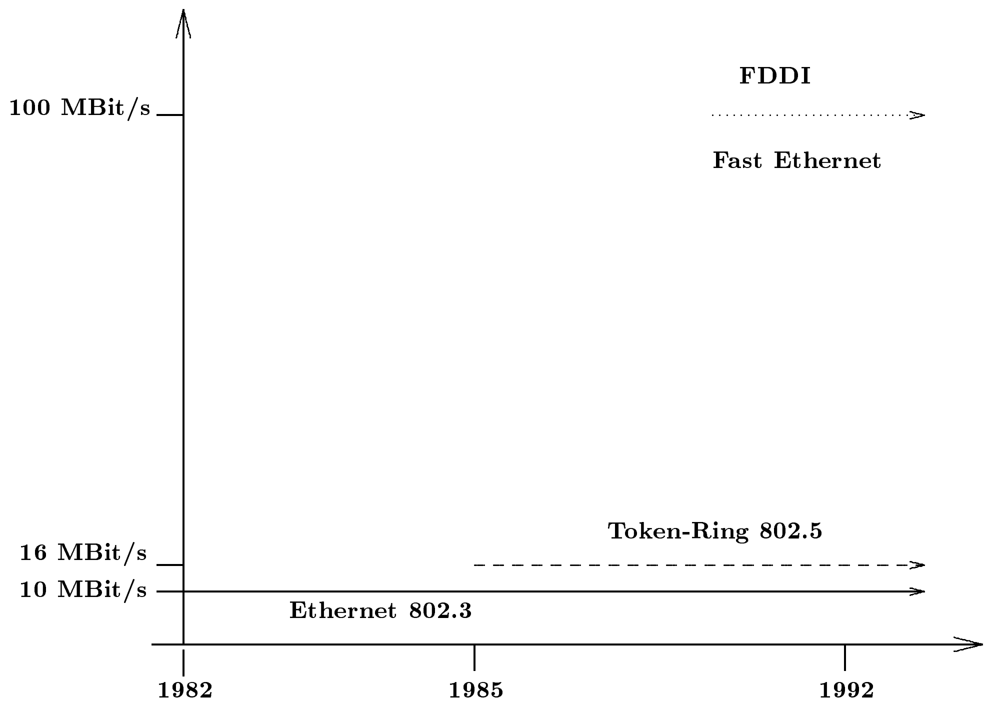


Abbildung 2: Entwicklung der Übertragungsgeschwindigkeiten im Netzwerkbereich

im MPEG-3 Format mit einer Auflösung von 1920 x 1080 eine Übertragungsbandbreite von 20 Mbit/s.

Die entscheidende Ursache für die Unfähigkeit der traditionellen Netzwerktechniken, den multimedialen Anwendungen der neunziger Jahre zu genügen liegt nicht allein an den zu geringen Übertragungsbandbreiten, sondern auch an ihrer Unfähigkeit Echtzeitverhalten zu garantieren. Abbildung 3 zeigt die Anforderungen moderner Datenkommunikation in Bezug auf Übertragungsbandbreite und Echtzeitverhalten.

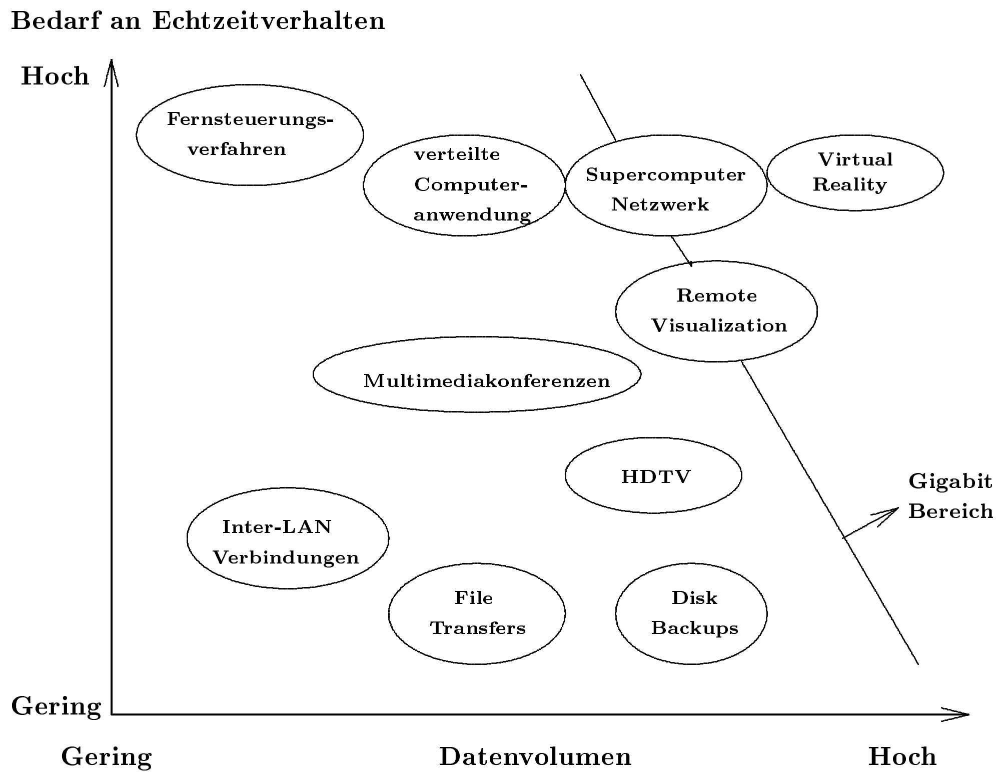


Abbildung 3: Entwicklung der Übertragungsgeschwindigkeiten im Netzwerkbereich

Endbenutzer verlangen eine immer größer werdende Anzahl von Diensten. Die bekanntesten Teledienste der Zukunft dürften wohl Video on Demand, Videokonferenz, Home Shopping, Hochgeschwindigkeits-Datentransfer, Telefonie, Telearbeit und HDTV sein. Alle diese Dienste können mit ATM verwirklicht werden.

1.3 Eigenschaften von ATM

1.3.1 Übertragungsprinzip des asynchronen Transfer Modus

Der asynchrone Transfer Modus (**ATM**) ist ein verbindungsorientiertes Übertragungsverfahren, welches auf asynchronem Zeit-Multiplexing unter Verwendung fester Datenpaketlängen basiert. Bei diesem Multiplexing werden die zu übertragenden Datenströme in Datenpakete fester Länge umgewandelt mit einem Header versehen und asynchron durch **ATM** -Switches und **ATM** -Cross-Connects übertragen. Die Abbildung 4 zeigt das **ATM** -Prinzip.

Die Datenpakete werden Zellen genannt und haben eine Länge von 53 Byte, wobei fünf Byte für den Zellen-Header reserviert sind.

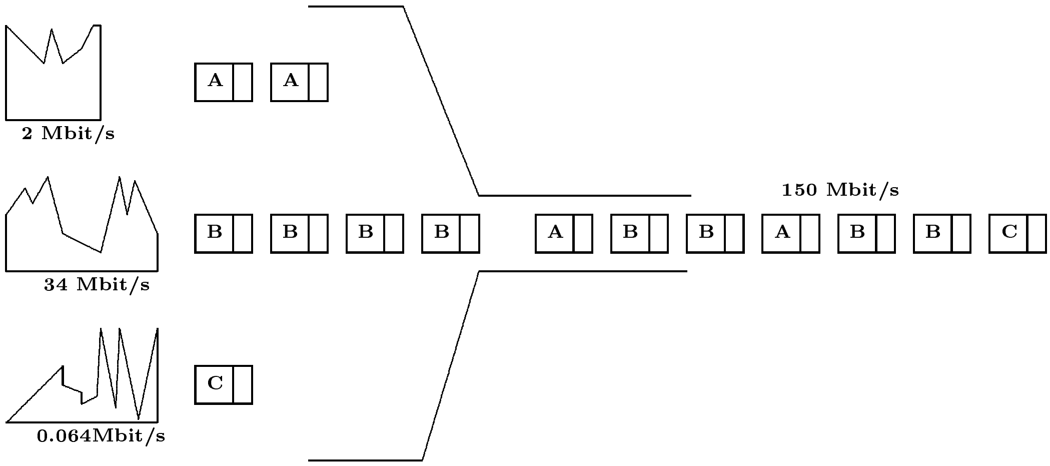


Abbildung 4: **ATM** -Prinzip

Wie kam es aber zu der Zellenlänge 53 Byte? Die Festlegung der Zellenlänge auf 53 Byte ist ein Kompromiß zwischen den Anforderungen von analoger Sprachübertragung und digitaler Datenübertragung. Für die Übertragung rein analoger Sprachsignale wäre eine möglichst geringe Zellenlänge wünschenswert. Die europäischen Ländern, in denen der reine Datenverkehr noch nicht die Größenordnung wie in USA erreicht hat, schlugen deshalb ursprünglich eine Zellenlänge von 32 Byte vor. Die USA versuchte eine Zellenlänge von 64 Byte durchzusetzen, da aufgrund des geringeren Overheads größere Datenmengen effizienter übertragen werden können. So einigte man sich auf die Zellenlänge von 53 Byte, die einerseits noch kurz genug ist um bei den hohen **ATM** -Übertragungsraten (Mega- und Gigabit/s-Bereich) für analoge Signale hervorragend geeignet zu sein, und andererseits aufgrund eines Overhead-Anteils von 9,4 % lang genug ist, um reine Datenströme ökonomisch zu übertragen.

Der 5 Byte große **ATM** -Zellen-Header enthält virtuelle Kanal- und Pfadinformationen (VCI/VPI) und wird mit Hilfe eines speziellen Feldes im Zellkopf, dem „Header Error Control" (HEC) Feld vor Übertragungsfehlern geschützt. Zu beachten ist, daß nur der Zellkopf geschützt wird, das Informationsfeld der Zelle bleibt ungeschützt.

Einzelne Verbindungen werden durch virtuelle Pfade und Kanäle gekennzeichnet. Dies bedeutet, daß zwischen Informationsquelle und Empfänger keine physikalisch Verbindung aufgebaut wird, sondern nur eine logische Verbindung, bei der die Vermittlungsstellen die VCI/VPI - Informationen im Zellen-Header auswerten. Abbildung 5 zeigt eine **ATM** -Verbindung.

Ein virtueller Kanal ist die kleinste Einheit für die Vermittlung von Zellen und wird zwischen zwei Benutzern aufgebaut. Ein virtueller Kanal ist immer unidirektional, es kann jedoch ein Kanalpaar zur gleichen Zeit erzeugt werden, diese werden dann über dieselbe Kennung angesprochen, so daß ein solcher virtueller Kanal in diesem Fall bidirektional ist. Alle Zellen die zu derselben virtuellen Verbindung gehören, werden mit der gleichen Kanal- und Pfadnummer (VCI) versehen.

Zellen verschiedener Verbindungen, die für einen Übertragungsabschnitt den gleichen Weg haben, werden zu einem virtuellen Kanalbündel, einem sogenannten virtuellen Pfad zusammengefaßt. Alle diese Zellen erhalten die gleiche Pfadnummer (VPI). Somit kann eine virtuelle Verbindung anhand des VPI /VCI Paares im Zellen-Header eindeutig identifiziert werden.

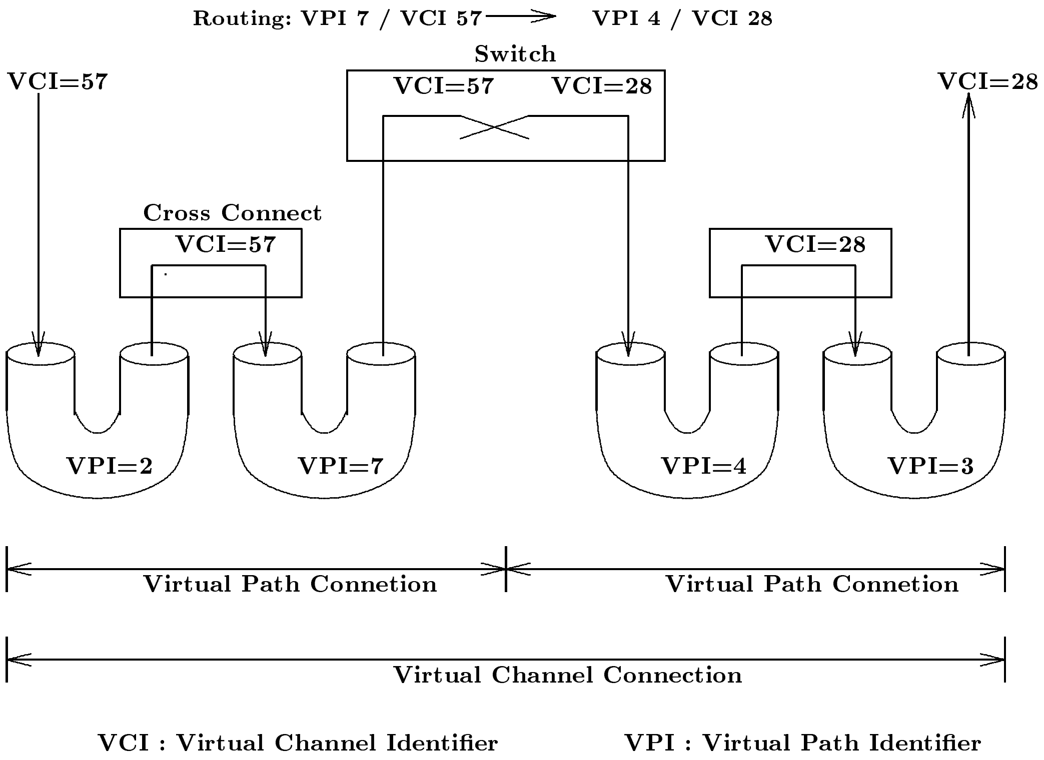


Abbildung 5: **ATM** -Verbindung

1.3.2 Verkehrskontrakt, Dienstgüteanforderung

Für jede Verbindung wird in der Aufbauphase ein Verkehrskontrakt vereinbart. Dieser Kontrakt beschreibt die Charakteristik des Zellstroms und die vom Netz zur Verfügung zu stellende Dienstgüte. Die Bestandteile des Verkehrskontrakts sind dementsprechend die verbindungsspezifischen Verkehrsparameter und die Dienstgüteanforderungen. Diese Parameter werden von den Endsystemen zur Verfügung gestellt.

Im Folgenden wird nun schematisch beschrieben wie der Verbindungsaufbau einer Point-to-Point Verbindung erfolgt. Generell ist der Verbindungsaufbau analog zum Schmalband-ISDN.

Will Teilnehmer A eine Verbindung mit Teilnehmer B aufbauen, so muß er zuerst ein SETUP auf den Signalisierungskanal schicken (alle Signalisierungsinformationen werden in VCI=5 und VCI=0 transportiert).Diese Nachricht enthält eine Reihe von Informationen wie z.B. die Adressen der anrufenden bzw. angerufenen Partei, die gewünschte Dienstgüte (Quality of Service - QoS) und einen Identifikator, der eine konkrete Verbindung eindeutig identifiziert. Der erste Switch im **ATM** -Netzwerk bestätigt einfach diesen Verbindungswunsch und liefert einen VPI/VCI - Wert für die bisher noch nicht existierende Verbindung in einer CALL PROCEEDING Nachricht zurück. Nun wird überprüft, ob genügend Ressourcen vorhanden sind um die Verbindung aufbauen zu können (Source-Routing). Dann schickt das **ATM** -Netz ein SETUP an den Teilnehmer B in der unter anderem der VPI/VCI - Wert für die aufzubauende Verbindung steht. Da der Teilnehmer B auch zuerst die eigenen Ressourcen überprüfen muß, schickt er zuerst ein CALL PROCEEDING and das Netzwerk. Kann die Verbindung aufgrund der freien Ressourcen aufgebaut werden, schickt B ein CONNECT an das Netzwerk, dieses wiederum antwortet mit einem CONNECT ACKNOWLEDGE. Der Teilnehmer A erhält nun auch ein CONNECT und antwortet mit einem CONNECT ACKNOWLEDGE. Nun steht die Verbindung zwischen Teilnehmer A und B und es können Daten übertragen werden. Abbildung 6 zeigt einen B-ISDN-Verbindungsaufbau.

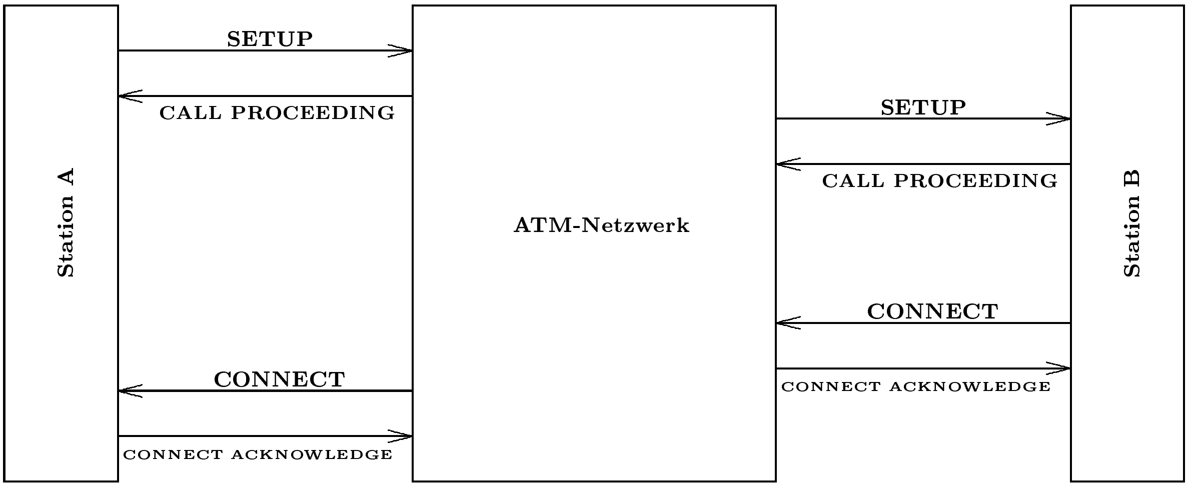


Abbildung 6: B-ISDN-Verbindungsaufbau

1.3.3 Verkehrsparameter

Wie im schematischen Verbindungsaufbau kurz erwähnt, wird bei **ATM** -Netzen die von Verbindungen geforderte Dienstgüte (QoS) durch eine Reihe von Parametern festgelegt. Diese beziehen sich auf die von den jeweiligen Endsystemen beobachtete Übertragungsqualität und werden beim Verbindungsaufbau zwischen Endsystem und Netz vereinbart.

Folgende QoS-Parameter, die bezüglich einer Verbindung vereinbart werden können, hat das **ATM** -Forum identifiziert:

- maximale Zellverzögerung (Maximum Cell Transfer Delay [maxCTD])
- Zellverzögerungsschwankung (Peak-to-Peak Cell Delay Variation [Peak-to-PeakCDV])
- Zellverlustrate (Cell Loss Ratio [CLR]).

Falls die Netzbelastung die Erfüllung eines geforderten QoS-Parameters nicht ermöglicht, wird der Aufbau der Verbindung verweigert!

Folgende QoS-Parameter beschreiben die Charakteristik eines **ATM** -Zellstroms

- maximale Zellrate (Peak Cell Rate [PCR])
- vertretbare Zellrate (aus der Sicht des Netzbetreibers) (Sustainable Cell Rate [SCR])
- maximale Burstlänge (Maximum Burst Size [MBR])
- minimale Zellrate (Minimum Cell Rate [MCR])

Diese Parameter kennzeichnen das Sendeverhalten der Endsysteme und werden deshalb der Gruppe der endsystemspezifischen Verkehrsparameter zugeordnet. Zusammen mit der Toleranz für die Zellverzögerungsschwankung (Cell Delay Variation Tolerance [CDTV]) bilden sie die verbindungsspezifischen Verkehrsparameter.

1.3.4 Dienstklassen

Der Zusammenhang zwischen Dienstgüteanforderungen bzw. Charakteristiken der Zellströme und den Aktionen des **ATM** -Netzes beim Verbindungsaufbau und während der Übertragung, wird durch die Dienstklassen hergestellt.

Das **ATM** -Forum hat fünf Dienstklassen (Service Categories) definiert:

- Constant Bit Rate (CBR)
- Real-Time Variable Bit Rate (rt-VBR)
- Non-Real-Time Variable Bit Rate (nrt- VBR)
- Unspecified Bit Rate (UBR)
- Available Bit Rate (ABR)

CBR-Dienstklasse:
Diese Dienstklasse ist für Verbindungen konzipiert, die während der gesamten Verbindungsdauer eine konstante Bandbreite benötigen. Diese Bandbreite wird durch die maximale Zellrate (PCR) charakterisiert. In dieser Dienstklasse werden Echtzeitanwendungen, Sprach- und Videoübertragungen sowie Circuit Emulation unterstützt. Zellen, für die eine maximale Zellverzögerung überschritten wurde, werden als für die Applikation als unbrauchbar eingestuft.

rt-VBR-Dienstklasse:
Dieser Klasse ist ebenfalls für Verbindungen mit strengen Anforderungen an die Übertragungsverzögerung und die Zellverzögerungsschwankung vorgesehen. Es wird jedoch davon ausgegangen, daß die Endsysteme mit einer variablen Zellrate senden, wie sie z.B. bei MPEG-kodierten Videoübertragungen vorliegt. Die Charakterisierung des Zellstroms erfolgt durch eine maximale Zellrate (PCR) und zusätzlich durch eine vertretbare Zellrate (SCR) in Verbindung mit einer maximalen Burstlänge (MBS).

nrt-VBR-Dienstklasse:
Die nrt-VBR-Dienstklasse steht für Dienste zur Verfügung, die eine variable Zellrate und keine Echtzeitanforderungen haben. Die Zellrate wird, wie bei der vorigen Dienstklasse, durch die maximale Zellrate (PCR), vertretbare Zellrate SCR und die maximale Burstlänge (MBR) charakterisiert. Als Dienstgüte wird eine geringe Zellverlustrate vereinbart, jedoch werden keine maximalen Verzögerungszeiten festgelegt.

UBR-Dienstklasse:
Diese Klasse ist für Dienste ohne Echtzeitanforderungen konzipiert. Bezüglich der Dienstgüte werden keine Parameter vereinbart, d.h. eine Übertragungsicherung muß auf höheren Protokollschichten stattfinden. Aus diesem Grund werden UBR-Dienste auch als Best-Effort-Dienste bezeichnet. Typische Applikationen sind Datenübertragung zwischen Rechnern, wie e-mail und File Transfer.

Attribute	*CBR*	*rt-VBR*	*nrt-VBR*	*UBR*	*ABR*
Parameter					
PCR	**ja**	**ja**	**ja**	**ja**	**ja**
SCR,MBR	**-**	**ja**	**ja**	**-**	**-**
MCR	**-**	**-**	**-**	**-**	**ja**
Dienstgüte					
CDV	**ja**	**ja**	**nein**	**nein**	**nein**
maxCTD	**ja**	**ja**	**nein**	**nein**	**nein**
CLR	**ja**	**ja**	**ja**	**nein**	**ja**
Andere					
Flußkontrolle	**nein**	**nein**	**nein**	**nein**	**ja**

Tabelle 1: Attribute der **ATM** -Dienstklassen im Überblick

ABR-Dienstklasse:
Die ABR-Dienstklasse unterstützt ebenfalls keine Echtzeitanwendungen, garantiert jedoch, im gegensatz zur UBR-Dienstklasse, eine geringe Zellverlustrate. Bei ABR-Diensten wird die Senderate der Endsysteme ständig an den aktuellen Lastzustand des Netzes angepaßt. Eine dynamische Aufteilung der für ABR-Dienste zur Verfügung stehenden Bandbreite unter den aktiven ABR-Verbindungen erlaubt eine gleichmäßig hohe Netzauslastung bei geringen Zellverlusten. Bezüglich Übertragungsverzögerung und deren Schwankung sind keine oberen Grenzen festgelegt. Beim Verbindungsaufbau werden eine maximale Zellrate PCR und eine minimal verfügbare Zellrate MCR vereinbart. Typische Applikationen für diese Dienstklasse sind der Datenaustausch zwischen lokalen Netzen und Remote-Terminal-Anwendungen.

Tabelle 1 zeigt die Attribute der **ATM** -Dienstklassen im Überblick.

Anzumerken ist hierbei noch, daß in der Praxis zum jetzigen Zeitpunkt bisher nur die CBR- und UBR-Dienstklasse implementiert und benutzt werden. Alle anderen sind nur teilweise implementiert.

1.3.5 Verkehrs- und Überlaststeuerungsmechanismen in ATM -Netzen

Zur Vermeidung und Auflösung von Überlastsituationen werden in **ATM** -Netzen verschiedene Mechanismen zur Netzverwaltung implementiert. Vorrangiges Ziel ist die Begrenzung der Dauer, Intensität und der Ausbreitung von momentanen Überlastsituationen und damit die Erhaltung der von den Verbindungen geforderten Dienstgüten. Folgende Mechanismen wurden standardisiert und können einzeln oder in beliebigen Kombinationen eingesetzt werden:

- Netzzugangskontrolle
- Überprüfung der Verkehrsparameter
- Verkehrsformung
- Selektives Löschen von Zellen / Protokollrahmen
- ABR-Datenflußsteuerung

Netzzugangskontrolle :
Signalisiert ein Endsystem einen Verbindungswunsch, so entscheidet die Netzzugangskontrolle, ob dieser Verbindungswunsch angenommen werden kann oder aufgrund mangelnder Ressourcen abgelehnt werden muß. Zu den Ressourcen zählen Pufferspeicher in Vermittlungsknoten, Bandbreite auf Übertragungsstrecken und Rechenkapazität zur Bewältigung des anfallenden Signalisierungsaufwandes. Auch Informationen aus dem Verkehrskontrakt und Messungen der aktuellen Netzlast fließen in die Entscheidung über die Annahme eines Verbindungswunsch mit ein.

Überprüfung der Verkehrsparameter :
Die Überprüfung der im Verkehrskontrakt festgelegten Verkehrsparameter dient zum Schützen der Netzressourcen vor absichtlichem oder unbeabsichtigtem Fehlverhalten der Endsysteme und damit zur Aufrechterhaltung der von anderen aktiven Verbindungen benötigten Dienstgüte. Bei einer Verletzung des Verkehrskontrakts wird notfalls die Einhaltung der vereinbarten Verkehrsparameter erzwungen, dies kann dadurch erreicht werden, daß Zellen mit niedriger Priorität verworfen werden, falls sie nicht zu den Verkehrsparametern konform sind. Außerdem kann bei besonders groben Verstößen die Auflösung einer Verbindung veranlaßt werden. Desweiteren ist die Überprüfung der Nummer des virtuellen Kanal Identifikators (VCI) und des virtuellen Pfad Identifikators (VPI) eine Aufgabe der Parameterkontrolle. Ist eine dieser Nummern ungültig, d.h. die Zelle kann keiner gültigen Verbindung zugeordnet werden, so wird die Zelle ebenfalls verworfen (vergleiche HEC).

Verkehrsformung :
Verkehrsformung wird eingesetzt um die Charakteristik des Zellstroms einer Verbindung so zu verändern, daß eine höhere Effizienz des Netzes bzgl. der Ressourcenauslastung bei gleichbleibender Dienstgüte erreicht werden kann. Sie kann bereits in den Endsystemen zur Anpassung der Zellströme oder zwischen Netzen verschiedener Betreiber eingesetzt werden. Beispiele für Verkehrsformung sind die Reduzierung der Spitzenzellrate, die Beschränkung der Burstlängen und die Reduzierung der Zellverzögerungsschwankung. Dies wird durch eine geeignete Verzögerung der Zellen erzielt, wobei die Reihenfolge der zur einer Verbindung gehörenden Zellen erhalten bleibt.

Selektives Löschen von Zellen / Protokollrahmen :
Befindet sich ein Netzelement, z.B. ein Verbindungsknoten in einem momentanen Überlastzustand, so kann er selektiv Zellen von Verbindungen verwerfen, die ihre Verkehrsparameter nicht einhalten. Weiterhin können Zellen mit niedriger Priorität verworfen werden, wobei die beim Verbindungsaufbau vereinbarte Zellverlustrate nicht überschritten werden darf. Falls Zellen verworfen werden, ist es in vielen Fällen effizienter, den gesamten zugehörigen Rahmen aus höheren Protokollschichten zu verwerfen. Der Grund dafür ist, daß nur komplett empfangene Rahmen von höheren Protokollschichten wieder zusammengesetzt werden können. Die Übertragung der restlichen Zellen von teilweise zerstörten Rahmen bedeutet somit eine Verschwendung von Netzressourcen. Das Löschen von ganzen Protokollrahmen unterstützt die Vermeidung von Überlastsituationen und kann zur Verbesserung des Netzdurchsatzes und der Fairneß unter aktiven Verbindungen beitragen. Die Anwendung dieses reaktiven Verfahrens ist allerdings auf Verbindungen beschränkt, die dies ausdrücklich zulassen und unterstützen.

ABR-Datenflußsteuerung :
Die ABR-Datenflußsteuerung wird ausschließlich für Verbindungen aus der ABR-Dienstklasse implementiert und findet zwischen sendendem und empfangendem Endsystem statt (End-to-End

Flow Control). Die Steuerung erfolgt durch einen Flußkontrollmechanismus, der den Endsystemen mittels sogenannter Ressource-Management-Zellen (RM) aktuelle Informationen über den Lastzustand des Netzes liefert. Generell werden die RM-Zellen von den sendenden Endsystemen periodisch generiert und zusammen mit den Datenzellen an die empfangenden Endsysteme übertragen, welche diese wieder zurück an den Sender schicken. Während die RM-Zellen die virtuelle Verbindung in Vorwärts- und Rückwärtsrichtung durchlaufen, modifizieren die Vermittlungsknoten, und optional das empfangende Endsystem, den Inhalt der RM-Zellen, um momentane Überlastungen zu signalisieren. Die Sendestation paßt daraufhin, aufgrund der empfangenen Informationen, die Senderate an. In kritischen Situationen können RM-Zellen auch von Vermittlungsknoten generiert werden. Es wird garantiert, daß Endsysteme, die ihre Senderate entsprechend der in den RM-Zellen signalisierten Informationen anpassen, geringe Zellverluste erfahren und einen fairen Anteil an der für ABR-Dienste zur Verfügung stehenden Bandbreite erhalten.

1.4 ATM im Vergleich zu anderen Datenübertragungstechniken

1.4.1 Ethernet / Fast Ethernet

Die Produkte Ethernet und Fast-Ethernet sind in IEEE-Norm 802.3 bzw. IEEE-Norm 802.3u definiert und arbeitet mit einer Übertragungsgeschwindigkeit von 10Mbps bzw. 100Mbps. Sie werden ausschließlich im LAN-Bereich benutzt.

Die Übertragungsmethode von Ethernet-Netzwerken ist ein Halb-Duplex-Verfahren. Dies bedeutet, daß das Senden und Empfangen von Daten nicht gleichzeitig stattfinden kann. Will eine Station übertragen, so hört sie das Kabel ab. Ist das Kabel belegt, wartet sie, bis es frei ist, ansonsten sendet sie sofort. Greifen zwei Stationen gleichzeitig auf ein freies Kabel zu, so erfolgt eine Kollision. Jede dieser Stationen unterbricht dann die Übertragung, wartet eine zufallsgesteuerte Zeitspanne und wiederholt den ganzen Vorgang. Diese Art eines Trägererkennungsprotokolls nennt man auch 1-Persistent CSMA (Carrier Sense Multiple Access). Abbildung 7 zeigt verschiedene physikalische Realisierungen von Ethernet-Netzwerken. Die Tabelle 2 zeigt vier Kabeltypen die bei Ethernet verwendet werden.

Bezeichnung	*Kabeltyp*	*Max. Seg.*	*Kn./Seg.*	*Vorteile*
10Base5	**Dickes Koax**	**500 m**	**100**	**Gut für Backbones**
10Base2	**Dünnes Koax**	**200 m**	**30**	**Kostengünstig**
10Base-T	**Verdrilltes Paar**	**100 m**	**1024**	**Einfache Wartung**
10Base-F	**Glasfaser**	**2000 m**	**1024**	**Ideal zwischen Gebäuden**
100Base-T	**Verdrilltes Paar**	**100 m**	**1**	**Schneller - teuer (HUB)**
1000Base-T	**Kategorie ≥ 6**	**25 m**	**??**	**Schnell, bekannte Technik**

Tabelle 2: Kabeltypen von Ethernet

Welche Nachteile hat Ethernet ?

- Erstens wächst die Wahrscheinlichkeit von Kollisionen mit zunehmender Anzahl von Netzteilnehmern. Ab einer Netzlast von 40-50% bei mehr als 10 Teilnehmern nimmt die Ausnutzung der zur Verfügung stehenden Bandbreite von 10Mbit/s drastisch ab.

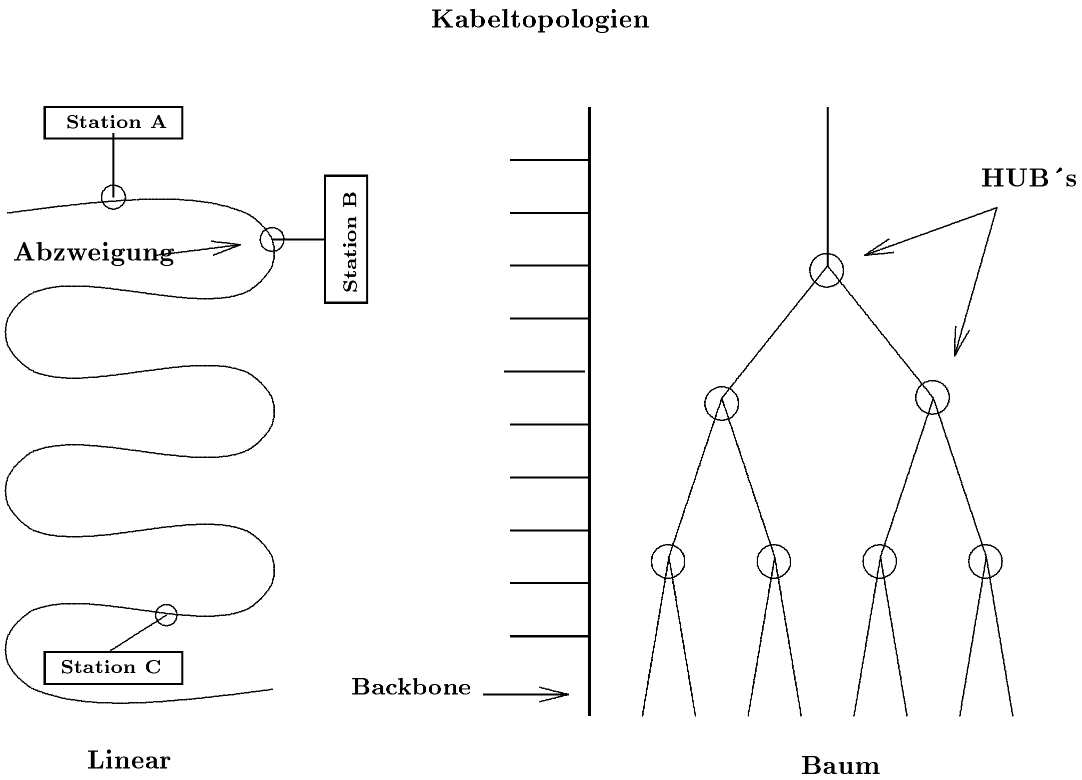


Abbildung 7: Physikalische Realisierungen von Ethernet-Netzwerken

- Zweitens ist auch die Länge des Netzsegmentes von entscheidender Bedeutung, denn mit zunehmender Länge wird nämlich der zentrale Parameter des CSMA/CD-Algorithmus - die Slotzeit - größer. Ihr Wert beträgt das Doppelte der Signallaufzeit zwischen den beiden am weitesten voneinander entfernten Stationen.
- Und drittens ist die Leistungsfähigkeit um so geringer, je kürzer die übertragenen Pakete sind.

Diese drei Faktoren zeigen deutlich, daß Ethernet (ausgenommen switched Ethernet) für die Übertragung von Multimedia-Applikationen denkbar ungeeignet ist, da u. a. für die Übertragung von analogen Signalen gerade kurze Pakete besser geeignet sind.

1.4.2 Token-Bus / Token-Ring und FDDI

Die Produkte Token-Bus und Token-Ring sind in IEEE-Norm 802.4 bzw. IEEE-Norm 802.5 definiert. Token-Bus und Token-Ring werden ebenfalls wie Ethernet im LAN-Bereich eingesetzt und haben Übertragungsgeschwindigkeiten von 1,5 Mbit/s bis 16 (neu 100) Mbit/s. FDDI hingegen wird zusätzlich noch im MAN-Bereich (bis zu 200km) eingesetzt und unterstützt Übertragungsgeschwindigkeiten von bis zu 100 Mbit/s.

Token-Bus :
Physikalisch besteht der Token-Bus aus einem linearen oder baumförmigen Breitbandkoaxialkabel mit 75 Ohm, an dem die Stationen angeschlossen sind. Von der Logik her sind die Stationen

als Ring organisiert. Abbildung 8 zeigt einen Token-Bus.

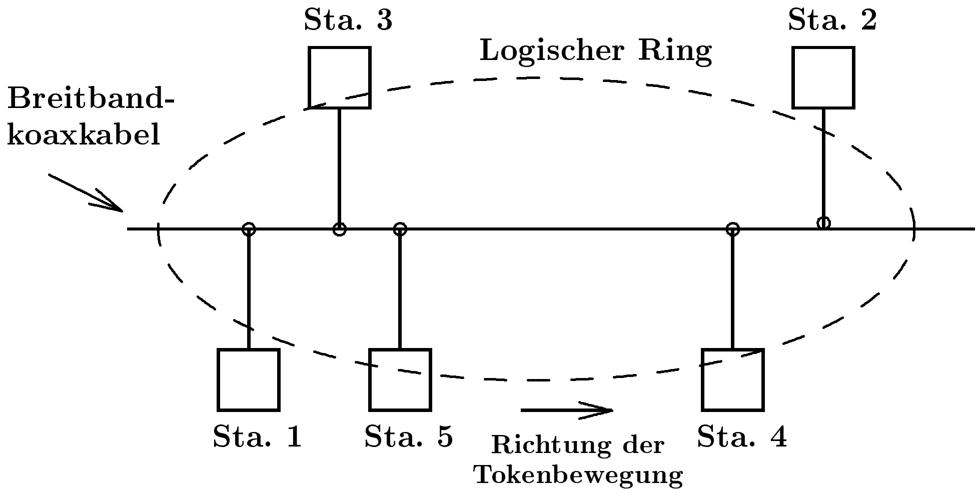


Abbildung 8: Token-Bus

Jede Station kennt die Adresse ihres linken und rechten Nachbarn. Wenn der logische Ring initialisiert wird, schickt die Station mit der höchsten Nummer den ersten Rahmen. Dann übergibt sie die Erlaubnis an ihre unmittelbare Nachbarin, indem sie dieser einen speziellen Steuerrahmen das sogenannte Token sendet. Jedesmal, wenn eine Station das Token erhält, darf sie für eine bestimmte Zeitspanne Rahmen übertragen. Dann muß sie das Token sofort weitergeben. Erhält eine Station das Token, hat aber nichts zu senden, so gibt sie es sofort weiter. Da immer nur eine Station das Token besitzt, treten keine Kollisionen auf. Wichtig bei Token-Bus ist, daß die physische Reihenfolge, in der die Stationen verbunden sind, keine Rolle spielt. Da das Übertragungskabel von Natur aus ein Broadcast-Medium ist, empfängt jede Station jeden Rahmen und läßt denjenigen weg, der nicht für sie bestimmt ist. Gibt eine Station das Token weiter, so sendet sie einen Token-Rahmen an die Adresse ihrer Ringnachbarin gleichgültig, wo sich diese physikalisch im Netz befindet.

Token-Ring :
Der Token-Ring besteht physikalisch aus einer Ansammlung von Stationen die in einem Ring angeordnet sind und durch Punkt-zu-Punkt-Kabel verbunden sind. Jedes Bit, das an einer Station ankommt, wird in einen 1-Bit-Puffer und dann wieder auf den Ring kopiert. Während sich das Bit im Puffer befindet, kann es untersucht und eventuell verändert werden, bevor es wieder ausgelesen wird. Wenn alle Stationen im Token-Ring untätig sind kreist ein spezielles Bitmuster - das Token - um den Ring. Falls eine Station übertragen will, muß sie zuerst das Token erlangen und aus dem Ring entfernen. Das Übertragen von Daten erfolgt wie beim Token-Bus, wobei die zurückkommenden Bits, die durch den ganzen Ring gewandert sind, wieder durch den Absender vom Ring entfernt werden müssen. Abbildung 9 zeigt ein Ringnetz.

Ist die Auslastung im Token-Ring sehr hoch, so wird jedesmal wenn eine Station eine Übertragung beendet hat und das Token weiterlaufen läßt, die nachfolgende Station das Token sofort wieder an sich reißen. Auf diese Art rotiert die Sendeberechtigung ohne Unterbrechung im Ring. Die Netzeffizienz nähert sich bei hoher Auslastung 100%.

FDDI (Fiber Distributed Data Interface) :
FDDI ist ein leistungsstarkes Token-Ring-LAN mit Lichtwellenleitern, welches bis zu 1000 Sta-

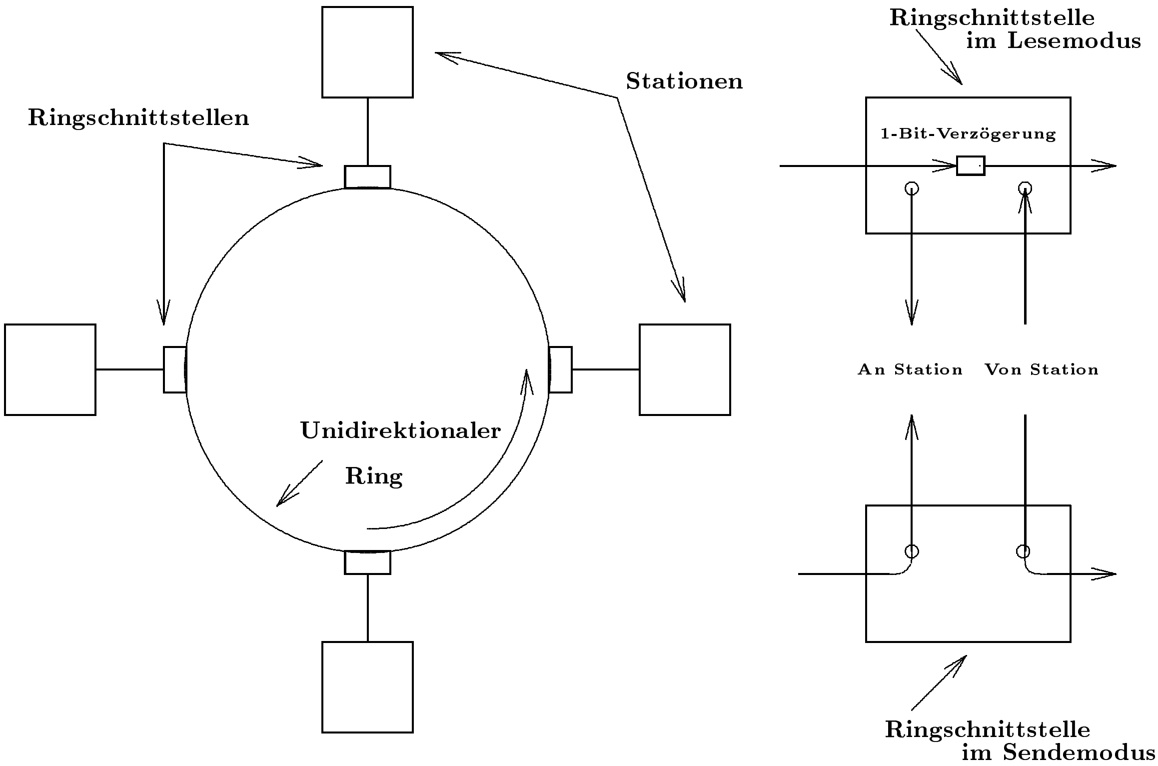


Abbildung 9: Token-Ring

tionen unterstützt und bis zu einem gewissen Grad echtzeitfähig ist. Es kann auf die gleiche Weise benutzt werden wie Token-Bus und Token-Ring. Häufig wird es aufgrund seiner hohen Bandbreite aber als Backbone für LANs mit Kupferkabeln benutzt. Die FDDI-Verkabelung besteht aus zwei Glasfaserringen. Einer überträgt im Uhrzeigersinn, der andere entgegen dem Uhrzeigersinn. Abbildung 10 zeigt ein FDDI-Netzwerk.

Bricht ein Ring, so kann der andere als Reserve, bzw. als sogenannter Schutzring, benutzt werden. Brechen beide am gleichen Punkt, z.B. durch einen Brand oder Unfall, können die zwei Ringe zu einem zusammengefaßt werden, welcher etwa die doppelte Länge hat. Jede Station enthält Relais, die benutzt werden können, um die zwei Ringe zusammenzufügen. Abbildung 11 zeigt diese Situation.

Die grundlegenden FDDI-Protokolle lehnen sich eng an den Token-Ring an. Es gibt allerdings den Unterschied, daß bei FDDI allen Stationen gewährt wird, das Token wieder in den Ring zustellen, sobald sie mit der Übertragung ihrer Rahmen fertig sind. In einem Ring können sich mehrere Rahmen gleichzeitig befinden, da es bei potentiell 1000 Stationen und einer Faserstrecke von 200 km eine Verschwendung wäre auf den abgesendeten Rahmen zu warten bis er den gesamten Ring durchlaufen hat.

Nachteile von Token-Bus / Token-Ring und FDDI :
Der Token-Bus benutzt Material aus dem Bereich des Kabelfernsehens. So verwenden Breitbandsysteme sehr viel Analogtechnik, unter anderem auch Breitbandverstärker und Modems. Das Protokoll ist sehr komplex und erzeugt bei geringer Auslastung hohe Verzögerung (die Stationen müssen immer auf das Token warten, auch in einem ansonsten untätigen System).

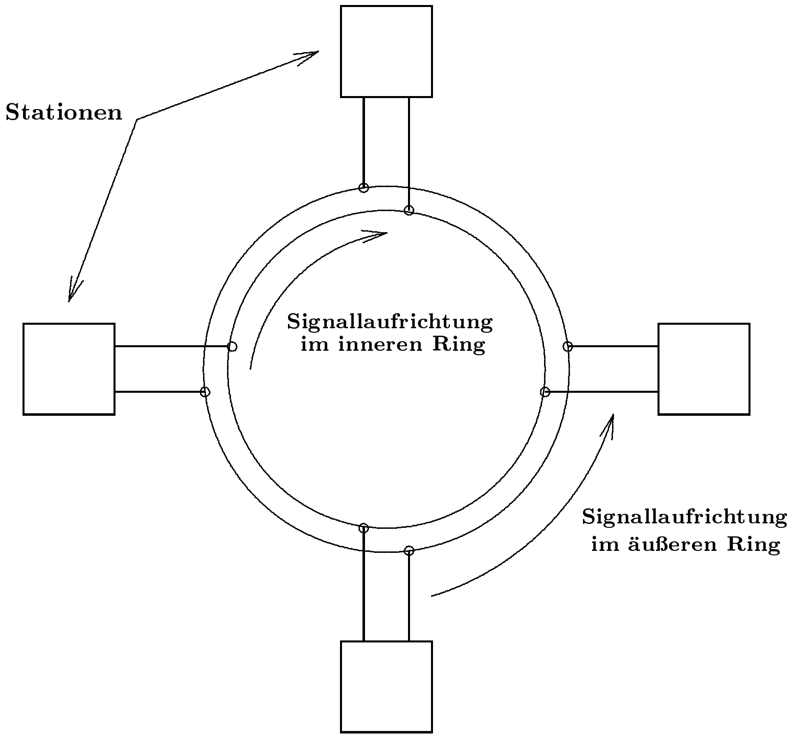


Abbildung 10: FDDI-Netzwerk

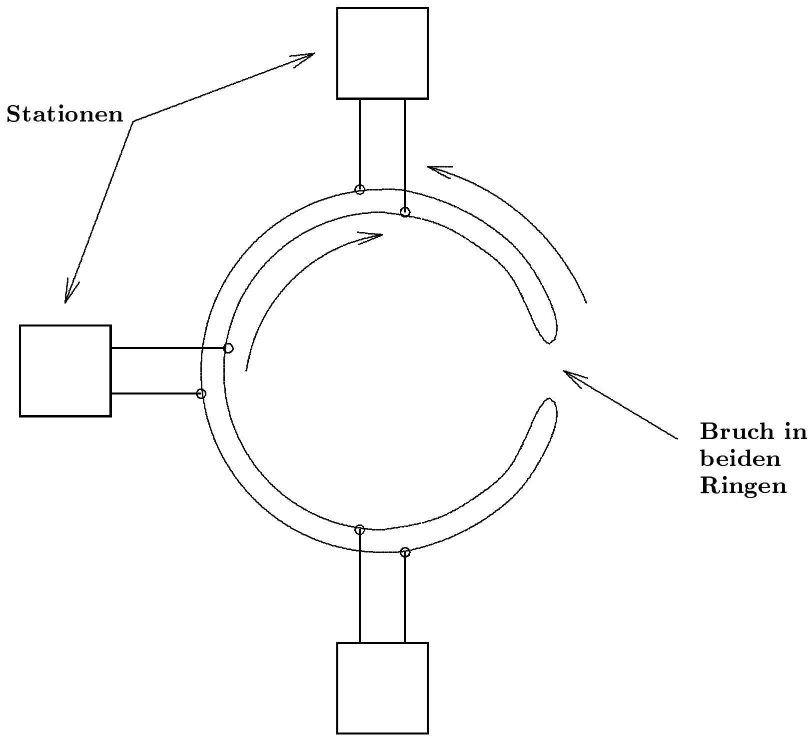


Abbildung 11: FDDI-Netzwerk mit Ausfall einer Station

Schließlich ist diese Technologie generell teurer als Ethernet.

Der größte Minuspunkt bei Token-Ring ist die Existenz einer zentralen Überwachungsfunktion, die eine unsichere Komponente darstellt. Obwohl eine abgestürzte Überwachungsstation ersetzt werden kann, führt eine fehlerhafte Überwachungsstation meist zu größeren Schwierigkeiten. Auch beim Token-Ring entstehen bei geringer Auslastung des Netzes Verzögerungen.

Der Nachteil von FDDI ist das zu komplizierte Stationmanagement, was zu komplexen Chips und hohen Preisen führt.

1.4.3 X.25 und Frame-Relay

X.25 :
Der Standard X.25 wurde in den siebziger Jahren vom CCITT entwickelt und ist noch in vielen älteren Netzen zu finden. Er wurde entwickelt um eine Schnittstelle zwischen öffentlichen paketvermittelnden Netzen und den Kunden bereitzustellen. X.25 ist verbindungsorientiert und unterstützt sowohl vermittelte, als auch permanente, virtuelle Leitungen.

Eine vermittelte virtuelle Leitung wird erzeugt, wenn ein Rechner ein Paket an das Netz schickt und eine Verbindung zu einem entfernten Rechner anfordert. Nach dem Aufbau der Leitung können über diese Verbindung Pakete versendet werden, die immer in der richtigen Reihenfolge ankommen. X.25 unterstützt Flußsteuerung und stellt somit sicher, daß ein schneller Sender nicht einen langsameren Empfänger mit Daten überflutet.

Eine permanente virtuelle Leitung basiert auf dem gleichen Prinzip wie eine vermittelte, jedoch wird sie im voraus zwischen dem Kunden und dem Anbieter vereinbart. Sie ist immer vorhanden, und für die einzelnen Übertragungen ist kein Verbindungsaufbau erforderlich. Dies entspricht einer Standleitung.

Die meisten X.25 Netze arbeiten mit einer Geschwindigkeit von bis zu 64 Kbps und erlauben dem Benutzer Pakete mit einer maximalen Länge von 128 Byte zu übertragen.

Eine zentrale Aufgabe der Sicherungsschicht von X.25 ist es, die Übertragungsfehler der Telefonleitung zwischen Endeinrichtung und dem öffentlichen Netz zu beheben.

Frame Relay :
Frame Relay besitzt auch wie X.25 einen verbindungsorientierten Übertragungsmechanismus, d.h. für jede Übertragung muß eine virtuelle Verbindung aufgebaut werden. Außerdem multiplext Frame Relay wie X.25 die Übertragungsrahmen verschiedener Sende- bzw. Empfangsstationen nach statistischen Gesichtspunkten über eine Leitung. Es unterstützt dabei Geschwindigkeiten zwischen 56 Kbit/s und 45 Mbit/s.
In Abbildung 12 ist ein Frame Relay Übertragungsrahmen dargestellt.

Der wesentliche Unterschied zwischen Frame Relay und X.25 besteht in den Fehlerkorrekturmechanismen. Im Gegensatz zu X.25 bietet Frame Relay keine Möglichkeit, verlorene oder fehlerhaft übertragene Datenpakete wiederholt zu übertragen, da heutige Weitverkehrsstrecken nur noch geringe Fehlerraten besitzen. Der Benutzer muß selbst feststellen, ob ein Rahmen fehlt, und die erforderlichen Maßnahmen einleiten. Lediglich die Gültigkeit der Adressen sowie das Auftreten von Bitfehlern wird überprüft.

Der Aufbau eines Frame Relay-Übertragungsrahmens ist sehr einfach, da zwischen dem Beginn- und dem Ende-Flag sich außer den Benutzerdaten lediglich ein 2 - 4 Byte langes Headerfeld befindet. Innerhalb des Informationsfeldes können bis zu 8 KByte Benutzerdaten übertragen werden.

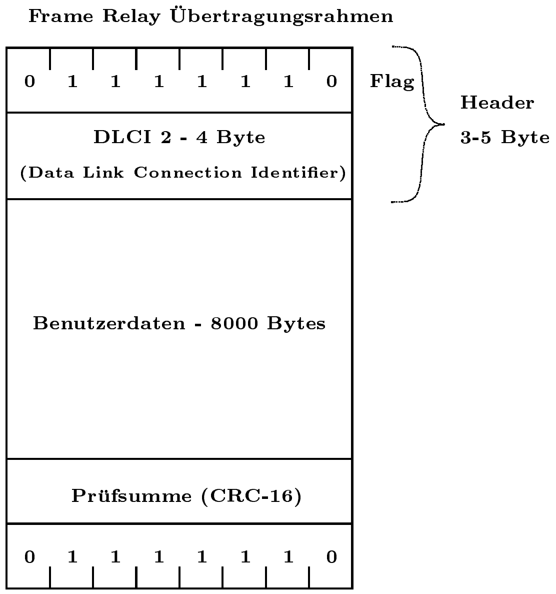


Abbildung 12: Übertragungsrahmen bei Frame Relay

Nachteil von Frame Relay :
Frame Relay ist zwar außerordentlich effizient bei der Übertragung von traditionellem Datenverkehr, aber es ist von seinem Aufbau her nicht in der Lage stark wechselnde Verkehrsprofile, wie sie bei Multimedia-Anwendungen auftreten, zu bewältigen. Müßten mit Frame Relay Daten und Sprachsignale gleichzeitig übertragen werden, so würden Sprachsignale mit starken Jitter das Resultat sein.

1.4.4 DQDB

Das Produkt DQDB (Distributed Queue Dual Bus) ist in IEEE-Norm 802.6 definiert und arbeitet mit Übertragungsgeschwindigkeiten von 1,5 Mbit/s bis 2048 Mbit/s und wird vorwiegend im MAN-Bereich eingesetzt. In der Regel werden DQDB-Netze mit 44,736 Mbit/s betrieben und decken bis zu 160 Kilometer ab. Zur Zeit wird DQDB bei der Telekom unter dem Dienst Datex-M angeboten. Die Grundlegende Geometrie von DQDB wird in Abbildung 13 gezeigt.

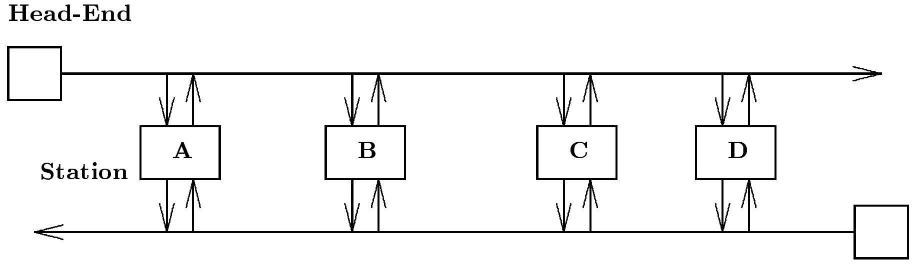


Abbildung 13: Dualbus-Prinzip von DQDB

Zwei parallele unidirektionale Busse schlängeln sich ihren Weg durch eine Stadt. Jede Station ist an beide Busse gleichzeitig angeschlossen. Jeder Bus hat ein Head-End, das einen ständigen Strom von Zellen erzeugt. Jede Zelle fließt vom Head-End weg - erreicht sie das Ende, fällt sie vom Bus.

Jede Zelle besteht aus einem 5 Byte großen Header, 48 Byte Nutzdaten und wird in 125 μs langen Datenrahmen übertragen. Jeder Rahmen besteht aus einem Header, N Zellen und nach Bedarf aus Füllbits. Wieviele Zellen nun ein solcher Datenrahmen enthält, ist von der Bitrate des MAN abhängig. Bei einer Übertragungsrate von 34,368 Mbit/s ist ein 125 μs-Rahmen 4296 Bits lang, und kann somit zehn 53-Byte-Zellen transportieren.

Um eine Zelle übertragen zu können, muß eine Station wissen, ob sich das Ziel links oder rechts von ihr befindet. Liegt das Ziel rechts, benutzt die Station Bus A, andernfalls Bus B. Daten werden über eine spezielle Schaltung auf einen Bus eingefügt, so daß der Ausfall einer Station nicht den Ausfall des gesamten Netzes bewirkt.

Im Gegensatz zu den anderen 802 Protokollen ist DQDB nicht gierig. Hier reihen sich die Stationen in eine Warteschlange in der Reihenfolge ihrer Anfrage ein, d. h. sie senden und empfangen nach der FIFO-Reihenfolge. Um eine FIFO-Warteschlange zu simulieren, besitzt jede Station zwei Zähler, RC (Request Counter) und CC (Countdown Counter). Will nun eine Station Daten in eine Richtung übertragen (Bus A), so muß sie dazu zunächst auf dem jeweiligen anderen Bus (B), eine Reservierung vornehmen. Dies geschieht dadurch, daß die Station auf Bus B, auf einen nicht reservierten Datenrahmen wartet und diesen reserviert. Jede weiter stromabwärts liegende Station erhöht nun wenn sie diesen Rahmen liest ihren RC. Die sendewillige Station kopiert nun den Wert des RC in den CC, anschließend wird RC auf Null gesetzt. Der CC greift auf den Bus A zu, und wird beim Einlaufen jedes freien Rahmen erniedrigt. Ist der CC bei Null angelangt, so ist jener Rahmen der anschließend die Station passiert, derjenige, der zuvor reserviert wurde. Die Station kann nun diesen mit Datenzellen füllen.

Einen Nachteil besitzt DQDB bezüglich der Fairneß. Die Stationen die im mittleren Abschnitt des Busses liegen sind gegenüber den am Ende liegenden Stationen benachteiligt, denn sie erhalten im Mittel am wenigsten freie Datenrahmen.

Da DQDB-Zellen dieselbe Länge und einen ähnlichen Aufbau wie ATM-Zellen besitzen, ist eine Migration zu ATM möglich. Abbildung 14 zeigt die beiden Zellen im Vergleich.

1.4.5 SONET / SDH

In den Anfängen der Glasfasertechnik hatte jede Telefongesellschaft ihr eigenes optisches Übertragungssystem. 1985 begann man einen Standard namens SONET (Synchronous Optical NETwork) auszuarbeiten. Später trat CCITT in die Bemühungen mit ein, was 1989 zu einem SONET-Standard und mehreren parallelen CCITT-Empfehlungen führte. Diese CCITT-Empfehlungsreihe (G.707, G.708 und G.709) heißt SDH (Synchronous Digital Hierarchy) und unterscheidet sich von SONET nur geringfügig, deshalb ist im Folgenden, bei Erwähnung von SONET, auch immer SDH gemeint. Bei SONET wird die gesamte Bandbreite des Glasfaserkabels einem Kanal mit Zeitschlitzen für die verschiedenen Unterkanäle zugeordnet. Als solches ist SONET ein synchrones System. Es wird von einem Mastertakt mit einer Genauigkeit von etwa einem Teil in 10^9 gesteuert. Über eine SONET-Leitung werden Bits in sehr genauen durch den Mastertakt gesteuerten Intervallen übertragen. Das SONET-System besteht aus

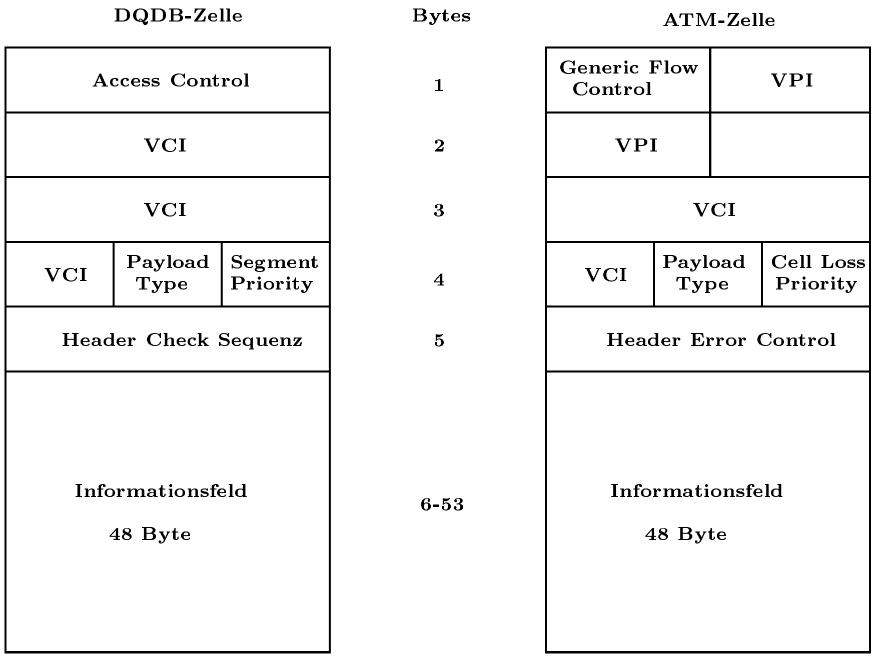


Abbildung 14: DQDB-Zelle und ATM-Zelle im Vergleich

Vermittlern, Multiplexern und Repeatern, die alle über Glasfaser miteinander verbunden sind. SONET kann eine Maschentopologie aufweisen, wird aber meist als Dualring konfiguriert. Der SONET-Basisrahmen ist ein Block von 810 Byte alle 125 μs. Da SONET synchron ist, werden Rahmen ungeachtet dessen ausgegeben, ob es Nutzdaten zu senden gibt oder nicht. Die Bruttodatenrate beträgt 51,84 Mbit/s, dies entspricht dem SONET-Basiskanal namens STS-1 (Synchronous Transport Signal-1). Alle SONET-Bündel sind ein Vielfaches von STS-1. Die Abbildung 15 zeigt den Multiplexvorgang bei SONET.

Das Multiplexen mehrerer Datenströme, die hier als Teilströme bezeichnet werden, spielt in SONET eine wichtige Rolle. Links beginnt das Beispiel mit verschiedenen Eingabeströmen (T1 und T3) in niedrigen Geschwindigkeiten, die auf die Grundrate STS-1 konvertiert werden. Dann werden drei STS-1-Teilbitströme auf einen STS-3-Ausgabestrom mit 155,52 Mbit/s gemultiplext, der die 12fache Kapazität des STS-1-Stroms hat. An diesem Punkt wird das Signal zerlegt, um zu verhindern, daß lange Null- oder Einserreihen die Taktgabe stören. Schließlich wird das elektrische Signal in ein optisches umgewandelt.

Das Multiplexen erfolgt Byte für Byte. Wenn beispielsweise drei STS-1-Teilbitströme zu 51,84 Mbit/s in einen STS-3-Strom zu 155,52 Mbit/s zusammengefügt werden, gibt der Multiplexer zuerst ein Byte von Teilbitstrom 1, dann eines von Teilbitstrom 2 und schließlich eines von Teilbitstrom 3 aus, dann macht er wieder mit dem ersten Strom weiter.

Die SONET-Multiplexhierarchie geht aus Tabelle 3 hervor. Definiert sind Raten von STS-1 bis STS-48. Der optische Träger, der STS-n entspricht, heißt OC-n (Optical Carrier), ist aber abgesehen von dem in Abbildung 15 aufgezeigten Scrambling Bit für Bit gleich. Die SDH-Bezeichnungen unterscheiden sich, da sie erst mit OC-3 beginnen, denn CCITT-basierte Systeme haben keine Raten die an 51,84 Mbit/s herankommen.

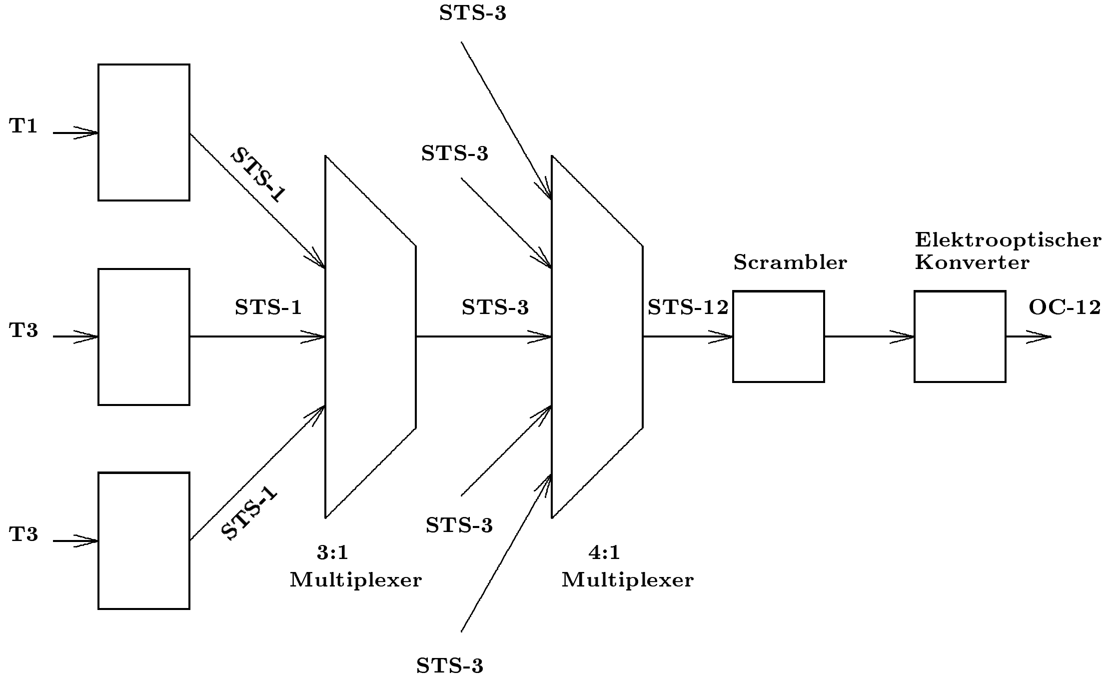


Abbildung 15: Multiplexvorgang in SONET

Sonet Elektrisch	*Sonet Optisch*	*SDH Optisch*	*Bruttodatenrate in Mbit/s*
STS-1	**OC-1**		**51,84**
STS-3	**OC-3**	**STM-1**	**155,52**
STS-9	**OC-9**	**STM-3**	**466,56**
STS-12	**OC-12**	**STM-4**	**622,08**
STS-18	**OC-18**	**STM-6**	**933,12**
STS-24	**OC-24**	**STM-8**	**1244,16**
STS-36	**OC-36**	**STM-12**	**1866,24**
STS-48	**OC-48**	**STM-16**	**2488,32**

Tabelle 3: SONET-Multiplexheriarchie

Der Hauptvorteil von SONET / SDH liegt im transparenten Multiplexverfahren. Das heißt, daß ein 64Kbit/s Kanal direkt aus der höchsten SDH-Multiplexhierarchie von derzeit 2,488 Gbit/s entkoppelt werden kann. Man spricht deshalb auch vom Ein-Stufen-Multiplexing innerhalb von SDH-Netzwerken. Schließlich kann SDH von Übertragungsdiensten wie ATM als physikalisches Transportmedium genutzt werden.

1.4.6 Vergleich in Tabellenform

In der Tabelle 4 werden alle besprochenen Datenübertragungstechniken miteinander verglichen.

Produkt	*LAN*	*MAN*	*WAN*	*Grenz-durchsatz in Mbit/s*	*Effizienz bei schwankenden Paketströmen?*	*Sonstige Vorteile*
Ethernet	**ja**	**nein**	**nein**	**10/100**	**ja**	**kostengünstig**
Token-Bus	**ja**	**nein**	**nein**	**10**	**nein**	**Bei Vollast-effizient**
Token-Ring	**ja**	**nein**	**nein**	**10**	**nein**	**Bei Vollast-effizient**
FDDI	**ja**	**ja**	**nein**	**100/200**	**ja**	**billige Anschlüsse**
SDH	**nein**	**ja**	**ja**	**2488**	**nein**	**wenig Overhead**
DQDB	**nein**	**ja**	**ja**	**2048**	**ja**	**Brückentechnologie**
ATM	**ja**	**ja**	**ja**	**2488**	**ja**	**Universell**

Tabelle 4: Datenübertragungstechniken im Vergleich

1.5 Zusammenfassung und Gründe für den Einsatz von ATM

ATM ist in der Lage, eine Reihe von Anforderungen zu erfüllen, die heutzutage und auch zukünftig an Netzwerke gestellt werden.

Folgende Gründe sprechen für den Einsatz von **ATM** :

- Zum einen basiert **ATM** auf weltweiten Standards, was ein Zusammenarbeiten einzelner Netze, die sich an diese Standards halten, gewährleistet.
- Die historische Unterscheidung in Lokale und Weitverkehrs-Datenübertragung und die dadurch resultierende Netzwerkkomplexität mit einer Vielzahl von Netzwerkskomponenten (Brücken, Router) wird durch **ATM** beendet . Mit **ATM** kann ein nahezu grenzenloses Informationsnetzwerk mit nahtlosen Übergängen zwischen lokalen Unternehmensnetzwerken und weltweiten Verbindungsnetzwerken entstehen.
- Heute werden die unterschiedlichen Datendienste wie Sprache, Daten und Video in der Regel noch in getrennten Netzwerken übertragen. **ATM** ist die einzige standardisierte Technologie, die in der Lage ist, die gleichzeitige Übertragung aller Datendienste in effizienter Weise (schnelles Routing auf Ebene 1) durchzuführen.
- Da **ATM** skalierbar ist und Geschwindigkeitsklassen von wenigen Mbit/s bis hin zu Gbit/s anbietet, wird diese Technologie in der Lage sein, die Anforderungen an die Datenkommunikation bis ins nächste Jahrtausend zu erfüllen.

ATM hat aufgrund dieser Eigenschaften sicherlich gute Chancen sich als Netz der Zukunft zu etablieren. So ist es für den Einsatz im MAN und WAN-Bereich hervorragend geeignet. Es müssen allerdings noch erhebliche finanzielle Mittel freigesetzt werden, um ein flächendeckendes **ATM** -Netz aufzubauen. Genau diese Kosten sind es auch, die derzeit eine weite Verbreitung im LAN-Bereich behindern. Ein **ATM** -Anschluß für den privaten Endbenutzer ist derzeit leider immer noch nicht finanzierbar bzw. wird noch überhaupt nicht von der Telekom angeboten.

2 Die ATM-Schicht

Dieses Kapitel behandelt die ATM-Schicht. Sie ist die Schlüsselschicht im ATM-Netzwerk, da sie die Zellstruktur definiert und die Verbindungen aufbaut. In ihr werden die Netzauslastung und die Einhaltung der Übertragungsvereinbarungen überwacht. Sie erkennt falsche oder illegale Zellen und kann diese zur Räson bringen. Die innovative Idee des ATM-Layer-Protokolls liegt im Verbindungsaufbau durch virtuelle Channels und virtuelle Pfade, die eine schnelle Verbindung zwischen den Teilnehmern ermöglichen.

2.1 Zellformat

Eine ATM-Zelle besteht aus 53 Bytes, die sich aus 5 Byte Headerinformationen und 48 Byte Nutzlast, Daten oder Information zusammensetzt. Diese Größe ist ein Kompromiß aus den Vorteilen für sehr datenintensive Anwendungen, wie Video, die bei größerer Bytezahl weniger Overhead transportieren müssen und 'Telnet'-Anwendungen die bei geringerer Größe wenig ungenutzte Information mitschleppen müssen.

Im Folgenden soll das Format des Zellheaders vorgestellt werden. Es gibt zwei leicht voneinander abweichende Typen, den Zellheader für Verbindungen im Netzwerk (NNI, Network Node Interface) und für Verbindungen zwischen Netzwerk und End-User (UNI, User Network Interface).

Der UNI Header besteht aus:

1. Datenflußkontrollfeld (4 Bit)
2. Adressierungsfeld (24 Bit) (8 Bit Kanalidentifikation und 16 Bit Pfadidentifikation)
3. Zellverlustprioritätsfeld (1 Bit)
4. Nutzlastidentifikationsfeld (3 Bit)
5. Header Prüfsummenfeld (8 Bit)

Das Datenflußkontrollfeld soll zur Steuerung lokaler Funktionen, wie zum Beispiel die Regelung von Zugriffs- und Übertragungsrechten verwendet werden. Es ist derzeit jedoch noch nicht standardisiert.

Das Adressierungsfeld spezifiziert die virtuellen Pfade und Kanäle zu denen die aktuelle Zelle gehört.

Ein virtueller Kanal bezeichnet die Verbindung zwischen Sender und Empfänger, die sich ihren Weg durch verschiedene Netzwerkknotenpunkte bahnt. Ein virtueller Pfad bündelt Kanäle die zwischen einem oder mehreren Verbindungspunkten den gleichen Weg haben. So können virtuelle Netzwerke verwirklicht werden, indem man einen entfernten Rechner einfach über einen virtuellen Kanal, ohne die faktischen Topologien beachten zu müssen, an das lokale Netz anschließt.

Das Nutzlastidentifikationsfeld (PT, Payload Type) identifiziert den Zellentyp. So können Benutzerzellen von Nicht-Benutzerzellen unterschieden werden. Letztere können zum Beispiel OAM-Zellen sein, die für das ATM-Management verwendet werden.

Mit dem Zell-Verlust-Prioritätsfeld kann zwischen 'legalen' und 'illegalen' Zellen unterschieden werden. Illegale Zellen sind solche die dem Verkehrs-Vertrag zuwider laufen. Daten, die trotz vereinbarter schmalerer Bandbreite im Vertrag noch bei Überkapazität übertragen werden, haben Priorität 1. Sie werden bei Engpässen zuerst und ohne Vorwarnung verworfen.

Aufbau des UNI Zellenheaders

8	7	6	5	4	3	2	1	Bit/Byte
Datenflußkontrolle (GFC)				Kanalidentifikation (VPI)				1
Kanalidentifikation (VPI)				Pfadidentifikation (VCI)				2
Pfadidentifikation (VCI)								3
Pfadidentifikation (VCI)				PT			CLP	4
Headerprüfsumme (HEC)								5

Tabelle 5: Die ATM Zelle (UNI)

Im Header-Prüfsummen-Feld (HEC) steht eine über die gesamte Zelle errechnete Prüfsumme. Sie wird zur Synchronisation einer ATM-Zelle aus dem Datenstrom verwendet.

Zellen mit NNI-Headern sind denen mit UNI-Headern sehr ähnlich, jedoch weisen sie nicht das für lokale Funktionen nötige GFC-Feld auf. Dessen Platz wird nun vom Adressierungsfeld (VPI) mitbelegt.

Es gibt noch andere spezielle Header, die sich durch spezifizierte Werte im PT-Feld oder Adressierungsfeld auszeichnen:

Idle Zellen sind im System mitfließende Zellen, die den Datenfluß auffüllen, bis die vorgegebene physikalische Bandbreite belegt ist. Sie gelangen nicht einmal an die ATM-Schicht, erkannt werden sie durch Nullwerte im Adressierungsfeld und einer 1 im CLP-Feld.

Im Gegensatz dazu gibt es auch noch Leerzellen, die VPI- und VCI-Werte besitzen, jedoch keinen Informationsgehalt besitzen. Sie werden bis in die ATM-Schicht durchgereicht und gehören auch zu einem bestimmten virtuellen Pfad und Kanal.

In der physikalischen Ebene gibt es auch noch einige besondere Zellen, die durch eigene Belegungen im Zellheader ausgewiesen werden, sie werden für OAM Zwecke verwendet:

- Übertragung von Benutzerdaten
- Metasignalisierung
- Broadband Signalisierung
- VC OAM Zellen
- SMDS/CBDS Zellen
- ILMI Zellen (Interim local management Interface Specification) (Verbindungen zwischen Enduser und Netzwerk, bzw. zwischen öffentlichem und privatem Netzwerk)

2.2 Synchronisation

Die Synchronisation versucht aus einem laufendem Bitstrom einzelne Zellen zu erkennen, und sich dann auf den Zellenstrom zu synchronisieren. Dazu sucht sie nach bekannten Mustern, hier dem Headerfeld (HEC), das eine Prüfsumme über den gesamten Zellheader beinhaltet.

Die Synchronisation des Bitstroms auf den Zellenanfang funktioniert über drei Zustände:

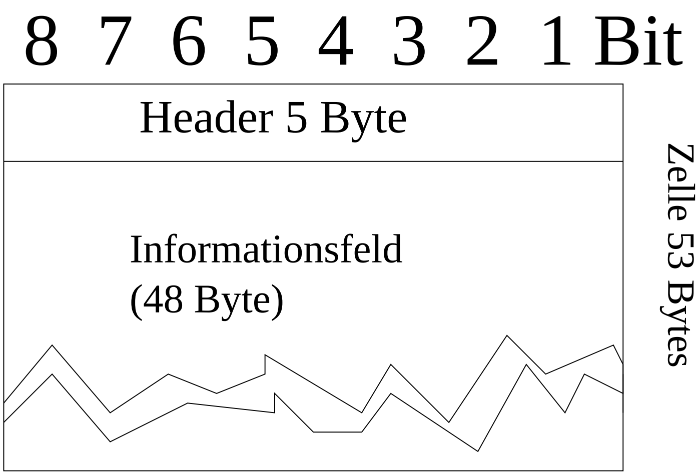


Abbildung 16: Die ATM Zelle

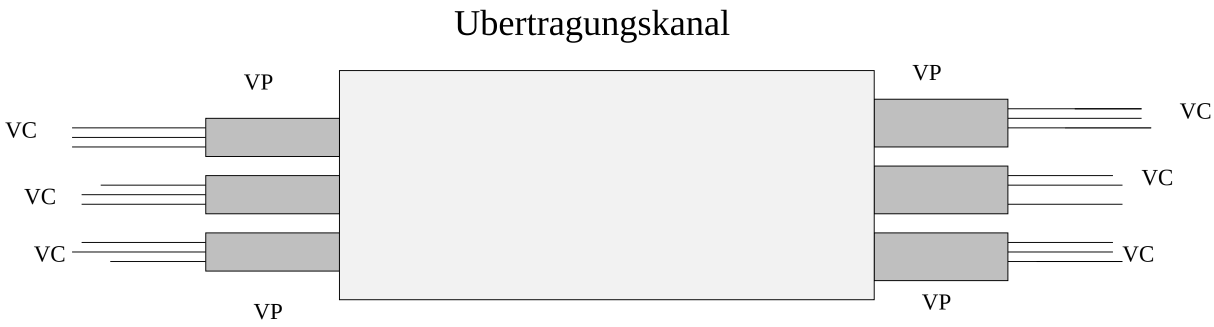


Abbildung 17: Übertragungskanal

- Hunt: Testet ob ein HEC-Muster in den nächsten 8 Bit erkennbar ist, d.h. ob das Muster zur aktuell angenommenen Zelle paßt.
- Presync: Nach gefundenem Zellenanfang im 'Hunt' Stadium werden die nächsten sechs Zellen betrachtet und bei richtigen Prüfsummen wird in den 'Synchronisations-Zustand' gewechselt. Andernfalls fällt das System in das 'Hunt' Stadium zurück.
- Sync: Synchronisation gilt als verloren, sobald sieben Zellen falsche Prüfsummen aufweisen. Auch hier wird in den 'Hunt'-Zustand geschaltet.

Der Informationsteil der ATM-Zelle wird verschlüsselt übertragen, um das ungünstige Auftreten gleicher aufeinanderfolgender Bits zu vermeiden. So wird die Payload im HUNT Zustand nicht entschlüsselt, um die HEC Felder besser erkennen zu können.

2.3 Reservierte Headerbelegungen

Zur Organisation des Zellenstroms, also um z.B. Verbindungen aufzubauen oder diese zu überwachen, werden spezielle Zellen verwendet. Um diese leicht erkennen zu können, werden vorbelegte, reservierte Header verwendet.

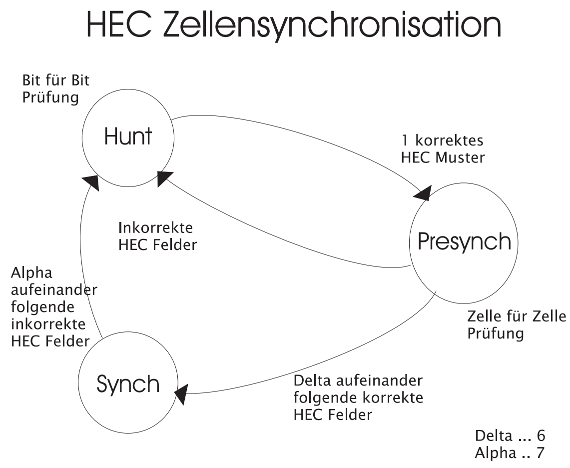


Abbildung 18: Zellensynchronisation

- Die ersten 16 VCI-Werte sind für spezielle Funktionen reserviert.
- Auch drei Werte des PT-Feldes sind indiziert, sie beschreiben OAM-Zellen.

2.4 Traffic Shaping und Policing

Die Verfahren Traffic Shaping und Policing dienen dazu, das ATM Netz optimal auszulasten. Es werden nur soviele Verbindungen von einem ATM-Switch angenommen, wie Ressourcen zur Verfügung stehen. Dabei kann durch das Multiplexing mehr Daten übertragen werden, als die Summe der Peaks der einzelnen Verbindungen erlauben würden. Maxima und Minima verschiedener Kanäle werden geschickt anhand geeigneter Algorithmen gegeneinander aufgerechnet. Die Funktionalität dieses Mechanismuß müssen die in den einzelnen ATM-Switches eingesetzten Algorithmen garantieren können.

Es gibt zwei unterschiedliche Arten zur Regelung der Netzlast. Erstens die 'Regelung bei der Verbindungsaufnahme' (CAC, Connection Admission Control) und zweitens die 'Steuerung der Benutzerkenngrößen' (UPC, Usage Parameter Control).

Sinn und Zweck ist es, Verbindungen zu erkennen, die den 'Traffic Contract' brechen.

Im Traffic Contract wird die maximale Verkehrslast ausgehandelt. Für die ausgehandelten Parameter muß das Netzwerk ausreichend Ressourcen zur Verfügung stellen können. Um Benutzer zu identifizieren, die im Begriff sind, den Kontrakt zu brechen, gibt es den 'Generic Cell Rate Algorithm', dessen Umsetzung bekannt ist als 'Leaky Bucket':

Bei Ankunft einer Zelle berechnet der Algorithmus anhand der Verkehrslastparameter, wann frühestens die nächste Zelle eintreffen darf. Kommt die Zelle zu früh, wird sie als nicht konform gekennzeichnet und kann so bei Bedarf gelöscht werden.

Die 'Connection Admission Control' CAC entscheidet über die Aufnahme oder die Ablehnung einer neuen ATM-Verbindung anhand der zur Vefügung stehenden Netz-Ressourcen. Dazu tauschen Benutzer und Netzwerk folgende Informationen aus:

Reservierte Zellenbytes in der ATM-Spezifikation									
Funktion	Pfadidentifikation		Kanalidentifikation				PT/CLP		VCI-Werte
Leerzellen	0000	0000	0000	0000	0000	0000	0000	xxx0	VCI=0
Metasignalisierung	0000	yyyy	yyyy	0000	0000	0000	0001	0a0c	VCI=1
Broadcast Signalisierung	0000	yyyy	yyyy	0000	0000	0000	0010	0aac	VCI=2
Punkt zu Punkt Sig.	0000	yyyy	yyyy	0000	0000	0000	0101	0aac	VCI=5
Ungültige Sequence	xxxx	0000	0000	0000	0000	0000	0000	xxx1	VCI=0
ILMI Message	0000	0000	0000	0000	0000	0001	0000	aaa0	VCI=16
OAM F4 Segment Zelle	0000	aaaa	aaaa	0000	0000	0000	0011	0a0a	VCI=3
OAM F4 EndtoEnd Zelle	0000	aaaa	aaaa	0000	0000	0000	0100	0a0a	VCI=4

- a .. dieses Bit steht für die Signalisierungsschicht zur Verfügung
- x .. don't care bits
- y .. jede VPI außer 0000 0000
- c .. Sendestation soll CLP Bit auf 0 setzen.

Tabelle 6: Reservierte Zellenbytes

- bestimmte Beschränkungen am Datenverkehrsvolumen, gemessen an Verkehrslastparametern
  - quantitativ: durchschnittliche oder maximale Zellrate
  - qualitativ: Quellenart (Telefon, Video)
- Dienstqualität bezüglich Übertragungsverzögerung, Verzögerungsschwankung, Zellverlust, Verlust von Burst-Zellen
- Toleranz für
  - die Abweichung der Zellverzögerung
  - die erwarteten Werte für den Durchsatz

CAC- Strategien sind nicht standardisiert und dem Netzwerk-Betreiber überlassen.

Die Schwierigkeit besteht darin, einen Mittelweg zu finden zwischen 'Überlast vermeiden' und 'Ausnutzung des Netzes'.

Die Auslastungs-Kontrolle (UPC, Usage Parameter Control) ist vorgesehen, um durch besondere Maßnahmen das Netz vor hoher Belastung, zum Beispiel durch Benutzer, die 'ihren' 'Traffic Kontrakt' brechen, zu schützen. Sie kann jede Verbindung in ihre ausgehandelten Schranken zwingen:

Es gibt drei Instanzen zur Regulation des Zellenstroms:

- Passing: Zellen verhalten sich konform zum Netzlastvertrag und werden durchgelassen.
- Tagging: Eine gewisse Menge der Zellen überschreit die der Verbindung zugewiesene Bandbreite, jedoch steht noch ungenutztes Potential zur Verfügung. Die Zellen werden für einen eventuellen Engpaß markiert.

- Discarding: Zellen werden sofort verworfen, wenn sie nicht konform zum Netzlastvertrag sind.

'Traffic Shaping' (Nutzlastformung) ist eine wichtige Funktion der ATM-Endpunkte, um einen gewünschten 'Quality of Service' zu erreichen. Bei Verwendung dieses Mechanismuß schon im Anfangspunkt eines Übertragungskanals kann eine gewünschte Charakteristik des Datenstroms erreicht werden.

Beispiele für 'Traffic Shaping' sind 'Peak Cell Rate Reduction' (Reduzierung der Spitzenzellrate), Beschränkung der Burstlängen, Reduzierung der Zellverzögerungsschwankung (Zellen werden so verzögert, daß die Reihenfolge der Zellen erhalten bleibt). 'Traffic Shaping' ist nur eine optionale Funktion. Es gibt keine spezifizierten Standardfunktionen dafür. Die Algorithmenauswahl steht dem Netzwerkbetreiber frei. Die Funktion wird bei der Parameterübergabe(Traffic Contract) mit ausgehandelt.

Hält sich ein Teilnehmer nicht an die im Netzlastvertag ausgehandelten Parameter, so kann die Zellverlustpriorität seiner Zellen (d.h. die Zellen, die seine Daten übertragen) von Null ('Alle Funktionen gemäß der Parameter') auf Eins ('Im Notfall zuerst löschen') gesetzt werden. Diese können dann bei Netzauslastung sofort verworfen werden.

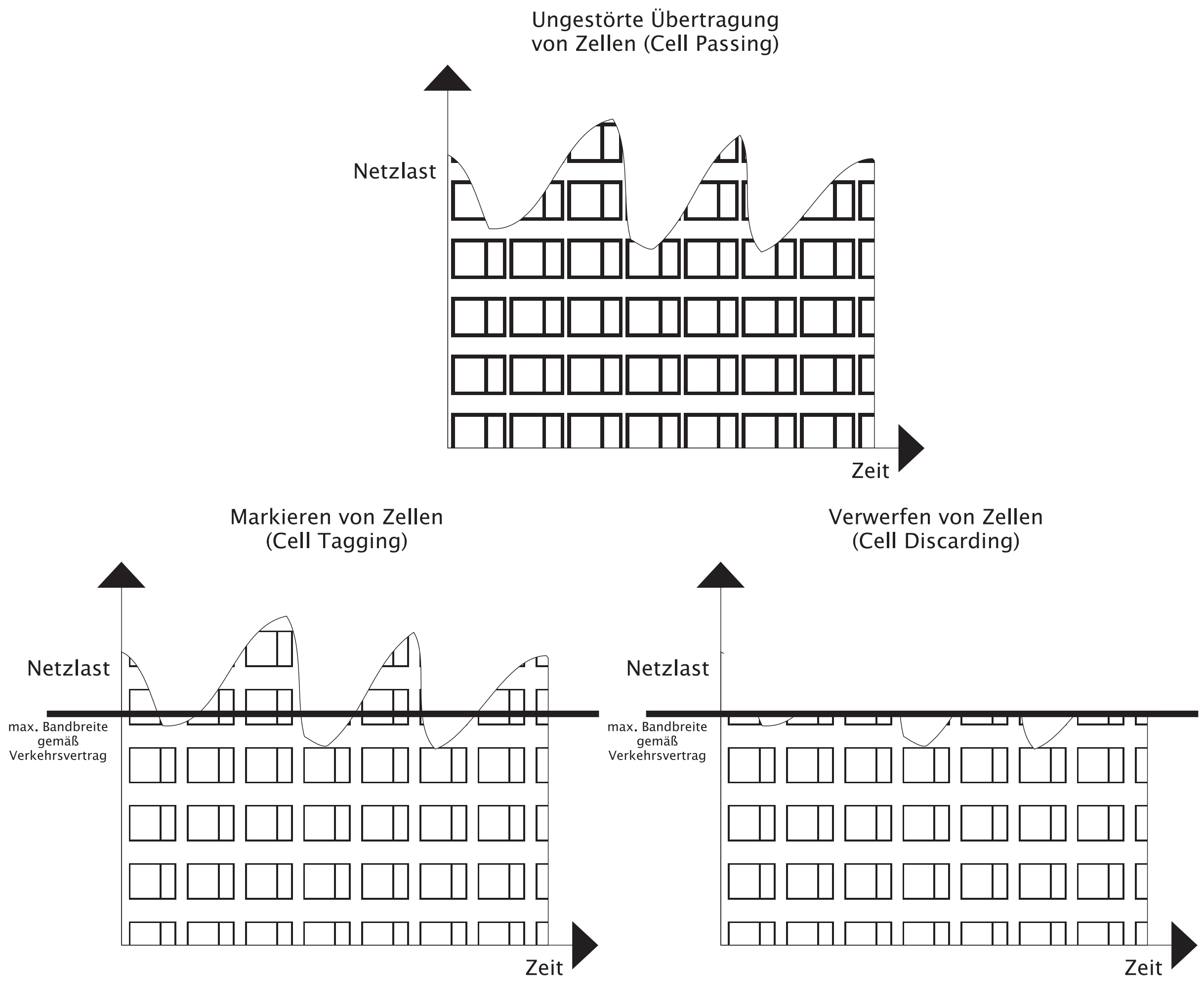


Abbildung 19: Usage Parameter Control

2.5 Serviceparameter

Jeder Verbindung im ATM-Netzwerk werden bei ihrem Aufbau Serviceparameter zugeordnet, welche die Eigenschaften der Verbindung näher spezifizieren. Von sieben Parametern zur Zellenübertragungsleistung sind drei als Serviceparameter vorgesehen:

- Zellenfehlerrate: (CER - Cell Error Ratio)
- Rate schwerwiegender Zellen-Block-Fehler
- Zellenverlustrate (CLR - Cell Loss Rate)
- Cell Missinsertion Rate (CMR)
- Zellenverzögerung (CDR)
- durchschnittliche Zellenverzögerung
- Schwankung der durchschnittlichen Zellenverzögerung (CDV, Cell Delay Variation)

Die Serviceparameter verhandeln die Netzwerkbenutzer mit dem Netzwerk im Netzlastabkommen (Traffic Contract). Nach Abschluß garantiert die Netzwerkseite die Einhaltung der Serviceparameter. Es gibt Quality of Service (QoS) Klassen mit und ohne Leistungsparameter. Klassen mit Serviceparametern weisen mindestens zwei auf, die für jeweils einen der beiden CLP-Werte steht. Für die in ITU I.362 definierten Serviceklassen, sind folgende QoS vorgesehen:

1. 'Video, Standleitung, konstante Bitrate' - 'QoS Klasse 1 für Klasse A Anforderungen'
2. 'A/V mit variabler Bandbreite' - 'QoS Klasse 2 für Klasse B Anforderungen'
3. 'Verbindungsorientierte DÜ' - 'QoS Klasse 3 für Klasse C Anforderungen'
4. 'Verbindungslose DÜ' - 'QoS Klasse 4 für Klasse D Anforderungen'

Klassen ohne Parameter lassen sich verbalisieren mit 'bestmögliche Nutzleistungsparameter', hier ist keine expliziter Netzwerkleistungsgarantie nötig.

2.6 Metasignalisierung

Signalisierung erfolgt in Signalisierungskanälen, welche von den Nutzkanälen getrennt sind. Aufgrund hoher Bandbreiten und unterschiedlichen Diensten ist ein flexibles Signalisierungskonzept nötig: Die Signalisierungskanäle werden durch die sogenannte Metasignalisierung eingerichtet. Das läßt neue Konzepte wie zum Beispiel 'Punkt zu Mehrpunkt' zu. Die Metasignalisierung erfolgt recht einfach. Die dazu nötigen Befehle werden in einer ATM-Zelle untergebracht, und über einen reservierten ATM-Kanal (VCI=0, VPI=0) versendet.

Aufgaben bei der Signalisierung:

- Aufbau und Abbau
- Überprüfung von Signalisierungskanälen

Dafür gibt es spezielle Befehle:

- assign request - remove request
- assigned - denied
- check request
- check response

2.7 Aufbau und Funktion von OAM Zellen

Den ATM-Management Informationsfluß übernehmen die OAM Zellen. Sie überwachen Netzwerkleistung und die Verfügbarkeit der virtuellen Pfade und Kanäle. Die Informationen regeln auf der Pfadebene das 'Segment'- oder 'End-to-End'-Management. (Segment= Verbindung zw. zwei Netzwerk-Knoten, 'End-to-End' = Enden eines einzelnen ATM-Pfades).

Auf der ATM-Schicht sind für Pfad- und Kanalverbindungen jeweils verschiedene ATM-OAM Zellen zuständig:

- F4 OAM Zellen: Sie dienen zur Überwachung von Netzwerkleistung und der Pfadverbindungen, und sind durch die reservierten VCI-Werte drei und vier ausgezeichnet, der VPI-Wert bleibt mit den übrigen Benutzerzellen gleich.
- F5 OAM Zellen: Sie überwachen die Netzwerkleistung, VCI- und VPI-Werte sind identisch mit denen des Zellenstroms des Kanals. Sie sind markiert durch das PT-Feld. Der Wert 100 steht für F5-Segment OAM-Zellen, 101 für F5 End to End OAM Zellen.
  - 'End to End' OAM Zellen passieren die zwischenliegenden Netzwerkknoten unverändert, sie werden erst von den Endstellen entfernt.
  - Segment OAM Zellen werden am Ende des jeweiligen Segments entfernt.

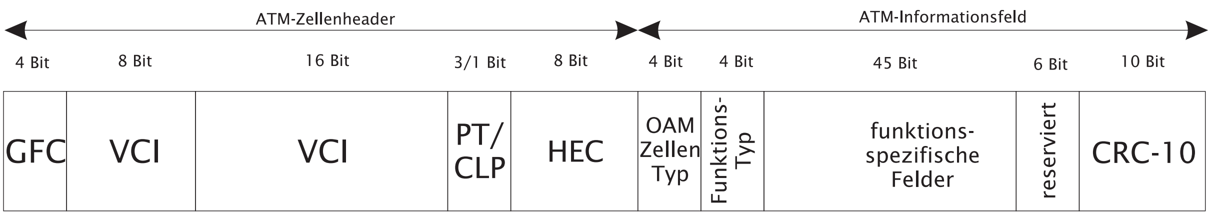


Abbildung 20: OAM-Zelle

2.7.1 Fehlermanagement

Es gibt zwei Arten von Alarmen:

1. Alarm ist aufgetreten (AIS - Alarm Indication Signal)
2. Empfänger meldet Signal (FERF - Far End Receive Failure)

AIS wird generiert um einem 'stromabwärts gelegenem Knoten einen Fehler anzuzeigen (VCC/VPC Verbindungsfehler, Fehler in physikalischer Schicht).

Nach Empfang eines AIS wird ein FERF 'stromaufwärts' geschickt, um stromabwärts liegende Fehler mitzuteilen.

- VP-Fehlermeldungen (AIS, FERF) werden mit VCI-Wert 4 spezifiziert.
- VC Fehlermeldungen (AIS, FERF) mit PT=101.

2.7.2 Leistungsmanagement

Durch geeignetes Einfügen von OAM Zellen können Rückschlüsse zur Leistungsmessung auf den Betriebszustand der ATM Verbindungen gezogen werden.

3 Die AAL-Schicht

3.1 Einleitung

Die ATM-Adaptions-Schicht hat die Funktion, die Datenströme der Höheren Schichten auf ATM-Zellen abzubilden und die dazugehörigen Steuer- und Managementfunktionen zur Verfügung zu stellen. Da die AAL-Schicht die unterschiedlichen Eigenschaften der Datenströme berücksichtigen muß, wurden die Datenströme ursprünglich in vier Dienstklassen eingeteilt. Für jede dieser Dienstklassen wurde von der ITU ein eigener AAL-Typ definiert. Man hat festgestellt, daß sich die AAL-Typen 3 (für verbindungsorientierten Datenverkehr) und 4 (für verbindungslosen Datenverkehr) sehr ähnlich sind und diese deshalb zum AAL-Typ 3/4 zusammengefaßt. Die Computerindustrie war mit den Definitionen der ITU nicht zufrieden und entwichelte ein eigenes AAL-Protokoll (SEAL, Simple Efficient Adaption Layer), das jetzt den AAL-Typ 5 bildet.

	Class A	Class B	Class C	Class D
Zeit-Kompensation	erforderlich		nicht erforderlich	
Bit Rate	konstant	variabel		
Verbindungsmodus	verbindungsorientiert			verbindungslos
AAL Typ	Typ 1	Typ 2	Typ 3,5	Typ 4

Abbildung 21: Dienstklassen und AAL-Typen

Inzwischen ist man von der Einteilung in Dienstklassen abgekommen und unterscheidet bei den Datenströmen zwischen den verschiedenen Verkehrsklassen (CBR, VBR, ABR und UBR).

Hier zusammengefaßt die wichtigsten Aufgaben der AAL-Schicht (nach [42])

- Abbildung der Nutzdaten auf ATM-Zellen
- Rückgewinnung der Nutzdaten (und ggf. Taktinformationen) aus ATM-Zellen
- AAL-Typ 1 für zeitkontinuierliche Daten mit fester Übermittlungsrate
- AAL-Typ 2 für kleine Dateneinheiten mit effektiven Übertragungen und geringerer Übermittlungsgeschwindigkeit
- AAL-Typ 3/4 für eine gesicherte, verbindungsorientierte Datenübertragung
- AAL-Typ 5 für eine einfache Form der Datenübertragung

3.2 AAL Teilschichten

Die AAL-Schicht besteht aus der Konvergenzteilschicht (Convergence Sublayer, CS) und der Segmentierungs/Reassemblierungsteilschicht (SAR). Wobei die Konvergenzteilschicht in einen gemeinsamen Teil (Common Part Convergence Sublayer, CPCS) und einen anwendungsspezifischen Teil (Service Specific Convergence Sublayer, SSCS) unterteilt ist. Einige dieser Teilschichten können auch leer sein.

AAL	SSCS Service Specific Convergence Sublayer
	CPCS Common Part Convergence Sublayer
	SAR Segmentation and Reassembly Sublayer

Einen Datenblock, den eine Teilschicht von einer übergeordneten Teilschicht erhält nennt man Service Data Unit, kurz SDU. Diese SDU kann dann mit einem Header und einem Trailer versehen werden. Der daraus resultierende Datenblock wird Protocol Data Unit (PDU) genannt.

3.3 AAL-Typ 1

Der AAL-Typ 1 dient zur Übertragung von Daten mit konstanter Bitrate wie etwa Leitungsschnittstellen oder Video/Audio-Daten mit konstanter Bitrate. Dabei wird sichergestellt, daß die AAL-Schicht auf der Empfängerseite die Daten mit derselben Frequenz weitergibt mit der sie der AAL-Schicht auf der Senderseite übergeben werden. Außerdem können zu den Daten Strukturinformationen übertragen werden.

3.3.1 Die Segmentierungs/Reassemblierungsteilschicht (SAR)

Für jede ATM-SDU werden folgende Operationen durchgeführt

Die SAR-Teilschicht auf der Senderseite erhält von der Konvergenzteilschicht einen 47 Byte langen Datenblock, versieht ihn mit einem ein Byte langen SAR-PDU-Header und übergibt diese 48 Byte lange SAR-PDU an die ATM Schicht.

Die SAR-Teilschicht auf der Empfängerseite entfernt den SAR-PDU-Header von dem 48 Byte langen Datenblock, den sie von der ATM Schicht erhält. Der daraus resultierende 47 Byte lange Datenblock wird an die Konvergenzteilschicht weitergegeben.

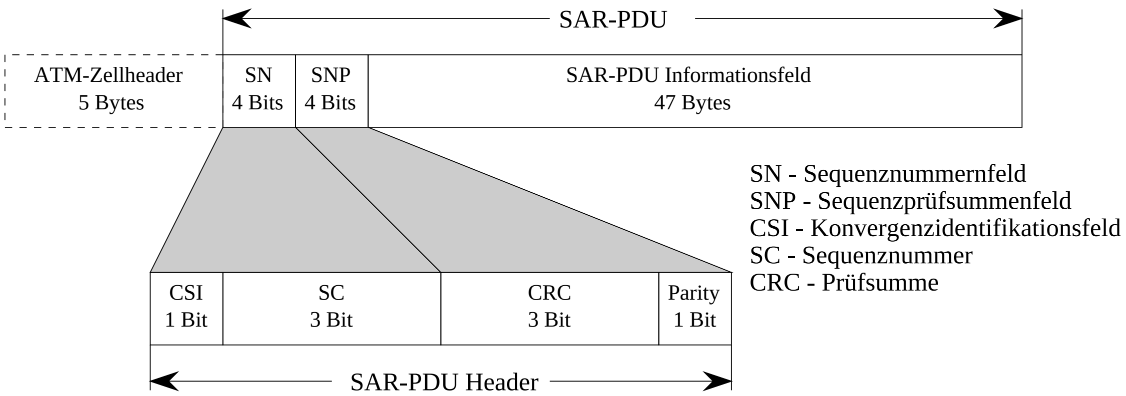


Abbildung 22: Aufbau einer AAL-1 SAR-PDU

Sequenznummernfeld Die SAR-Teilschicht erhält von der Konvergenzteilschicht zu jedem 47 Bytes langen Datenblock ein Sequenznummernfeld. Das Sequenznummernfeld besteht aus der eigentlichen Sequenznummer, um auf der Empfängerseite Zellverluste zu erkennen, und dem Konvergenzidentifikationsfeld, in dem unter anderem Taktinformationen oder Informationen zur Struktur der Sendedaten übertragen wird.

Sequenzprüfsummenfeld Die SAR-Teilschicht berechnet eine 3 Bit lange CRC über das Sequenznummernfeld. Die 7 Bit des Sequenznummernfeldes und der CRC wird zusätzlich mit einem Paritätsbit gesichert. Durch diese Verfahren soll eine fehlerfreie Übertragung des Sequenznummernfeldes gewährleistet werden.

3.3.2 Die Konvergenzteilschicht

Die Konvergenzteilschicht stellt sicher, daß die AAL-SDUs mit konstanter Bitrate an den Empfänger weitergeleitet werden. Außerdem stellt die Konvergenzteilschicht Verfahren zur Sendefrequenzrückgewinnung, zur Übertragung strukturierter Daten und einer Fehlerkorrektur der Benutzerdaten zur Verfügung.

Bei der Emulation von Leitungsschnittstellen haben die AAL-SDUs eine Größe von einem Bit, bei der Übertragung von Video- oder Audiodaten eine Größe von einem Byte.

Mit diesen Daten werden die Informationsfelder der CS-PDUs aufgefüllt. Die CS-PDUs werden dann in 47 Bytes langen Datenblöcken zusammen mit dem Sequenznummernfeld an die SAR-Teilschicht weitergegeben.

Daten, die zur Bereitstellung der Zusatzfunktionen nötig sind, werden mit Hilfe des seriellen Datenstroms, der aus den CSI-Bits aufeinanderfolgender SAR-PDUs gebildet wird, übertragen.

Sendefrequenzrückgewinnung

Synchronous residual time stamp (SRTS) Bei der SRTS-Methode wird die Frequenzdifferenz zwischen der Netzwerkdatenrate (network clock) und der Benutzerdatenrate (service clock) gemessen. Dabei wird vorausgesetzt, daß die feste Netzwerkdatenrate an der Sender- und Empfängerseite bekannt ist. Um diese Frequenzdifferenz mit ausreichender Genauigkeit zu repräsentieren, genügt es, alle 8 ATM-Zellen den 4-Bit langen Residual Time Stamp zu übertragen. Der RTS wird seriell in den CSI-Bits der SAR-PDUs mit den Sequenznummern 1, 3, 5 und 7 übertragen.

Adaptive Clock Method Bei diesem Verfahren wird auf der Empfängerseite ein Puffer eingerichtet. Die empfangenen Daten werden in diesen Puffer geschrieben und mit der lokalen Ausgabedatenrate gelesen. Die Füllhöhe des Puffers wird ständig überprüft. Ist sie zu hoch, wird die Ausgabedatenrate erhöht, ist sie zu niedrig wird die Ausgabedatenrate verringert. Im Gegensatz zum SRTS-Verfahren muß hier keine zusätzliche Information übertragen werden.

Übertragung strukturierter Daten

Bei der Übertragung von strukturierten Daten werden Pointer benutzt, um die Strukturgrenzen zu kennzeichnen. Dazu wird beim SAR-PDU Informationsfeld zwischen dem P-Format (mit Pointer) und dem non-P-Format (ohne Pointer) unterschieden. Nur SAR-PDUs mit den Sequenznummern 0,2,4 und 6 können einen Pointer enthalten, was durch das CSI-Bit im SAR-PDU-Header gekennzeichnet wird (Abbildung 22).

Das Offsetfeld enthält die Anzahl Bytes, die zwischen dem Ende des Pointerfeldes und dem Anfang des nächsten strukturierten Datenblocks innerhalb der nächsten 93 Informationsfeld-Bytes liegen. Die 93 Bytes bestehen aus den restlichen 46 Bytes der Pointer-PDU und den 47 Informationsfeld-Bytes der darauffolgenden Nicht-Pointer-PDU. Falls das Ende der 93 Bytes mit dem Ende eines strukturierten Datenblocks, der nicht in diesen 93 Bytes beginnt, zusammenfällt, enthält das Offsetfeld den Wert 93. Das Offsetfeld kann also Werte von 0 bis 93 annehmen.

Zusätzlich zur Strukturinformation kann eine Sendefrequenzrückgewinnung mittels SRTS erfolgen. Dabei wird für jede SAR-SDU mit den Sequenznummern 0, 2, 4 und 6 das Pointer-Format benutzt, um die Sendefrequenzrückgewinnung korrekt durchführen zu können. Falls das Offsetfeld nicht für die Strukturinformation benötigt wird, wird es mit 7 Füllbits mit dem Wert eins belegt.

Korrekturverhalten für Bitfehler und Zellverluste

Für die unidirektionale Übertragung von Videosignalen, für die die Zellverlust- und Bitfehlerraten der ATM-Schicht zu hoch sind, ist ein besonderes Fehlerkorrekturverfahren vorgesehen. Es besteht aus einer Kombination aus Reed-Solomon-Codierung und Byte-Umschichtung. Über 124 Datenbytes wird dabei ein vier Byte langer Reed-Solomon-Code berechnet. Diese 128 Byte langen Datenblöcke werden in eine Matrix mit 128 Spalten und 47 Zeilen eingetragen. Diese Matrix wird zeilenweise geschrieben und spaltenweise ausgelesen. Jede Spalte entspricht dem Informationsfeld einer SAR-PDU. Um den Beginn einer CS-PDU zu erkennen, wird das CSI Bit der ersten SAR-PDU auf den Wert 1 gesetzt.

In jeder CS-PDU können damit folgende Fehlerkombinationen korrigiert werden:

- 4 Zellverluste, oder
- 2 Zellverluste und ein fehlerhaftes Byte pro Zeile, oder
- 2 fehlerhafte Bytes pro Zeile, falls kein Zellverlust auftritt

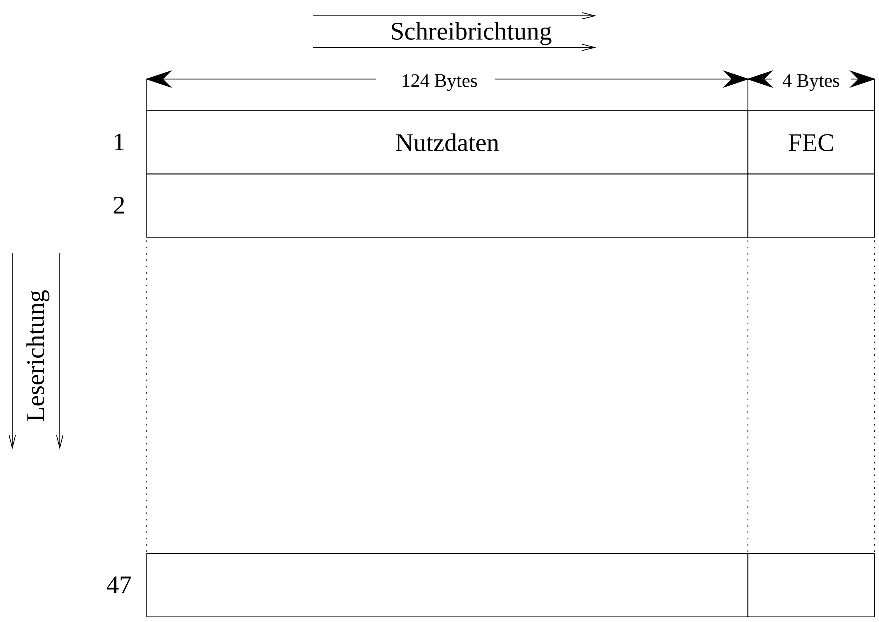


Abbildung 23: Matrix zur Byteumschichtung

3.4 AAL-Typ 2

Der AAL-Typ 2 dient zur Übertragung von Daten mit variabler Bitrate und Echtzeitanforderungen (Realtime Variable Bitrate, rt-VBR). Es werden also hohe Anforderungen an die Zellverzögerungszeiten und die Zellverzögerungsschwankungen gestellt. Beispiele hierfür sind komprimierte Audio- und Videodaten. Gelegentliche Zellverluste oder fehlerhaft übertragene Zellen können hingegen toleriert werden.

Bei AAL 2 ist die AAL-Schicht in ein Service Specific Convergence Sublayer (SSCS) und ein Common Part Sublayer (CPS) unterteilt. Eine SAR-Teilschicht ist nicht vorgesehen.

Die CPS-SDUs haben normalerweise eine Länge von 1 bis 45 Bytes, bei speziellen Anwendungen auch 64 Bytes. Aus den CPS-SDUs werden CPS-Pakete gebildet, indem sie mit einem 3 Bytes langen CPS-Paket-Header versehen werden.

Die CPS-Pakete werden mit CPS-PDUs transportiert. Das 47 Bytes lange Informationsfeld der CPS-PDUs wird dabei mit CPS-Paketen aufgefüllt. Ein CPS-Paket kann dabei auf mehrere CPS-PDUs aufgeteilt werden. Ist es nicht möglich, das ganze Informationsfeld mit Daten zu füllen, wird an die Daten ein PAD-Feld angehängt.

Die CPS-PDUs bestehen aus einem 1 Byte langen Startfeld (STF) und einem 47 Bytes langen Informationsfeld. Im Startfeld wird durch das 6 Bits lange Offsetfeld (OSF) angegeben, wieviele Bytes sich zwischen dem Offsetfeld und dem Anfang des ersten in dieser CPS-PDU übertragenen CPS-Paket befinden. Ein Wert von 47 zeigt an, daß kein Anfang eines CPS-Paketes mit dieser CPS-PDU übertragen wird. Das Startfeld enthält außerdem eine 1 Bit lange Sequenznummer und wird durch ein Parity-Bit gesichert.

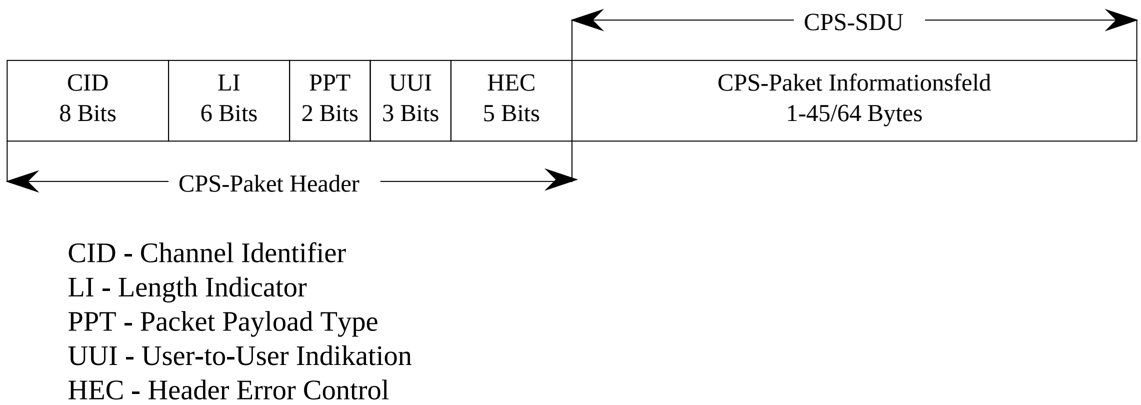


Abbildung 24: Aufbau eines AAL-2 CPS-Paketes

Channel Identifier (CID) Beim AAL-Typ 2 können mehrere AAL-Verbindungen dieselbe ATM-Verbindung nutzen. Jede AAL-Verbindung hat einen eigenen Channel Identifier, der beim Verbindungsaufbau dynamisch vergeben wird.

Length Indicator (LI) Das Längenfeld enthält die Länge des Informationsfeldes

Packet Payload Type (PPT) Bei Paketen mit OAM-Funktionen enthält dieses Feld den Wert 3. Alle anderen Werte kennzeichnen Pakete mit Nutzdaten im Informationsfeld.

User-to-User Indication (UUI) In diesem Feld können Zusatzinformationen (z.B. OAM-Informationen) zwischen den AAL-Benutzern ausgetauscht werden.

Header Error Control (HEC) Das HEC-Feld sichert die Informationen im CPS-Paket-Header.

3.5 AAL-Typ 3/4

Der AAL-Typ 3/4 dient zur verbindungsorientierten und verbindungslosen Übertragung von Datenpaketen variabler Länge.

Folgende Parameter werden unterschieden:

Streaming-Modus/Message-Modus Im Streaming-Modus werden bereits Datenpakete an die ATM-Schicht übergeben, bevor die vollständige AAL-SDU vom AAL-Benutzer übergeben wurde. Die Übertragung einer AAL-SDU kann auch abgebrochen werden. Dazu wurde in der SAR-Teilschicht eine spezielle Abbruch-PDU definiert.

garantierte / nicht garantierte Datenübertragung Bei der garantierten Datenübertragung werden verlorengegangene oder fehlerhaft übertragene AAL-SDUs erneut übertragen. Für die garantierte Datenübertragung ist eine SSCS-Teilschicht (z.B. SSCOP für die Signalisierung) notwendig.

Point-to-Point / Point-to-Multipoint Verbindungen Die Daten können bei AAL-3/4 von einem Sender zu einem oder mehreren Empfängern übertragen werden. Die Unterstützung von Point-to-Multipoint Verbindungen wird dabei vom CPCS bereitgestellt.

3.5.1 Die Segmentierungs/Reassemblierungsteilschicht (SAR)

Eine SAR-PDU besteht aus einem zwei Bytes langen Header, einem zwei Bytes langen Trailer und einem 44 Byte langen Informationsfeld.

Um den Abbruch der Übertragung einer SAR-SDU anzuzeigen, wird eine Abbruch-PDU gesendet. Ihr Segmenttypenfeld enthält den Wert EOM und ihr Längenfeld den Wert 63.

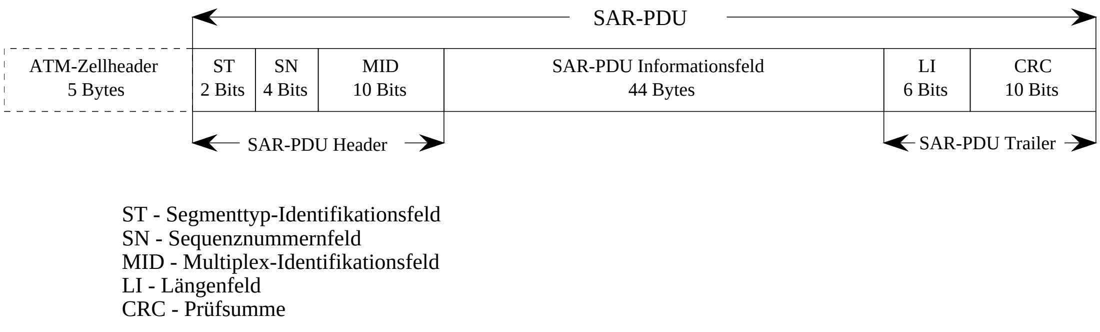


Abbildung 25: Aufbau einer AAL-3/4 SAR-PDU

Segmenttypenfeld Im Segmenttypenfeld wird angegeben, welchen Teil der SAR-SDU die betreffende SAR-PDU enthält.

Segmenttyp	Codierung	Bedeutung
BOM	10	Begin of Message
COM	00	Continuation of Message
EOM	01	End of Message
SSM	11	Single Segment Message

Sequenznummernfeld Mit Hilfe der Sequenznummern kann beim Empfänger ein Zellverlust erkannt werden. Alle SAR-PDUs einer SAR-SDU erhalten fortlaufende Sequenznummern modulo 16.

Multiplex-Identifikationsfeld Wenn ein Sender mehrere AAL-Verbindungen zu einem Empfänger unterhält, ist es bei AAL 3/4 möglich, daß diese AAL-Verbindungen dieselbe ATM-Verbindung nutzen. Alle SAR-PDUs einer Verbindung enthalten dabei denselben MID-Wert.

Längenfeld Das Längenfeld gibt an, wieviele SAR-SDU-Bytes sich im Informationsfeld der SAR-PDU befinden.

Segmenttyp	Gültige Werte
BOM	44
COM	44
EOM	4 .. 44, 63
SSM	8..44

Prüfsummenfeld Über den SAR-PDU-Header, das Informationsfeld und das Längenfeld wird eine 10 Bit lange CRC berechnet.

3.5.2 Die Konvergenzteilschicht

Bei AAL 3/4 ist die Konvergenzteilschicht in einen gemeinsamen Teil (CPCS Common Part Convergence Sublayer) und einen anwendungsspezifischen Teil (SSCS Service Specific Convergence Sublayer) unterteilt. Für die verbindungslose Datenübertragung (Dienstklasse C, z.B. CL-NAP) ist eine SSCS-Teilschicht nicht notwendig, bei verbindungsorientierter Datenübertragung (Dienstklasse D, z.B. Frame Relay) wird sie benötigt.

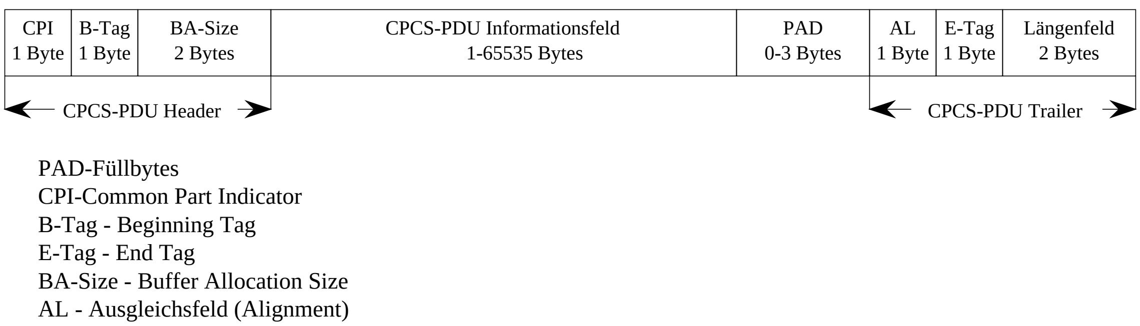


Abbildung 26: Aufbau einer AAL-3/4 CPCS-PDU

CPI Der Common Part Indicator kann zur Leistungs- und Fehlerüberwachung, zur MID-Festlegung, zur Kennzeichnung von OAM-Nachrichten verwendet werden oder weitere Hinweise zur Interpretation der restlichen Felder der CPCS-Dateneinheiten enthalten.

B-Tag und E-Tag Die Werte des B-Tag und des E-Tag einer CPCS-PDU sind identisch. Die Werte in aufeinanderfolgenden CPCS-PDUs müssen unterschiedlich sein.

BA-Size Das BA-Size-Feld teilt der Empfangsstation mit, welche Pufferspeichergröße für den Empfang der CPCS-SDU benötigt wird. Im Message-Mode entspricht das der Länge der CPCS-PDU. Im Streaming-Mode kann es auch größere Werte annehmen, da hier eine SDU mittels mehrerer PDUs übertragen werden kann.

AL Dieses Ausgleichsfeld enthält keine Information und wird nur benutzt, um die Länge des CPCS-Trailers auf 32 Bits aufzufüllen.

Längenfeld Das Längenfeld enthält die Länge des CPCS-PDU-Informationsfeldes.

3.6 AAL-Typ 5

AAL-5 ist eine Vereinfachung des AAL-Typs 3/4. Der AAL-Typ-5 wurde im Gegensatz zu den übrigen AAL-Typen nicht von der ITU-T, sondern von der Industrie entwickelt. Dabei wurde besonderer Wert darauf gelegt, den Protokoll-Overhead möglichst gering zu halten. Dazu wurden die meisten Aufgaben, die bei AAL 3/4 von der SAR-Teilschicht übernommen werden, in die Konvergenzteilschicht, die in CPCS und SSCS aufgeteilt ist, verlegt.

Auch bei AAL-5 wird zwischen Message-Modus und Streaming-Modus, garantierter und nicht garantierter Übertragung, Point-to-Point und Point-to-Multipoint Verbindungen unterschieden. Die Definitionen entsprechen denen bei AAL 3/4. Allerdings werden Point-to-Multipoint Verbindungen bei AAL 5 durch das SSCS und nicht, wie bei AAL 3/4, durch das CPCS unterstützt. Bei AAL 5 ist es, wie bei AAL 3/4, möglich , daß mehrere AAL-Verbindungen dieselbe ATM-Verbindung nutzen. Allerdings findet bei AAL 5 das Multiplexen nicht auf SAR-PDU-Ebene sondern auf CPCS-PDU-Ebene statt und für das Multiplexen wird eine SSCS-Teilschicht benötigt.

3.6.1 Die Segmentierungs/Reassemblierungsteilschicht (SAR)

Die AAL-5-SAR-Teilschicht kann nur SAR-SDUs verarbeiten, deren Länge ein ganzzahliges Vielfaches von 48 Bytes ist. Es werden keine Header oder Trailer an die Daten angefügt. Die AAL-5-SAR-Teilschicht benutzt das AUU-Bit im ATM-Header, mit dem Benutzer der ATM-Schicht Zusatzinformationen austauschen können, um die letzte SAR-PDU einer SAR-SDU zu kennzeichnen.

3.6.2 Die Konvergenzteilschicht

Die Konvergenzteilschicht besteht aus einem gemeinsamen Teil (CPCS) und einem anwendungsspezifischen Teil (SSCS). Eine AAL5-CPCS-PDU enthält keinen Header, sondern nur einen Trailer.

PAD-Feld Die CPCS-PDU-Länge muß ein ganzzahliges Vielfaches von 48 Bytes sein. Dazu werden 0 bis 47 Füllbytes eingefügt. Die Werte der Füllbytes können beliebig gewählt werden und werden auf der Empfängerseite nur zur Berechnung der CRC benutzt.

CPCS-PDU Informationsfeld 1-65535 Bytes	PAD 0-47 Bytes	CPCS-UU 1 Byte	CPI 1 Byte	Längenfeld 2 Bytes	CRC 4 Bytes
		CPCS-PDU Trailer			

PAD - Füllbytes
CPCS-UU - CPCS User-to-User-Indication
CPI - Common Part Indicator
CRC - Prüfsummenfeld

Abbildung 27: Aufbau einer AAL-5 CPCS-PDU

CPCS-UU Dieses Feld enthält die ein Byte lange Benutzer-Benutzer-Information. Die CPCS-Teilschicht auf der Senderseite erhält es zusammen mit der CPCS-SDU vom CPCS-Nutzer (also von der SSCS-Teilschicht oder direkt vom AAL-Nutzer) und die CPCS-Teilschicht auf der Empfängerseite leitet es an den dortigen CPCS-Nutzer weiter.

CPI Das CPI-Feld hat derzeit nur die Funktion, den CPCS-Trailer auf ein vielfaches von vier Bytes zu ergänzen.

Längenfeld Das Längenfeld enthält die Länge des CPCS-PDU-Informationsfeldes in Bytes. Ein Wert von Null signalisiert den Abbruch einer laufenden CPCS-SDU-Übertragung.

CRC-Feld Zur Überwachung von Bitfehlern wird über die gesamte CPCS-PDU (also das Informationsfeld, das PAD-Feld und die ersten vier Bytes des CPCS-Trailers) eine CRC-32-Prüfsumme gebildet.

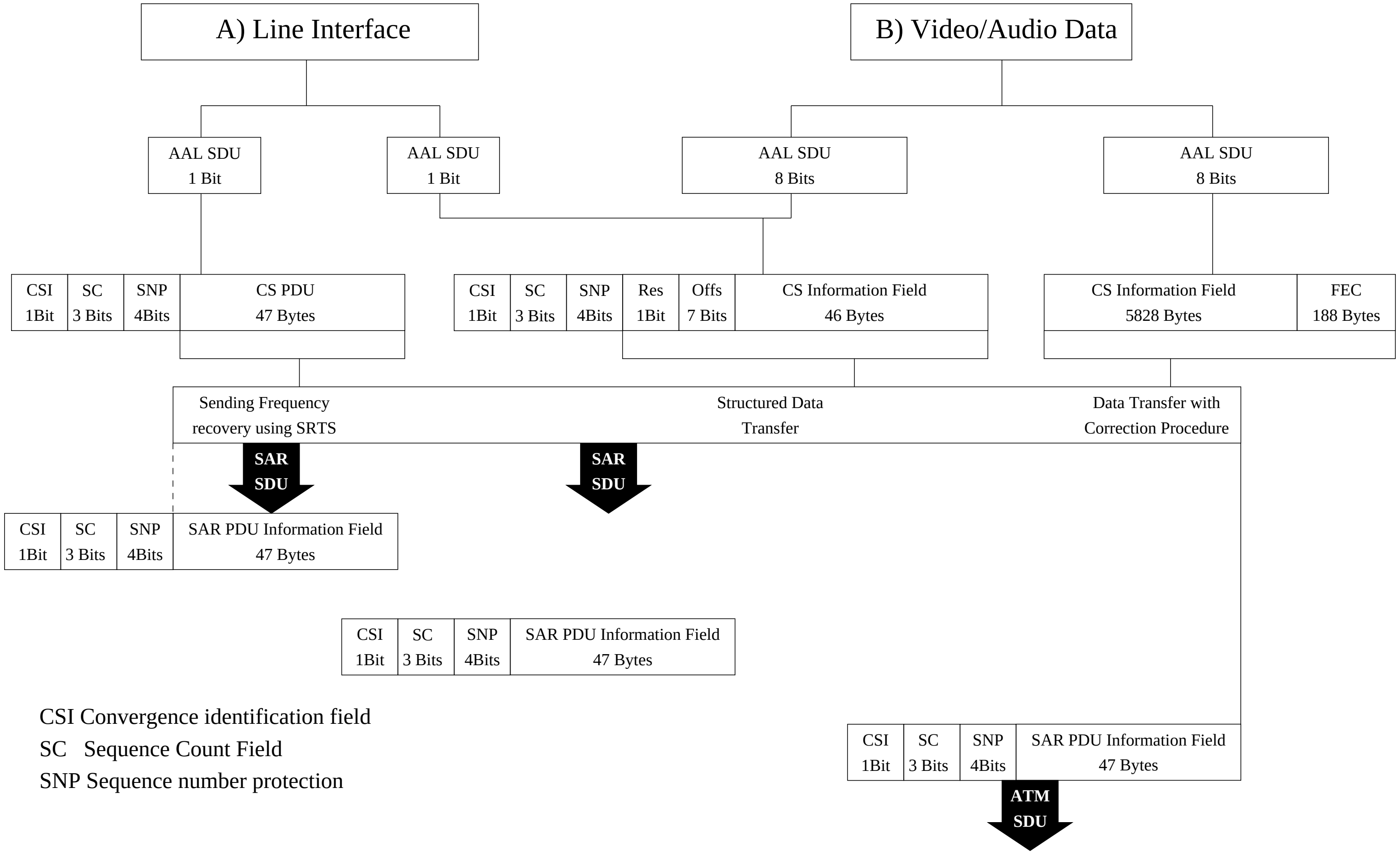


Abbildung 28: Die Struktur von AAL-Typ 1

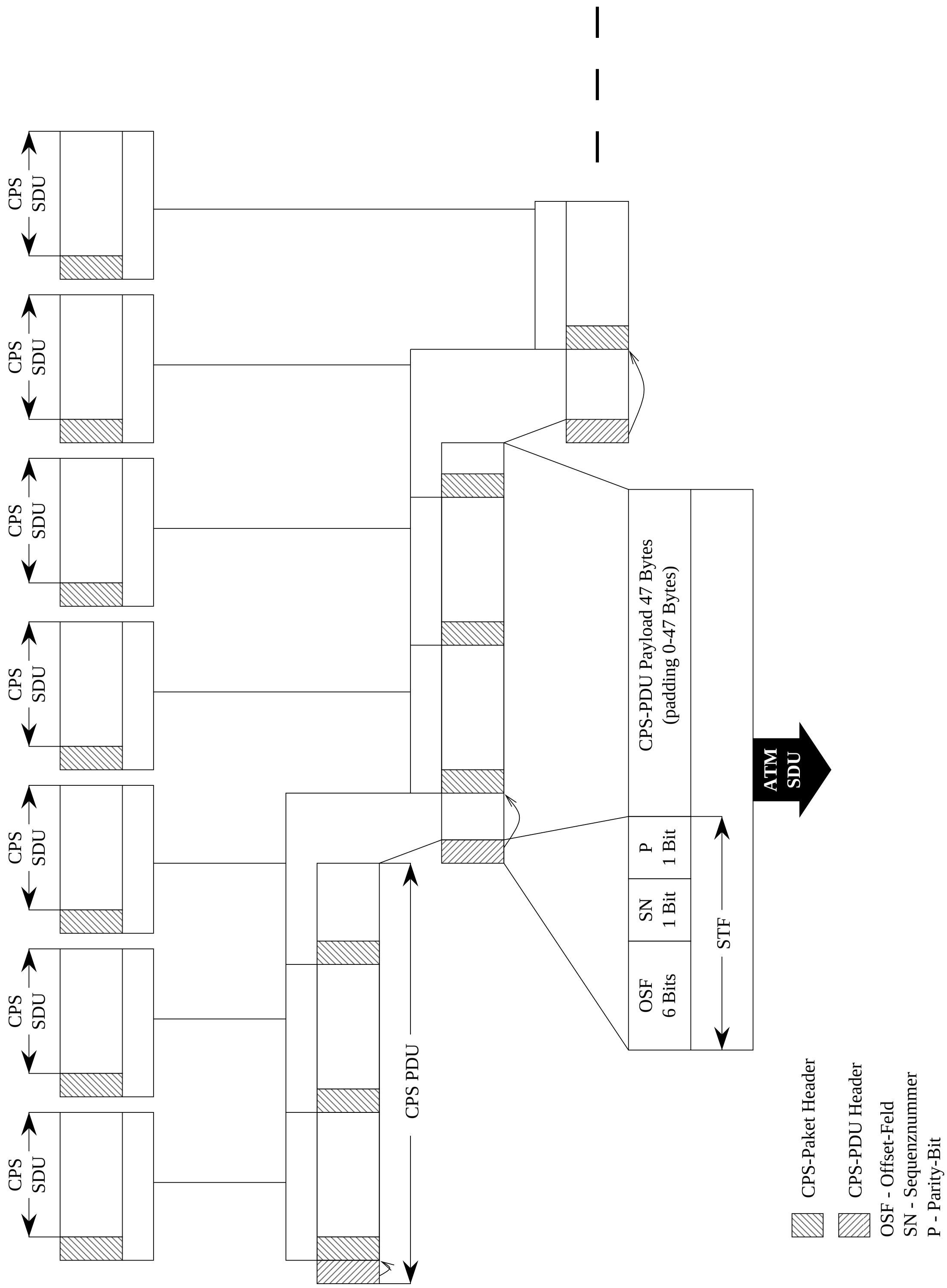


Abbildung 29: Die Struktur von AAL-Typ 2

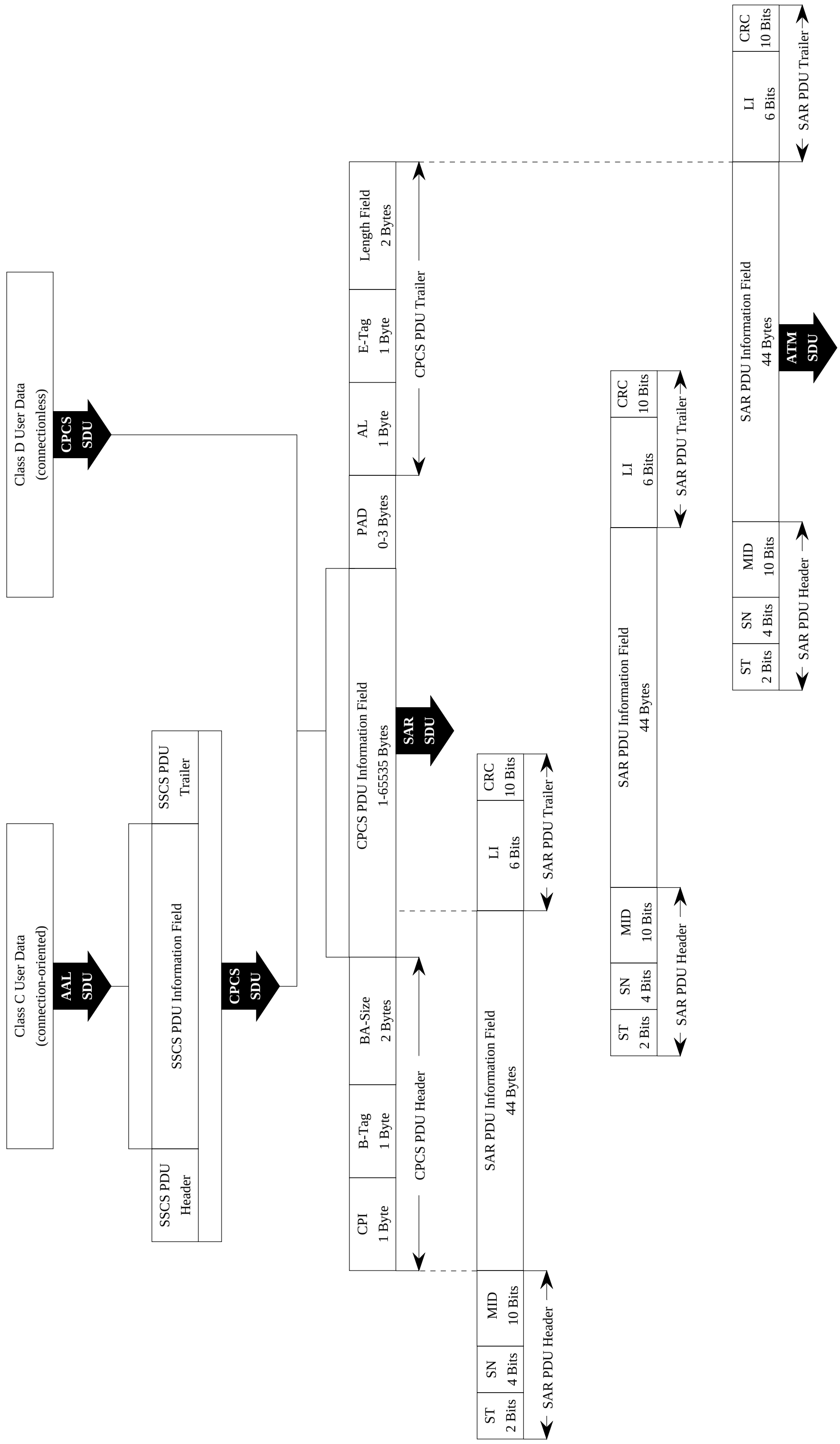
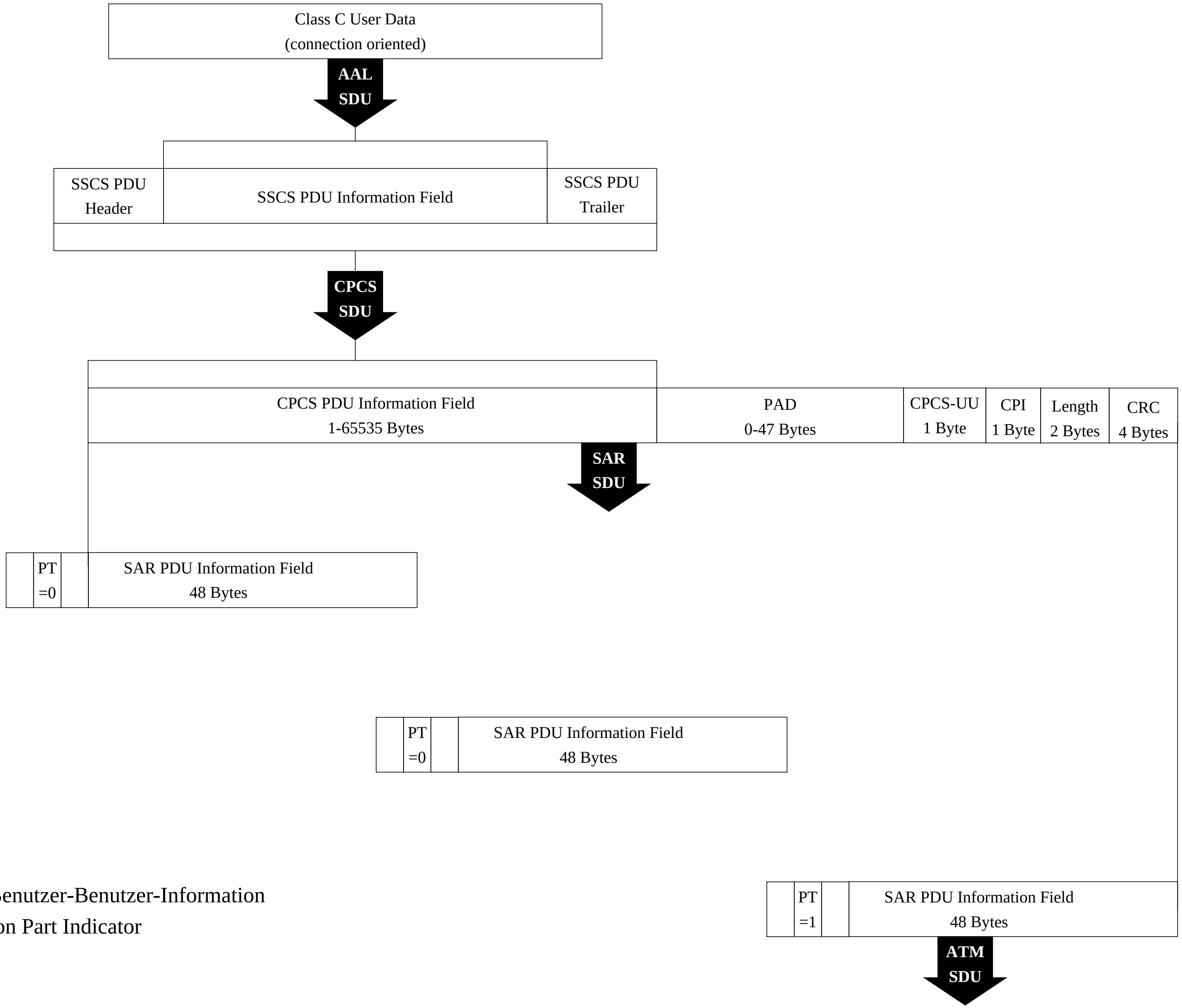


Abbildung 30: Die Struktur von AAL-Typ 3/4



CPCS-UU Benutzer-Benutzer-Information
CPI Common Part Indicator

Abbildung 31: Die Struktur von AAL-Typ 5

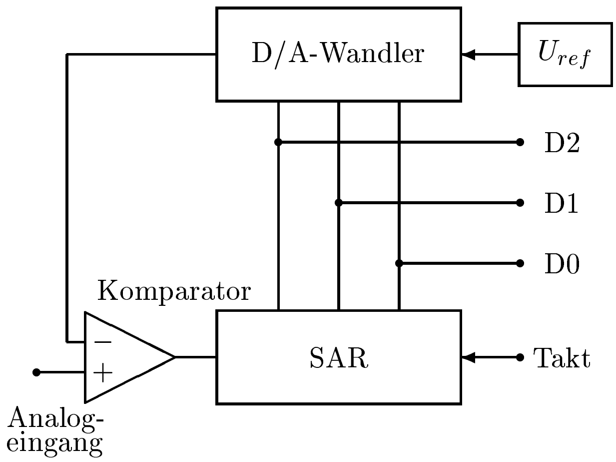
4 Signalisierung und ATM-Routing (1)

4.1 Die Eingliederung von ATM in die bestehende Netzwerklandschaft

Zu Beginn möchte ich einen kurzen Überblick darüber geben wie sich ATM in die bestehende Netzwerklandschaft einfügt. Anhand eines Einblicks in bestehende ISDN-Technik wird die Idee des ATM-Netzes leicht verständlich.

4.1.1 N-ISDN - Kurzüberblick

Mit der Einführung von ISDN (Integrated Services Digital Network) sollte ein Netz entstehen, das die Integration aller Netzwerkdienste in einem einheitlichen Netz gewährleistet. ISDN ist heute noch integraler Bestandteil des Fernsprechnetzes und bietet daher nur eingeschränkte Netzwerkdienste an. Mit seinen 64 Kbit/s wird es auch als Schmalband-ISDN bezeichnet. Das reicht z.B. um eine Telefonverbindung aufzubauen, indem man ein Kompressionsverfahren wie das von der ITU spezifizierte G.711 nutzt. Dieses Quantisierungs- und Kompressionsverfahren wird in fast allen ISDN-Telefonen benutzt. Es beruht auf einer Abtastrate von 8 kHz und einer Quantisierung mit 14 Bit. Hierzu kann beispielsweise ein billiger A/D-Wandler nach dem Prinzip des Wägeverfahrens zum Einsatz kommen, da die Wandlungszeit keine große technische Herausforderung darstellt. Damit beim Leser eine Vorstellung davon entsteht wie so ein A/D-Wandler funktionieren kann, will ich im Folgenden einen kurzen Einblick in das Verfahren der sukzessiven Approximation anhand eines Drei-Bit-Wandlers geben:



A/D-Wandler nach dem Prinzip der sukzessiven Approximation

Die Eingangsspannung wird mit einer, über einen D/A-Wandler erzeugten Spannung verglichen. Stimmen die Spannungen überein, entspricht die Ansteuerung des D/A-Wandlers der Eingangsspannung. Die Annäherung der intern erzeugten Spannung erfolgt über ein spezielles Register zur sukzessiven Approximation (SAR). Dazu wird die Anpassung mit dem höchstwertigen Bit (MSB) begonnen. Ist die erzeugte Spannung im Vergleich zur Eingangsspannung nach dem Setzen dieses Bits immer noch kleiner, so bleibt es gesetzt, ansonsten wird es wieder gelöscht. Dieses "Abwägen" wird mit den niederwertigeren Bits bis zum LSB fortgesetzt. Danach stehen die Digitalausgänge zur Verarbeitung bereit.

Würde man nun die 14 Bit mit 8 kHz übertragen, so entstünde ein Datenstrom von 8 KHz x 14 Bit = 112 Kbit/s. Das ist mit ISDN noch nicht möglich. Die ITU hat in der Spezifizierung G.711 deshalb eine weitere Quantisierung (Komprimierung) vorgesehen.

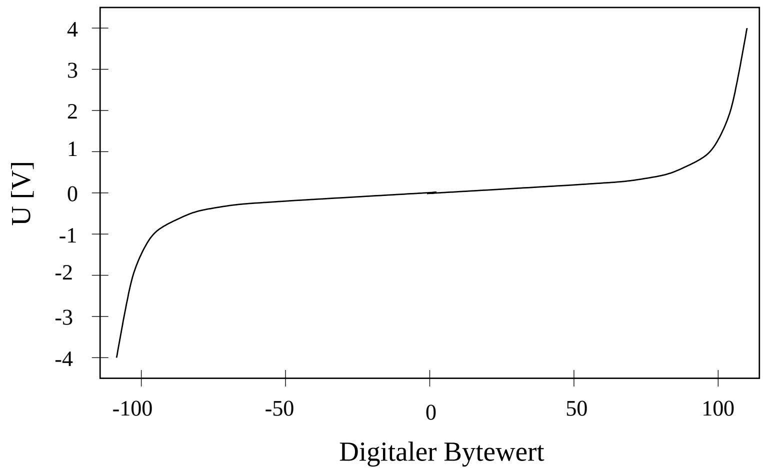


Abbildung 32: Datenreduktion nach G.711

Die Auflösung des Wandlers von 14 Bit wird, wie in Abbildung 32 dargestellt, auf 8 Bit reduziert. Daraus resultiert ein Datenstrom von konstant 8 kHz x 8 Bit = 64 Kbit/s.

Die ITU hat auch andere Komprimierungsverfahren spezifiziert, die in Qualität und Datenrate variieren. Die Abbildung 33 zeigt weitere Standards für die Sprachübermittlung. Mit G.726-32 ist es beispielsweise auch möglich zwei Sprechverbindungen über eine ISDN-Verbindung zu führen.

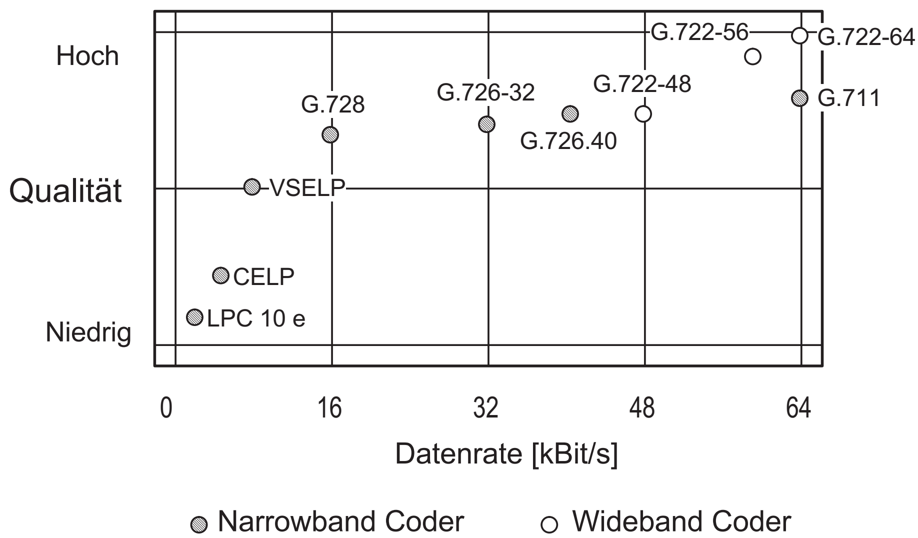


Abbildung 33: Komprimierungsmethoden für ISDN nach ITU

Um aber nun Musik Stereo in CD-Qualität übertragen zu können, ist schon eine Bandbreite von 2 x 44,1 kHz x 16 Bit ≈ 1,4 MBit/s nötig. Auch wenn gute Kompressionsverfahren eingesetzt werden, bleiben immer noch höhere Datenraten als sie N-ISDN erlaubt. Soll beispielsweise auch noch ein Videoclip zur Musik übertragen werden, sind schnell Datenraten bis über 100 MBit/s erreicht. All diese Anwendungsgebiete wie Breitband-Videokonferenz, Fernunterricht, Filme auf Abruf usw. verlangen ein Netz, das sehr breitbandigen Anwendungen gerecht wird. Dazu hat die ITU-T (ITU - International Telekommunication Union, T - Telecommunication Standardization, frühere CCITT) ein ATM - Verfahren (Asynchronous Transfer Mode) vorgeschlagen, das auch als Breitband-ISDN bezeichnet wird.

4.1.2 Die Vision: B-ISDN

Mit der Einführung von Breitband-ISDN werden verschiedene Serviceklassen spezifiziert. Je nach

	Klasse A	Klasse B	Klasse C	Klasse D
Zeitbezug	zeitkontinuierlich		nicht zeitkontinuierlich	
Bitrate	konstant	variabel		
Verbindungs-art	verbindungsorientiert			verbindungslos
Beispiele	Emulation der synchronen Durch-schaltevermittlung (Sprachübertragung)	variable Bitrate (komprimierte Sprache oder Video)	verbindungs-orientierte Datenübertragung	verbindungslose Datenübertragung
Diensttyp	AAL 1	AAL 2	AAL 3/5	AAL 4

Abbildung 34: Serviceklassen im ATM-Netz

Klasse können hier konstante Bitraten, maximale Verzögerungen usw. garantiert werden. Eine weitere wesentliche Neuerung ist die Skalierbarkeit der Bandbreite. Diese reicht von 2 MBit bis 2,4 GBit. Das ISDN-Protokoll DSS1 (Digital Subscriber Signalling System 1, spezifiziert in Q.930 u. Q931) wird durch DSS2 erweitert (Spezifikation: Q.2931), der Netzabschluß NT (Network Terminator) wird durch einen NT2 ersetzt und anstatt eines S_0-Bus wird ein S_B-Bus verlegt. Das klingt alles recht einfach, doch dahinter steckt eine völlig neue Netzwerktechnologie.

4.1.3 Der Asynchrone Transfermodus

Gegenüber dem Paket-Transfer-Modus[3] bietet ATM bei höherer Netzlast die volle Bandbreite. Das wird durch die festen Zellrahmen erreicht. Der Vermittlungsaufwand kann durch den einfachen Aufbau des Zellheaders stark vereinfacht und somit beschleunigt werden. Die Paketvermittlung findet nicht in höheren Schichten statt, sondern schon auf der Schicht 2 (ATM-Schicht).

Im Vergleich zum synchronen Transfermodus[4] (STM), bei dem synchrone Zeitschlitze als Übertragungskanäle verwendet werden, bietet ATM aber auch die Flexibilität eines paketvermitteln-

[3] Der Paket-Transfer-Modus wird beispielsweise beim Ethernet verwendet

[4] Das Multiplexen von Telefonverbindungen funktioniert nach diesen Prinzip

den Netzes. Bursts können auf evtl. noch freie Zellen verteilt werden, wozu in einem STM-Netz erst weitere Zeitschlitze allokiert werden müßten.

Der einfache und schnell zu vermittelnde Aufbau der ATM-Zellen hat aber auch den Nachteil, daß der Verwaltungsaufwand und die Protokolle in höheren Ebenen um einiges komplexer werden. Sychronisierung, Leistungsverteilung usw. müssen dabei ausgehandelt und überwacht werden.

4.2 Netzwerkschnittstellen

Zur Durchleuchtung der Signalisierung im einzelnen ist es zuerst einmal nötig die ATM-Netzstruktur genauer zu betrachten. In einem ATM-Netz gibt es zwei verschiedene, den Aufga-

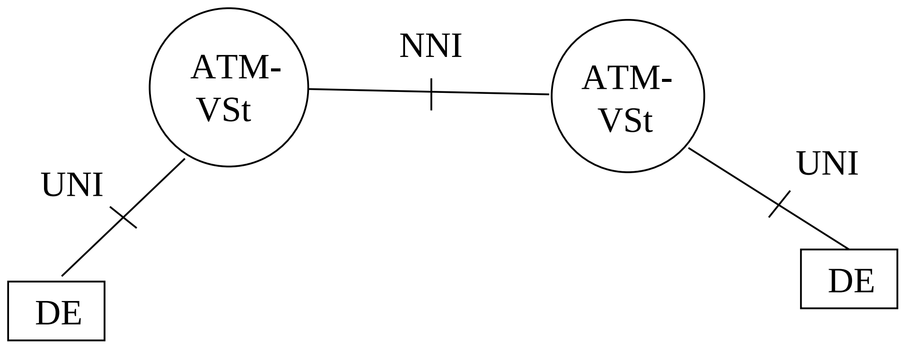


Abbildung 35: Netzwerkschnittstellen im ATM-Netzwerk

ben angepaßte Schnittstellen. Die Verbindung eines Netzwerkbenutzers (bzw. seiner DE: Datenendeinrichtung) mit dem Netz[5] ist das User-Network Interface (UNI). Die Endgeräte oder Privatnetze werden auch als Customer Premises Equipment (CPE) bezeichnet. Zwischen den Netzwerkknoten befindet sich das Network-Network Interface (NNI). Bildet ein NNI einen Übergang zwischen zwei Netzbetreibern, so spricht man auch von einem Broadband Intercarrier Interface (B-ICI). Da beim NNI zwischen privaten und öffentlichen Übergängen unterschieden werden kann, spricht man auch häufig von P-NNI. Das ist eine Abkürzung für private NNI und stellt die Übergänge innerhalb von Netzen dar. Ein P-UNI hat sich dagegen im Sprachgebrauch nicht durchgesetzt.

4.2.1 UNI

Das User-Network Interface wurde von der ITU-T 1992 in Q.2931 spezifiziert. Es stellt die Schnittstelle (Protokoll) des Netz-Benutzers zum Netz dar. Q.2931 ist eine Erweiterung der Signalisierung in Schmalband-ISDN - der Q.931 (DSS1). Die Signalisierungszellen werden im Netzwerk behandelt wie Zellen mit Nutzinformation. Ein eigener Signalisierungskanal steht wie im Schmalband-ISDN virtuell zur Verfügung. Beim Aufbau einer Verbindung müssen u.a. die Qualitätsmerkmale ausgehandelt werden. Dazu gehören die Zellverzögerung, der mittlere Durchsatz, der Spitzendurchsatz, usw. Dabei kommt eine feste Vereinbarung (der Traffic Contract) zustande oder der Verbindungsaufbau ist fehlgeschlagen. Die Signalisierungsschicht baut auf der

[5] Der Netzwerkbenutzer ist üblicherweise an eine ATM-Vermittlungsstelle angeschlossen. Hier wird in der Regel ein ATM-Switch die Verbindungen weiterleiten.

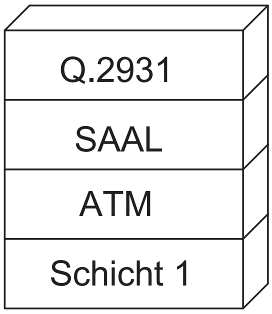


Abbildung 36: Der UNI-Protokoll-Stack

SAAL-Schicht auf. Diese ATM-Adaptionsschicht für die Signalisierung realisiert alle für das Signalisierungsprotokoll notwendigen Funktionen und stellt somit einen Interpreter des Protokolls dar. Die SAAL-Schicht wird in einem folgenden Kapitel noch genauer dargestellt.

4.2.2 P-NNI

Im P-NNI Protokoll sind nicht nur alle nötigen Eigenschaften zum Auf- und Abbau einer Verbindung enthalten, sondern auch die Protokolle zum Routing, zur Adreßauflösung, zur Verkehrsüberwachung usw. Diese Protokolle bauen auch auf die SAAL-Schicht auf. Da dieses The-

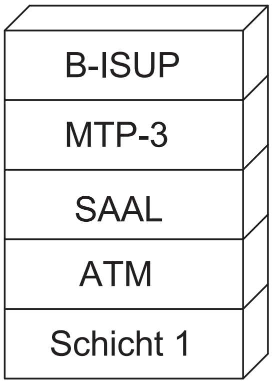


Abbildung 37: Der P-NNI-Protokoll-Stack

mengebiet für sich wieder sehr umfangreich und komplex ist, widmet sich der nächste Vortrag ausschließlich diesem Gebiet. Deshalb möchte ich im Weiteren nicht auf die einzelnen Protokolle des Routings eingehen, sondern versuchen ein grundlegendes Verständnis des Aufbaus von ATM zu vermitteln.

4.2.3 B-ICI

Das Broadband Intercarrier Interface wurde hauptsächlich eingeführt um definierte Netzübergabestellen zu realisieren. Über die Netzinternas wird ein Betreiber nur ungern Auskunft geben.

4.3 Der ATM Protokollstack

Das ATM-Referenzmodell basiert auf dem OSI-Referenzmodell. Es lassen sich aber einige Funktionen nicht auf das OSI-Modell abbilden. Die ATM-Schicht und die ATM-Adaptionsschicht sind speziell für den asynchronen Transfermodus notwendig. Die Managementebene zieht sich über alle Schichten und überwacht die Koordination der einzelnen Schichtenaufgaben (Layer Management) und die Koordination der Aufgaben im Netz (Plane Management). Die Nutzerebene stellt

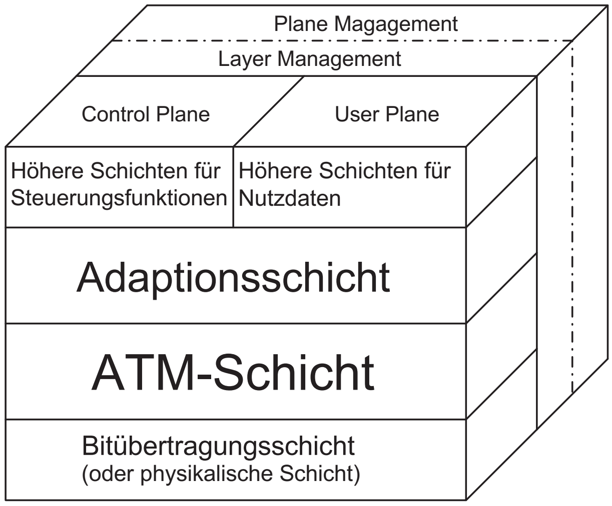


Abbildung 38: Das ATM-Referenzmodell

den Transport zur Verfügung und reicht die Daten zur ATM-Anpassungsschicht weiter. Dabei können die Nutzdaten über vier verschiedene Adaptionsdienste versendet werden. Wie in Abbil-

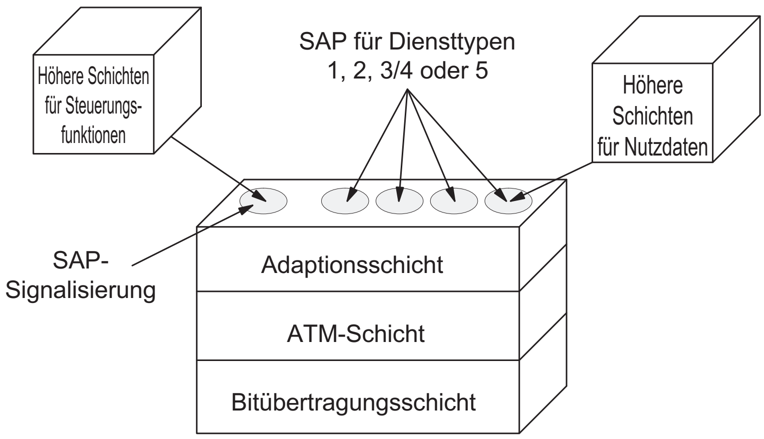


Abbildung 39: SAPs im ATM Protokollstack

dung 39 verdeutlicht werden auch die Signalisierungspakete über Service Access Points (SAP) an die tieferliegenden Schichten weitergereicht. Diese Dienstzugangspunkte stellen die Verbindung zwischen den höheren Schichten und den Übertragungsschichten mit ihren verschiedenen Leistungsmerkmalen zur Verfügung. Für die Signalisierung ist in erster Linie nur der Zugang für die Steuerungsfunktionen von Interesse. Die Handhabung der anderen SAPs wird abhängig von

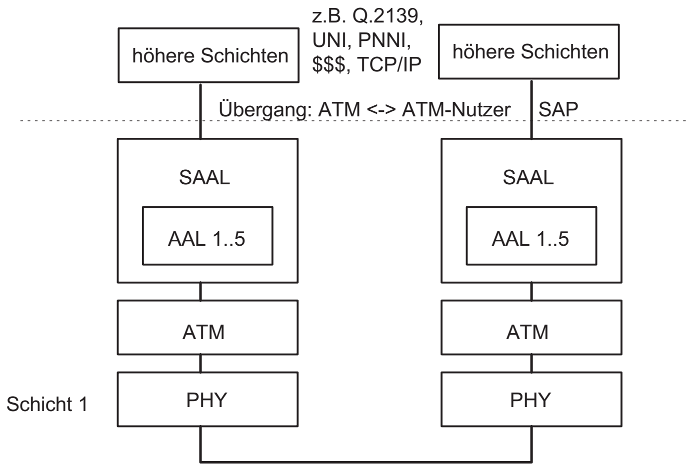


Abbildung 40: Der ATM-Signalisierungs-Stack

der Signalisierung gesteuert. Die Abbildung des ATM-Signalisierungs-Stack zeigt nun im Überblick wie die Signalisierung im ATM-Netzwerk aufgebaut ist. Alle Protokolle höherer Schichten (von der Integration von TCP/IP bis zur Abrechnungsfunktion der Dienstanbieter) nutzen diesen Stack. Dabei spielt die AAL-Schicht für die Signalisierung eine entscheidende Rolle. Sie soll deswegen etwas genauer unter die Lupe genommen werden.

4.4 SAAL: Die Signalisierungs-AAL

Die Signalisierungs-AAL setzt sich nach dem Vorschlag der ITU aus drei einzelnen Teilschichten zusammen. Dabei kann zuerst grob in zwei Teilschichten unterschieden werden. Die unterste

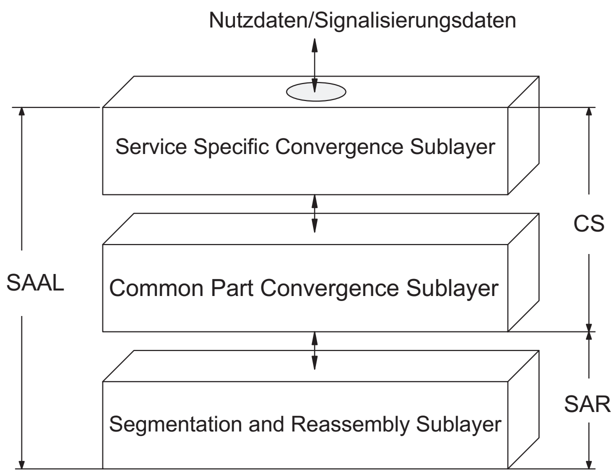


Abbildung 41: Die ATM Signalisierungs-AAL

Teilschicht ist der Segmentation and Reassembly Sublayer. Dieser wird nicht nur zur Signalisierung, sondern auch bei den anderen Diensttypen benötigt. Er dient bei der Übertragung von Signalisierungs- (Nutz-) Information auf der Senderseite zur Aufteilung auf die Länge eines

Nutzdatenfeldes einer ATM-Zelle und auf der Empfängerseite, um die Daten wieder zu rekonstruieren. Auf der SAR-Teilschicht baut der Convergence Sublayer (CS) auf. Dieser gliedert sich dienstspezifisch. Für die Signalisierungs-AAL beinhaltet die ITU-Empfehlung zwei Teilschichten: das Service Specific Connection Oriented Peer-to-Peer Protocol (SSCOP) und darauf aufbauend die Service Specific Coordination Funktion (SSCF).

4.4.1 SSCOP

Das Service Specific Connection Oriented Peer-to-Peer Protocol stellt den Common Part CS. Diese Schicht sichert die Übertragung von Signalisierungsinformationen, und erkennt falsch übertragene Signalisierungsdatenpakete.

4.4.2 SSCF

Der Service Specific CS wird mit der Service Specific Coordination Funktion (SSCF) realisiert. Diese Schicht stellt einen Interpreter/Übersetzer der UNI-Protokolle (z.B. DSS2) dar.

4.5 Adressierung

Zur Adressierung der ATM-Endgeräte hat die ITU drei Rufnummernformate definiert. Die Rufnummern sind 20 Byte lang , was den Nummernraum der bisherigen Netze um ein gutes Stück erweitert. Das allgemeine Format für Rufnummern ist in E.164 spezifiziert. Daneben gibt es auch internationale Rufnummern, die für international agierende Institutionen gedacht sind. Um welche Rufnummernart es sich handelt ist im ersten Byte, dem Authority and Format Identifier (AFI) abgelegt. Danach folgt der Initial Domain Identifier (IDI) und der Domain Specific Part (DSP). Der erste Teil entspricht der Rufnummer im Netz (Telefonnummer), der zweite Teil stellt eine ISDN-Subadresse dar.

4.5.1 E.164 ATM-Format

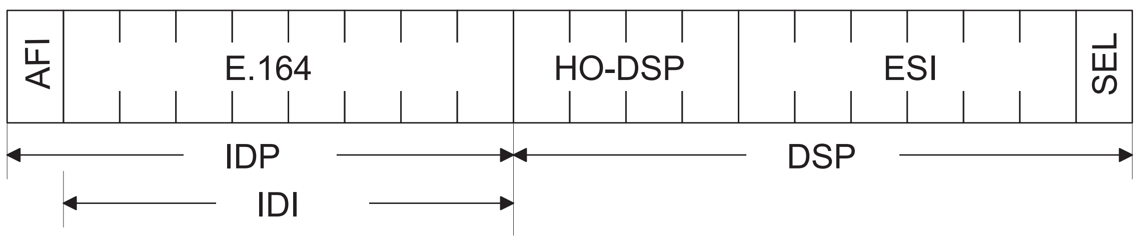


Abbildung 42: Der E.164 Rufnummernaufbau

Mit den acht Byte des IDI -Teils der Rufnummer ist es möglich maximal 15 Ziffern anzugeben. Diese beinhalten die Länderkennzahl, die Ortsnetzkennzahl und die Teilnehmernummer. Die weiteren 11 Byte stellen eine Subadresse dar, mit der z.B. in einem angeschlossenem LAN ein einzelnes Endgerät angesprochen werden kann. Diese setzt sich aus dem High Order Domain Specific Part (HO-DSP) dem End System Identifier (ESI, z.B. MAC) und einem Selector (SEL) zusammen.

4.5.2 Internationale Rufnummern

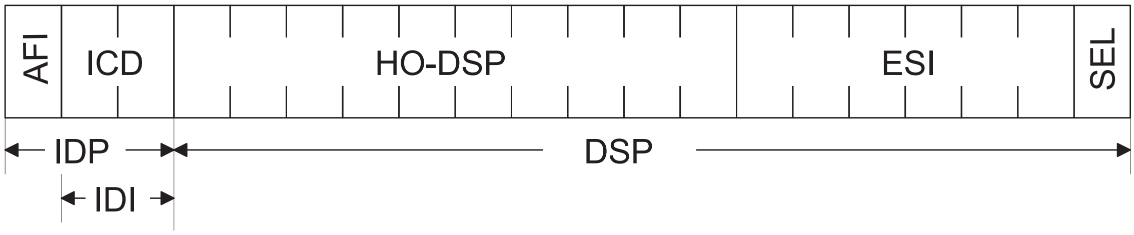


Abbildung 43: Format der ICD-Rufnummern

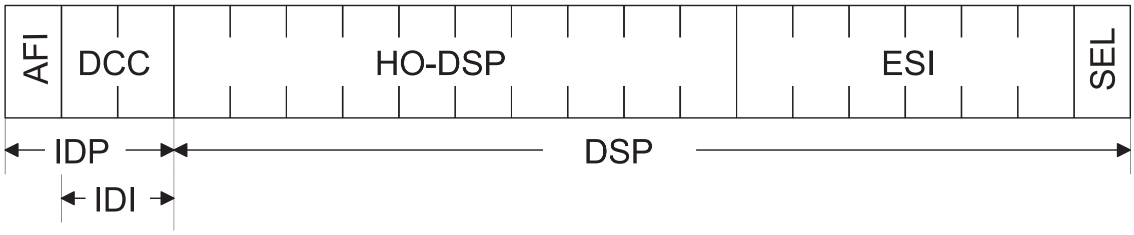


Abbildung 44: Das Format der DCC-Rufnummern

Bei den internationalen Rufnummernformaten entfällt der eigentliche Rufnummernteil. An deren Stelle tritt ein Data Country Code (DCC) oder ein International Code Designator (ICD) mit dem ein Land oder eine Organisation ausgewählt wird. Alle weiteren Bytes der Rufnummer werden dann der Organisation zur internen Auswahl übergeben. Diese Rufnummernarten sind aber noch nicht einheitlich spezifiziert.

4.6 Verbindungsauf- und abbau

Wie nun eine Verbindung zwischen zwei Teilnehmern aufgebaut wird soll mit folgendem Beispiel gezeigt werden.

4.6.1 Q.2931

In Abbildung 45 wird der Weg einer Signalisierungsverbindung gezeigt. Hierbei unterhalten sich zwei Teilnehmer mittels der in Q.2931 spezifizierten Signalisierungsnachrichten über einen von der Metasignalisierung (ATM-Schicht) geöffneten Kanal.

4.6.2 Signalisierungspakete

Eine Signalisierungsnachricht beginnt immer mit einem Protokolldiskriminator, in dem festgelegt wird, um welches Protokoll es sich bei dieser Nachricht handelt. Danach folgt die Call Reference. Diese ist nötig um zu kennzeichnen, welche Nachricht zu dieser Signalisierung gehört, denn es können mehrere Signalisierungsaktivitäten gleichzeitig über einen Kanal stattfinden. Als nächstes kommt die Angabe über den Message Type. Sie enthält die eigentliche Nachricht des Schicht-3-Protokolls (SETUP, CONNECT usw.). Letzter Bestandteil des Nachrichtenkopfes ist

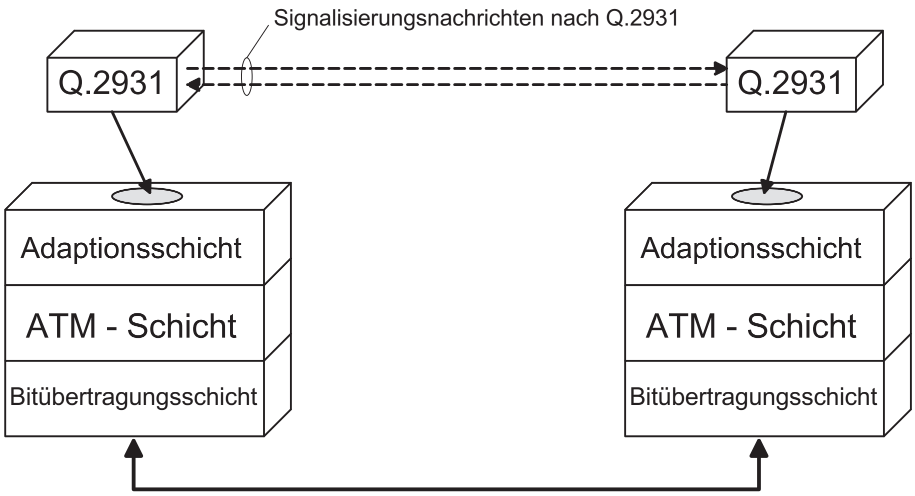


Abbildung 45: Der Weg einer ATM-Signalisierungsnachricht

die Längenangabe der folgenden Informationselemente. In den folgenden Informationselementen müssen alle für den Verbindungsauf- bzw. abbau notwendigen Daten enthalten sein. Im Fall einer SETUP-Nachricht sind dies u.a. die AAL-Parameter, der Traffic Descriptor, alle Quality of Service Parameter und natürlich die Zieladresse.

Wie auf der dritten Schicht ein Verbindungsaufbau funktioniert wird in folgendem Beispiel exemplarisch dargestellt.

4.6.3 Die Vermittlung der Verbindung

In diesem Beispiel ist ein Verbindungsaufbau über das DSS2-Protokoll über zwei Vermittlungsstellen gezeigt. Eine ALERT-Signalisierung ist dabei nur nötig, wenn es sich nicht um ein automatisch antwortendes Gerät handelt (Klingelzeichen beim Telefon).

4.6.4 Metasignalisierung

Da auch die Signalisierung über eine virtuelle Verbindung stattfindet, muß diese auch auf- und abgebaut werden. Die Steuerung dieses Auf- und Abbaus wird als Metasignalisierung bezeichnet. Für diese Verbindung wird ein fester, für die Signalisierung reservierter VCI[6] (Virtual Channel Identifier) verwendet. Der Mechanismus selbst ist Teil der ATM-Schicht. Wie in Abbildung 47 verdeutlicht werden Signalisierungspakete von den unteren Schichten selbständig über den dafür vorgesehenen Weg durchgereicht. Der Übertragungsschicht erscheint ein Signalisierungspaket genauso, wie jedes andere Datenpaket. Allein an der VCI erkennt die Gegenstelle, ob es sich um ein Nutzdaten-Paket handelt oder um Management-Informationen.

[6] Bei Punkt-zu-Punkt-Verbindungen: VCI = 5

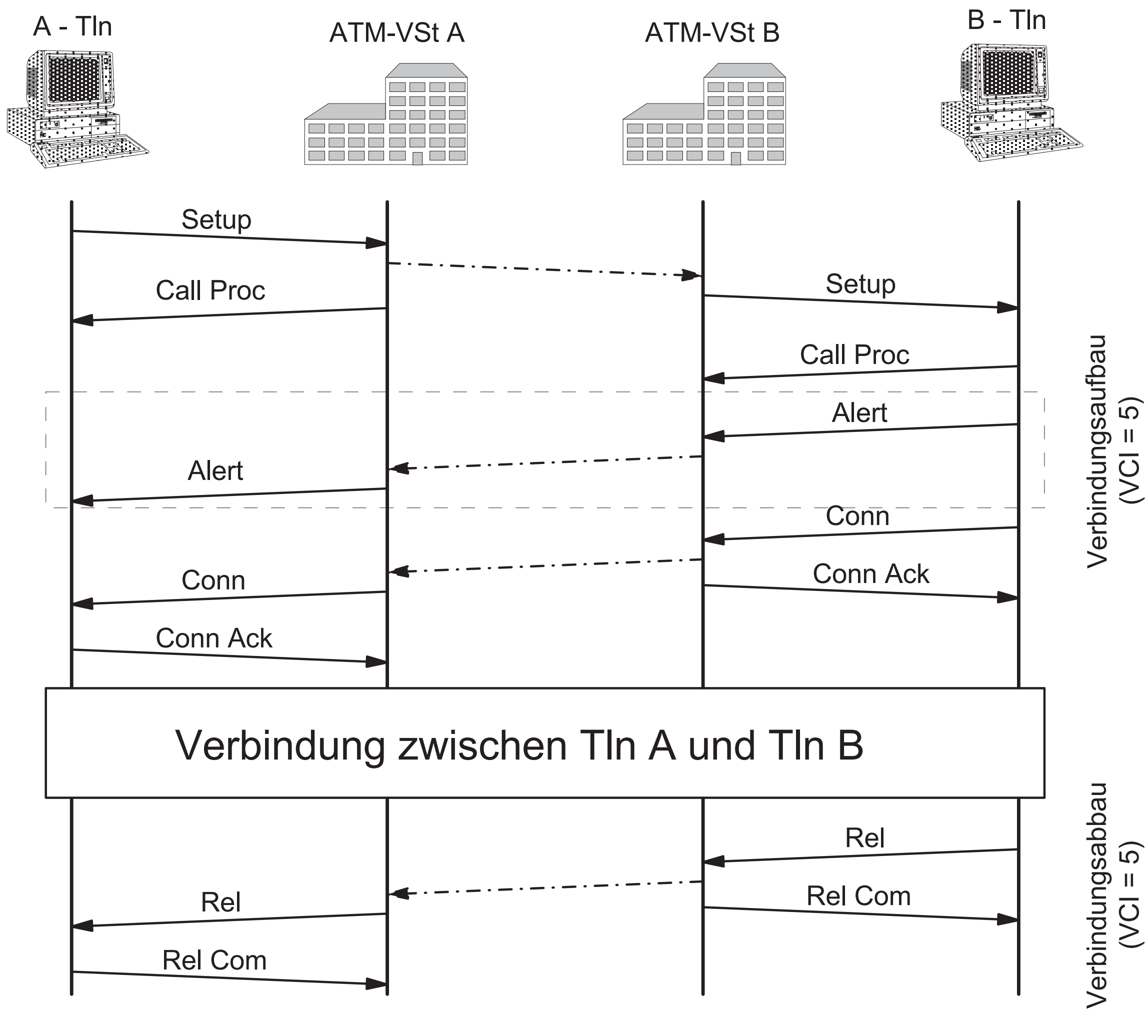


Abbildung 46: Der ATM-Verbindungsauf- und abbau

4.7 Verbindungskonfigurationen

Das generelle Referenzmodell für eine ATM-Nutzverbindung über einen ATM-Cross-Connect (Switch) zeigt die Abbildung 48. Dabei werden die ATM-Zellen auf der ATM-Schicht weitergereicht. Als Resultat des einfachen Aufbaus einer ATM-Zelle und der geringen Stackhöhe gewährleistet das Netz gute Übertragungszeiten. Es müssen während einer ATM-Verbindung keine Signalisierungsdaten mehr bearbeitet werden. Der Nachteil dieser schnellen Verbindung ist die Komplexität, die in die höheren Schichten verlagert wird. So muß bei der Signalisierung des Aufbaus und Abbaus, aber auch zur Überwachung, ein großer Aufwand getrieben werden. Auch die Vermittlung und Gebührenabrechnung durch die Netzbetreiber wird auf höheren Schichten als der ATM-Schicht stattfinden müssen.

Wenn jedoch eine Verbindung aufgebaut ist, werden die meisten Cross-Connects in der Verbindung keine Zeit mehr mit Auspacken und Analysieren der Signalisierungsinformationen verbrauchen.

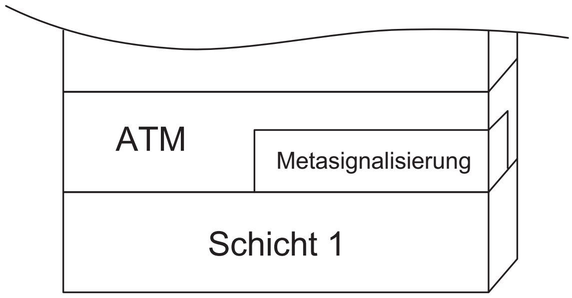


Abbildung 47: Die Metasignalisierung

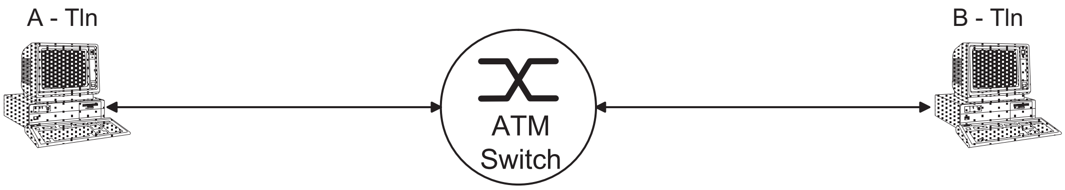


Abbildung 48: ATM-Verbindung über einen Cross-Connect(Switch)

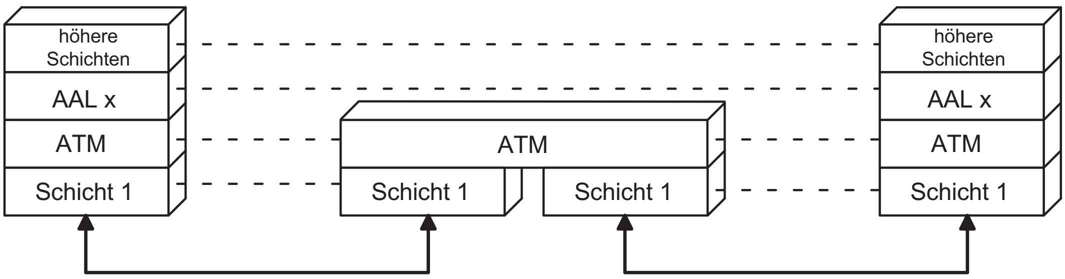


Abbildung 49: Eine ATM-Verbindung für Nutzdaten

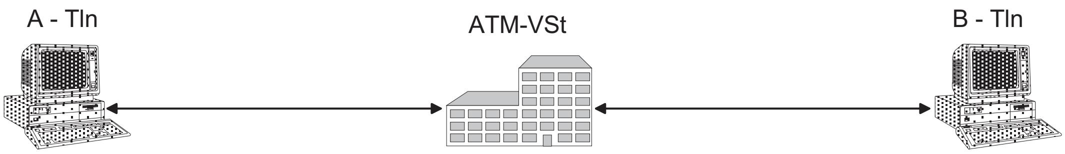


Abbildung 50: Eine ATM-Verbindung in der Vermittlungsstelle

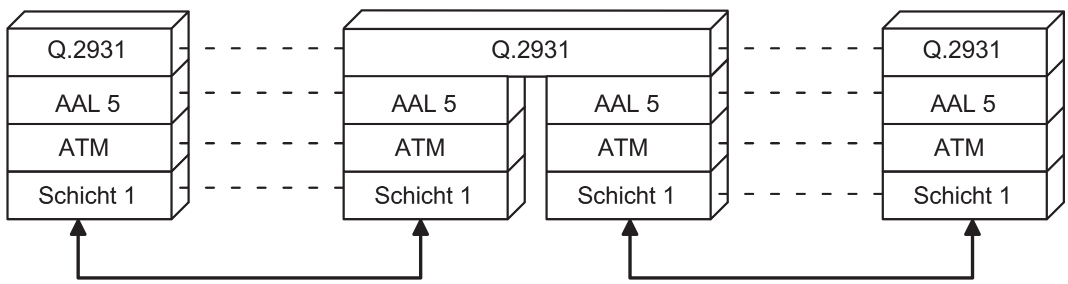


Abbildung 51: Eine ATM-Signalisierungsverbindung

5 Signalisierung und ATM-Routing (2)

5.1 ILMI

ILMI steht für Integrated Local Management Interface. Früher stand das erste I wohl einmal für Interim, also für provisorisch. Aber wie so oft wurde aus der Übergangslösungs eine Dauerlösung.

Nach der ATM - Spezifikation hat jeder Switch und jedes ATM - Endgerät eine rudimentäre Datenbank, die nach dem Schlüssel - Datensatz Prinzip arbeitet. Diese Datenbank muß zumindest den standardisierten Teil haben, der in den ATM Spezifikationen festgelegt ist. Außerdem kann jeder Hersteller noch einen herstellerspezifischen Teil an die Datenbank anfügen, in dem er dann weitere Daten speichern kann. Diese Datenbank heißt MIB (Management Information Base). Im standardisierten Teil der MIB gibt es für jede ATM Schnittstelle eine IME, eine Interface Management Entity. Man kann eine IME auch als eine Instanzierung des Objektes ATM - Schnittstellendatenstruktur sehen. In dieser IME stehen folgende Informationen:

- Physical Layer
  Interface Index
- ATM Layer
  Interface Index
- Virtual Path Connection (VPC)
  Interface Index + VPI
- Virtual Channel Connection (VCC)
  Interface Index + VPI + VCI
- Network Prefix
  Interface Index + Prefix
- Address
  Interface Index + Address
- Service Registry
  Interface Index + Srvd D + Srvc Index

Wie man am Beispiel der Virtual Path Connections sieht, müssen die IMEs an den Enden einer ATM Verbindung synchronisiert werden, sonst könnte ein Gerät versuchen, etwas über einen virtuellen Pfad zu senden, den das andere Gerät noch gar nicht kennt. Dazu ist das ILMI - Protokoll da.
Da ILMI als Übergangslösung geplant war, hat man sich für das einfache, bereits vorhandenes Protokoll SNMP entschieden. SNMP steht für Simple Network Management Protocol, und ist ein Service Protokoll aus der Internet Protokoll Familie (vgl. RFC 1157). Über das SNMP Protokoll können immer nur zwei Partner gleichzeitig verbunden werden, es ist also ein Peer-to-Peer Protokoll. Dabei ist einer der Master und der andere der Slave. Der Slave ist passiv, d.h. von sich aus sendet er nichts ohne Aufforderung durch den Master. Jeder Datenaustausch wird also von Master initiiert. Dazu hat er 4 Befehle zur Verfügung:

- get(...)
  Mit dem get(...) - Befehl fordert der Master einen Datensatz zu einem bestimmten Schlüssel an. Der Schlüssel wird als Argument des get - Befehls übergeben. Der Slave liefert daraufhin den Datensatz.

- get-next
  Mit dem get-next - Befehl kann man die Datenbank durchlaufen und so zum Beispiel nach einem get(...) - Befehl, der den Datensatz über den VPI=0 abgefragt hat, auch die Datensätze der VPIs 1, 2, 3 usw. abfragen.

- set(...)
  Der set(...) - Befehl hat als Argumente einen Schlüssel und einen Wert. Damit kann der Master in der Datenbank des Slaves den Wert, auf den der Schlüssel zeigt, neu setzen.

- trap(...)
  Mit dem trap(...) - Befehl kann der Master den Slave anweisen, Alarm auszulösen, wenn ein bestimmter Wert (durch einen Schlüssel festgelegt) eine Grenze erreicht oder überschreitet. Damit kann der Slave dem Master z.B. mitteilen, wenn er kurz vor einer Überlastsituation steht.

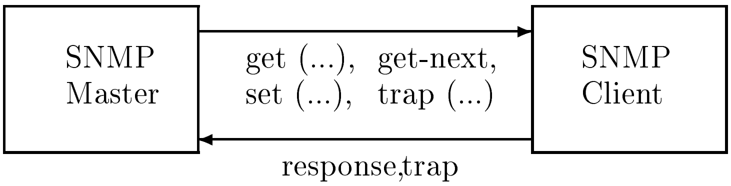


Abbildung 52: SNMP Kommunikation

Für das ILMI mußte man sich noch entscheiden, welches ATM Gerät nun die Rolle des Masters und welches die Rolle des Slaves übernehmen soll. Die Lösung bestand darin, daß beide Geräte einer ATM Verbindung beide Funktionen wahrnehmen. Jedes Gerät frägt als Master die Datenbank des anderen Gerätes ab. Das Gerät muß gleichzeitig die Anfragen des anderen Gerätes als Slave beantworten. Die SNMP Nachrichten werden in AAL-5 Pakete verpackt und über VCC = 16 übertragen.
Mit dem ILMI ist es jetzt z.B. möglich die Adressregistrierung zu automatisieren. Nehmen wir an, wir haben einen Switch und ein ATM - Endgerät (Computer mit ATM Karte). Dem Switch haben wir den Adressprefix einprogrammiert, dem ATM Endgerät die Endgerätekennung (ESI = End System Identifier). Bei der Datenbanksynchronisation werden Prefix und Endgerätekennung ausgetauscht. Das Endgerät kennt nun seine vollständige ATM Adresse und der Switch weiß jetzt, daß er ein erreichbares Endgerät mehr hat und an welchem physikalischen Interface es hängt.
Das ILMI bietet noch weitere Funktionalität. So kann man z.B. periodisch die ATM Geräte ID des anderen Geräts abfragen. Ändert sie sich, muß auch das andere Gerät gewechselt worden sein, und die IME Datenbank muß neu aufgebaut werden.

5.2 Routing

5.2.1 Grundlagen

- Verbindungsorientierung
  In ATM Netzwerken werden die Verbindungen geroutet. Die Arbeit des "Routens" muß also nur bei Verbindungsaufbau geleistet werden. Steht eine Verbindung, so laufen die Zellen sozusagen immer dem Vorgänger hinterher.
  Der Vorteil: Es muß nicht für jede einzelne Zelle geroutet werden, der Arbeitsaufwand ist

geringer.
Der Nachteil: Steht eine Verbindung, kann man an ihr nichts mehr verändern. Bei einem Ausfall eines Switches werden alle Verbindungen, die über ihn laufen, unterbrochen. Auch lassen sich QoS - Anforderungen nachträglich nicht mehr ändern.

- Adressierung
  ATM Adressen bestehen aus 20 Bytes, die für das Routing folgende Bedeutung haben: Fangen wir beim letzten Byte an, dem SEL, dem Selektor. Dieses Byte wird beim Routing

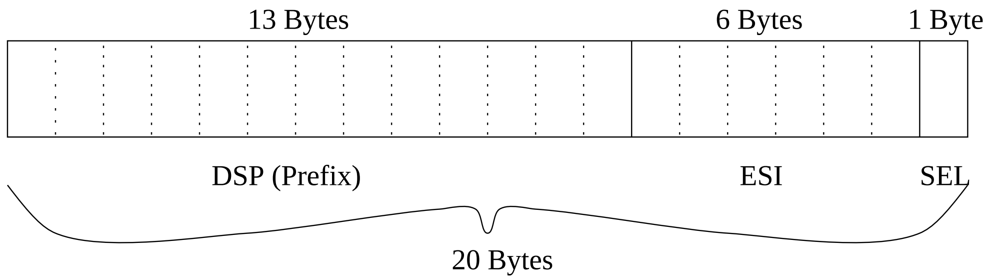


Abbildung 53: ATM Adressierungsschema

  nicht berücksichtigt. Es dient dazu, im Endgerät, noch einmal verschiedene Einheiten ansprechen zu können. Das Beispiel hierzu ist die Telefonanlage, die bis zu 256 analoge Telefone verwalten kann. Mit den ersten 19 Bytes routet man den Anruf zur Telefonanlage, und das 20. Byte teilt der Telefonanlage mit, welches der 256 Telefone zu klingeln hat.
  Vor dem SEL steht der ESI, der End System Identifier. Er besteht aus 6 Bytes, identifiziert damit das Endgerät und ist das Analogon zur MAC Adresse beim Ethernet.
  Die ersten 13 Bytes sind der DSP, der Domain Specific Part. Dieser Teil der Adresse wird bei der hierarchischen Netztopologie dazu verwendet, um in die einzelnen Domains und tieferen Hierarchiestufen zu verzweigen.

  Ein paar Worte zur Eindeutigkeit von Adressen: Für das ATM Routing ist jede Kombination der ersten 19 Bytes eine eigene, eindeutige Adresse. Habe ich also zwei Adressen, die sich in den ersten 19 Bytes um ein Bit unterscheiden, werden die Adressen unterschieden und korrekt an zwei verschiedene Anschlüsse geroutet.
  In der Praxis jedoch sollte man die Unterscheidung der Endgeräte anhand der 6 Bytes ESI treffen. Somit kann ein Endgerät in eine neue Domain umziehen, ohne daß man darauf achten muß, ob nicht dessen ESI in der neuen Domain schon vergeben ist.

5.2.2 IISP

IISP steht für Interim Inter-switch Signaling Protocol. Wie das Interim schon sagt, ist es ein provisorisches Routing Protokoll, bis das eigentliche (P-NNI) fertiggestellt ist. IISP routet hop-by-hop, d.h. jeder Switch überlegt sich nur, zu welchem seiner Nachbarn er die Verbindung weiterreichen soll. Dieser muß sich dann wiederum überlegen, an wen er die Verbindung weitergibt. Das Routing in einem Switch geschieht über eine Routing Tabelle. Diese Tabelle muß manuell vom Netzwerkadministrator verwaltet werden. Jeder Eintrag in dieser Tabelle hat folgenden Aufbau: Im Feld Adresse steht eine 20 Byte ATM Adresse. Im Feld Adresslänge ein Integerwert mit einem Wertebereich von 0 bis 104 oder dem Wert 152. Dieses Feld gibt an,

Adresse	Adresslänge	Port

Abbildung 54: IISP Routingtabelle

wieviele Bits der Adresse im Adreßfeld gültig sind. Im letzten Feld steht ein Identifikator entweder für einen physikalischen Port oder für eine logische Verbindung zu einem anderen Switch (Permanent Virtual Path). Bei einem Routingvorgang vergleicht der Switch die Zieladresse, die er mit dem Setup-Request bekommen hat, Eintrag für Eintrag mit der im ersten Feld gespeicherten Adresse. Stimmen die Adressen in allen gültigen Bits überein (die Anzahl der gültigen Bits steht im zweiten Feld) kommt dieser Eintrag in die nähere Auswahl. Von allen Einträgen, die in die nähere Auswahl gekommen sind, wird dann der Eintrag gewählt, der mit den meisten Bits mit der Zieladresse übereinstimmt.
Die Bedeutung der Anzahl der gültigen Bits:

- 152 Bits
  Alle 19 Bytes sind gültig. Die Adresse bezeichnet ein Endgerät, das an dem Port angeschlossen ist, der im letzten Feld steht. Wenn ein Setup-Request mit dieser Adresse kommt, wird die Verbindung direkt an dieses Endgerät geroutet.

- 104 Bits und weniger
  13 Bytes oder weniger sind gültig. Die Adresse ist nur noch ein Prefix für eine bestimmte Domain. Diese Domain ist durch den Port im letzten Feld des Eintrags erreichbar. Hat eine Zieladresse diesen Prefix und stimmt sie mit keinem weiteren Eintrag in mehr Bits überein, wird sie durch diesen Port dorthin geroutet.

- 0 gültige Bits
  Die Defaultroute. Kommt nur dann zum Zug, wenn kein anderer Eintrag paßt. Hinter diesem Port sollte ein Switch sein, der weiß, wohin die restlichen Adressen zu routen sind.

Fazit: Für die Zwecke, für die IISP konzipiert war, nämlich erste Gehversuche mit der damals noch neuen ATM Technologie zu machen, war es völlig ausreichend. 30 Switches unter Laborbedingungen zu verbinden und die Routingtabellen aufzustellen ist kein Problem. Versucht man aber IISP für größere Netzwerke einzusetzen, wird der Verwaltungsaufwand sehr hoch. Und durch kleine Konfigurationsfehler können Routingendlosschleifen entstehen, die einen Setup-Request immer im Kreis laufen lassen. Da hierbei immer neue VCIs belegt werden, sind die beteiligten Switches nach kurzer Zeit blockiert.

5.2.3 P-NNI

P-NNI = Private Network-Network Interface. Das P-NNI ist nun das "richtige" ATM Routingprotokoll. Die Spezifikation der Version 1.0 gibt es seit März 1996. Was bietet es?

- hohe Skalierbarkeit

- automatische Umgehung von ausgefallenen Switches

- automatische Distribution von erreichbaren Adressen

- Möglichkeit, verschiedene Routingstrategien parallel zu nutzen.

P-NNI arbeitet mit Source-Routing, d.h. der erste Switch der den Setup-Request bekommt, plant einen Weg durch das Netzwerk zum Ziel. Dann wird der Setup-Request diesen Weg entlang durch das Netzwerk zum Ziel geschickt.
Um derart routen zu können, brauchen die Switches eine mehr oder minder genaue Vorstellung vom Aufbau des Netzwerks. Sie müssen zunächst wissen, wo sich die Zieladresse überhaupt befindet und über welche Switches Wege dort hinführen. Da ATM auch QoS Anforderungen stellt, müssen sie genaueres über die Verbindungen auf dem Weg wissen. Wie groß ist die Bandbreite, Latenzzeit, usw. Sie müssen auch wissen, wieviele Verbindungen andere Switches noch aufnehmen können.

5.2.4 Topologiedistribution

Es muß also einen Mechanismus geben, der diese Informationen im Netz verteilt. Sehen wir uns dazu ein Beispielnetz (Abb. 55) an: Die Kringel stellen Switches ohne angeschlossene End-

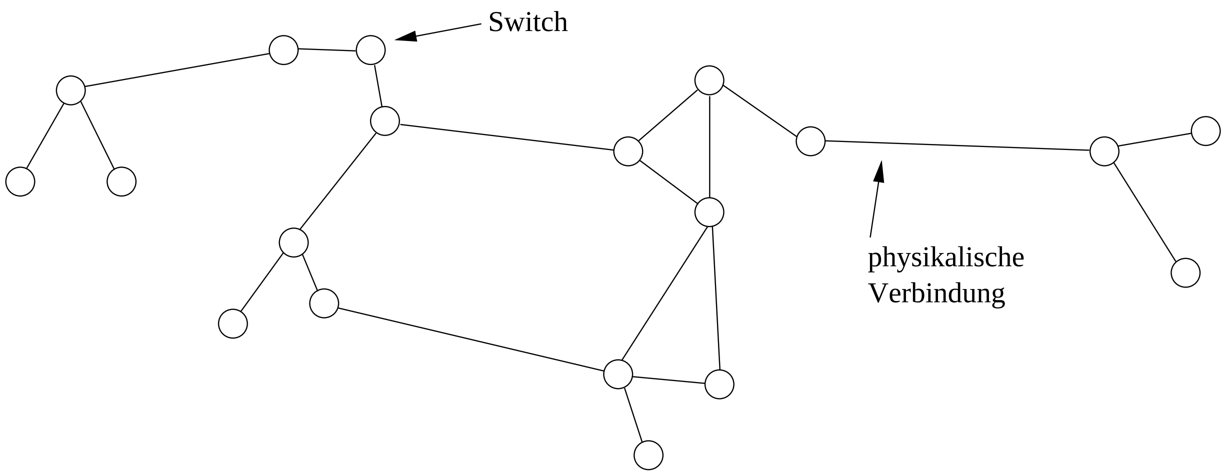


Abbildung 55: Physikalische Topologie

geräte dar. Jeder Switch hat die Adressen der Endgeräte, die an ihn angeschlossen sind über das ILMI-Protokoll erfahren. Hier im Beispielnetz ist die Anzahl der Switches natürlich noch überschaubar. Es würde kein Problem darstellen, die komplette Topologieinformation einschließlich aller Adressen und QoS Informationen im ganzen Netz zu verbreiten. In größeren Netzen jedoch muß man versuchen, die Informationsflut einzuschränken, ohne wichtige Daten zu unterschlagen. Dazu wird über der physikalischen Netzschicht eine logische Hierarchie errichtet. Zunächst werden benachbarte Switches zu sogenannten “Peergroups” zusammengefaßt.
Die Aufteilung der Switches auf die Peergroups muß der Administrator festlegen. Innerhalb der Peergroups werden Topologieinformationen durch PTSEs (PTSE = PNNI Topology State Element) ausgetauscht. PTSEs transportieren einzelne Topologieinformationen, z.B. daß A.1.1 eine Verbindung zu A.1.2 hat (dazu die Bandbreite, Verzögerung, etc.). Die Nodes setzen in ihrer Datenbank diese PTSEs zusammen und erhalten somit ein genaues Bild ihrer Peergroup. Die PTSEs werden von den Nodes entweder in regelmäßigen Abständen oder durch bestimmte Ereignisse automatisch generiert (wenn z.B. eine Leitung abgeschaltet wird).
Jede Peergroup wählt unter sich einen Peergroupleader. Dieser stellt für die Peergroup die *logische* Verbindung nach aussen dar. Auch er sammelt die Topologieinformationen seiner Peergroup

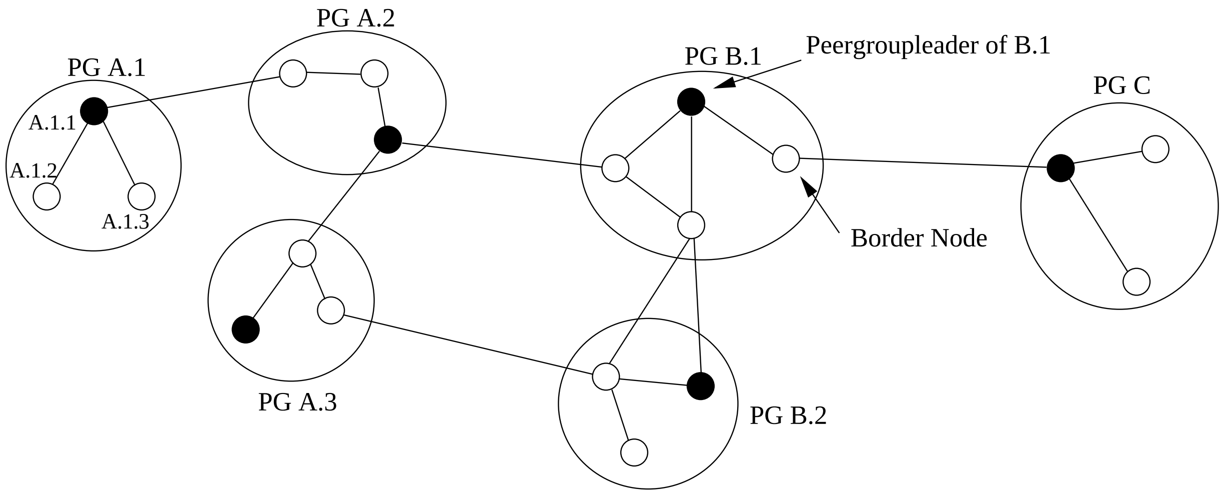


Abbildung 56: Peergroups

in einer Datenbank. Zusätzlich generiert er aus dieser eine vereinfachte Darstellung der Peergroup für die nächst höhere logische Netzwerkebene. Dort stellt er seine Peergroup als einzelnen logischen Knoten (Node) dar. In der höheren logischen Ebene erhält man eine neue Darstellung des Netzwerks, in der nun alle Mitglieder der Peergroups verschwunden sind und die Peergroups selber sozusagen die kleinsten Bausteine des Netzes sind. Verbindungen zwischen Peergroups der unteren Ebene werden zu Verbindungen zwischen logischen Nodes der höheren Ebene. Haben zwei Peergroups mehrere Verbindung untereinander, werden diese auf der höheren Ebene zu einer Verbindung zusammengefaßt, natürlich mit angepaßten Statistiken, wie z.B. einer höheren Bandbreite. (siehe Abb. 57 zwischen PG(B.1) und PG(B.2))

Um eine weitere Reduktion der Netzwerkkomplexität zu erreichen, können wiederum Gruppen von logischen Nodes zu Peergroups zusammengefaßt werden. In denen tauschen die Nodes ihrerseits Topologieinformationen mittels PTSEs aus. In diesen Peergroups können dann wieder Peergroupleader gewählt werden, die die Peergroups in einer nächsthöheren logischen Ebene vertreten usw. Siehe dazu Abb. 57.
Die Peergroupleader agieren als eine Art Gateway für Topologieinformationen zwischen den verschiedenen logischen Ebenen des Netzwerks. Sie lauschen auf der unteren Ebene und fassen die PTSEs für die nächsthöhere Ebene zusammen. Gleichzeitig lauschen sie auch auf der höheren Ebene und leiten alle PTSEs von hier *unverändert* nach unten weiter, wo sie in der Peergroup verteilt werden. Die Nodes innerhalb einer Peergroup (also “unten”) wissen dadurch wie die darüberliegenden logischen Netzwerkschichten aufgebaut sind.

Das Netzwerk aus der Sicht einer Node

Ein Switch der untersten Ebene hat nun folgendes Bild vom Netzwerk: Die detailliertesten Informationen hat er von seiner unmittelbaren Umgebung, seiner Peergroup. Hier kennt er jede Verbindung, jeden Switch und seine physikalischen Parameter. Das restliche Netz kennt er nur in einer mehr oder minder abstrakten Darstellung. Von seinen Nachbarpeergroups weiß er noch, daß es sie gibt und wie er zu ihnen hinkommt[7], aber nicht mehr, wie sie im Detail aufgebaut sind. Peergroups, die weiter entfernt sind, werden mit anderen Peergroups zusammengefaßt und

[7]siehe Abschnitt über Uplinks

erscheinen als logischer Knoten auf einer höheren Ebene.
Beispiel: Betrachten wir in Abb. 57 den Knoten A.1.2 . Zusammen mit A.1.1 und A.1.3 tauscht er PTSEs aus und weiß so von ihrer Existenz, wie sie verbunden sind und wie er sie erreichen kann. Zusammen bilden sie die Peergroup PG(A.1). Über A.1.1 als Peergroupleader wird A.1.2 mit Informationen über die Peergroup PG(A) versorgt. Er weiß somit, daß es in PG(A) noch die logischen Knoten A.2 und A.3 gibt, wie sie verbunden sind und wie er sie erreichen kann. Über A.1 als Peergroupleader von PG(A), wird PG(A) mit Informationen über die höchste logische Ebene versorgt. Diese Informationen werden von A.1.1 nach "unten" in PG(A.1) durchgereicht und kommen so auch zu A.1.2. A.1.2 kennt nun PG(A.1), PG(A) und die höchste logische Ebene. Siehe dazu Abb. 58. Vielleicht ist ein praxisnahes Beispiel anschaulicher:
An der TU München steht an einem Lehrstuhl ein Rechner mit ATM-Anschluß. Das Lehrstuhlnetz ist als Peergroup konfiguriert. Innerhalb einer Peergroup fließen Topologieinformationen frei umher. Also kennt er sämtliche anderen Lehrstuhlrechner/-switches und wie sie verbunden sind. In der nächsthöheren Peergroup sind alle Lehrstühle zusammengefaßt. Hier weiß er nur noch, daß es diese Lehrstühle gibt und über welche Bordernodes er sie erreichen kann, aber nicht mehr, welche Rechner und Switches dort im einzelnen stehen. Die nächste Hierarchiestufe könnte alle Fakultäten der TU zu einer Peergroup zusammenfassen. Auch hier weiß er zwar, daß es eine Fakultät für z.B. Elektrotechnik gibt, aber nicht wieviele Lehrstühle sie hat oder gar aus welchen Rechnern und Switches sich ein bestimmter Lehrstuhl dort zusammensetzt. Eine weitere Stufe könnte das DFN zusammenfassen: In ihr würde die UNI Regensburg nur als ein logischer Knoten auftauchen, ohne Information darüber, wie er sich zusammensetzt. Dann könnte man so noch die europäischen Netzbetreiber zu einer Peergroup zusammenfassen und die letzte Stufe würde schon die Kontinente verbinden.

Uplinks

Ein wichtiger Punkt, der bereits erwähnt wurde, ist die Rolle der "Bordernodes" und deren "Uplinks". Bordernodes sind Switches, die sich am "Rand" einer Peergroup befinden und eine physikalische Verbindung nach "aussen", also zu einer anderen Peergroup haben (siehe Abb. 56). Auch sie beteiligen sich an dem Topologieaustausch innerhalb der Peergroup. Zusätzlich generieren sie PTSEs, die aussagen, daß sie noch eine zusätzliche Verbindung nach "aussen" haben. Tatsächlich handelt es sich dabei um die physikalische Verbindung zur benachbarten Peergroup. Die logische Sicht einer Peergroup kennt jedoch die Nachbarpeergroups nicht als Peergroups, sondern nur als logische Nodes in der nächsthöheren Hierarchieebene. Deswegen wird aus der Verbindung nach "aussen" eine Verbindung nach "oben", also ein "Uplink" zu einem logischen Node, der auf einer höheren Netzwerkebene existiert und die tatsächlich vorhandene physikalische Peergroup dort repräsentiert.
Sehen wir uns dazu in Abb. 57 und 58 die PG(A.1) an. Wie wir bereits wissen, kennen die Mitglieder der Peergroup die Topologie ihrer eigenen Peergroup, also PG(A.1), die Topologie der nächsthöheren Peergroup, also PG(A) und die Topologie der höchsten Peergroup, mit A, B und C als Mitgliedern. A.1.1 hat eine physikalische Verbindung zu A.2.1 und somit zur Peergroup PG(A.2). Die PG(A.1) kennt aber die PG(A.2) nicht, sondern nur die Zusammenfassung A.2, deswegen ist die Verbindung von A.1.1 für die PG(A.1) ein Uplink zu A.2.
Für das Routing sind Uplinks Verbindungen zwischen den Hierarchieebenen. Durch den Uplink A.1.1 $\leftrightarrow$ A.2 besteht nun eine logische Verbindung zwischen PG(A.1) und PG(A), über die dann später Verbindungen *logisch* geroutet werden können. In Abb. 58 sind alle Uplinks, die PG(A.1) kennt, eingezeichnet und die Abbildung zeigt die endgültige Struktur des Netzes aus der Sicht von PG(A.1).

Adressdistribution

Die Distribution erreichbarer Adressen geschieht parallel zur Distribution der Topologie über PTSEs. ATM Adressen sind hierarchisch gegliedert (siehe Grundlagen). Die ersten Bytes sind die wichtigsten, sie sagen aus, in welchen Teil der Welt die Adresse ungefähr hingehört (ob nach Amerika, Europa, Asien, usw.). Die weiteren Bytes legen in diesem Teil dann genauer fest, wo sie liegt. Die letzten Bytes, genauer der ESI-Teil der Adresse, identifizieren zum Schluß die Endgeräte in den untersten Peergroups.
In diesen untersten Peergroups werden die vollen ATM Adressen mit den PTSEs ausgetauscht. Die Switches wissen also explizit, welche Adressen in ihrer Peergroup vorhanden und erreichbar sind. Der Peergroupleader hat nun die Aufgabe, die höheren Netzwerkschichten über die erreichbaren Adressen in seiner Peergroup zu informieren. Würde er alle Adressen einzeln weitergeben, wäre die Flut an Information über erreichbare Adressen in den höheren Netzwerkschichten nicht mehr zu bewältigen. Deswegen geht man davon aus, daß viele Mitglieder einer Peergroup den gleichen Adressprefix haben und daß es die einzigen mit diesem Prefix sind. Jetzt kann der Peergroupleader mit einigen wenigen Adreßprefixen alle bei ihm erreichbaren Adressen beschreiben. Adreßprefixe können von einem Bit bis zu 13 Bytes lang sein und müssen entweder explizit konfiguriert werden oder nehmen einen Standardwert an. Der Standardwert ist der Identifikator für die Peergroup, die Peergroup-ID. So einen Adressprefix nennt man in diesem Zusammenhang "summary address". Adressen, die sich durch "summary addresses" zusammenfassen lassen, heißen "native addresses".
Manchmal ist es aber auch nötig, aus dieser strengen Ordnung auszubrechen, z.B. wenn man das E.164 Adressierungsschema benutzt. Derartige internationale ISDN-Nummern sollen weltweit eindeutig einer Person zugeordnet werden können. Wenn diese Person nun umzieht, zieht die Nummer mit um, d.h. das ATM-Gerät mit der E.164 Nummer wechselt die Peergroup. Die neue Peergroup hat aber einen anderen Adressprefix als die E.164 Nummer und die E.164 Nummer ist "zu Hause", also in der Peergroup mit dem Prefix der E.164 Nummer, nicht mehr zu erreichen. Für diese Fälle gibt es die sog. "foreign addresses". Diese werden explizit nach "oben" weitergereicht, bis sie in eine Peergroup kommen, mit der sie eine "summary address" gemeinsam haben. Man sieht leicht, daß "foreign addresses" ihre Nachteile haben. Für jedes Endgerät, das die Peergroup wechselt, muß eine Adresse explizit bekanntgegeben werden. Je weiter so ein Endgerät umzieht, umso höher muß die "foreign address" in der Netzwerkhierarchie hinaufgereicht werden. Zieht eine Person z.B. von Asien nach Amerika um und nimmt ihre ATM Adresse mit, dann wird die "foreign address" bis auf internationale Ebene hinaufgereicht, da sie sich ja schon in der nationalen Kennung von den amerikanischen Adressen unterscheidet. Dadurch, daß alle PTSEs ungefiltert die Peergroups "hinunter" laufen, wird jeder Switch auf der Erde von dieser einen "foreign address" informiert. Eine wirksame Methode, alle Adreßtabellen auf der Welt zu füllen...
Wenn bei einem Routingvorgang eine Zieladresse gesucht wird, sucht man sie zunächst in der Liste der "foreign addresses". Ist sie dort nicht zu finden, sucht man die "summary address", die den längsten Prefix der Zieladresse darstellt. Dieses "longest match wins"-Prinzip arbeitet analog zum IISP-Routingverfahren.

Das (P-NNI) Hello-Protokoll[8]

Wenn ein Switch eingeschaltet wird, kennt er nur das, was in seiner permanenten Datenbank konfiguriert ist. Das sind Daten, wie z.B. seine Endgeräteadresse (zu Konfigurationszwecken haben auch Switches welche), seine Peergroup-ID und die Port-IDs, die Kontakt zu anderen Switches haben sollen usw. Als nächstes versucht der Switch festzustellen, mit welchen anderen

[8] nicht zu verwechseln mit dem OSI-Hello-Protokoll; im folgenden ist das P-NNI Hello-Protokoll gemeint

Switches er wirklich verbunden ist. Dazu schickt er aus allen Ports, die eine Verbindung zu anderen Switches sein sollen, in regelmäßigen Intervallen Hello-Pakete hinaus. Ist auf der anderen Seite ein aktiver Switch, beginnen die beiden Informationen ausizutauschen.

Dazu gehören u.a. die Versionen des P-NNI Protokolls, die von den Switches unterstützt werden, die Switch-/Node-ID des Nachbarswitches und die jeweilige Peergroup-ID des Nachbars. Anhand der Peergroup-ID können die Switches feststellen, ob sie in der gleichen Peergroup sind oder ob ihre Verbindung ein "Outside Link" ist, also eine Verbindung zwischen zwei Peergroups. Wenn sie in der gleichen Peergroup sind, beginnen sie Topologieinformationen mit PTSEs auszutauschen. Sind sie in verschiedenen Peergroups, dann tragen sie die Verbindung in ihrer Datenbank als Uplink ein. In beiden Fällen schicken sie sich weiterhin regelmäßig Pakete, um festzustellen, ob die Verbindung noch steht.

Bemerkungen:

- Die hierarchische Struktur eines ATM-Netzwerks ist nicht festgelegt. Theoretisch könnte ein ATM-Netzwerk aus nur einer Peergroup bestehen, oder es könnte mit jedem Bit der Adresse in zwei neue Peergroups verzweigen; das Netzwerk hätte dann die Struktur eines Binärbaums. Die Praxis liegt irgendwo dazwischen. Dabei sind auch unsymmetrische Hierarchien möglich. Ich habe versucht, das in Abb. 57 mit der PG(C) zu demonstrieren. Sowohl PG(A.1)und PG(C) sind Lowlevel-Peergroups, die beide in der höchsten Peergroup vertreten werden, einmal durch A und einmal durch C. Zwischen PG(A.1) und A liegt aber noch PG(A), während PG(C) direkt von C repräsentiert wird. Der Hierarchiebaum ist unsymmetrisch, zwischen PG(C) und C "fehlt" eine Hierarchiestufe.

- Die Adressen in den Abbildungen sind von der Form A.x.x oder PG(A.x). Die Adressen sind als abstrakt zu sehen. Es handelt sich dabei um keine vollständigen Endgeräteadressen. PG(A.x)-Adressen sind symbolische Peergroupadressen und A.x.x-Adressen sollen eine Repräsentation aller Adressen der Endgeräte sein, die an einen Switch angeschlossen sind. A ist jeweils ein Prefix der kompletten Adresse, die durch das "A" für die Abbildungen "handlicher" gemacht wurde.

- Damit sich Peergroupleader untereinander "unterhalten" können, um z.B. PTSEs auszutauschen, bauen sie untereinander mit sog. SVCCs (SVCC = Switched Virtual Control Channel) ein virtuelles Netz, das die logische Topologie der jeweiligen Ebene hat, auf. Auch hier läuft über die SVCCs das Helloprotokoll, um Informationen über gemeinsame Peergroup-IDs auszutauschen und die Verbindung durch regelmäßige Pakete zu testen.

- In der höchsten logischen Ebene gibt es keinen Peergroupleader. Er ist auch nicht nötig, es gibt ja keine höhere Ebene, mit der Topologieinformationen ausgetauscht werden muß.

- Der physikalische Switch A.1.1 in Abb. 57 ist gleichzeitig der "normale" Switch A.1.1, der Peergroupleader (PGL) von PG(A.1) und der Peergroupleader von PG(A). Prinzipbedingt ist jeder Knoten in der n-ten Hierarchiestufe selber n-mal Peergroupleader. Dadurch könnte es zu Performanceproblemen kommen. Da die Aufgaben der Peergroupleader aber sehr gut parallelisierbar sind (man kann die PGL - Tätigkeit jeder Ebene von einander trennen), sollte es im allgemeinen reichen, wenn man ein oder mehrere zusätzlichen PGL - Module in so einen Switch einbaut.

- Bei Uplinks kann es Vererbung geben. In Abb. 57 hat A.2.3 eine physikalische Verbindung zu B.1.2. Für A.2.3 und den Rest von PG(A.2) ist das ein Uplink zu B, denn erst die höchste Hierarchieebene haben sie mit B.1.2 gemeinsam und dort stellt diese Verbindung eine Verbindung nach B dar. A.2 als Verkörperung von PG(A.2) hat natürlich auch einen Uplink zu B, diesen hat A.2 von A.2.3 "geerbt".

Das eigentliche Routing

Bis jetzt ging es nur um die Verteilung von Topologieinformationen, nicht um das eigentliche Routing oder um den Verbindungsaufbau. Nachdem die Netzhierarchie konfiguriert ist und die Topologiedatenbanken aufgebaut sind, haben die Switches eine einigermaßen genaue Vorstellung von der Situation im Netz. Ihre Sicht des Netzes ist natürlich nicht immer ganz aktuell. Wenn im Netz weiter entfernt auf einmal massiv neue Verbindungen geschaltet werden, sinkt dort die Aufnahmekapazität der Switches und Leitungen. Da es eine Weile dauert, bis die Meldung über die Änderungen durch das Netz wandern, stimmen die Informationen der Switches mit der tatsächlichen Lage dort kurzzeitig nicht mehr überein. Wenn nun eine Verbindung durch diesen Bereich geschaltet werden soll, kann es so zu Fehlkalkulationen kommen, die man jedoch in Kauf nimmt. Um dieses Problem etwas zu entschärfen, gibt es ja neben dem periodischen Update der Datenbanken, die gezielte Versendung von PTSEs mit wichtigen Ereignissen (Leitungsabschaltung, Switch nicht mehr aufnahmefähig, usw.).
Der Routingvorgang beginnt mit einem Setup-Request, der über ein UNI (User Network Interface) an einen Switch geschickt wird. In diesem Setup-Request steht die Zieladresse und die QoS-Anforderungen der gewünschten Verbindung. Ausgehend von seinem Wissen über das Netz, versucht der Switch eine möglichst optimale Route durch das Netz zu planen. Wie er das genau macht, ist nicht festgelegt, doch dazu mehr in dem Abschnitt Routingstrategien.
Diese Route packt er in eine DTL, eine **D**esignated **T**ransit **L**ist. Eine DTL bestimmt einen kompletten Pfad durch eine Peergroup. DTLs sind als Stack organisiert. Für jede Routinghierarchiestufe existiert eine DTL. Jeder Eintrag ist eine Liste von Switches oder Knoten, die in der jeweiligen Peergroup durchlaufen werden sollen und ein Zeiger auf den aktuellen Knoten (wo sich der Verbindungsaufbau gerade befindet). Ganz oben auf dem Stack steht immer die Liste für die aktuelle Routinghierarchie. Diese DTL läuft nun mit dem Setup-Request durch das Netz. Auf unterster Ebene führt jeder Switch eine CAC (Connection Admission Control) aus, ob er die Verbindung noch aufnehmen kann. Wenn ja, setzt er den Pointer in der DTL um eins weiter und leitet den Setup-Request und die neue DTL an den nächsten Switch. Wenn nicht, schickt er die Verbindung zu der letzten Routinginstanz zurück **(Crankback)**, die dann versuchen kann, einen besseren Weg zu finden. Kommt ein Verbindungsaufbau an den Rand einer Peergroup, läuft die aktuelle DTL aus und die Bordernode nimmt sie vom Stack. Der nächste Eintrag auf dem Stack ist die DTL für die nächsthöhere Peergroup und der Verbindungsaufbau wird auf der nächsten Ebene fortgesetzt.
In einer höheren Peergroup repräsentiert jeder Knoten eine untere Peergroup. Passiert ein Verbindungsaufbau einen Knoten in einer höheren Peergroup, muß ein Weg durch die zugehörige Peergroup gefunden werden. Dieses Routing wird lokal von der Bordernode ausgeführt, bei der die Verbindung in die Peergroup eintritt. Sie weiß durch die DTL der übergeordneten Peergroup, zu welchem Uplink die Verbindung hingeroutet werden soll. Sie hat die vollständige Topologie der lokalen Peergroup in ihrer Datenbank und kann so eine neue DTL für diese Peergroup berechnen. Nachdem sie die DTL auf den DTL-Stack "gepushed" hat, kann der Verbindungsaufbau durch diese Peergroup stattfinden. Dabei ist zu beachten, daß weitere Routingvorgänge (rekursiv) nötig sind, wenn in dieser Peergroup Knoten sind, die wiederum Peergroups repräsentieren. Am Ende der Peergroup verläßt der Verbindungsaufbau die Peergroup und die lokale DTL wird vom Stack "gepoppt". Die Verbindung folgt nun wieder der DTL der übergeordneten Peergroup.

Routingstrategien

Konkrete Routingstrategien werden vom ATM-Forum nicht vorgegeben. Da jeder Routingvorgang einen "kompletten" Pfad bestimmt, der von anderen nur noch lokal ergänzt werden muß, ohne daß sie die ursprüngliche Routingstrategie kennen müssen, können die Hersteller verschiedenste Routingalgorithmen einsetzen. Man sollte nicht verschweigen, daß effizientes QoS-basiertes Routing immer noch ein aktuelles Forschungsthema ist. Das ATM Forum gibt zwar einen Vorschlag für einen Routingalgorithmus an, aber dieser berücksichtigt nur einen Gewichtungsparameter für Verbindungen. Es handelt sich dabei um den Dijkstraalgorithmus für minimale Wege in einem gewichteten Graphen. Ein vollständig QoS-sensitives Routing müßte aber viel mehr Parameter berücksichtigen (Bandbreite, Latenzzeit, Burstverhalten usw.) und damit steigt die Komplexität der Algorithmen. Ob sie sich in der Zukunft genügend weit drücken läßt, wird sich zeigen ...

Crankback

Schon beim Routing sollte sichergestellt werden, daß die Verbindung die QoS-Anforderungen erfüllen kann. Diese Entscheidungen kann die Routinginstanz aber nur anhand ihrer Topologiedatenbank treffen. Durch Verzögerung bei der Verteilung von PTSEs oder durch Informationsverlust bei der Topologiezusammenfassung können in der Datenbank veraltete oder einfach falsche Daten stehen. Daneben kann es sein, daß der verwendete Routingalgorithmus aus Effizienzgründen einige weniger wichtige QoS-Parameter einfach ignoriert. Das Ergebnis kann also ein Pfad sein, der den QoS-Anforderungen nicht gerecht wird, deswegen prüft jeder Switch beim Verbindungsaufbau mit einer CAC (Connection Admission Control), ob er die Verbindung noch aufnehmen kann. Schlägt die CAC fehl, passiert ein Crankback. Der Switch schickt die Verbindung mit einer Crankbackmessage und aktualisierten Topologie-PTSEs zurück zu der letzten Routinginstanz. Das kann entweder der Switch sein, von dem der Verbindungsaufbau ausgegangen ist oder es ist eine Bordernode der lokalen Peergroup, die für die Verbindung einen Weg durch diese Peergroup gesucht hat. Die Routinginstanz kann nun mit aktualisierter Datenbank versuchen, einen neuen Weg durch die lokale Peergroup zu finden oder sie erkennt, daß für die gegebenen QoS-Parameter kein Weg durch diese Peergroup führt. In diesem Fall schickt sie ihrerseits an die vorhergehende Routinginstanz eine Crankbackmeldung ("kein Weg durch diese Peergroup"). Und so läuft die Verbindung rekursiv zurück, bis entweder ein akzeptabler Weg gefunden wurde oder der Crankback an dem Switch ankommt, von dem die Verbindung ausgegangen ist. In diesem Fall meldet der Switch über die UNI, daß eine Verbindung mit diesen QoS-Parametern nicht möglich ist und der Verbindungsaufbau ist gescheitert.

5.2.5 Ein Beispiel für einen Verbindungsaufbau mit Crankback

In diesem Beispiel soll ein Verbindungsaufbau exemplarisch vorgeführt werden. Wir nehmen das Netzwerk aus Abb. 57 als Vorlage. A.1.2 soll von einem seiner Endgeräte über das UNI einen Setup-Request bekommen, der eine Verbindung zu C.3 anfordert. Weiter sollen in diesem Beispiel die physikalischen Leitungen A.3.2 $\rightarrow$ A.3.3 und B.1.2 $\rightarrow$ B.1.1 die QoS-Parameter der gewünschten Verbindung nicht erfüllen können. Die Information darüber soll irgendwo in der Topologiedistribution "hängengeblieben" sein. In Abb. 57 sehen wir das Netzwerk aus der Sicht von A.1.2. Da die Adresse nirgends als "foreign address" deklariert ist, sucht A.1.2 den geeigneten Prefix für C.3 und wird auf der höchsten Ebene bei "C" fündig. Aufgrund seiner Topologiedatenbank und seines Routingalgorithmus entscheidet sich A.1.2 für folgenden Weg:
A.1.2 $\rightarrow$ A.1.1 $\nearrow$ A.2 $\rightarrow$ A.3 $\nearrow$ B $\rightarrow$ C
Daraus generiert er einen DTL - Stack:

In der obersten Zeile sind alle Switches aufgeführt, die in PG(A.1) durchlaufen werden müssen. In der zweiten Zeile darunter steht der Weg durch PG(A) und in der letzten Zeile der Weg durch die höchste Netzwerkebene. Bevor A.1.2 den Setup-Request zu A.1.1 schickt, setzt er den Pointer der aktuellen DTL auf den Eintrag für A.1.1 (siehe Abb. 60).

Der Setup-Request läuft nun in PG(A.1) von A.1.2 zu A.1.1. A.1.1 macht einen CAC und entscheidet, daß er die Verbindung noch aufnehmen kann. Die oberste DTL ist zu Ende und wird "gepoppt". Die nächste DTL beschreibt den Weg durch PG(A). Dort läuft der Setup-Request von A.1 (bereits erledigt) zu A.2. Physikalisch wird dabei die Verbindung von A.1.1 zu A.2.1 weitergereicht, von Bordernode zu Bordernode. Als nächstes soll A.2 passiert werden. Dazu muß in der zugehörigen PG(A.2) ein Weg von A.2.1 zum Uplink nach A.3 gefunden werden. A.2.1 routet entlang dem einzig möglichen Weg:
A.2.1 → A.2.2 → A.2.3
Die DTL für die lokale Peergroup wird auf den Stack "gepushed":

Nehmen wir an, daß alle CACs auf dem Weg glücken und damit steht die Verbindung physikalisch bei A.2.3. Die DTL für PG(A.2) ist zu Ende und wird "gepoppt". Die nächste Station in der DTL für PG(A) ist A.3. A.3.2 routet durch PG(A.3):
A.3.2 → A.3.3
Bei der lokalen CAC für die physikalische Verbindung von A.3.2 nach A.3.3 gibt es den vorher angesprochenen Fehler. Die QoS-Anforderungen können nicht erfüllt werden. ⇒ Crankback. Die Verbindung läuft zur letzten Routinginstanz zurück, in diesem Fall zu A.3.2. A.3.2 kann keinen alternativen Pfad durch PG(A.3) festlegen, der Verbindungsaufbau durch diese Peergroup ist gescheitert. A.3.2 reicht den Crankback weiter zurück, zu der aus seiner Sicht letzten Routinginstanz. Das ist in diesem Fall A.1.2. Hier ist zu beachten, daß A.2.1 zwar als letztes geroutet hat, aber nur die Route durch PG(A.2) festgelegt hat, und nicht an der Entscheidung beteiligt war, die Verbindung durch PG(A.3) zu routen. Diese Entscheidung hat A.1.2 getroffen und muß nun den Crankback auffangen. Der Crankback läuft also zurück:
A.3.2 → A.2.3 → A.2.2 → A.2.1 → A.1.1 → A.1.2
A.1.2 weiß nun, daß der Weg durch A.3 blockiert ist. Wie aus Abb. 58 ersichtlich ist, führt der einzig andere mögliche Weg über:
A.1.2 → A.1.1 ↗ A.2 ↗ B → C
Durch PG(A.1) und PG(A.2) läuft der Setup-Request wie beim ersten Mal. Bei A.2.3 läuft die lokale DTL aus und wird gepoppt. Die nächste DTL ist die DTL, die den Weg durch PG(A) beschreibt. Auch sie ist zu Ende und wird gepoppt. Die letzte DTL auf dem Stack beschreibt den Weg in der höchsten Peergroup. A ist durchlaufen und der nächste Knoten ist B. A.2.3 hat über die physikalische Verbindung A.2.3 → B.1.2 einen Uplink zu B. Physikalisch wandert der Setup-Request von A.2.3 nach B.1.2. Logisch wechselt er in der höchsten Ebene von A nach B. B soll durchlaufen werden, also muß PG(B) durchquert werden. Die Verbindung tritt bei B.1 in PG(B) ein. B.1 soll als Bordernode einen Weg zu C routen. In diesem Fall hat B.1 selbst einen Uplink zu C. Es muß also nur B.1 selbst durchquert werden. Dazu muß B.1.2 einen Weg durch PG(B.1) zu einem Uplink nach C finden. B.1.2 routet nach seiner Topologiedatenbank und legt seine DTL auf den DTL-Stack:

Der Setup-Request läuft von B.1.2 nach B.1.1. B.1.1 führt seine CAC aus und merkt, daß sein Eingangsport die QoS-Anforderungen nicht erfüllen kann. Er schickt die Verbindung mit einem Crankback zu B.1.2 zurück. B.1.2 errechnet eine Ausweichroute:
B.1.2 → B.1.4 → B.1.1 → B.1.3

Diese Route führt zwar auch durch B.1.1, aber nicht durch den überlasteten Port und so klappt diesmal der Verbindungsaufbau. Bei B.1.3 laufen dann sowohl die DTL der lokalen Peergroup als auch die DTL für PG(B) aus. Die Verbindung wechselt in der höchsten Peergroup nach C und da C ein Prefix von C.3 ist, ist die Verbindung aus der Sicht der höchsten Peergroup am Ziel. C.1 hat nun noch die Aufgabe, die Verbindung in PG(C) zu C.1 zu routen. Die resultierende DTL: Die CACs glücken, der Setup-Request kommt bei C.3 an und die Verbindung steht.

5.3 Ausblick

Natürlich war das nicht alles, was es zu den Themen ILMI, IISP und P-NNI zu sagen gibt. Neben vielen Büchern sind die ergiebigsten Informationsquellen die Spezifikationen des ATM-Forums. Für meinen Vortrag und meine Ausarbeitung habe ich hauptsächlich folgende Quellen benutzt: [6], [3] und [7]. In [7] stehen noch ein paar interessante Dinge, die ich hier aus Platz- und Zeitgründen weggelassen habe:

Peergroupleader-Election-Algorithm Dabei wird der Peergroupleader nach demokratischem Vorbild gewählt

Complex Node Representation wie die Struktur von Peergroups effizient zusammengefaßt werden kann

Address Scoping Adressen können in der Erreichbarkeit eingeschränkt werden, z.B. für firmeninterne Server

Dinge, die P-NNI nicht beherrscht:

1. Keine Meldung über verfügbare Netzressourcen. Zur Zeit kann man nicht abfragen, welche QoS-Anforderungen das Netz für eine Verbindung zu einer bestimmten Adresse erfüllen kann. Man muß sich mit wiederholten Verbindungsversuchen herantasten. Nicht sehr effizient ...

2. Dynamische Umlegung von Verbindungen zum Lastausgleich. Steht eine Verbindung einmal, kann sie nicht mehr auf andere Switches verschoben werden, auch wenn die alternative Route die gleichen QoS-Anforderungen erfüllt und die momentane für andere Verbindungen dringender gebraucht wird.

3. Loadbalancing. Loadbalancing könnte in ATM nur von höheren Protokollschichten implementiert werden, da die einzelnen ATM-Zellen keine Seriennummer tragen. P-NNI bietet aber keine Möglichkeit mehrere Routen zu einem Ziel gleichzeitig zu finden. Die höheren Protokollschichten stehen wieder vor dem gleichen Problem wie in 1. Sie müßten die Verbindungen durchprobieren, auch nicht effizient ...

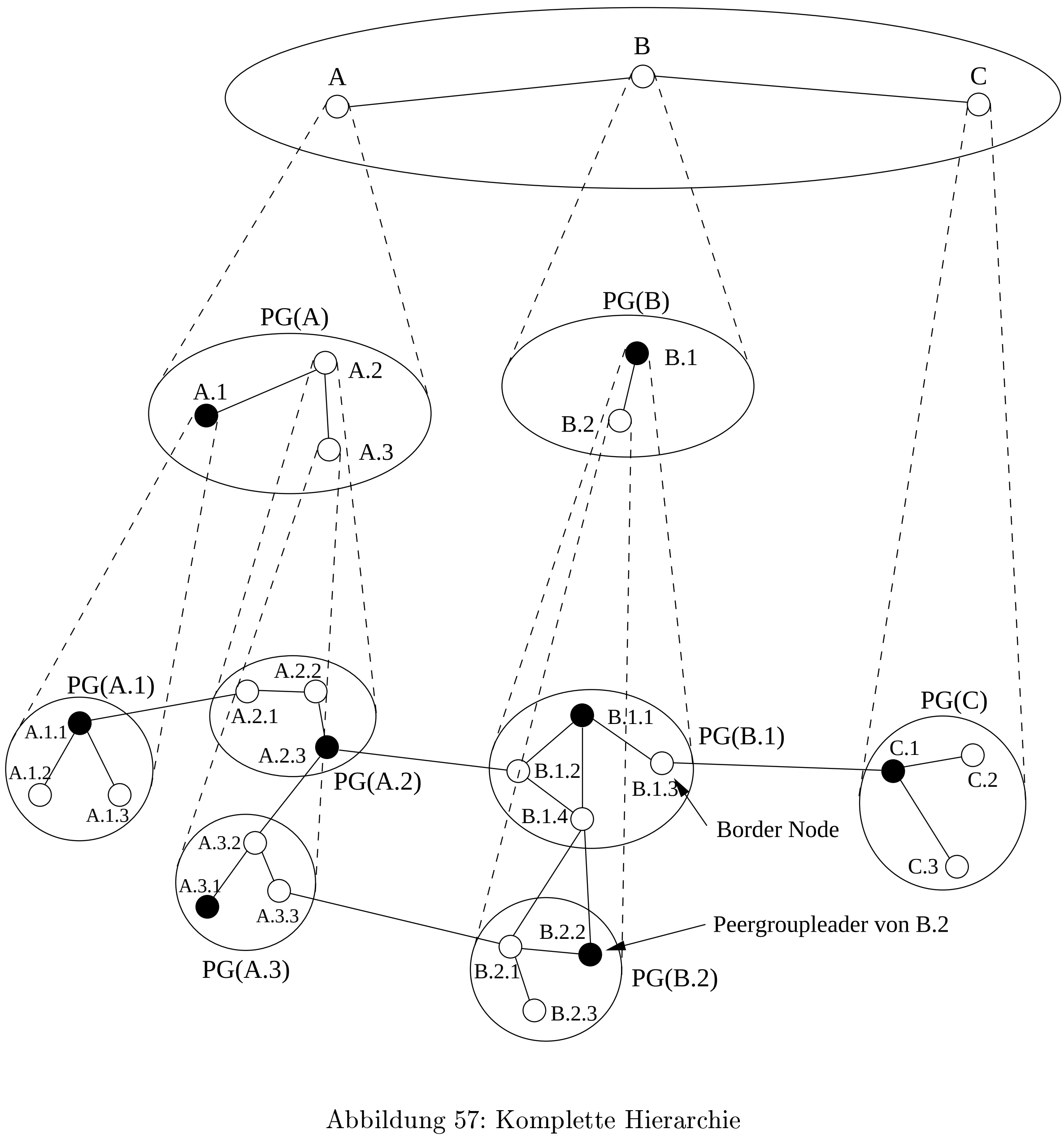


Abbildung 57: Komplette Hierarchie

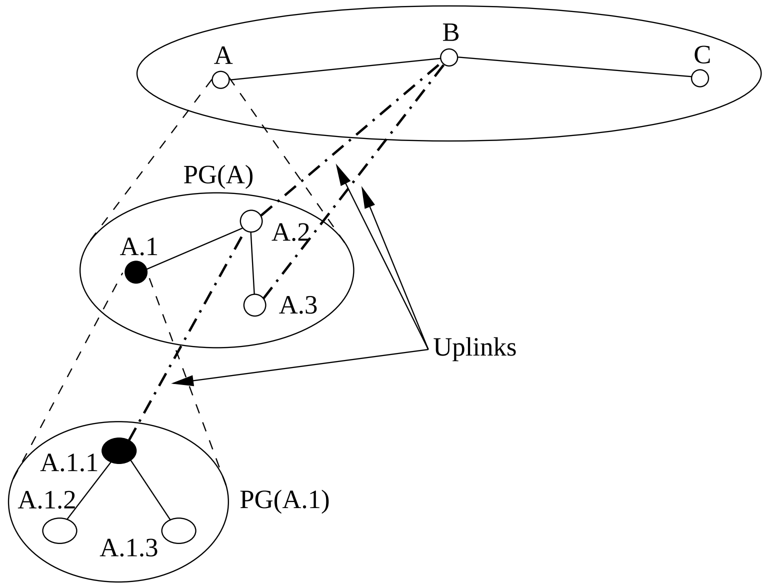


Abbildung 58: Das Netzwerk aus der Sicht einer Node

DTL:	[A.1.2, A.1.1]	pointer-1
DTL:	[A.1, A.2, A.3]	pointer-1
DTL:	[A, B, C]	pointer-1

Abbildung 59: DTL - Stack Original

DTL:	[A.1.2, A.1.1]	pointer-2
DTL:	[A.1, A.2, A.3]	pointer-1
DTL:	[A, B, C]	pointer-1

Abbildung 60: DTL - Stack bei A.1.1

DTL:	[A.2.1, A.2.2, A.2.3]	pointer-1
DTL:	[A.1, A.2, A.3]	pointer-2
DTL:	[A, B, C]	pointer-1

Abbildung 61: DTL - Stack in PG(A.2)

DTL:	[B.1.2, B.1.1, B.1.3]	pointer-1
DTL:	[B.1]	pointer-1
DTL:	[A, B, C]	pointer-2

Abbildung 62: DTL - Stack bei B.1.2

DTL:	[C.1, C.3]	pointer-1
DTL:	[A, B, C]	pointer-3

Abbildung 63: DTL - Stack in PG(C)

6 Protokolle der Anwendungsschicht

Mit Hilfe von ATM Netzwerken wird es möglich sein viele verschiedene Dienste zu integrieren. Die dafür benötigten, dienstabhängigen Protokolle werden in der Anwendungsschicht zusammengefaßt.

Die Protokolle der Anwendungsschicht lassen sich grob in drei Gruppen einteilen: Protokolle zur Datenkommunikation, zur Sprachkommunikation und zur Videoübertragung. Ein weiterer interessanter Dienst sind Videokonferenzen, dafür werden jedoch meist nur Sprach- und Videoübertragungsprotokolle gemultiplexed.

6.1 Datenkommunikation

Die Protokolle zur Datenkommunikation werde ich nur kurz behandeln, da diese in Kapitel 9 detailiert besprochen werden.

Ich werde mich hier auf die drei Protokollgruppen zur „Portierung" vorhandener LANs beschränken.

6.1.1 Classical IP over ATM

Dieses, durch die IETF[9] standardisierte Protokoll dient dazu, IP-basierende Netze auf ATM-Strecken zu ermöglichen. Es ist das einzige Protokoll, bei dem es möglich ist gewisse ATM Eigenschaften auszunutzen.

Ein „Classical IP over ATM" Netz besteht aus einem oder mehreren „Logical IP Subnets" (LIS) in dem die Clients ähnlich einem normalen IP-Subnets miteinander kommunizieren können. Dennoch kann zwischen zwei LIS nur mittels eines normalen IP-Routers kommuniziert werden.

Innerhalb eines jedem LIS gibt es einen Server, bei dem sich jeder Client anmelden muß. Dieser Server speichert eine vollständige Tabelle, die alle IP-Nummern und die dazugehörige ATM-Adressen enthält. Wenn ein Client jetzt mit einem anderen kommunizieren will, erfragt er die ATM-Adresse seines Partners und baut eine SVC Verbindung zu ihm auf. Dabei handelt es sich um eine ABR Verbindung. Es ist auch möglich andere Dienstklassen zu benutzen (CBR, VBR oder UBR), dann ist es jedoch nötig eine permanente Verbindung zwischen den Clients einzurichten.

Auch bei Ethernet-basierendem IP wird vor einer Verbindung erst, mittels ARP[10], die MAC-Adresse des Partners ermittelt. Für ATM wurde mit ATMARP ein analoges Verfahren entwickelt und es besteht im Protokollstack kein Unterschied zum normalen IP (siehe Abb. 64).

Da es sich bei ATM jedoch nicht um ein „shared medium" handelt, sind Broadcasts nicht möglich. Diese sind jedoch bei IP in den seltensten Fällen notwendig, womit „Classical IP over ATM" eine einfache Möglichkeit bereitstellt IP-basierte Netze auf ATM abzubilden.

6.1.2 LAN-Emulation

Bei LAN-Emulation (LANE) handelt es sich um eine Protokollgruppe, die es ermöglicht jedes Protokoll über ATM zu betreiben. Dabei bleibt es dem eigentlichen Protokoll vollkommen verborgen, daß es sich nicht mehr um ein Ethernet, sondern um ATM handelt. Es können also

[9] IETF: Internet Engineering Task Force
[10] Address Resolution Protocol

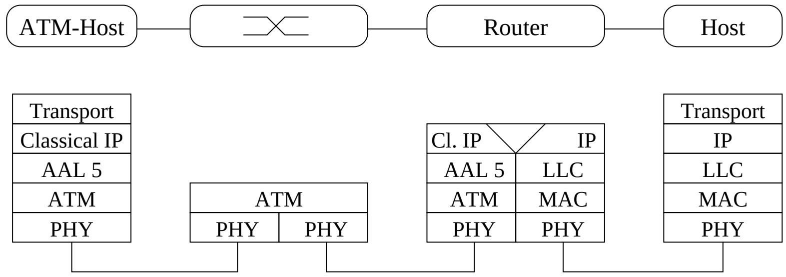


Abbildung 64: Koppelung Classical IP over ATM und konventionelles LAN

keinerlei ATM-Eigenschaften ausgenutzt werden, dafür kann jedoch jedes Protokoll über ATM benutzt werden.

Auch bei LANE werden auf dem ATM-Netzwerk virtuelle Netzwerke aufgebaut. Diese werden als ELANs (emulierte LANs) bezeichnet. Wie bei „Classical IP“ ist es nur möglich mittels eines externen Routers zwischen zwei ELANs zu kommunizieren.

Der Aufbau eines ELANs ist jedoch weitaus komplizierter. Ein neuer Client, als LANE-Client (LEC) bezeichnet, muß sich am Anfang bei einem LANE-Server (LES) anmelden. Dabei überträgt er die MAC-Adresse, die er emulieren will, und etwaige Routing-Informationen zum LES. Vom LES erhält er die Adressen des BUS und LECS.

Beim BUS handelt es sich um einen Server für „Broadcast and Unknown Services“. Zu diesem werden alle Pakete geschickt, deren Empfänger nicht bekannt ist. Der BUS baut dann seinerseits zu jedem LEC einen Verbindung auf und schickt das Paket weiter. Da das verwendete LAN Protokoll nicht weiß, daß es nicht mehr an ein „shared medium“ angeschlossen ist, erzeugt der BUS sehr viel Verkehr, auch wenn das Paket nur für einen LEC interessant ist.

Der LANE Configuration Server (LECS) verwaltet die Zugehörigkeit der LECs zu verschiedenen ELANs. Damit kann ein LEC verschiedenen ELANs angehören.

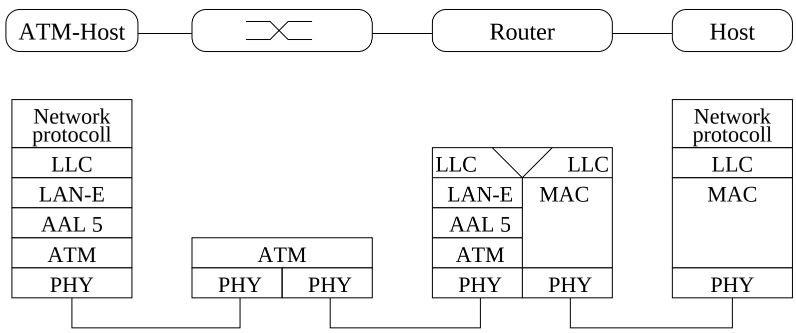


Abbildung 65: Koppelung LAN-E und konventionelles LAN

Wie man in Abb. 65 sehen kann, werden zwei weitere Schichten in den Protokollstack eingefügt. Dies erhöht den Protokoll-Overhead. Da jeder Server nur einmal vorkommen darf, sind solche ELANs leider fehleranfälliger als nötig.

Im Moment wird an einer verbesserten Version, LANE 2.0, gearbeitet (siehe Kapitel 9).

6.1.3 MPOA

„Multi Protocol over ATM“ (MPOA) ist ein Ansatz die Eigenschaften von „Classical IP“ und LANE zu verbinden und eine bessere Ausnutzung der Fähigkeiten von ATM zu erreichen.

Neben der Möglichkeit zur Schaffung von Redundanz und der direkten Kommunikation zwischen LIS bzw. ELANs wurde auch ein spezielles Protokoll (NHRP, Next Hop Resolution Protokoll) entwickelt mit dem zwei MPOA-Clients, wie bei „Classical IP“, eine eigene Verbindung zwischeneinander aufbauen können. Diese Verbindung kann, im Gegensatz zu IP auch zwischen verschiedenen ELANs verlaufen.

6.2 Sprachkommunikation

Für Sprachkommunikation über ATM werden normalerweise Verfahren benutzt, die schon seit langem von der ITU standardisiert sind. Diese Verfahren, ITU G.72x, erzeugen alle einen konstanten Datenstrom und sind somit für CBR-Verbindungen ideal geeignet. Bis jetzt gibt es noch kein Verfahren, das Gesprächspausen erkennt und die Datenrate in diesem Fall senkt. Dies war, da bis jetzt immer nur dedizierte Kanäle einer gegebenen Bandbreite verwendet wurden, nicht notwendig.

6.2.1 Sampling, PCM

Das Umwandeln des analogen Eingangssignals in ein digitales Signal wird als „Sampling“ bezeichnet. Dabei unterscheidet man nach Bit-Tiefe, auch Quantisierungsstufe, und Bandbreite. Je größer die Bandbreite, desto höhere Frequenzen können digitalisiert werden. Generell gilt: die Bandbreite muß mindestens doppelt so hoch sein, wie die höchste zu digitalisierende Frequenz (Abtasttheorem). Die Bit-Tiefe gibt an, wieviele verschiedene Stufen aufgezeichnet werden können.

Die Datenrate ergibt sich direkt als ein Produkt aus Bit-Tiefe und Bandbreite.

Ein Sampling in CD-Qualität, also mit 16 Bit und 44.1 KHz Stereo, kann somit Frequenzen bis zu 22.05 KHz aufzeichnen und erzeugt eine Datenrate von 1411.2 Kbit/s = 176.4 KByte/s.

Die durch Sampling gewonnen digitalen Werte liegen im Puls Code Modulation Format (PCM) vor. Dabei handelt es sich um eine lineare Repräsentation der analogen Signale.

Angenommen unser Mikrofon liefert Werte zwischen 0V und 1V, wobei 0.5V die Grundspannung („silence“) ist, so wird dies entweder durch $0 \ldots 255$ (mit $0 \equiv 0V$, $128 \equiv 0.5V$, $255 \equiv 1V$) oder $-128 \ldots 127$ (mit $-128 \equiv 0V$, $0 \equiv 0.5V$, $127 \equiv 1V$) repräsentiert.

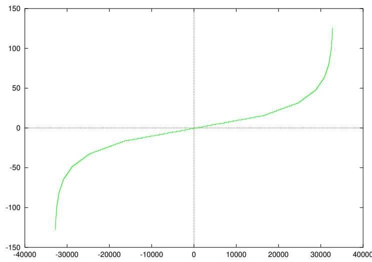
6.2.2 a-law bzw. μ-law Kodierung, G.711

Bei der a-law bzw. μ-law Kodierung werden die Eingangssignale mittels einer logarithmischen Kurve kodiert. Dies entspricht dem menschlichen Gehör, das laute Geräusche nicht so gut unterscheiden kann, besser als die lineare Kodierung von PCM. Es ist also möglich mit einer geringeren Datenrate dieselbe Qualität zu übertragen.

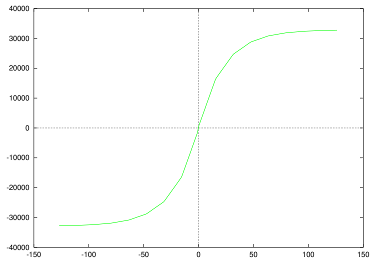
Bei G.711 wird ein 14 Bit Sample mittels a-law, in Amerika und Asien, bzw. μ-law, in Europa, auf 8 Bit reduziert. Während bei μ-law „silence“ durch Null und Signale im Bereich ± 127 um Null liegen, wird bei a-law 127 als „silence“ interpretiert, und die Signale liegen im Bereich von

127 ± 127. Auch bei der Kodierung selbst, die meist mittels Tabellen implementiert ist, gibt es kleine Unterschiede.

Abb. 66 zeigen die Umwandlungsfunktionen von 16 Bit PCM nach 8 Bit μ-law, bzw. umgekehrt, gemäß [1].



(a) linear $\rightarrow$ μ-law



(b) μ-law $\rightarrow$ linear

Abbildung 66: μ-law Kodierung

Dieses Verfahren wird momentan im ISDN verwendet und ist zur Übertragung von Sprache gut geeignet. Mit 64 Kbit/s ist die Datenrate jedoch relativ hoch.

6.2.3 Adaptiv-Differential-PCM, G.721

Bei G.721 wird nicht direkt der eigentliche Wert gespeichert, sondern nur die Differenz zum vorhergehenden Wert. Dadurch kann der Datenstrom auf 32 Kbit/s reduziert werden. Somit bleiben für ein Sample, bei 8 KHz, nur 4 Bit. Würde man jetzt einfach den Unterschied zum vorherigen Sample speichern, dürfte sich das Signal nur um ± 8 ändern. Ein solches Verfahren existiert und wird als Differential-PCM (DPCM) bezeichnet.

Bei der Adaptiv-Differential-PCM wird statt dessen ein Index auf eine Tabelle gespeichert, der auf eine zweite Tabelle verweist, in der die eigentliche Differenz gespeichert ist. Der Aufbau der Tabellen ist auf Sender- und Empfängerseite möglich, da er sich durch die übertragenen Indizes ergibt. Ein weiterer Vorteil des verwendeten Verfahrens, gegenüber PCM ist die eingeschränkte Fehlertoleranz. Nach 88 Samples hat sich das Signal wieder rekonstruiert.

Die Eingangsdaten für ADPCM müssen als lineare PCM vorliegen. Es können 16 Bit Daten mit dem Faktor 4:1 komprimiert werden. Aus 16 Bit, 8 KHz wird somit ein Datenstrom mit 32 Kbit/s. Dieses Verfahren ist primär auf die Übertragung von Sprache ausgelegt und erreicht, obwohl es sehr einfach ist, ISDN Qualität.

6.2.4 Sub-Band-ADPCM, G.722

Gegenüber G.721 wird bei G.722 mit doppelter Bandbreite gesamplet (16 Bit, 16 KHz). Somit wird eine bessere Qualität erreicht, die sogar für die Übertragung von Musik geeignet ist.

Das Verfahren basiert auf einer Zerlegung des Eingangssignals in ein oberes Frequenzband (8 KHz – 16 KHz) und ein unteres Frequenzband (0 KHz – 8 KHz). Da das menschliche Ohr im oberen Frequenzband nicht so empfindlich ist, kann dieses stärker komprimiert werden (Faktor 8:1, von 64Kbit/s auf 16 Kbit/s).

Das untere Frequenzband wird entweder 2:1, 8:5, 4:3 komprimiert (aus 64 Kbit/s werden 32, 40 oder 48 Kbit/s).

Da zur Übertragung ein 64 Kbit/s Kanal benutzt wird, bleiben noch 0, 8 oder 16 Kbit/s für die Übertragung eines Datenkanals.

Abbildung 67 stellt den Algorithmus schematisch dar, nach [39].

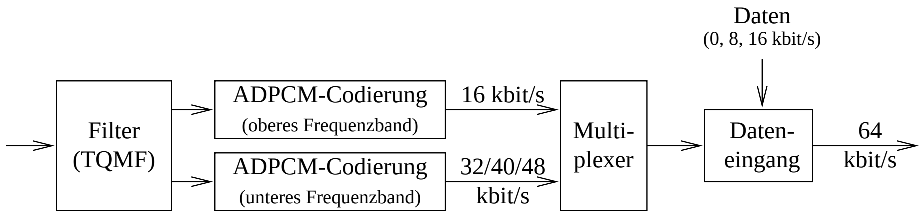


Abbildung 67: SB-ADPCM Kodierung

6.2.5 Advanced Code Excited Linear Predictor, G.723

Bei G.723 handelt es sich um eine Kombination aus ACELP und TrueSpeech. Beide Verfahren wurden zur Übertragung von Sprache mittels eines Analogmodems entwickelt.

Es wird versucht das aufgenommene Signal mittels eines Sprach-Modells zu beschreiben. Bei dem ursprünglichen Verfahren (CELP) wurden nur die Parameter für das Modell übertragen. Da das Ganze wie eine einfache Computerstimme klang, wird bei G.723 neben den Modell-Parametern noch ein Fehlerparameter übertragen, der beschreibt wie sehr sich das Modell vom aufgenommenen Signal unterscheidet. Damit wird allemal eine Qualität erreicht, die für eine Audiokonferenz ausreicht.

Der große Vorteil ist jedoch die geringe Datenrate von nur 4.8 Kbit/s (bei 8 Bit μ-law PCM mit 8 KHz Eingangsdaten).

6.2.6 Low Delay - CELP, G.728

Auf ACELP basierend, benutzt LD-CELP ein deutlich größeres Bewertungsfenster. Es werden die letzten 5 Samples betrachtet, dadurch ist eine bessere Vorhersage möglich. Durch eine Rückkopplung, des letzten Ausgabewertes, und einer gewichteten Filterung, der Eingangsdaten, wird das Sprachmodell weiter verbessert.

Im Vergleich zu G.723 wird versucht, den Mean Square Error[11] möglichst klein zu halten. Somit kommen unhörbare, kleine Abweichungen häufiger vor als hörbare, große. Bei G.723 wurde versucht den absoluten Fehler gering zu halten, wodurch große und kleine Fehler gleich häufig vorkommen.

Der resultierende Datenstrom von 16 Kbit/s (auch 12.8 und 14.4 Kbit/s möglich) erreicht für Sprache eine deutlich bessere Qualität als ISDN.

[11] Summe der Fehlerquadrate: $\frac{1}{n}\sum_{i=1}^{n} e_i^2$ mit $e_i = i$-ter Fehler

6.2.7 MPEG Audio

Bei MPEG handelt es sich um eine Gruppe von Standards zur Komprimierung von Video. Da die MPEG-Audio-Codecs auch einzeln Verwendung finden, bespreche ich sie hier.

Im Vergleich zu G.723 und G.728 wird hierbei nicht versucht das Signal gemäß der menschlichen Sprache zu modellieren, sondern gemäß des menschlichen Gehörs.

Das menschliche Gehör ist fähig Signale zwischen 20 Hz und 20 KHz wahrzunehmen. Dabei ist es um Bereich von 2 KHz bis 4 KHz am empfindlichsten. Abbildung 68 zeigt ab welcher Lautstärke (in [dB]) ein Mensch eine gegebene Frequenz hört.

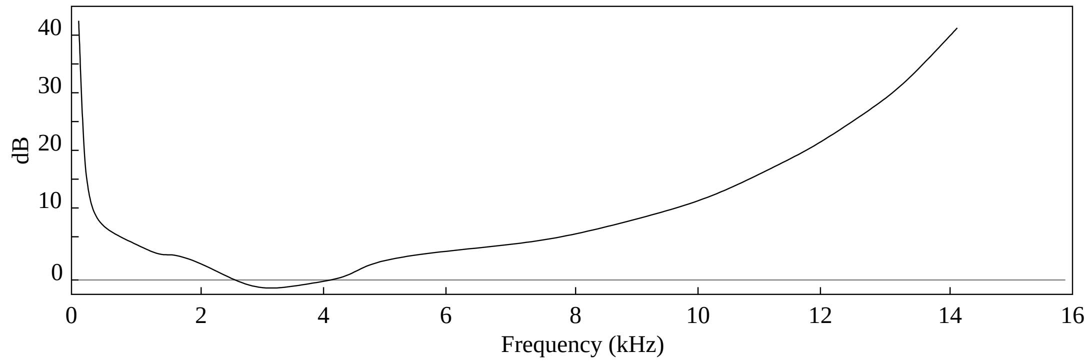


Abbildung 68: Empfindlichkeit des menschlichen Ohres

MPEG-Audio benutzt weitere Eigenheiten des menschliche Ohres. So verdecken laute Geräusche leisere Geräusche in ihrer Nähe (benachbarte Frequenzen). Darüber hinaus ist das Ohr nach einem lauten Geräusch für eine Zeit nicht mehr empfindlich genug um leise Geräusche wahrzunehmen.

All diese psychoakkustischen Merkmale werden bei MPEG-Audio benutzt um den Datenstrom möglichst effektiv zu verringern.

Das Eingangssignal, meist 16 Bit, 44.1 KHz, wird zuerst in 32 signifikante Frequenzbänder aufgespalten. Für jedes Band wird berechnet, wie sehr es die Nachbarbereiche übertönt. Bänder die komplett von Nachbarbändern überlagert werden, werden nicht weiter betrachtet, und auch nicht kodiert. Jetzt wird bewertet, wieviele Bits gebraucht werden, um den hörbaren Anteil eines jeden Bandes zu kodieren. Damit kann dann der Bitstrom kodiert werden. Des weiteren wird z.B. die Ähnlichkeit der beiden Kanäle bei Stereo Signalen beachtet.

Mit dem neueren Standard MPEG-2-Audio sind neue Samplingfrequenzen (neben 32, 44.1 und 48 KHz auch 8, 11, 16, 22 und 24 KHz) und mehr Kanäle (zusätzlich zu Mono, dual-Channel[12], Stereo und joined-Stereo[13] sind jetzt 5 Kanäle und ein spezieller Baß-Kanal möglich) hinzugekommen. Dies war nötig, da MPEG-2 für digitales Home-Kino (HDTV) gedacht ist.

Sowohl bei MPEG-1, als auch bei MPEG-2 werden 3 verschiedene Kodierungslayer unterstützt, die alle zueinander kompatibel sind. Je höher der Layer, desto besser kann bei gleicher Qualität komprimiert werden. Dies wird durch immer bessere Ausnutzung des psychoakkustischen Hörmodelles erreicht. Natürlich steigt mit jedem Layer der Kodierungs- und Dekodierungsaufwand. Da immer mehr Samples pro Schritt bewertet werden, steigt auch die theoretische

[12] 2 unabhängige Mono Kanäle

[13] Bewertung der Ähnlichkeit des linken und rechten Signals

minimale Verzögerung[14]

6.2.8 Advanced Audio Codec

Bei diesem sehr neuen Verfahren, erste Implementierungen werden erst 1998 erwartet, wird ein ähnliches Vorgehen wie bei MPEG-Audio verwendet. Da jedoch auf sämtliche Kompatibilität verzichtet wurde, sind bessere Ergebnisse möglich. Angestrebt wurde, mit einer Datenrate von 64i Kbit/s CD-Qualität zu erreichen.

Eine frühe Implementierung von AT&T benötigt noch 96 Kbit/s. Allerdings hat AT&T noch weitere Features integriert (wie Liedertexte). Eine genaue Analyse dieses Codecs ist jedoch aufgrund fehlender Dokumentation und Quelltexte nicht möglich.

6.2.9 Zusammenfassung

Alle der hier vorgestellten Kodierungsverfahren erzeugen einen Datenstrom mit fester Datenrate. Somit eignen sie sich für die Benutzung der ATM CBR-Dienste. Da meist PCs oder Workstations zur Kodierung benutzt werden, wird AAL5 verwendet. Manche Geräte benutzen auch AAL1.

Man sollte noch anmerken, daß für Audioübertragung neben der reinen Datenrate noch andere Parameter wichtig sind. So sollte die Verzögerung (end-to-end) 400 ms nicht überschreiten, auch nicht bei starkem Jitter[15]. Da Rauschen bei Audio viel stärker auffällt als bei Video sollten es möglichst keine Verluste geben. Sollte es zu Schwankungen in der Datenrate kommen, so ist eine Pufferung beim Empfänger nötig.

Ein weiteres Problem bei Audioübertragung ist das Echo, das vor allem bei langen Verzögerungen immer problematischer wird. Die Echobehebung ist jedoch Aufgabe des Codecs.

Bei den gut komprimierenden Codecs muß man noch bedenken, daß die Kodierung Zeit kostet, was die Verzögerung erhöht. Da die Daten besser gepackt sind, dauert es auch länger eine Zelle zu füllen.

6.3 Videoübertragung

Übertragung von digitalen Videodaten wird immer populärer, sei es für Video on Demand oder hochauflösende Videobilder für TeleMedizin. Hier werde ich die gebräuchlichen kurz beschreiben.

6.3.1 JPEG

Bei JPEG handelt es sich eigentlich um einen Standard zur Komprimierung von Standbildern. Da JPEG-Komprimierung jedoch die Grundlage für alle weiteren Video-Codecs bildet, werde ich sie hier beschreiben.

Ähnlich wie MPEG-Audio benutzt JPEG Eigenheiten des menschlichen Auges, um die Datenmenge eines Bildes, verlustbehaftet aber unmerklich, zu verringern.

Ausgehend von einem RGB Bild, wird dieses zuerst in den YUV-Farbraum konvertiert. Die Y Komponente speichert die Helligkeit, U und V (auch als Cr und Cb bezeichnet) speichern die dazugehörige Farbinformation. Da das Auge jedoch für Helligkeitsunterschiede empfindlicher ist,

[14] Layer 1: 19 ms, Layer 2: 35 ms, Layer 3: 59 ms. Die effektive Verzögerung ist nochmal um den Faktor 3 größer.

[15] Schwanken der Verzögerung

als für Farbunterschiede wird einem 1:1:1 RGB Bild meist als 4:1:1 YUV, für 4 RGB-Pixel also 4 Y-Werte und je ein U- und V-Wert, gespeichert.

Die folgenden Schritte (Abb. 69) werden für jede Ebene (Y, U und V), die auch als „Plane“ bezeichnet werden, unabhängig durchgeführt.

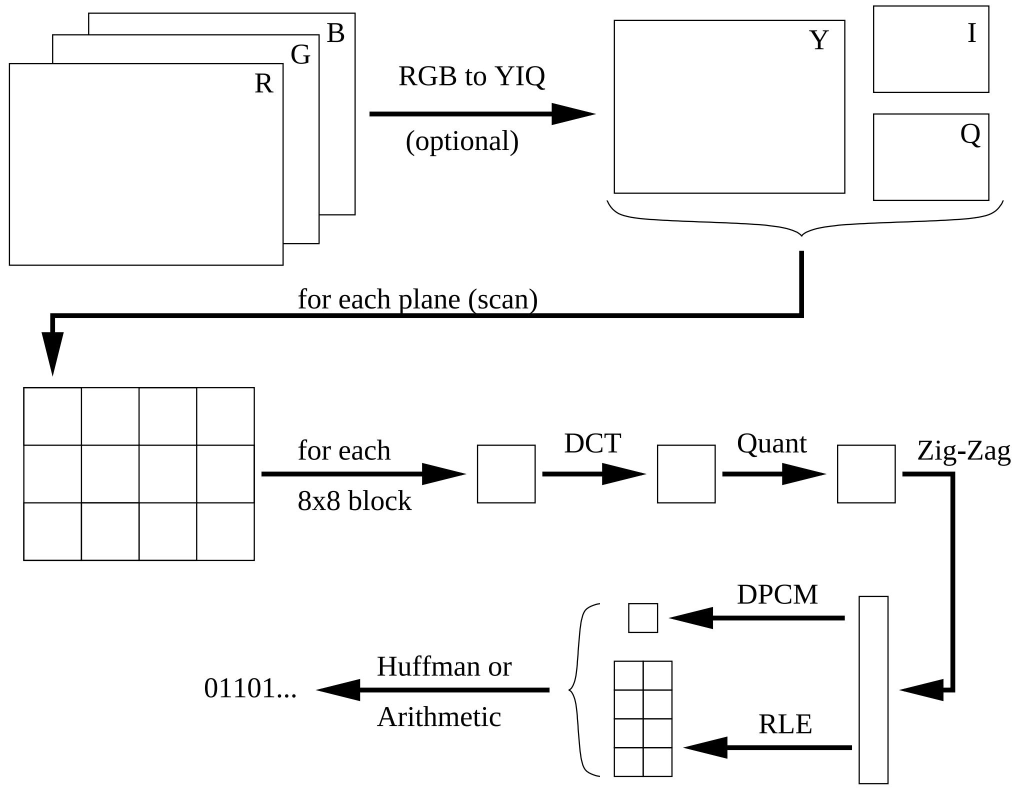


Abbildung 69: JPEG Kodierung

Zuerst wird die Ebene in 8×8 Pixel Blöcke unterteilt. Jeder dieser Blöcke wird mittels einer Diskreten Cosinus Transformation (DCT) in seine Frequenzkomponenten zerlegt. Da die DCT 64 Basisfunktionen hat, erhält man ein 8×8 Feld. Das Auge ist für verschiedene Frequenzen unterschiedlich empfindlich, deshalb werden die DCT-Komponenten jetzt quantisiert. Dabei handelt es sich um eine Division ohne Rest.

Als nächstes wird das 8×8 Feld mittels Zig-Zag-Scan[16] zu einem 64 Element Vektor umgewandelt. Das erste Element wird als AC-Komponente bezeichnet. Von den restlichen Elementen wird mittels DPCM nur der Unterschied zum vorherigen Element gespeichert. Durch die Quantisierung und die DPCM erhält man jetzt sehr viele gleiche Werte in Folge. Diese werden mittels RLE[17] (Run length Encoding) zusammengefaßt. Der AC-Wert wird gemeinsam mit dem RLE-Code noch Huffman[18] kodiert.

Am Ende erhält man für jeden 8×8 Pixel Block einen eigenen Huffman-Stream.

Indem man dies für jeden Block und jede Ebene macht, erhält man ein JPEG-Bild.

[16] Eine spezielle Art, die 8×8-Matrix in einen Vektor zu überführen, bei der ähnliche Frequenzen nacheinander kommen.

[17] RLE: Für eine Folge von gleichen Werten, wird nur die Anzahl und einmal der Wert gespeichert

[18] Huffman: Häufige Werte werden durch weniger Bits kodiert als seltene.

Neben dem hier beschriebenen „Baseline“ Verfahren, gibt es noch andere, die jedoch in der Videotechnik kaum verwendet werden. Z.B. ein verlustfreies JPEG-Verfahren oder „progressive JPEG“, bei dem ein geringer aufgelöstes Bild stufenweise verbessert wird.

6.3.2 M-JPEG

Auch das Motion-JPEG ist eigentlich kein Video-Codec, da nur einfach JPEG-Bilder aneinandergereiht werden.

Eigentlich für den Video-Schnitt entwickelt, findet es dennoch Anwendung in der Videokonferenz.

Neben den einzelnen Bildern enthält ein M-JPEG Strom noch „Time-Stamps“. Diese sind notwendig, um die, unabhängig übertragenen, Audio-Daten synchronisieren zu können.

Da keinerlei Abhängigkeiten der Bilder untereinander ausgenutzt werden, ist der Datenstrom recht groß. Der Vorteil ist aber, daß jedes Bild übertragen werden kann, sobald es aufgenommen wurde. Neben der JPEG-Kodierung sind keinerlei weitere Schritte nötig, dies senkt die Verzögerung deutlich.

Im Vergleich zu den im Folgenden vorgestellten Verfahren, kann M-JPEG keine sinnvolle maximale Datenrate garantieren. Es ist also nicht ohne weiteres für CBR-Dienste geeignet.

6.3.3 H.261

Bei H.261 handelt es sich um ein Kodierungsverfahren, das für Bildtelefonie und Videoconferencing entwickelt wurde. Als Übertragungsmedium wird meist ISDN verwendet, mit $n \times 64$ Kbit/s.

Es wurden zwei Formate definiert, das „Common Interchange Format“ (CIF) mit 352×288 Pixel und das „Quarter-CIF“ (QCIF) mit 176×144. Für beide Formate wird eine Bildwiederholfrequenz von 30 Bildern/s angepeilt. Diese wird jedoch so gut wie nie erreicht.

Der Videostrom selber besteht aus I-Frames, vollständigen Bildern, und P-Frames bei denen nur der Unterschied zum letzten Bild übertragen wird (siehe Abb.: 70).

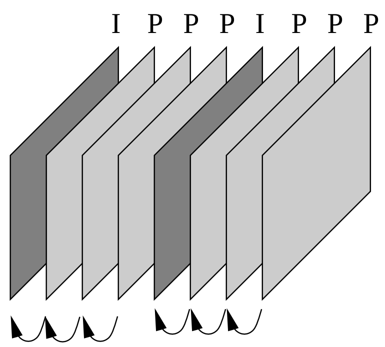


Abbildung 70: H.261 Videostrom

Dabei werden die I-Frames mit JPEG komprimiert. Sobald das Bild kodiert ist, wird es übertragen. Bei den P-Frames wird eine Motion-Estimation durchgeführt. Dazu wird versucht für jeden Macroblock (entspricht den 8×8 Blöcke von JPEG) zu ermitteln wo er sich jetzt befindet (vgl. Abb. 71). Die Differenz zu dem so errechneten Bild und dem eigentlichen Bild wird, nach dem Bewegungsvektor, im P-Frame übermittelt.

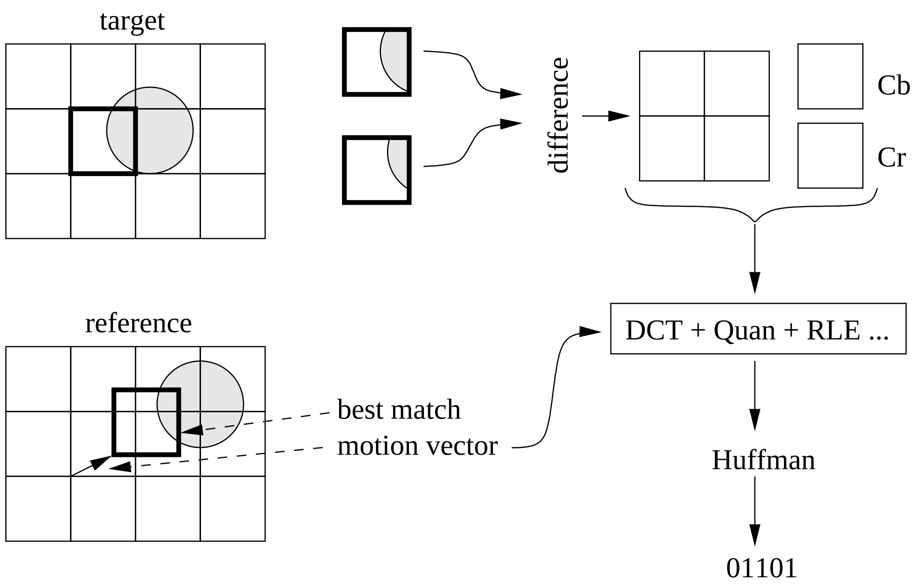


Abbildung 71: P-Frame Kodierung

Da die einzelnen P-Frames von einander abhängen, kann sich ein Übertragungsfehler auf die gesamte Bilderfolge, bis zum nächsten I-Frame, auswirken. Deshalb wird der H.261 Bitstream mittels einer Forward-Error-Correction geschützt, die es ermöglicht 1 und 2 Bit Fehler zu korrigieren und 3 Bit Fehler zu erkennen.

6.3.4 MPEG

Der Video-Codec der Motion Picture Expert Group basiert auf H.261, beinhaltet jedoch die Codierung eines Audio-Signals (siehe 6.2.7).

MPEG-1 ist für eine höhere Auflösung vorgesehen. Es ist dazu gedacht Video in PAL bzw. NTSC Auflösung zu kodieren (726×576 bei 25 bzw. 30 fps). Für den Datenstrom stehen 1.5 Mbit/s zur Verfügung, wovon 0.3 Mbit/s für den MPEG-Audio-Stream vorgesehen sind.

Um eine bessere Komprimierung zu erreichen, wird ein neuer Frame-Type, die B-Frames, eingeführt (siehe Abb. 72).

Bei den „Bidirectional-Frames“ wird nicht nur Information des aktuellen Bildes und des letzten I- bzw. P-Frames betrachtet, sondern auch der nächste P- bzw. I-Frame. Da dieser bereits aufgenommen sein muß, wird dadurch der Delay, bei der Kodierung, deutlich erhöht. Die Information in den B-Frames ist für die Video-Sequenz unerheblich, verbessert das Bild jedoch deutlich. Bei der Kodierung des B-Frames wird eine gewichtete Mittelung des letzten I- bzw. P-Frames und des nächsten P- bzw. I-Frames durchgeführt.

Gegenüber H.261 wurden noch weitere Verbesserungen eingeführt. So kann der Bewegungsvektor in 1/2-Pixelschritten angegeben werden. Es ist möglich wahlfrei auf den Videostrom zuzugreifen und es wurde die Robustheit gegen Übertragungsfehler erhöht. Dies wurde zum einen durch eine Forward-Error-Correction mittels Reed-Solomon und durch die Einteilung des Bildes in „Slices“, die voneinander unabhängig sind, erreicht.

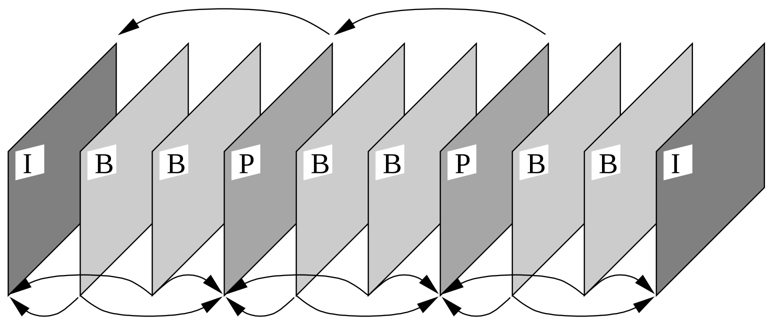


Abbildung 72: MPEG Videostrom

level	size	pixel/s	bit-rate[19]	application
low	352×240	3M	4	consumer tape equip.
main	720×480	10M	15	studio TV
high 1440	1440×1152	47M	60	consumer HDTV
high	1920×1080	63M	80	film production

Tabelle 7: MPEG-2 target applications

6.3.5 MPEG-2

Da MPEG für den professionellen Bereich mehrere Unzulänglichkeiten aufwies, wurde mittels MPEG-2 ein erweiterter Standard geschaffen. Dieser wurde als H.262 von der ITU standardisiert.

Neben höheren Auflösungen, bis 16383×16383 Pixel, sind auch andere YUV Modelle möglich, so 4:2:2 oder 4:4:4. Letzteres ist besonders für Studioaufnahmen interessant.

Es wurden vier Standard Anwendungen definiert (siehe Tabelle 6.3.5), zu denen auch HDTV mit einem 16 : 9 Bildformat gehört. Da nicht jedes Wiedergabegerät das 16 : 9 Format unterstützt, ist es möglich ein „Pan&Scan“ Fenster in 4 : 3 Auslösung zu definieren. Dieses kann in jedem Frame an einer anderen Stelle liegen.

Um eine weitere Qualitätssteigerung zu erreichen, ohne einfachere Abspielgeräte zu überlasten, wurde ein optionaler „Enhanced Layer“ definiert. Dadurch sind zwei Skalierungsmöglichkeiten gegeben. Bei der „Temporal-Scalability“ wird zwischen je zwei Frames ein weiterer B-Frame eingeschoben. Dahingegen wird bei der „Quality-Scalability“ zwischen jeden Frame ein P-Frame eingeschoben.

Bei MPEG-2 wurde eine Übertragung mittels ATM vorgesehen. Der MPEG-2 „Transport Stream“ besteht aus 188 Bit, die exakt die Größe von 4 ALL1-CS-PDU haben.

Der Draft für MPEG-3, höhere Auflösungen für medizinische Anwendungen, wurde in den MPEG-2 Standard aufgenommen.

Im Moment wird ein weiterer Standard, MPEG-4 entwickelt, der für schmalbandigere Übertragungsstrecken gedacht ist, z.B. mittels eines UKW-Senders.

6.4 Motion-Vector-Search

Beim Besprechen der Video-Codecs H.261, MPEG und H.262 wurde immer erwähnt, daß für die P- und B-Frames ein Bewegungs-Vektor ermittelt wird. Auch wenn es für einen Menschen nicht schwer ist, die Bewegung eines Objektes zwischen zwei Bildern nachzuvollziehen, ist dies der komplexeste Teil dieser Video-Codecs.

Für die Motion-Vector-Search gibt es drei verbreitete Verfahren. Jedes Verfahren betrachtet für einen Block, meist mit Kantenlänge 8, die Umgebung mit einer Kantenlänge p. Es wird immer versucht den MAE[20] zu minimieren.

Full Search: Jede mögliche Position wird als Ziel getestet. Dies ist einfach, braucht jedoch viele Operationen.

Two-Dimensional Logarithmic Search: An den neun im Abb. 73 markierten Punkten wird der mit dem geringsten MAE ausgewählt, dies wird solange wiederholt, bis das abzusuchende Gebiet nur noch 1 Pixel groß ist.

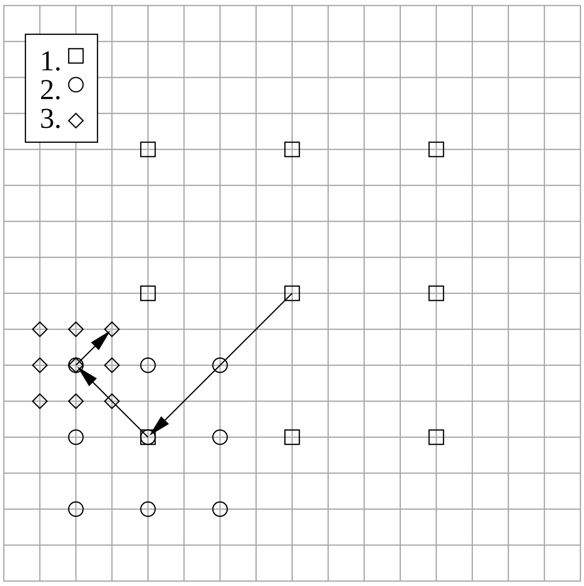


Abbildung 73: 2-Dimensional logarithmic Search

Das Verfahren ist schwieriger zu implementieren, bringt jedoch einen deutlichen Geschwindigkeitsvorteil.

Hierarchical Motion Estimation: Hierbei wird eine geringer auflösende Version der Two-dimensional logarithmic search angewendet. Dazu werden zwei Bilder, je mit einem viertel der Auflösung des detaillierteren Bildes berechnet. Der dort gefundene Bewegungsvektor wird in der höher auflösenden Version schrittweise verbessert.

Auch wenn das Verfahren nochmals komplizierter ist, kann wieder eine Geschwindigkeitssteigerung bemerkt werden.

[20] Mean Average Error: Summe der Fehlerbeträge $\frac{1}{n^2}\sum_{i,j=1}^{n}|e_{i,j}|$ bei einer Kantenlänge n und dem Fehler $e_{i,j}$ an Position i, j

Search Method	$p = 15$	$p = 7$
Full Search	29.89 GOPS	6.99 GOPS
Logarithmic	1.02 GOPS	777.60 MOPS
Hierarchical	507.38 MOPS	398.52 MOPS

Tabelle 8: Vergleich der Motion-Estimation Verfahren

Tabelle 8 zeigt einen Vergleich der drei Verfahren, für ein Video mit einer Auflösung von 720×480 bei 30 Bilder/s, abhängig von der Größe des Umgebungsbereiches p. Dabei wird angenommen, das für einen Vergleich drei Operationen nötig sind: Differenzbildung, Betragbildung und Aufaddieren.

6.5 Videokonferenzen

Nachdem ich jetzt die wichtigsten Audio- (siehe 6.2) und Video-Codecs (siehe 6.3) vorgestellt habe, fehlt nicht mehr viel und man kann Videokonferenzen durchführen.

Neben vielen proprietären Lösungen, z.B. von CellStack, Siemens oder SUN, gibt es auch zwei verbreitete Standards. Zu den nicht standardisierten Lösungen muß man noch sagen, daß sie zwar untereinander inkompatibel sind, ansonsten aber meist problemlos funktionieren.

6.5.1 H.320 / H.321

Die beiden Protokollgruppen H.320 und H.321 beschrieben ähnliche Verfahren zur Videokommunikation. Während H.320 für schmalbandiges System, N-ISDN mit $n \times 64$ Kbit/s, $n = 1 \ldots 30$, gedacht ist, ist H.321 für breitbandige System ausgelegt.

Allgemein wird Audio und Video getrennt übertragen. Für Audio wird dabei eines der Protokolle G.711, G.722 oder G.728 verwendet. Video wird immer nach H.261 kodiert. Zur Synchronisation und zum Multiplexing wird das Protokoll I.211 verwendet.

Bei H.320 wird für die Sprache meist ein eigener B-Kanal verwendet, da bei einem normalen ISDN-Anschluß dann für Video nur 64 Kbit/s zur Verfügung stehen, sind die Ergebnisse im Moment noch recht unbefriedigend.

Bei Videokonferenzen mit mehr als 2 Personen, wird bei H.320 immer nur ein Bild[21] und die gemischten Audiodaten übertragen. Dafür ist jedoch ein spezieller Server nötig. Bei H.321 wird ebenfalls ein gemischtes Audio-Signal übertragen, dank Multicast-Diensten ist es jedoch möglich das Videobild eines jeden Teilnehmers zu übertragen.

Darüber hinaus beschreibt H.321 noch die Einbindung von H.320 Terminals in eine H.321 Videokonferenz.

Wenn das Audiosignal nach G.722 kodiert wird, kann der Datenkanal für sog. „White-Board“ Applikationen verwendet werden. Dabei handelt es sich meist um eine Möglichkeit untereinander Notizen auszutauschen.

[21] normalerweise das Bild des Teilnehmers mit dem lautesten Audiosignal

6.5.2 MPEG

Bei MPEG hat man den großen Vorteil, daß der Codec bereits auf Video Übertragung, Bild und Ton, ausgelegt ist, und man sich somit eine aufwendige Synchronisation sparen kann.

Auch wenn bei MPEG darauf geachtet wurde, daß man den Algorithmus theoretisch in Hardware gießen kann, sind vor allem MPEG-Encoder nur sehr selten anzutreffen. Dies liegt primär am hohen Aufwand für die Motion-Estimation.

Da auch moderne Pentium-II 333 nicht imstande sind selbstständig MPEG-2 Filme (Main-Level) zu dekodieren wird eine weite Verbreitung von MPEG-Videolösungen auch noch eine Weile auf sich warten lassen.

6.6 Videokonferenz Pilotprojekt des LRZ

Seit dem Wintersemester 1995/96 betreibt das LRZ eine Videokonferenz-Verbindung zwischen der TU-München und der Uni Erlangen. Über das dazu durchgeführten Pilot-Projekt wird in [10] berichtet.

Es werden proprietäre CellStack-Codecs verwendet. Diese ATM-Native Videokonferenz Lösung bietet M-JPEG Kodierung in einer Auflösung von 720×288 bei 25 Bildern/sec. Die Audio-Signale werden in DAT-Qualität, 48kHz 16Bit Stereo, aufgezeichnet und PCM-Kodiert übertragen. Im Vergleich zu anderen Verfahren zeichnen sich die CellStacks durch einen geringen Delay, da M-JPEG, und eine einfache Handhabung aus – „Einschalten und losgeht's“.

Auch wenn der Videostrom im Mittel nur 15 Mbit/s braucht, erwies er sich doch als sehr burstartig, mit Spitzen über 40 Mbit/s. Auch der Audiostrom, der im Mittel 1.74 Mbit/s benötigt, wies Bursts auf.

Da die verwendeten ATM-Strecken des B-WiN für IP-Verkehr benutzt werden, wurde befürchtet, daß diese Bursts auf die Performance des IP-Verkehrs eine negative Auswirkung haben.

Um die Datenrate der CellStack gleichmäßiger zu gestalten, wurde mittels Fore-ATM-Switches ein Trafficshaping durchgeführt. Dabei wird mittels des Leaky-Bucket Verfahrens der Datenstrom auf 20.076 Mbit/s begrenzt. Bei dieser Rate entsteht keine Qualitätsverlust für den Videostrom und für den IP-Verkehr des B-WiNs ist keinerlei Beeinträchtigung merkbar.

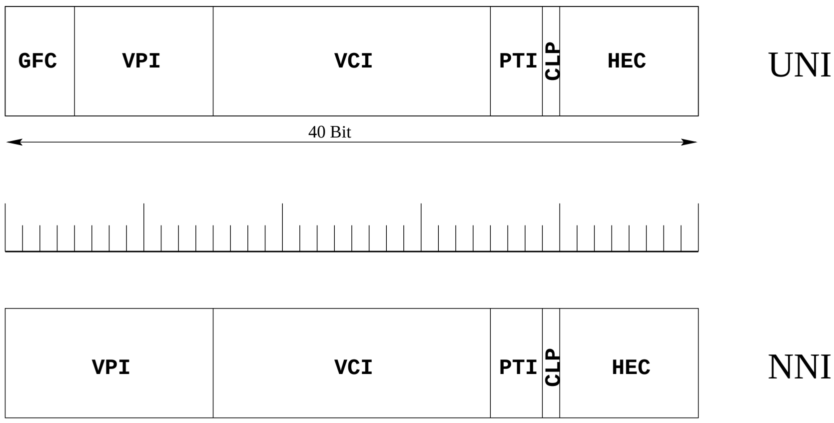
Dank des Traffic-Shapings war es sogar möglich eine Videokonferenzverbindung zwischen München und Augsburg aufzubauen. Im Vergleich zu der 155 Mbit/s Strecke zwischen München und Erlangen, ist die Uni Augsburg nur mit 34 Mbit/s an das Leibnitzrechenzentrum angeschlossen.

Mittels der CellStacks wird jetzt regelmäßig eine Vorlesung der TU-München nach Erlangen übertragen. Auch Vorlesungen aus Erlangen kann man im Hörsaal S1128 miterleben.

7 ATM-Switches

7.1 Einführung

ATM-Switches sorgen im ATM-Netzwerk dafür, daß darin befindliche Zellen auf der Reise vom Ursprung zum Ziel den richtigen Weg durch die verschiedenen Netzwerksegmente finden und schließlich und endlich beim richtigen Empfänger landen. Die Tatsache, daß alle ATM-Zellen gleich groß sind kann dabei durch eine massiv parallele Architektur genutzt werden, was Zelldurchsatzraten im Giga- und Terabitbereich und damit die Realisierung von Hochgeschwindigkeitsnetzen wie B-ISDN ermöglicht. Damit alle an einem Switch angeschlossenen Benutzerkanäle gleichzeitig bearbeitet werden können ist es notwendig daß die Vermittlungsgeschwindigkeit in ATM-Schalteinheiten die Übertragungsraten der angeschlossenen Stationen um ein vielfaches übertrifft.



GFC ... Generic Flow Control
VPI ... Virtual Path Identifier
VCI ... Virtual Cannel Identification
PTI ... Payload Type
CLP ... Cell Loss Priority
HEC ... Header Error Check

Abbildung 74: Aufbau der ATM-Zellheader

Zum besseren Verständnis folgt ein kurzer Rückgriff auf bereits vorangegangene Vorträge dieses Hauptseminars: Eine ATM-Zelle hat eine Länge von 53 Byte, wovon 5 Byte auf den Header und 48 Byte auf das Payload-Feld entfallen. Die ATM-Standardisierungsgremien haben für den Header zwei verschiedene Formate vorgesehen, das UNI-header-format und das NNI-Header-Format, der genaue Aufbau der ATM-Zellheader kann Abbildung 74 entnommen werden. Dabei legen die UNI-Spezifikationen die Kommunikation zwischen ATM-Endsystemen fest (beispielsweise Router oder Workstations), die NNI-Spezifikation definiert die Kommunikation zwischen verschiedenen ATM-Switches. Die in diesem Zusammenhang wichtigen Header-Komponenten sind die beiden Felder VPI und VCI. Im UNI-Header sind für VPI 8 Bits, für VCI 16 Bits reserviert, im NNI-Header sind 12 Bits für VPI und 16 Bits für VCI vorgesehen.

Einem ATM-Switch fallen grundsätzlich 2 Aufgaben zu:

- Die Identifizierung und Auswertung der VPI und VCI-Felder, also der Kanal- und Pfadidentifikation in der Zelle

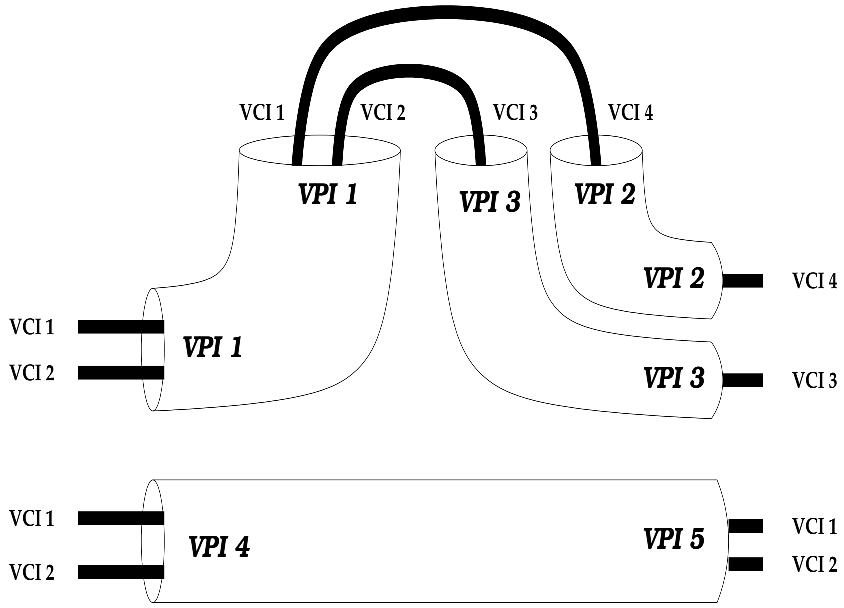


Abbildung 75: ATM Pfad- und Kanalvermittlung

- Der Transport einer Zelle von einem Eingangsport auf den aufgrund ihrer VCI/VPI-Information ermittelten Ausgangsport

7.2 Unterscheidungsmerkmale

Nach ihrer Funktion können Switches in ATM-Pfadvermittlungen (VP-Switches, Crossconnects) und in ATM-Kanalvermittlungen (VC-Switches) unterschieden werden.

- ATM-Pfadvermittlungen (Crossconnects): Hier werden ankommende Pfade beendet und mit allen darin befindlichen Kanälen in einen anderen Pfad umgeleitet. Die einzelnen ATM-Kanäle bleiben von diesem Vorgang unberührt, es ändern sich hier jeweils nur die VPI's, die VCI-Werte werden beibehalten.

- ATM-Kanalvermittlungen: An diesen Vermittlungsstellen werden sowohl die eingehenden Pfade als auch die eingehenden Kanäle beendet und in andere abgehende Pfade bzw. Kanäle umgeleitet. Wird eine Kanalverbindung beendet heißt dies daß gleichzeitig auch die Pfadverbindung getrennt werden muß, umgekehrt ist dies nicht der Fall. Eine Pfadvermittlung ist somit ein 'Spezialfall' einer Kanalvermittlung, was bedeutet daß Kanalvermittlungen auch als Pfadvermittlungen benutzt werden können.

Abbildung 75 soll den grundsätzlichen Unterschied dieser beiden Techniken veranschaulichen.

7.3 Ein allgemeines Switch-Modell

Um einen ersten Einblick in die innere Organisation eines Switches zu bekommen soll an dieser Stelle ein allgemeines Switch-Modell gezeigt werden, d.h. es werden hier funktionelle Anforderungen an einen Switch und mögliche Realisierungen im System vorgestellt. Für diese Betrachtungen

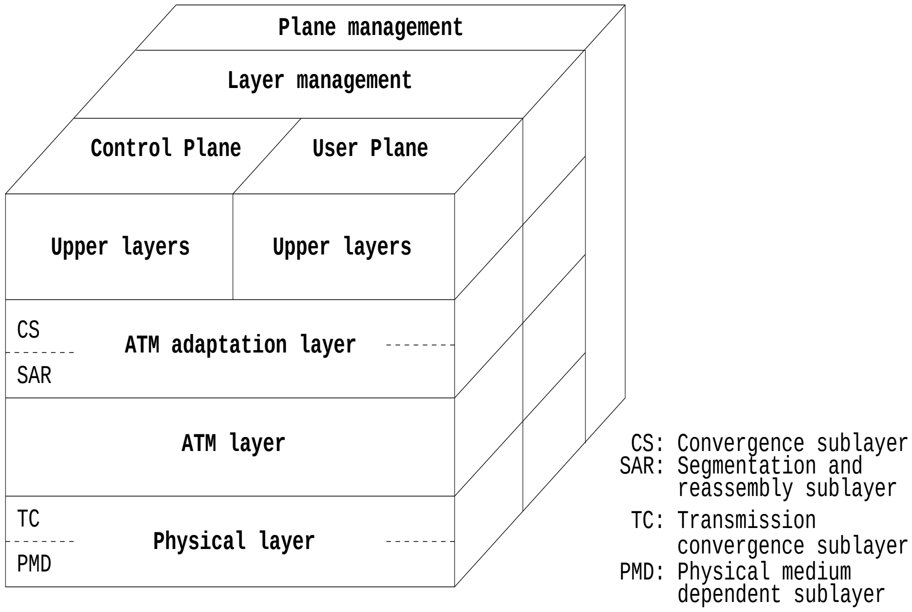


Abbildung 76: B-ISDN Referenzmodell

wird angenommen, daß der Switch mit standardisierten ATM UNI oder NNI-Schnittstellen ausgestattet wird. Zuerst stellt sich die Frage, welche verschiedenen Daten einem Switch eigentlich begegnen können. Um dies festzustellen sollte man das B-ISDN-ATM-Referenzmodell (Abb. 76) noch einmal genauer betrachten:

Im Gegensatz zu früheren Referenzmodellen ist das ATM-Modell dreidimensional, was im momentanen Zusammenhang besonders interessiert sind die Benutzerebene (User Plane), die Steuerungsebene (Control Plane) und die Managementebene (Management Plane). Die Benutzerebene betrifft Datentransport, Flußsteuerung, Fehlerkorrektur und andere Benutzerfunktionen, auf der Steuerebene findet demgegenüber das Verbindungsmanagement statt. Die Managementfunktionen der Schichten und Ebenen beziehen sich auf das Ressourcenmanagement bzw. die schichtübergreifende Koordination.
Aus den verschieden Datentypen die einem Switch begegnen können, Benutzer-Daten, Kontroll-Daten (Signalisierung) und Managementdaten, ergeben sich also verschiedene funktionelle Anforderungen an den Switch.

7.3.1 Anforderungen der Benutzerebene

Die Hauptfunktion eines ATM-Switches ist hier Benutzer-Datenzellen von seinen Input-Ports an seine Output-Ports weiterzuleiten. Die Information in den Zell-Payloads geht transparent durch den Switch, d.h. vom Switch werden also nur die Zellheader verarbeitet. Wenn die Zellen am Input-Port des Switches empfangen werden, wird aus den Headern die VPI/VCI-Information gewonnen, die in der weiteren Folge dazu verwendet wird um die Zellen durch den Switch an die entsprechenden Output-Ports zu leiten. An den Output-Ports werden die Zellen, bereit zur Physical-Layer-Übertragung, bereitgestellt. Dieser Zellvermittlungsservice umfaßt also 3 verschiedene Funktionen die in den funktionalen Blöcken der Abbildung 76 dargestellt werden: Die Zellen werden von Input- Modulen an den Input-Ports empfangen und für die Weiterleitung durch den Switch vorbereitet. Das eigentliche Routing passiert in der Switching-Fabric, auch cell switch fabric **(CSF)** genannt, danach werden die Zellen für die Weiterleitung zu den Output-Modulen an die Output-Ports des Switches vorbereitet. Die Input- und Outputmodule sowie die cell switch fabric übernehmen also die grundlegenden Zellrouting und -buffering Funktionen, die in einem ATM-Switch benötigt werden.

7.3.2 Anforderungen der Steuerungsebene

Wie bereits festgestellt ist die Steuerungsebene im B-ISDN Referenzmodell für die Einrichtung und Kontrolle von VPCs/VCCs verantwortlich. Verbindungskontrollinformation wird in Signalisierungszellen transportiert, welche durch ihre VPI/VCI-Felder gekennzeichnet sind. Anders als bei Benutzerdatenzellen ist die Payload-Information in den Signalisierungszellen für das Netzwerk nicht mehr transparent, die Signalisierungsinformation muß vom ATM-Switch verarbeitet und interpretiert werden. Dazu ist es notwendig, daß der Switch eingehende Signalisierungszellen identifiziert und die darin enthaltene Information interpretiert. Wenn der Switch selbst Steuerungsinformationen produziert werden diese in Signalisierungszellen eingekapselt und in den ausgehenden Benutzerdaten-Zellstrom gemischt. Da die Signalisierungszellen den gleichen Transportweg wie Benutzerdatenzellen verwenden, gehen sie durch die Input-Module, die cell switch fabric und die Output-Module. Zusätzlich wird die Signalisierungsinformation von einem weiteren funktionellen Block verarbeitet, der als Call Admission Control **(CAC)** bezeichnet wird. Hier lassen sich zwei grundsätzlich verschiedene Methoden unterscheiden wie mit der Signalisierungsinformation umgegangen wird: Entweder wird diese Information von der CSF zur CAC geleitet oder, da es ist nicht unbedingt erforderlich ist die CSF zu verwenden, wird die Signalisierungsinformation direkt von den Input-Modulen über die CAC zu dem Output-Modulen geleitet.

7.3.3 Anforderungen der Managementebene

Die Managementebene im B-ISDN-Referenzmodell stellt Funktionen zur Verfügung die das ATM-Netz beobachten und kontrollieren, um eine korrekte und effiziente Funktion sicherzustellen. Die Managementfunktionen sind zur Zeit noch nicht vollständig standardisiert, von den bereits bestehenden Standards und Entwicklungen ausgehend kann man aber bereits einmal eine grobe Aufgliederung der in ATM-Switching-Systemen erforderlichen Managementfunktionen vornehmen:

- Fault Management: Fehlfunktionen aufzuspüren, isolieren und darüber Protokoll führen;
- Performance Management: Kontinuierliche Beobachtung, Auswertung und Protokoll über das Verhalten der einzelnen Switch-Komponenten;
- Configuration Management: Aktivieren oder Abschalten von Switchkomponenten;
- Security Management: Zugriffsschutz und Kontrolle der Switch-Datenbank;
- Accounting Management: Messung des Ressourcenverbrauches des Switches, beispielsweise für Tarifierung;
- Traffic Management: Beobachtung und Regulierung des Verkehrs um Überlastsituationen vorzubeugen;

Diese Anforderungen werden einem weiteren funktionellen Block zugeordnet, der im folgenden System Management (SM) genannt wird. Eine der Haupttätigkeiten des SM-Blocks ist die Verarbeitung von OAM-Zellen. Wie bei Signalisierungszellen muß die Information im Payload-Feld der OAM-Zellen gelesen und interpretiert werden. Der Switch muß also wiederum eingehende OAM-Zellen erkennen (anhand der VPI/VCI-Werte oder des PT-Feldes), diese von den anderen Zellen trennen und die notwendigen Aktionen ausführen, um die ATM-Layer-OAM-Prozeduren zu unterstützen. Der Switch kann auch hier wieder eigene OAM-Zellen generieren und in den ausgehenden Zellstrom mischen. Zusammen mit Benutzerdaten- und Signalisierungszellen fliessen OAM-Zellen durch die Input-Module, die CSF (cell switch fabric) und die Output-Module, die Verarbeitung erfolgt im SM (System Management). Es existieren hier wieder zwei Möglichkeiten um die OAM-Zellen durch den Switch zu bringen, wahlweise mit oder ohne Beteiligung der CSF (cell switch fabric). Eine weitere wichtige Funktion des SM (System Managements) ist die Unterstützung des Interim Local Management Interface (ILMI) für jedes UNI. ILMI erlaubt dem Benutzer Status- und Kontrollinformation über VPCs/VCCs über das UNI zu erhalten. Für jedes UNI enthält das SM eine UNI Management entity (UME), welche in der ATM UNI Management-Spezifikation standardisierte Werte beobachtet. Ebenso enthält jedes ATM Endnutzergerät eine UME. Nebeneinanderliegende UMEs können über die UNI miteinander kommunizieren indem sie in ILMI-Zellen eingebettete SNMP-Messages austauschen, welche wieder über ihre VPI/VCI-Werte identifiziert werden. Die ILMI-Zellen nehmen ihren Weg durch den Switch durch die Input-Module, das SM und die Output-Module, designabhängig können sie dabei auch die CSF passieren. Ferner ist manchmal noch die Möglichkeit vorgesehen, daß Netzwerk-Management-Information vom SM direkt mit einem Betriebssystem ausgetauscht wird.

7.3.4 Anforderungen an die benötigten Module

Aufgrund der gestellten Anforderungen an einen ATM-Switch werden also folgende Module als nötig erachtet:

- Input-Module: Eingehende Zellen empfangen und für den Weg durch die CSF vorbereiten
- Output-Module: Ausgehende Zellen für die weitere Übertragung vorbereiten
- Cell Switch Fabric: Transport der Datenzellen von den Input- zu den Outputports, ggf. auch Transport von Signalisierungs- und Management-Zellen zwischen anderen funktionellen Blöcken des Switches.
- Connection Admission Control: Verarbeitung und Interpretation von Signalisierungsinformation, Entscheidung über Verbindungsannahme oder Ablehnung.
- System Management: Durchführung aller Management- und Verkehrskontrollmaßnahmen um einen sicheren und effizienten Betrieb des Switches zu gewährleisten.

7.4 Switch-Topologien

Ein stark verallgemeinertes Modell eines ATM-Switches besteht aus den Komponenten: Input-Modules, Switching Fabric, Output-Modules. Dabei ist die Switching-Fabric der Teil des Switches der die eigentliche Arbeit verrichtet, nämlich dynamische Übertragungswege zwischen Input- und Outputports zur Verfügung stellt. Hierbei ist zu beachten daß bei diesem Vorgang Kollisionen auftreten können, die entstehen wenn sich zwei Zellen gleichzeitig an einer

Schaltstufe für den gleichen Output-Port bewerben. Dieses Problem kann durch geeignete Pufferstrategien entschärft werden, die dann etwas später in diesem Kapitel noch zur Sprache gebracht werden.
Doch zunächst zum Aufbau der Schaltelemente, aus denen die Switching-Fabric zusammengesetzt ist: Ein Schaltelement funktioniert in sich wie eine kleine Switching-Fabric, jedoch ist die Anzahl der Input- und Output-Ports zu gering um tatsächlich als Herzstück eines ATM-Switches eingesetzt zu werden. Dies ist auch der Grund weshalb im Normalfall zahlreiche Schaltelemente zu einer Switching-Fabric zusammengesetzt werden. Die Schaltelemente selbst bestehen aus einem Interconnection-Netzwerk, daß die Übertragungswege für die ATM-Zellen zur Verfügung stellt. Hierbei werden grundsätzlich zwei Arten von Interconnection-Netzwerken unterschieden:

- Netzwerke mit Matrixstruktur: Bei diesem Netzwerktyp werden die ATM-Zellen parallel über ein Netzwerkgitter, welches Ein- und Ausgänge des Schaltelementes miteinander verbindet, übertragen. Die Zellen die an den Input-Ports warten werden gleichzeitig und synchron zu einem lokalen Takt übertragen. Dabei ist zu beachten, daß es zu Blockierungen kommen kann wenn sich zwei Zellen gleichzeitig um den gleichen Output-Port bewerben. Um solche Situationen in denen es zu Zellverlusten kommen kann zu vermeiden, werden an den Input- oder Output-Ports oder auch an den Kreuzungsstellen der Übertragungswege Pufferspeicher angebracht. Bei der Dimensionierung dieser Pufferspeicher ist eine wichtige Größe zu beachten, nämlich der sogenannte Speed-Up-Faktor K. Mit diesem Wert wird die Zellenanzahl angegeben, die innerhalb eines Schaltzyklus vom Output-Pufferspeicher empfangen werden kann. In einem Schaltnetzwerk mit N Output-Ports und Output-Pufferspeichern mit einer Kapazität von OP_x sollte K kleiner oder gleich N bzw. OP_x sein, je nachdem welcher Wert kleiner ist. Zum Abschluß dieses Punktes noch eine Definition von der man häufig im Zusammenhang mit Schaltelementen liest: Nichtblockierend nennt man ein NxN-Matrixschaltelement dann, wenn es einen Speed-up-Faktor der Größe N besitzt. Ist der Speed-Up-Faktor kleiner als N, so müssen, um Zellverluste zu verhindern, an den Eingangsports Pufferspeicher zur Verfügung stehen.

- Netzwerke nach dem Prinzip des Time-Division-Multiplexing: Diese Art der Schaltelemente wird nach der vorliegenden Topologie in drei feinere Gruppen aufgeteilt: Schaltelemente mit Bus-Topologie (Bus Switching Elements), Schaltelemente mit Ring-Topologie (Ring Switching Elements) und Schaltelemente mit Zentralspeichertopologie (Central Memory Switching Elements).
  `Bus Switching Elements:` Kennzeichen dieser Klasse ist ein 16 oder 32 Bit breiter Hochgeschwindigkeitsbus, hierbei muß die Übertragungskapazität des Busses mind. gleich der Summe der Übertragungskapazitäten der Eingangsports sein um kollisionsfrei übertragen zu können. Staugefahr besteht an den Output-Ports, da bei diesen mit einer relativ zum Bus niedrigeren Übertragungsrate gearbeitet wird. Aus diesem Grund werden Output-Port-Pufferspeicher notwendig, dann kann jedoch die gleiche Leistung wie mit Matrix-Switching-Elementen erreicht werden.
  `Ring Switching Elements:` Bei dieser Technik sind IC und OC über einen Ring miteinander verbunden, was folgenden Vorteil gegenüber der Bus-Topologie bietet: Ein Time-Slot der über Ring läuft kann innerhalb eines Ringumlaufes von mehreren Controllern benutzt werden, dadurch ist eine effektive Auslastung von mehr als 100 % der Ringkapazität möglich, allerdings entsteht ein nicht zu vernachlässigender Overhead bei der Steuerung dieses Mechanismus.
  `Central Memory Switching Elements:` Bei den Central Memory Switching Elements

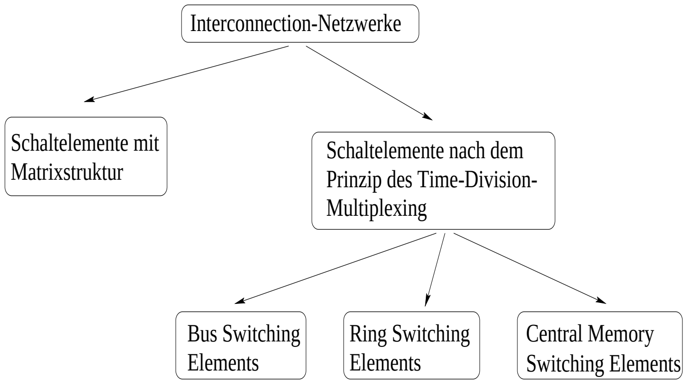


Abbildung 77: Interconnection-Netzwerke

werden alle Zellen von den Input-Controllern in einen gemeinsamen Speicherbereich geschrieben, von dort von den Output-Controllern wieder ausgelesen. Da sich die Pufferspeicher aller Output-Ports denselben Speicher teilen kann hier Speicherplatz eingespart werden. Ein Nachteil dieser Architektur entsteht durch Seriell-Parallel-Wandlung des ankommenden Zellstroms, es muß die Verarbeitungsfrequenz des Schaltelements soweit reduziert werden, daß Speicherlese- und schreibprozesse im Geschwindigkeitsbereich der Speicherbausteine liegen. Meist geschieht hier eine Umwandlung in acht parallele Bitströme, das ergibt bei einer Input-Port-Datenrate von 155 Mbit/s eine interne Verarbeitungsrate von ca. 20 Mbit/s. Central Memory Switching Elements werden besonders bei großen Schalteinheiten wegen der effizienten Speicherausnutzung eingesetzt.

7.5 Leistungsparameter

7.6 Angaben von Herstellern für kommerzielle Systeme

In den Produktspezifikationen der verschiedenen Hersteller findet man durchwegs die folgenden Leistungsangaben für die umworbenen Switches:

- Switch Fabrics (Blocking, Non-Blocking, Buffering)
- Wie viele Ports stehen zur Verfügung?
- Kapazität (avg. 2.4 Gbps Durchsatz)
- Aufbau (Modularität)
- Signalisierung (proprietär, standardisiert (UNI))
- Traffic Management (Traffic Policing, CLP Selective Discard)
- Konfiguration (Manuell, Plug-and-play)
- Switch Prozessor
- Management-Fähigkeiten
- Upgrade-Fähigkeit

7.7 Quantitative Leistungsparameter

- Zellendurchsatz: Dabei handelt es sich um jene Last, die die Schalteinheit bewältigt, und zwar ausgedrückt durch die Wahrscheinlichkeit, mit der während eines bestimmten Schaltzyklus (slot) eine Zelle einen Output-Port des Switches verläßt. Der maximale Zellendurchsatz r_m wird bei einer Last von p=1 ermittelt. Die Last p $(0 \leq p \leq 1)$ ist dabei jene Last (uniform random traffic), bei der mit der Wahrscheinlichkeit p während eines bestimmten Schaltzyklus eine Zelle an der Schalteinheit eintrifft.
- Durchschnittliche Zellenverzögerung: Dieser Parameter gibt die Verzögerung in Schaltzyklen (slots) an, die einer Zelle während der Übertragung durch eine Schalteinheit widerfährt.
- Zellenverlustwahrscheinlichkeit: Die Zellenverlustwahrscheinlichkeit gibt an, mit welcher Wahrscheinlichkeit eine Zelle die an einem Input-Port angekommen ist, während der Übertragung durch die Schalteinheit wieder verloren geht.

Man ist bestrebt solche Schalteinheiten zu entwickeln deren Zellendurchsatz nahe 1 ist, deren durchschnittliche Zellenverzögerung lediglich einige wenige Schaltzyklen beträgt und deren Zellenverlustrate (Stand: 02/98) bei 10^{-9} oder darunter liegt.

7.8 Grundlegende Konzepte in OPNET

OPNET (`Optimized Network Engineering Tools`) ist ein objektorientiertes Netzwerk-Simulations-Tool, das bereits mit zahlreichen fertig modellierten Komponenten ausgeliefert wird. Mit dem *OPNET PLANNER* ist es möglich aus den verschiedenen bestehenden Modellen eigene Netzwerksimulationen zu entwerfen, mit dem *OPNET MODELER* können darüber hinaus auch eigene Modelle, d.h. Komponenten, welche in einer Netzwerksimulation verwendet werden sollen von Grund auf selbst entworfen, oder von bereits vorhandenen Basismodellen abgeleitet werden. Die folgenden Punkte beziehen sich ausschließlich auf den *OPNET MODELER*, der auch bei uns zum Zweck der BWIN-Modellierung eingesetzt wird. Von der Idee zur fertigen Simulationskomponente müssen in *OPNET* drei verschiedene Abstraktionsstufen durchlaufen werden: Es sind dies der Process-Editor in dem eigener Proto-C-Code verwendet werden kann, der Node-Editor in dem Proto-C-Entwürfe kombiniert werden können und schließlich der Network-Editor in dem das Gesamtbild der Simulation entsteht. Nachfolgend eine kurze Beschreibung der einzelnen Editor-Ebenen:

7.8.1 Process-Editor

Im Herz der meisten OPNET-Simulationen findet sich ein Prozeß der mit dem Process-Editor erstellt wurde. Prozeßmodelle können Kommunikations-Hardware, Netzwerkprotokolle, verteilte Algorithmen oder auch Client-Server-Prozesse repräsentieren. OPNET vereinfacht die Modellierung von Prozeßmodellen indem es eine graphische Repräsentation dieser Modelle als endliche Automaten anbietet, Abbildung 78 zeigt ein Beispiel eines solchen Prozeßmodells.

Die Zustände der Automaten können danach mit Proto-C-Code angereichert werden. Dabei handelt es sich um die bekannte Programmiersprache C, die mit von *OPNET* zur Verfügung gestellten Funktionen versehen ist. An jedem Zustand lassen sich mehrere Codeblöcke einfügen, die sogenannten init-executables und die exit-executables. Die ersteren werden beim Eintritt in den Zustand ausgeführt, die letzteren werden aufgerufen falls der betreffende Zustand wieder

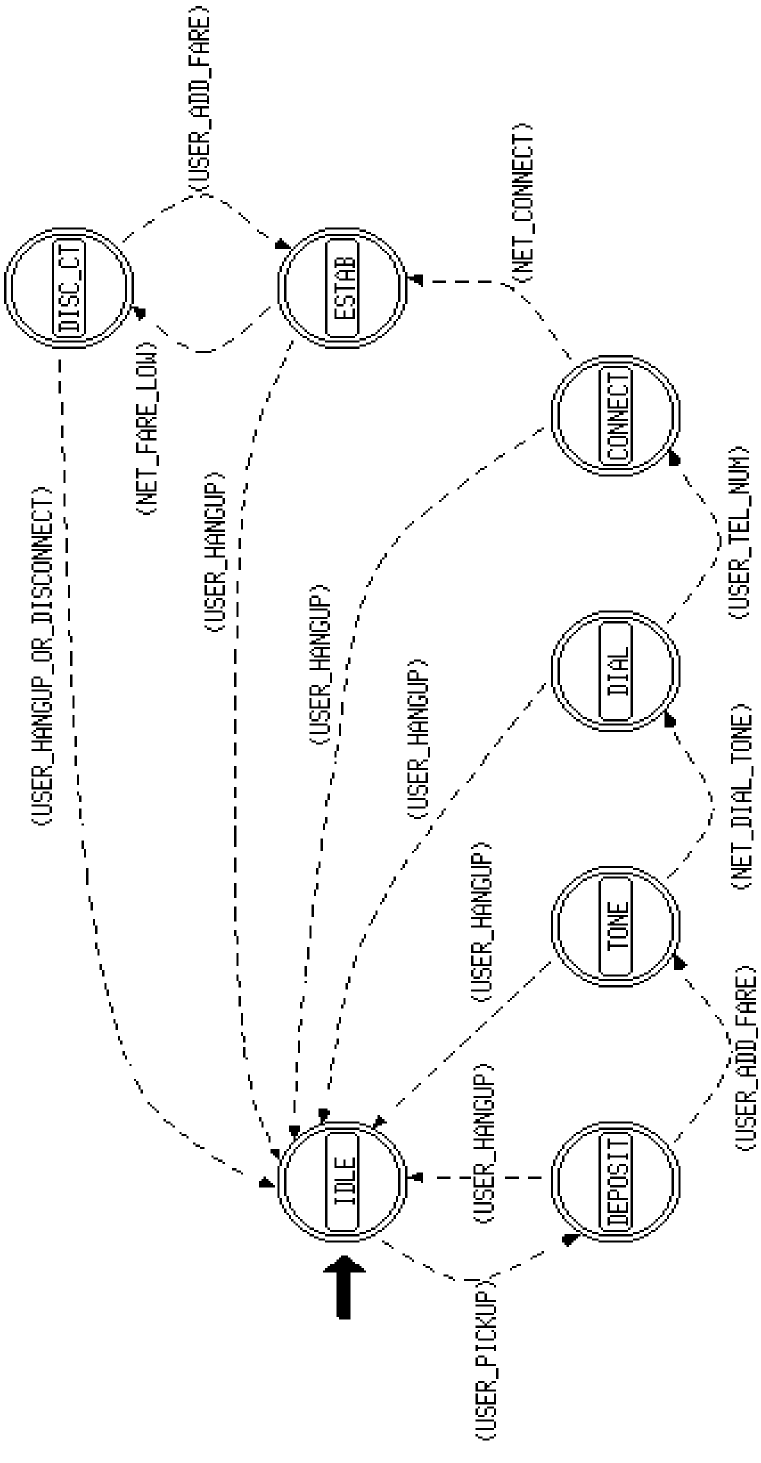


Abbildung 78: Der OPNET Process-Editor

verlassen wird. Als Zustandsübergangsbedingungen kann man beispielsweise bestimmte Simulationsereignisse verwenden. Ist das Prozeßmodell fertiggestellt und läuft zufriedenstellend, so kann es in einem sogenannten Node-Objekt verpackt werden. Dies entspricht der nächsthöheren Abstraktionsebene auf der im Node-Editor gearbeitet wird.

7.8.2 Node-Editor

In diesem Editor werden die verschiedenen Prozeßmodelle zu einem Gesamtmodell zusammengefügt. Als Beispiel in Abb. 79 soll eine Workstation dienen, die IP-Pakete über ein ATM-Netz verschickt. Wie man sich sicher leicht vorstellen kann sind hier, wenn man auf einen modularen Aufbau seines Modells achtet zahlreiche verschiedene Prozeßmodelle notwendig. Die Node-Modelle mit denen hier gearbeitet wird können untereinander, beispielsweise mit Packet Streams, verbunden werden. Freie Parameter die man eventuell im Prozeßmodell als solche gekennzeichnet hat können hier festgelegt werden, man kann sie aber auch höheren Ebenen zur Festlegung freihalten. So besteht die Möglichkeit Parameter erst auf der Network-Ebene einzugeben wo man bereits das gesamte Simulationsmodell vor sich sieht. Falls man sich auch hier noch nicht festlegen möchte können fehlende Parameter bei den Einstellungen zur Simulationsausführung erfragt werden, bzw. werden direkt zur Laufzeit der Simulation erfragt.

7.8.3 Network-Editor

Dieser Editor stellt die oberste Abstraktionsebene von OPNET dar. In ihm kann mit beliebigen kommunizierenden Objekten wie beispielsweise Workstations, Servern, Routern, etc. gearbeitet werden, welche durch Punkt-zu-Punkt-Verbindungen, Busverbindungen oder Funkkanäle verbunden werden können. In diesem Tool, das, wie in Abbildung 80 ersichtlich, einen sehr guten Überblick über die gesamte Simulation verschafft, bestehen Möglichkeiten um die vorhandenen Parameter mittels Objektmenüs einfach zu verändern, auch können Verbindungen und Modellkonfigurationen rasch mittels Mausklick verändert werden.

7.8.4 Messungen im Modell

Ist das Modell erstellt und lauffähig, kann man mit Hilfe des Probe-Editors (siehe Abb. 81) darin Beobachtungspunkte setzen und auf diese Weise verschiedenste Messungen in seiner Simulation vornehmen. Verfügbar sind hier Statistiken die in den mit dem MODELER ausgelieferten Modellen integriert sind, ferner gibt es die Möglichkeit selbst Statistiken zu erstellen oder Animationen aufzuzeichnen. Außer zu reinen Beobachtungszwecken kann der Probe-Editor auch als Hilfe beim Debugging eines neuen Modells helfen, will man die Zellströme in der Simulation beobachten, so kann man eine animierte Ansicht des Modells aufzeichnen welches man dann zur gleichen Zeit oder auch später in *op_vuanim*, einem zugehörigen Animationstool analysieren kann.

Beobachtungen sind mit diesem Tool nicht nur auf der Netzwerk-Ebene, also der höchsten Abstraktionsstufe möglich, auch darunterliegende Abstraktionsstufen können beleuchtet werden. Ein Beispiel dafür ist die Beobachtung des Bitdurchsatzes in ATM-Trunks nach der Verkehrsrichtung getrennt, ein Vorhaben welches auf der Netzwerkebene nicht durchführbar ist. Um diese Daten zu gewinnen, geht man zurück auf die Node-Ebene auf der Netzwerkkomponenten bis auf einzelne Receiver und Transmitter modular aufgegliedert werden und kann dort beobachten wie viele Bits über die betreffenden Transmitter gesendet, bzw. von den betreffenden Receivern

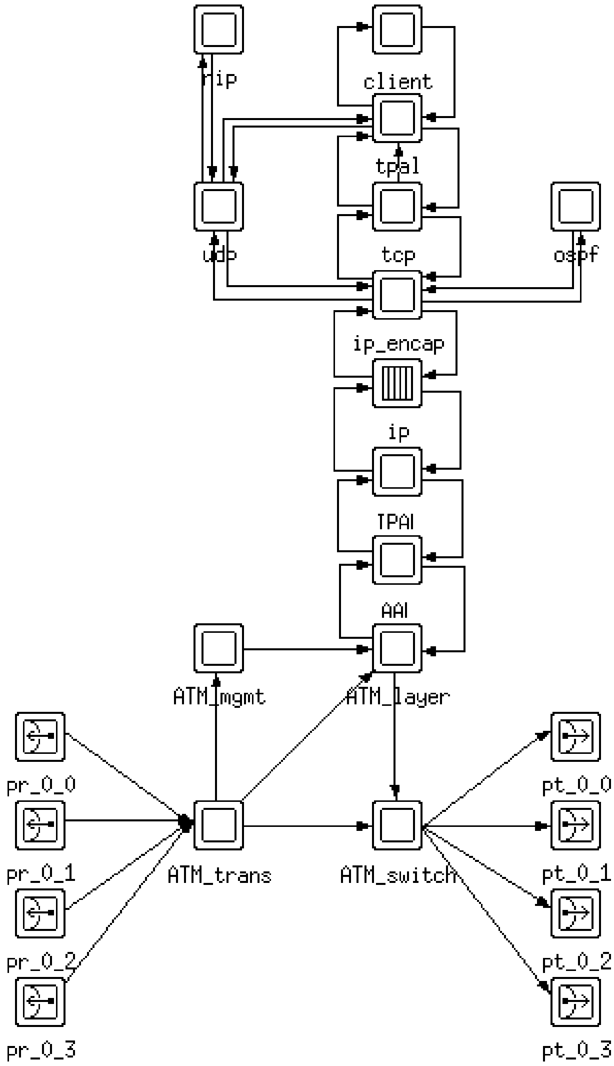


Abbildung 79: Der OPNET Node-Editor

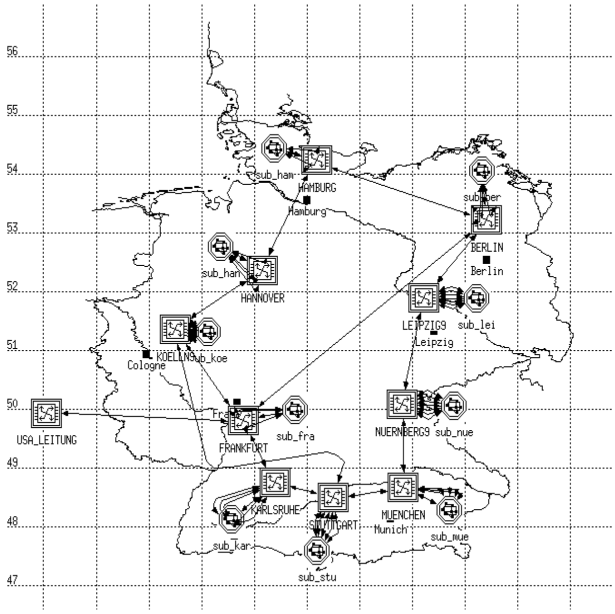


Abbildung 80: Der OPNET Network-Editor

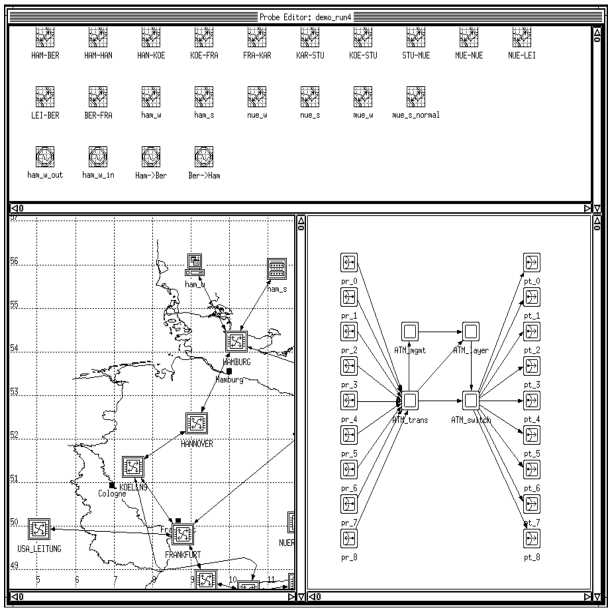


Abbildung 81: Der OPNET Probe-Editor

empfangen wurden. Noch eine Ebene tiefer, auf der Process-Ebene ist die Möglichkeit vorhanden die Zustände zu beobachten, welche die Automatenmodelle während des Simulationslaufs einnehmen.

7.9 Erstes BWIN-Modell

Das BWIN wurde im Frühjahr 1996 als Basisinfrastruktur des DFN in Betrieb genommen. Damit steht eine Breitband-Infrastruktur zur Verfügung, die Anschlüsse mit einer Kapazität von bis zu 155 Mbit/s im Anschlußbereich ermöglicht. Das BWIN wird als privates virtuelles Netz (VPN) auf dem ATM-Cross-Connect Netz der Deutschen Telekom betrieben. Zehn Zugangspunkte, die zentralen ATM Service-Switches (ZSS), bedienen die Kundenanschlüsse, die am Kundenstandort mit ATM Kunden Service Switches (KSS) abschließen. Die ZSS sind durch virtuelle Pfade (VP) mit einer Übertragungsrate von zunächst 34 Mbit/s verbunden, die in Schritten von 2 Mbit/s skalierbar sind. Zunächst wird die ATM-Schnittstelle des BWIN noch nicht freigegeben, sondern es wird ein IP-Dienst über DFN-eigene WiN-Router (WR) angeboten. Zur Realisierung dieses Dienstes sind auch an den ZSS-Standorten zentrale Router (ZR) im Einsatz. Dieser BWIN-IP-Dienst wird über ATM unter Nutzung von AAL5 ohne LLC/SNAP nach RFC1483 und RFC 1577 realisiert.

7.9.1 Zur Verfügung stehendes Material

Die Simulation basiert auf Durchsatzmessungen die uns vom BWIN-Labor in Erlangen zur Verfügung gestellt werden. Um die Aufzeichnung besser verstehen zu können, folgt eine feinere Erklärung der Topologie des BWIN-Modells. Man sollte auf den ersten Blick denken daß zur Bestimmung des Kernnetzdurchsatzes eine Auswertung der Linkstatistiken der ZSS genügt. Das Problem hierbei ist aber, daß viele Daten mehrfach gezählt werden! (So werden Daten von A nach C sowohl in der Linkstatistik von A nach B als auch von B nach C erfaßt). Durch das Fehlen von Information über Quelle und Senke in diesen Daten sind solche Mehrfachzählungen leider nicht zu vermeiden, allerdings kann man den Durchsatz auf eine andere Weise ermitteln. Für ein Kommunikationsnetz ist es ausreichend wenn man den Verkehr zählt der das Netz verläßt wenn man den Durchsatz bestimmen möchte. Die einzigen Zugangspunkte zum Backbone sind die an den jeweiligen Standorten angeschlossenen Zentralen Router. Ohne den Normalbetrieb empfindlich zu stören kann aber hier kein Accounting aktiviert werden. Deshalb wird das Accounting auch an den Interfaces der Win- und der X.25-Router durchgeführt, die von dort zu den darunterliegenden Netzen führen. Der Verkehr von WR/XR zu WR/XR wird hier zwar an zwei Stellen gezählt, bei der Weiterverarbeitung der Daten kann dieser Fall jedoch mittels eines Vergleiches von Quelle und Ziel bereinigt werden.

7.9.2 Zweigeteilte Modellentwicklung

Durch die Art des Simulationsablaufs in OPNET (jede einzelne zu versendende Zelle ist ein Simulationsobjekt, das einzeln behandelt werden muß) ist der Zeitbedarf zur Simulation eines so großen Netzes auch auf schnellen Rechnern entsprechend hoch. Deshalb wird in der Modellierung zweigleisig vorgegangen, auf der einen Seite entsteht ein genaueres BWIN-Modell in dem das Netz bis zu den Kundenserviceswitches (KSS) enthalten ist, auf der anderen Seite wird ein Kernnetz-Modell angefertigti, das nur aus Zentralen Serviceswitches (ZSS) und dem Backbone besteht. An die ZSS werden zur Verkehrserzeugung Lastgeneratoren angeschlossen die den Summenstrom der

KSS erzeugen. Weitere Vereinfachungen des Kernnetzmodells sind noch geplant. Man möchte damit ein Modell erhalten mit dem man Neukonfigurationen des Netzes ohne großen Zeitaufwand testen kann, genauere Betrachtungen können dann mit dem umfangreicheren BWIN-Modell vorgenommen werden.

8 ATM-Meßtechnik

8.1 Einführung

8.1.1 Warum messen?

Durch die zunehmende Verwendung von ATM-basierten Netzwerken muß ebenso die Meßtechnik Schritt halten um diese Technik beherrschen und kontrollieren zu können. Es ist daher wichtig, sich zu allererst die Gründe für die Anwendung von Meßverfahren zu vergegenwärtigen:

Messungen werden vom Netzbetreiber, seltener vom Benutzer durchgeführt. Die ausgeführten Messungen dienen hauptsächlich dem Nachweis und der Sicherstellung der Verfügbarkeit des Netzes. Der Betreiber will oder muß also damit den Nachweis führen, entsprechend des Vertrages die Netzwerkleistung bereitgestellt zu haben.

Ferner werden Messungen zur Fehlersuche oder Behebung in Netzwerken angewandt. Der Netzwerkbetreiber will möglichst schnell und effizient mögliche Fehlerquellen im Netz finden und beheben

Eine weitere Anwendung von Messungen im allgemeinen stellt die Forschung dar. Netzwerke sind meist äußerst komplexe Gebilde, die sich schlecht modellieren lassen. Um hier die bereits in der Entwicklung befindlichen Modelle mit der Wirklichkeit zu vergleichen, müssen die Ergebnisse der Messungen mit dem Modell verglichen werden. Es werden auch Statistiken zur verbesserten Lastmodellierung erfaßt.

8.1.2 Geräte und ihre Anwendung

So vielfältig wie die Meßaufgaben sind auch die angebotenen Meßgeräte. Diese Angebotsvielfalt resultiert aus den unterschiedlichen Anforderungen die mit den Messungen verknüpft werden. Bei den meisten Meßgeräten läßt sich allerdings der Prinzipaufbau wie in Abbildung 82 wiederfinden.

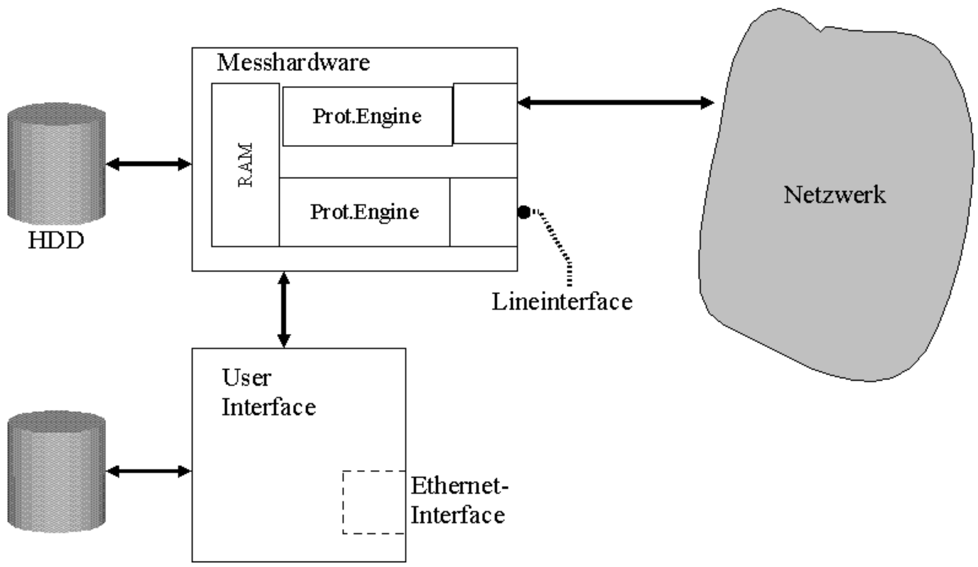


Abbildung 82: Schematischer Aufbau eines ATM-Meßgerätes

Das Meßgerät erhält über ein (meist austauschbares) Lineinterface den physikalischen Anschluß an das zu messende Netzwerk. Dieses Lineinterface ist der jeweils zu messenden Netzwerktechnik angepaßt (physikalische Anschlußart u.ä.) und übergibt die rohen Netzwerkdaten einer meßgeräteinternen Protokoll-Engine, die meist aus einem schnellem RISC-Prozessor besteht, wobei leistungsfähige Meßgeräte für jedes Lineinterface eine eigene RISC-CPU zur Verarbeitung einsetzen.

Die Protokollengine sorgt für die Dekodierung, Zusammensetzung und eventuelle Filterung der Rohdaten. Von dort gelangen die gemessenen Daten in einen Speicher, der meist als mehr oder weniger großer RAM-Bereich ausgeführt ist. Meist besteht auch die Möglichkeit diesen RAM-Bereich zusätzlich auf eine Festplatte zwischenzuspeichern. Manchmal kann der Datenstrom sogar direkt auf die Festplatte abgespeichert werden.

Gesteuert und angezeigt wird der Meßverlauf meist von „Standardhardware“ in Form eines an das Meßgerät angeschlossenen PC's oder Workstation, die unter heute üblichen Betriebssystemen laufen und zusätzlich einen LAN-Anschluß aufweisen. Hier kommen sowohl MS-Windows als auch UNIX-basierte Systeme zum Einsatz. Wichtig ist hierbei vor allem die Kompatibilität zum sonstigem Computer-Umfeld. So ist eine X-Display Umleitung auf eine entfernte Workstation unter UNIX ein leichtes Unterfangen, unter MS-Windows hingegen fast unmöglich.

Es ergeben sich bei diesem Meßgeräteaufbau einige Probleme die hier kurz angesprochen werden sollen:

- Hohe Datenraten erfordern viel schnellen Speicher (z.B. 32 MB RAM, 155 Mbit: alle 2,6 μs eine neue Zelle, 550.000 ATM-Zellen $\Rightarrow$ maximal 1,44 Sec. Aufnahmezeit bei voller Auslastung!)
- viel Speicher $\Rightarrow$ sehr teuer
- Festplatte daher meist zu langsam
- hohe Auflösung der Zellen-Zeitstempel gefordert

Es wird ersichtlich daß man bemüht ist den anfallenden Datenstrom möglichst klein zu halten, um mit dem vorhanden Speicher länger messen zu können, bzw. um die Meßhardware nicht zu überfordern. Hierzu verwendet man Filtermethoden, wie sie in Kapitel 8.3.4 noch besprochen werden.

Ferner ist für die Messungen ein Lastgenerator sehr hilfreich, der meist schon in das Meßgerät integriert ist. Dieser muß sowohl variable Last- und Fehlerzustände erzeugen, als auch eine Verbindung aufbauen können.

Das in Abbildung 83 beispielhaft gezeigte Gerät „InterWatch 9500“ erfüllt die obigen Anforderungen bestmöglichst und wird an der TU zu Forschungszwecken verwendet. Es besitzt 128 MB RAM, und wird unter UNIX betrieben. Die Eingliederung in das universitäre Rechnerumfeld ist somit gewährleistet, ferner die Möglichkeit relativ langer Messungen durch den großzügigen Speicherausbau.

8.2 Anschlußmöglichkeiten

8.2.1 Außer-Betrieb Messungen

Die Außer-Betrieb Messung erfolgt wie in Abbildung 84 zu sehen ist direkt am Testobjekt, ohne daß dieses am Netzwerk angeschlossen ist. Es können somit Tests durchgeführt werden, die im Betriebszustand nicht möglich wären.Wie zu erkennen ist, besteht das Netzwerk aus zwei

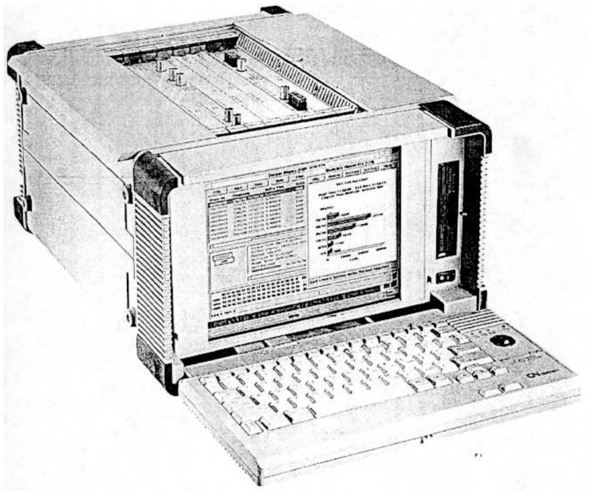


Abbildung 83: InterWatch 95000

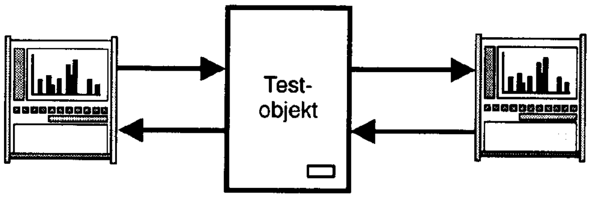


Abbildung 84: Anschluß bei Außer-Betrieb Messung

physikalischen Leitungen, die den Datenstrom entgegengerichtet transportieren. Die Leitung stellt eine „Einbahnstraße“ dar. Um jeweils sowohl das Ausgangssignal als auch die Antwort zu erhalten müssen beide Leitungen angezapft werden.

Bei diesen Test können auf unterster physikalischer Ebene die Parameter des Mediums gemessen werden. Dazu zählen zum Beispiel Jitter, Rauschen oder die Häufigkeit von Bitfehlern.

Komplexer werden die Messungen bei den „Install-Tests“. Hier wird geprüft ob die später in das Netzwerk eingesetzte Komponente sich auch wunschgemäß verhält. Bei ATM-Switches zählt hierzu beispielsweise die korrekte Pfadvermittlung. Durch Simulation der geplanten Vermittlungsvorgänge wird überprüft, ob die ATM-Zellen den Switch auch am richtigen Port wieder verlassen, also die Pfadtabelle im Switch richtig gesetzt ist.

Ferner wird das Verhalten bei normalem Lastniveau und in Überlastsituationen überprüft. Mit einem Lastgenerator wird der Switch möglichst ausgelastet und der Durchsatz beobachtet. Um mit heutigen Lastgeneratoren überhaupt eine volle Auslastung der Komponente zu erreichen ist es u. U. erforderlich den Zellstrom mehrfach durch den Switch zu führen, also jeweils den Ausgang wieder auf einen Eingang zu koppeln.

Für Überlasttests werden Zellen im Zellstrom als verwerfbar markiert und gleichzeitig eine Überlastsituation herbeigeführt. Währenddessen beobachtet man welche Zellen der Switch tatsächlich verwirft.

Für manche Tests werden nicht normale ATM- sondern spezielle Testzellen verwendet, was natürlich nur Außer-Betrieb möglich ist. So werden die Zellen mit Zeitstempel und Sequenz-

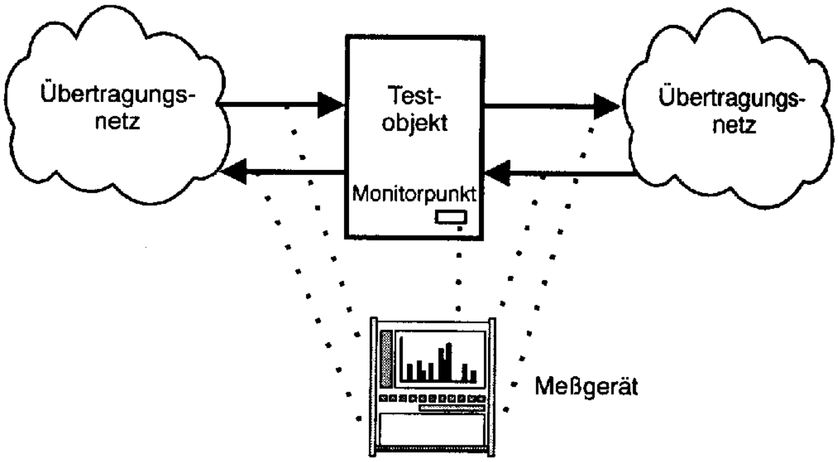


Abbildung 85: In-Betrieb Messung - Monitoring

nummer versehen und auf der Empfängerseite die Zahl der verlorenen oder verworfenen Zellen ausgewertet. Auch die Verzögerungszeit beim Durchlaufen der Komponente kann so ermittelt werden, dazu in Kap. 8.3.2 mehr.

Das generelle Problem der Tests zum Nachweis von Fehlerraten liegt im seltenen Auftreten der Fehlerereignisse: Will man eine Zellverlustrate von 10^{-10} bei 155 Mbit/s nachweisen, so ist eine Meßzeit von mindestens 7 Stunden bei Vollast nötig um auch nur einigermaßen verwertbare Aussagen zu bekommen.

8.2.2 In-Betrieb Messung

Die In-Betrieb Messung findet während des Betriebs statt, möglichst ohne diesen zu beeinflussen. In Abbildung 85 ist der Monitoring-Betrieb skizziert. Hierbei wird das Meßgerät an einen Testausgang am Switch oder an einem physikalischen Verteiler (siehe Kap. 8.2.3) angeschlossen.

Diese Meßmethode greift nicht in den Datenstrom ein. Auch ein eventuell ausgeschaltetes Meßgerät verursacht keine Unterbrechung des Datenstroms, wie dies im Durchgangsbetrieb der Fall ist. Daher ist diese Meßmethode für Langzeitmessungen vorzuziehen. Es können alle Parameter gemessen werden, die keinen Eingriff in den Datenstrom erfordern, z.B. Statistiken, Aufzeichnen des Datenstroms, passive Fehlerzählungen usw.

Will man hingegen in den Datenstrom aktiv eingreifen, so muß man den Durchgangsbetrieb wählen:

Hierbei ist wie in Abbildung 86 dargestellt das Meßgerät direkt in den Datenstrom eingeschleift. Um den vollständigen Datenstrom (in Hin- und Rückrichtung) zu erfassen sind also zwei Unterbrechungen notwendig. Die Daten laufen bei Rx in das Meßgerät hinein, werden dort für das Meßgerät aufbereitet, evtl. manipuliert und auf Tx wieder ausgegeben.

Ein Vorteil des Durchgangsbetriebs: man kann den Datenstrom manipulieren, also je nach Erfordernis zusätzliche Zellen in den Datenstrom einfügen oder daraus entfernen. Zu Meßzwecken fügt man meist spezielle OAM (Operation and Maintance) Zellen ein. Im allgemeinen versucht man diese Zellen möglichst ohne Beeinflussung des gerade stattfindenden Verkehrs nur in vorhandenen Freiräumen unterzubringen. Mit Hilfe der OAM-Zellen werden einzelne Netzwerkelemente nach

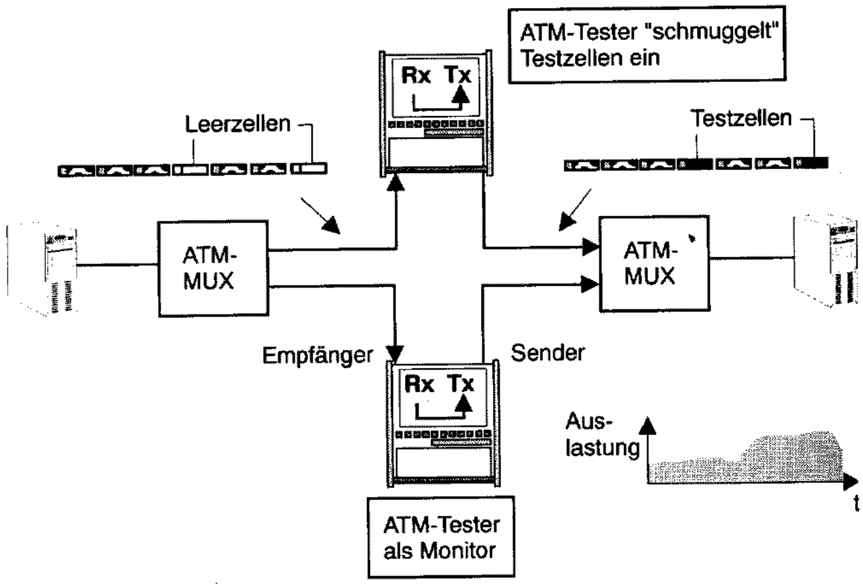


Abbildung 86: In-Betrieb Messung - Durchgangsbetrieb

ihrem Zustand befragt bzw. zu Reaktionen aufgefordert. Die Informationsübertragung findet somit in beide Richtungen statt. Es existieren ebenfalls Performance-Zellen, die mit gleichem VPI/VCI wie die Nutzdaten im Datenstrom „mitschwimmen“ und gewisse Parameter überwachen, z.B. Verzögerungszeiten. Folgende Typen von OAM-Zellen finden Verwendung (wobei F1-F3 hier nicht weiter betrachtet werden):

- F1 - Regenarotorebene: Auf unterster physikalischer Schicht
- F2 - Multiplexerebene
- F3 - Übermittlungsebene
- F4 - Virtuelle Pfadebene: gleicher VPI wie Nutzdaten, VPI aber festgelegt; überwacht einzelnen Pfad
- F5 - Virtuelle Kanalebene: Gleicher VPI/VCI wie Nutzdaten, Überwachung des kompletten Übertragungsweges

OAM F4/F5 Zellen werden durch ihren PTI (Payload Identifier) erkannt und bestehen u.a. aus der Anzahl der übertragenen Nutzzellen (um Zellverluste aufzudecken), sowie einem Zeitstempel um das Laufzeitverhalten untersuchen zu können. Gerade bei Laufzeit - oder Verlustmessungen sind jedoch wegen der geringen Sendefrequenz lange Meßzeiten nötig. Die OAM-Messung stellt daher nur einen Schätzwert dar.

8.2.3 Physikalisch

Wie bereits in Kapitel 8.2.2 angesprochen bedarf es eines physikalischen Verteilers, um das Signal sowohl an das Meßgerät als auch an das ursprüngliche Ziel weiterzuleiten. Da in ATM-Netzen meist optische Übertragung mittels Glasfaser zum Einsatz kommt müssen hier optische Splitter

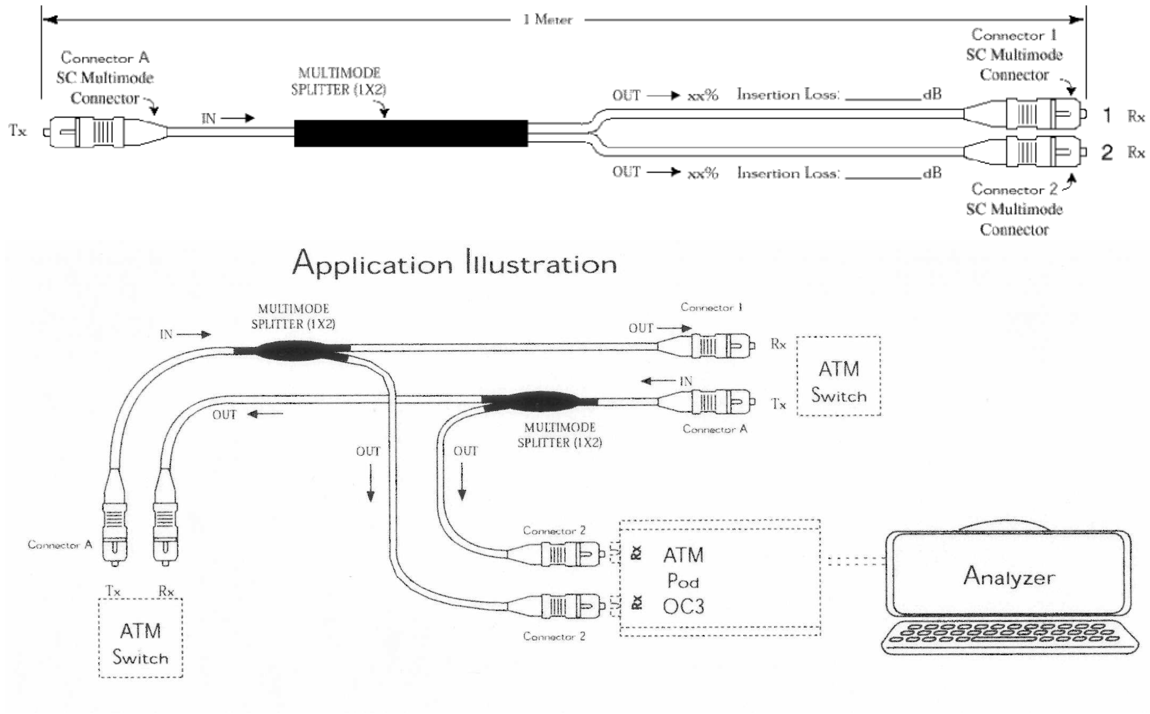


Abbildung 87: Optische Splitter - oben: Prinzip, unten: Anschluß

verwendet werden, wie sie in Abbildung 87 dargestellt sind. Diese Splitter teilen den Lichtstrom in einem variablen (bei der Herstellung bestimmten) Verhältnis auf.

Die Splitter stellen richtungsabhängige Abzweigungen dar, müssen also richtig herum eingesetzt werden. Beim Einsetzen des Splitters ist eine kurze Betriebsunterbrechung nötig, dafür wird aber der weitere Datenfluß vom Zustand des angeschlossenen Meßgerätes nicht weiter beeinflußt.

8.3 Messungen

8.3.1 Physikalische Schicht

Auf unterster physikalischer Ebene lassen sich zunächst einmal die Eigenschaften des Übertragungsmediums bestimmen. Hierzu zählen die Lichtleistung bzw. der elektrische Pegel, Dämpfung, Jitter usw.

Auf der Zellebene basierend lassen sich schon komplexere Zusammenhänge betrachten. Am meisten interessiert hier die „Bit Error Ratio“ (BER). Hierbei wird der Zellstrom vom Ziel zurück zur Quelle geleitet und dort die Fehlerfreiheit der zurückgekommen Zellen überprüft. Dies geschieht indem die Zellen als Payload eine Quasi-Zufallszahlenfolge erhalten, um einen normalen Datenstrom möglichst gut zu simulieren. Durch die Eigenschaften der Quasi-Zufallszahlen ist die Reihenfolge bekannt und es somit möglich, verfälschte Bits zu erkennen. Ein Problem stellt wiederum die lange Meßdauer dar, die bis zu 30 Tagen betragen kann, um beispielsweise bei Sonet/SDH eine BER von 10^{-12} nachzuweisen.

Desweiteren wird auf dieser Ebene noch die korrekte Zellstruktur sowie die Zellbegrenzungen überprüft.

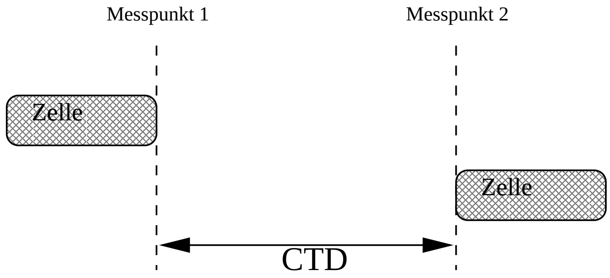


Abbildung 88: Definition der Zellaufzeit (CTD)

8.3.2 ATM-Schicht

Zellverluste, Zellfehler

Ein wichtiger Parameter auf ATM Ebene sind die Zellverlust- bzw. Zellfehlerrate. Die Zellverlustrate (Cell Loss Ratio, CLR) ist definiert als $CLR = \frac{\sum verlorene\,Zellen}{\sum uebertragene\,Zellen}$, die Definition der Zellfehlerrate ist analog. Zellverluste entstehen insbesondere bei hoher Auslastung der Komponenten. Das ATM-Netz sorgt nicht für eine Bandbreitenbegrenzung des Übertragungskanals, dies geschieht ausschließlich durch die Managementebene. Durch Pufferüberlauf in den beteiligten Komponenten kann es nun zu Zellverlusten kommen, insbesondere wenn die Managementebene zu spät eingreift. Normalerweise rechnet man bei einer 2 Mbit ATM-Strecke mit Zellverlustraten von $< 10^{-10}$, das bedeutet maximal eine verlorene Zelle in zwei Tagen.

Zellfehler hingegen entstehen wenn die Zelle zwar ankommt, dort aber gar nicht hingehört. Es handelt sich hierbei meist um Fehlvermittlungen und die Zelle wird vom Switch auf einem falschem Port ausgegeben. Hierzu können sowohl Fehler in der Hardware, als auch Headerfehler der ATM-Zelle beitragen.

Zellenlaufzeitfehler

Um Zellenlaufzeitfehler messen und verstehen zu können muß zuerst die Definition der normalen Zellenlaufzeit (Cell Transfer Delay, CTD) bekannt sein. Die CTD ist definiert als die Zeit zwischen dem Verlassen des 1. Bits von Meßpunkt 1 und der Ankunft des letzten Bits dieser Zelle an Meßpunkt 2, siehe Abbildung 88. Überschreitet die CTD eine gewisse Zeit t_{max} so gilt die Zelle als verloren. Die Verzögerungen entstehen durch die in den Komponenten enthaltenen Puffer sowie durch das Übertragungsmedium selbst. Bei ISDN (64 Kbit) verursacht das Auffüllen einer ATM-Zelle mit 47 Byte Payload bereits 5,8 ms Verzögerung! Bei Lichtwellenleiter rechnet man mit einer Verzögerungszeit von 5 $\mu s/km$, was sich bei größerer Entfernung zu ansehnlichen Zeiten summiert. Ein geostationärer Satellit sorgt sogar für eine Verzögerung von 290 ms (!), für heutige Netzwerke eine Ewigkeit.

Unter der Schwankung der Zellenlaufzeit (Cell Delay Variation, CDV) versteht man das Abweichen der Zellenlaufzeit von einem einheitlichem Durchschnittswert. Die CDV stellt also den Unterschied der Zellenlaufzeiten zwischen Zellen des gleichen Datenstroms dar. Die Schwankungen entstehen durch das Multiplexen und Zusammenfassen von Datenströmen; nicht immer wenn eine Zelle vollständig im Puffer steht kann die Komponente die Zelle auch sofort übertragen. Unter Umständen wird gerade eine andere Zelle verschickt und die bereitstehende muß warten bis sie an der Reihe ist. Diese Verzögerungen addieren sich dann zu dem unvermeidlichen CTD hinzu.

Die CDV läßt sich auf zwei verschiedene Arten messen:

- 2 Punkt CDV: Messung der CDV zwischen zwei Punkten im Netz ähnlich wie bei der CTD. Es werden die Eigenschaften der zwischen diesen Punkten liegenden Komponenten und Netzwerkabschnitten gemessen.
- 1 Punkt CDV: Messung der CDV an einem Punkt im Netz. In diese Messung gehen sowohl die Eigenschaften der Datenquelle als auch der dazwischenliegenden Komponenten und Netzteile mit ein. Es wird die CDV der Quelle mitgemessen.

Zum Messen der Laufzeit muß eine genaue Referenzzeit bekannt sein. Dieses Problem läßt sich auf unterschiedliche Weise lösen:

Die Zellen werden mit einem Zeitstempel versehen und an der Gegenstelle zurück zur Quelle geschickt. Da diese die zeitbestimmende Uhr besitzt kann die Zeitdifferenz zwischen senden und empfangen genau ermittelt werden. Allerdings wird dabei die Hin- und Rückstrecke gemessen, der Wert ist also zu halbieren, wobei eventuelle Unterschiede der beiden Richtungen unberücksichtigt bleiben müssen. Diese Messung ist daher nicht sehr genau durchführbar.

Man kann wie in Abb. 88 gezeigt an zwei verschiedenen Punkten im Netz messen, muß jedoch für synchrone Uhren in beiden Meßgeräten sorgen. Hier kann das GPS zu Hilfe genommen werden, das allerdings aufgrund der künstlichen Ungenauigkeiten und atmosphärischen Einflüsse nur eine Synchronisation im ms-Bereich zuläßt. Unter Verwendung von Differential-GPS (GPS unter Hinzunahme eines Senders an genau bekannter Position) läßt sich letztendlich eine ausreichende Synchronität erreichen.

Mithin werden auch Schätzmethoden verwendet, die ohne exakt synchronisierte Uhren auch hinreichend genaue Ergebnisse liefern.

Ein Beispiel für erlaubte CTD/CDV Werte: Bei nicht interaktivem MPEG ist End-to-End eine CTD von 1s erlaubt, die 2-Punkt CDV <1ms, maximal ein Zellverlust alle 30 Minuten und eine BER von 10^{-10}.

8.3.3 AAL-Schicht

Messungen auf AAL-Ebene sind meist nur von geringem Interesse, da hier erst wieder die höheren Protokollschichten aufschlußreiche Meßwerte liefern. Man kann auf AAL-Ebene die Fehlerbehandlung der AAL-Schicht testen, sowie die Segmentierungs- bzw. Reassemblierungsverzögerungen messen.

8.3.4 Höhere Protokollschichten

Es lassen sich hier im wesentlichen zwei Verfahren unterscheiden, nämlich den reinen Capture-Betrieb, bei dem der komplette eingehende Datenstrom im Meßgerät gespeichert wird, sowie die Online-Analyse.

Der Capture-Betrieb muß den kompletten Datenstrom speichern. Dies geschieht ohne vorherige Auswertung, was zwar Rechenleistung spart, aber dafür umso mehr schnellen und teuren Speicher erfordert. Der Capture-Betrieb ist meist nur in den RAM-Speicher sinnvoll, da eine evtl. vorhandene Festplatte die nötige Datenrate nicht bewältigen könnte. Trotzdem ergeben sich wegen der enormen Datenmengen und der Begrenztheit des Speichers sehr kurze Meßzeiten.

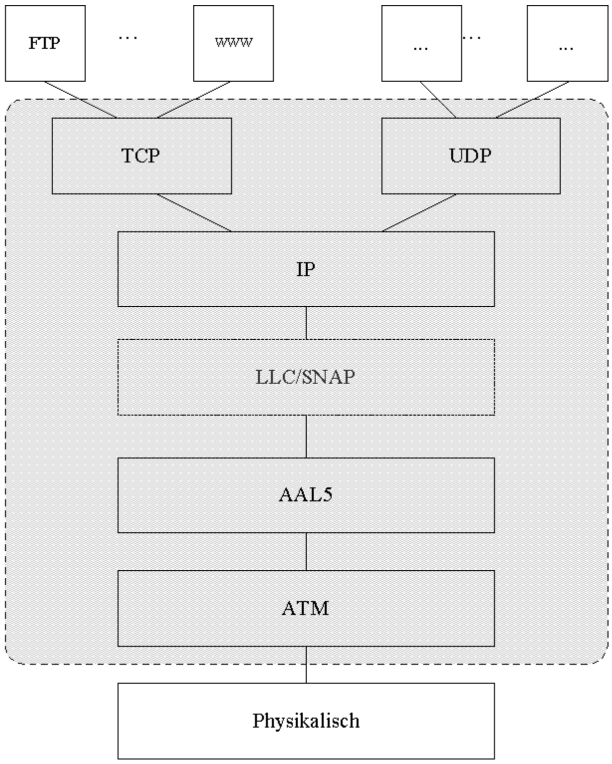


Abbildung 89: TCP/IP Protokollstack

Anschließend an den Capture-Lauf können die Rohdaten vom Rechner ohne Echzeitanforderungen ausgewertet werden, es können also auch kompliziertere und rechenzeitintensive Operationen vorgenommen werden.

Meist ist man allerdings nicht an der ganzen Datenmasse interessiert sondern will über längere Zeit nur bestimmte Arten von Daten beobachten. Hierzu muß man den eingehenden Datenstrom filtern und nur die relevanten Daten bzw. Header der Datenpakete speichern. Das Filtern kann auf unterster ATM-Ebene zunächst sehr einfach erfolgen, indem man nur einen bestimmten VPI/VCI betrachtet. Um allerdings auf Anwendungsebene die Daten filtern zu können, muß im Meßgerät der komplette Protokollstack implementiert werden. Dieser erfordert vom Meßgerät hohe Rechenleistung, ermöglicht aber dafür eine sehr gute Speicherausnutzung.

In Abb. 89 ist ein Stack für TCP/IP dargestellt. Er wird in dieser Form ohne die LLC/SNAP-Schicht im B-WIN eingesetzt. Um auf TCP-Ebene filtern zu können, muß das Meßgerät die von der ATM-Schicht kommenden Pakete gemäß AAL zu den kompletten IP-Paketen zusammensetzen. Danach können die IP-Pakete gefiltert werden, wobei für jedes Paket mindestens der Quell-Port, Quell-IP, Ziel-Port, die Ziel-IP sowie die Unterscheidung TCP/UDP vom Meßgerät ermittelt werden muß. Dies alles passiert online mit der gerade an der Leitung anliegenden Datenrate. Darauf aufbauend können die Pakete verschiedenen Applikationen zugeordnet werden.

Nach dem Ermitteln der obigen Daten können nun die uninteressanten Pakete verworfen werden und es muß zur weiteren Analyse nur noch der verbleibende Datenstrom gesichert werden. Trotz allem fallen noch reichlich Daten an, so daß die Meßzeit immer noch relativ beschränkt bleibt: Einen FTP-Datenstrom mit durchschnittlich 570 Kb/s (=2,4 Mbit/s) kann man mit 32 MB Meßgerätespeicher gerade einmal 50 Sekunden lang speichern. Auf einer 8 GB Festplatte sind

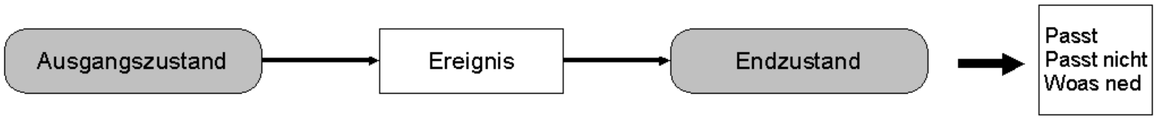


Abbildung 90: Anwendung von Testsuiten

immerhin schon 3,5 Stunden Meßzeit möglich. Es wird ersichtlich, daß die Datenmenge gerade bei länger währenden Messungen ein erhebliches Problem darstellt.

Daher führt man für Langzeitmessungen meist nur Statistiken oder Bandbreitenmessungen durch: Die Pakete werden gefiltert, danach quantisiert (gezählt o.ä.) und dann verworfen. So läßt sich auch über längere Zeit hinweg noch sinnvoll messen.

8.4 Testspezifikationen

Um Aussagen über das Zusammenwirken von Komponenten verschiedener Hersteller treffen zu können, wurden Testspezifikationen entworfen die dieses überprüfen. Es soll sowohl die Einhaltung der ATM-Spezifikationen als auch die Interoperabilität untereinander nachgewiesen werden. Dabei wird die zu überprüfende Komponente in einen Ausgangszustand versetzt, danach mit einem definiertem Ereignis zu einer Reaktion angeregt und im Anschluß überprüft ob der Endzustand konform zur Spezifikation erreicht wurde. Es ist auch ein unentschiedenes Ergebnis denkbar, wenn etwa noch weitere Ereignisse folgen müssen um einen entscheidbaren Endzustand herbeizuführen.

Um die Messungen zu automatisieren existieren für die verschiedenen Gerätetypen die „Abstract Test Suite“ (ATS). Hier sind die Testfälle spezifiziert, die zum Nachweis der Konformität bestanden werden müssen. In dieser Test-Suite sind noch keine Parameter enthalten, diese werden vor dem Test hinzugefügt und ergeben dann die „Executable Test Suite“ (ETS) die in dieser Form direkt in ein Meßgerät übertragen werden kann. Die weitere Messung führt dann das Meßgerät selbständig gemäß den Vorschriften der ETS durch.

Alle grundsätzlichen Protokollfunktionen und Möglichkeiten des Prüflings sind im PICS (Protocol Implementation Conformance Statement) definiert, wobei sich die konkret einzustellenden Parameter im PIXIT (Protocol Implementation Extra Information for Testing) befinden.

Basierend auf diesen Testmöglichkeiten wurden und werden diverse Testfälle zu Einheiten (Suiten) zusammengefaßt, die ganz bestimmte Bereiche abdecken. So enthält beispielsweise der UNI-Bearer Test (User Network Interface, Bearer = Träger) ca. 30 Testfälle mit verschiedenen Verbindungsarten, z.B. Punkt zu Punkt, Punkt zu Mehrpunkt, OAM-Transfer u.v.m.

Es liegen noch weitere Vorschläge zu Testsuiten vor, diese werden aktuell überarbeitet und nach und nach verabschiedet. Für die Zukunft wird es zunehmend wichtig sein, Produkte auf Testsuiten basierend zu spezifizieren um ein problemloses miteinander der verschiedenen Komponenten von unterschiedlichen Herstellern zu gewährleisten.

In dieser Beziehung befinden wir uns heute erst im sehr frühen Anfangsstadium, die oben genannten Suiten werden in nächster Zeit wohl um viele neue bereichert werden (müssen).

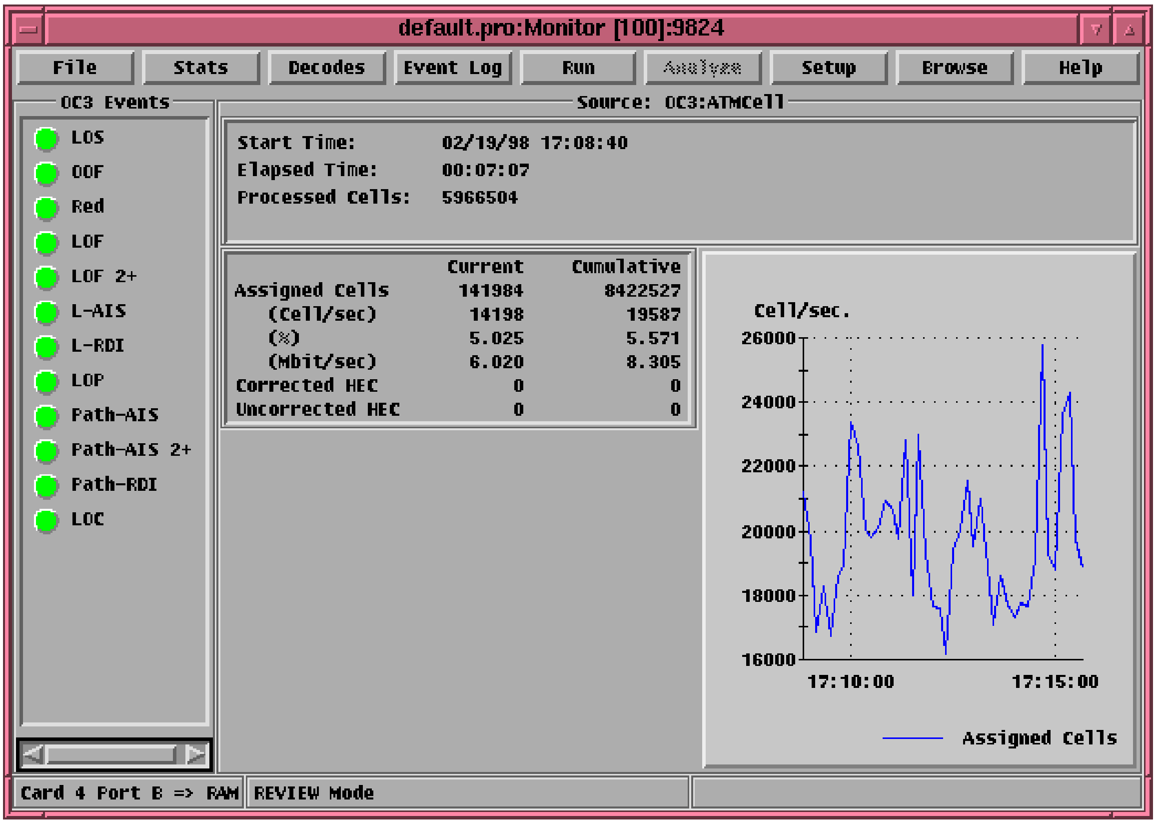


Abbildung 91: Darstellung einer Messung durch InterWatch 95000

8.5 Anwendungsbeispiele

Im Folgenden sollen zur Veranschaulichung einige Beispiele aus der Praxis gezeigt werden. Sie stammen von dem auf Seite 101 gezeigtem ATM-Tester InterWatch 95000. Wie bereits weiter oben ausgeführt ist dieses Gerät durch seine Unix-Bedienoberfläche komplett aus der Ferne zu bedienen. Der Benutzer bekommt auf seine X-Windows Workstation die grafischen Ausgaben des Messgerätes über das angeschlossene LAN exportiert. Somit ist es möglich den InterWatch auch an schwer zugänglichen (sowohl örtlich als auch sicherheitsrelevanten) Stellen einzusetzen.

In Abb. 91 ist die Bildschirmanzeige des InterWatch während einer Messung zu sehen. Das breite Fenster in der Mitte oben enthält die Grundlegenden Daten zur Messung (Datum, Zeit, Anzahl der Zellen). Darunter wird der aktuelle Datenstrom quantifiziert und sowohl die gegenwärtigen als auch die aufaddierten Daten für Zellenzahl, Zellen/sec, Mbit/sec. usw. dargestellt. Die grafische Anzeige rechts davon vermittelt einen ungefähren Überblick über die Verkehrsstatistik. Hier wird die Zellenrate (Cell/sec.) über der Zeit aufgetragen. Es sind hier sehr gut die kurzzeitigen Schwankungen im Zellstrom zu erkennen. Im Fenster „OC3 EVENTS“ leuchtet für jedes auftretende OC3-Ereignis ein symbolisiertes Lämpchen auf.

Abb. 92 zeigt eine andere Darstellungsart des InterWatch. Auf der linken Seite ist symbolhaft der für diese Messung verwendete Protokollstack dargestellt. Die Pakete durchlaufen in diesem Fall also die AAL-Reassemblierungsebene um dann an die IP- und von dort an die TCP-Ebene weitergereicht zu werden. Am Ende des Stacks stehen nun die einzelnen TCP/IP-Pakete zur weiteren Auswertung oder Filterung zur Verfügung. Im oberen Fenster wird jedes empfangene

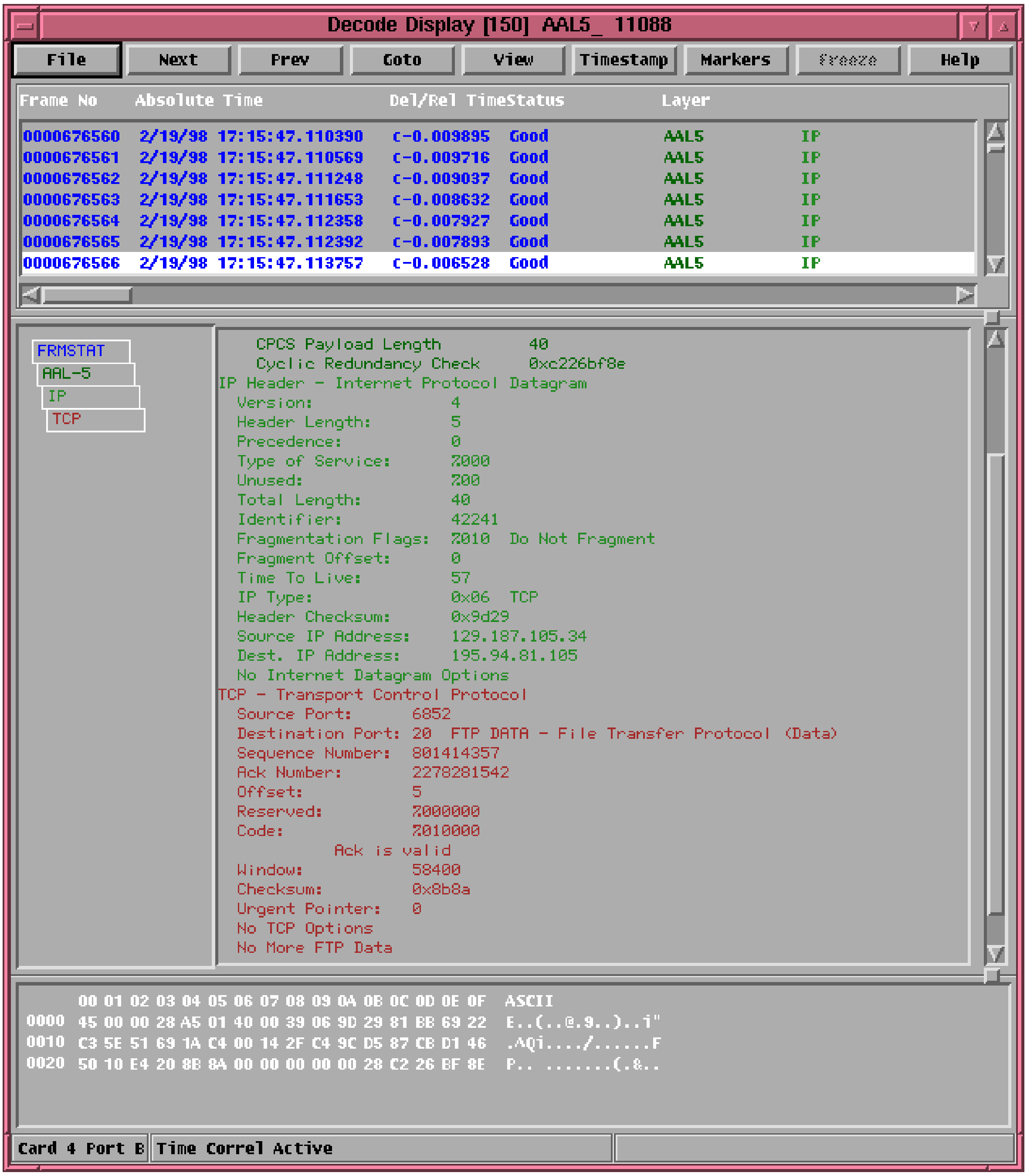


Abbildung 92: Beispielhafter Aufbau des Protokollstacks sowie Anzeige der Rohdaten

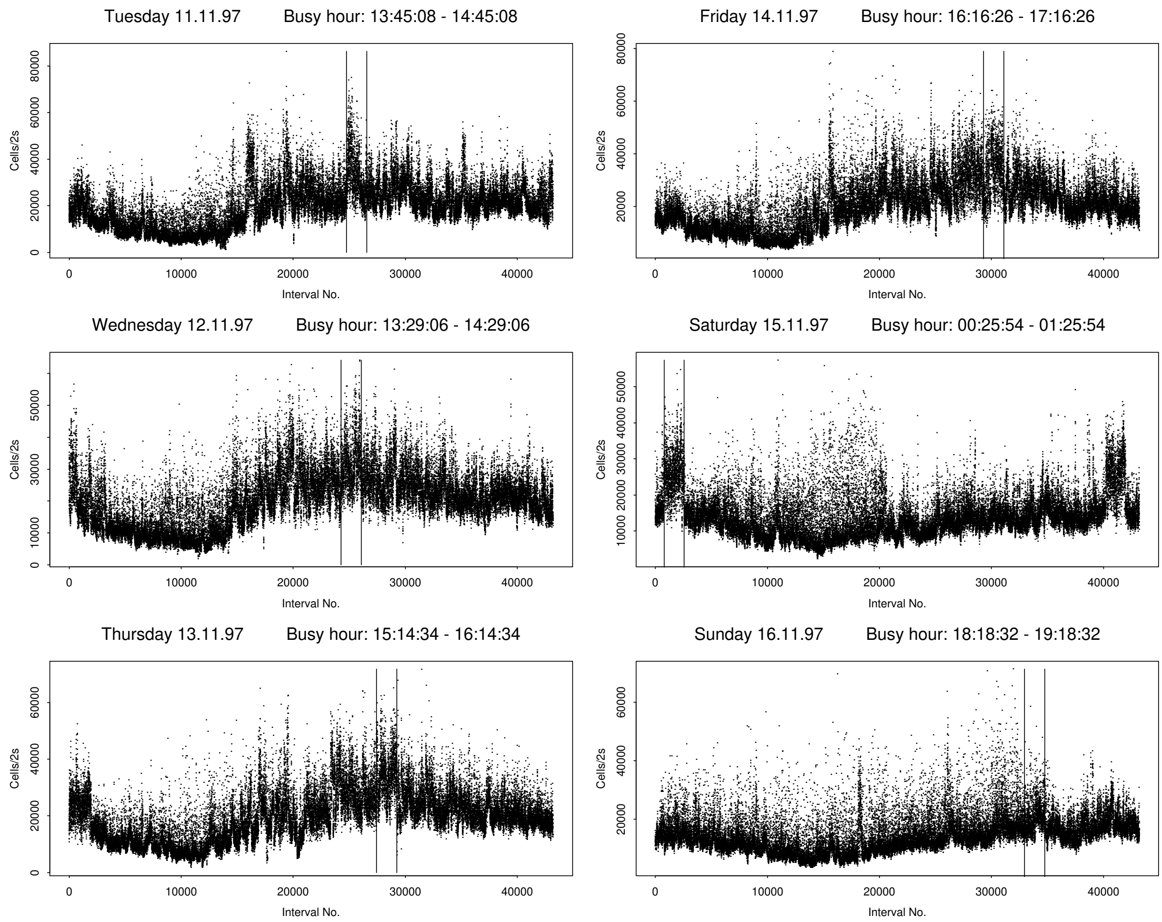


Abbildung 93: Untersuchung zur Verkehrsstatistik

TCP/IP-Paket mit den dazugehörigen Empfangsdaten gelistet. Dort kann man sich ein beliebiges Paket herauspicken und sich die zugehörigen Headerdaten (Quell-/Ziel-IP-Adresse, uvm.) im Klartext anzeigen lassen. Darunter ist der Inhalt des TCP/IP-Paket noch einmal als Hex-Dump zu sehen. Natürlich kann somit auch Einblick in den Dateninhalt des Pakets genommen werden, wobei es sogar möglich ist durch weitreichende Filtermaßnahmen nur die gewünschten Daten herauszufiltern.

Dies belegt wieder einmal sehr deutlich die Notwendigkeit für eine verschlüsselte Übertragung von sicherheitsrelevanten Inhalten in Netzen.

In Abbildung 93 ist abschließend eine Beispielmessung zur Untersuchung von Verkehrsströmen sichtbar. Sie wurde an der ausgehenden Leitung des LRZ zum DFN hin gemessen. Es wurde hierbei für jeden Wochentag ein eigenes Diagramm erstellt, das sich jeweils von 0 Uhr bis 24 Uhr erstreckt. Während dieser Zeit wurden die Anzahl der Übertragenen Zellen alle 2 Sekunden aufgezeichnet und in dieses Diagramm übertragen. Es wurde dann in jedem Diagramm die Stunde mit dem höchsten Verkehrsaufkommen durch zwei senkrechte Striche markiert. Man erkennt einen insgesamt ziemlich schwankenden Verkehrsfluß, der jedoch an den Wochentagen einer gewissen Regelmäßigkeit folgt. So liegt die Hauptverkehrszeit Wochentags am Nachmittag. Am Wochenende zeigt sich hingegen ein völlig anderes Profil: Die am Wochenende anlaufenden automatisierten Dienste (Mirror usw.) sorgen hier für einen starken Anstieg in der Nacht von

Samstag auf Sonntag. Am Sonntagabend wiederum wird die Last hauptsächlich von privaten Nutzern der Datendienste verursacht. Trotz dieser Regelmäßigkeit ist die starke Schwankung erstaunlich, bei der unmittelbar auf extreme Lastspitzen wieder Zeiten mit sehr viel weniger Datenverkehr folgen. Ein Grund hierfür liegt in der verwendeten Netzwerkart (IP over ATM), bei der jeweils für ein IP-Paket eine burstartige Folge von ATM-Paketen übertragen werden muß. Auf diese Periode der höchsten Auslastung schließt sich unmittelbar die Wartezeit auf das nächste IP-Paket an.

Die oben gezeigten Beispiele zeigen nur ein kleines Spektrum der mit dieser Meßtechnik möglichen Untersuchungen. Eine ausschweifendere Darstellung würde diesen Rahmen sprengen; es wird daher auf die umfangreiche Literaturliste verwiesen.

9 IP über ATM / LAN Emulation / MPOA

9.1 LAN-Emulation (LANE)

Bei LANE handelt es sich um eine Technik, die auf MAC Ebene aufsetzt um ein konventionelles LAN, basierend auf Ethernet oder Token Ring Technologie, mit all seinen Funktionen zu emulieren.

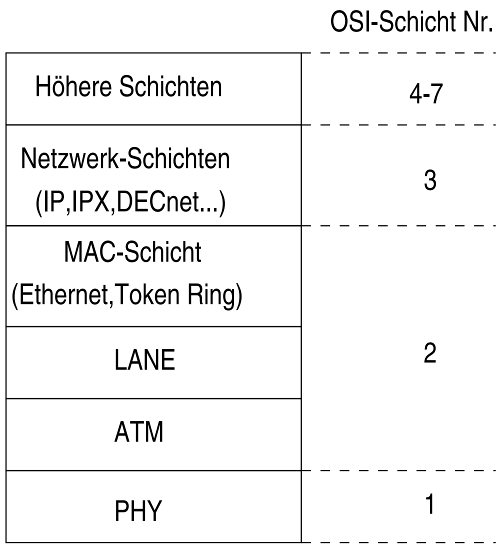


Abbildung 94: LANE im OSI-Schichtenmodell

Das LANE-Konzept V 1.0 stützt sich auf vier Komponenten:

1. LES (LAN Emulation Server)
2. LECS (LAN Emulation Configuration Server)
3. BUS (Broadcast and Unknown Server)
4. LEC (LAN Emulation Client)

Wobei LES, LECS und BUS zum sogenannten LANE Service zusammengefaßt werden. Man sieht also, daß die LAN Emulation eine Client-Server Hierarchie zwischen LECs und dem LANE Service aufweist.

9.1.1 Die Aufgaben der einzelnen Komponenten

• LANE Client (LEC)
Jedes Endsystem (Workstation u.ä.), das die Dienste der LANE nutzen möchte, muß einen LEC besitzen. Seine Aufgabe ist es eine MAC-Schnittstelle für höhere Protokollschichten bereitzustellen. Demzufolge besitzt der LEC zwei Adressen:

1. eine 6 Byte lange MAC-Adresse

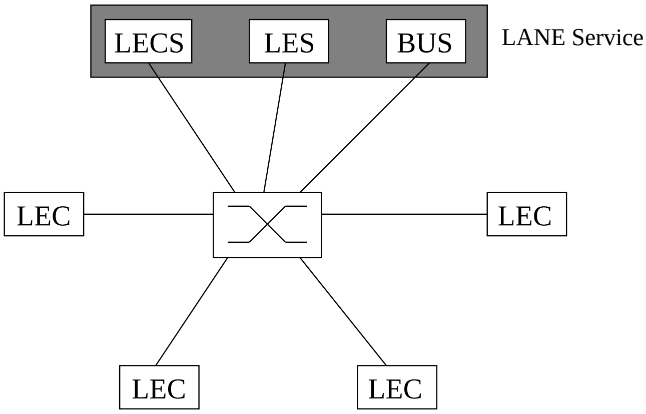


Abbildung 95: Prinzipieller LANE-Aufbau

2. eine 20 Byte lange ATM-Adresse

Je nach Gerät (z.B. LAN-Switch) können einem LEC auch mehrere MAC-Adressen zugewiesen werden.

- **LANE Server (LES)**
Der LES ist die Hauptzentrale des Emulierten LANs. Er beinhaltet die Tabellen der Adreßumsetzung von MAC-Adressen zu ATM-Adressen. Die Adreßauflösung wird über das "LAN Emulation Address Resolution Protocol" (LE-ARP) abgewickelt. Zusätzlich ist er die zentrale Stelle, die die LECs registriert. Es gibt nur einen LES pro emuliertem LAN.

- **Broadcast and Unknown Server (BUS)**
Ist eine ATM-Zieladresse beim LES nicht bekannt, so wird die Adressenzuordnung durch eine "Broadcast Message" vorgenommen. Für diese ist der BUS zuständig. Er verteilt Broadcast- und Multicast-Pakete an die LECs. Zusätzlich sind Routing-Mechanismen implementiert. Jeder LEC kann nur die Funktionen eines einzigen BUS nutzen. Es dürfen aber mehrere BU-Server in einem emuliertem LAN vorhanden sein. Zu welchem BUS ein LEC gehört wird im LES festgelegt.

- **LANE Configuration Server (LECS)**
Es gibt nur einen LECS in einer "Administrativen Domäne". Mit seiner Hilfe lassen sich mehrere virtuelle LANs zentral bilden, überwachen und konfigurieren. Er ordnet den LECs ihren jeweils spezifischen LES zu.

9.1.2 Kommunikation in einem emulierten LAN

In diesem Abschnitt sollen die prinzipiellen Abläufe, die bei der Kommunikation zwischen den LANE-Komponenten auftreten erläutert werden. Hierbei sollen aber der Übersichtlichkeit halber die benötigten ATM-Verbindungen unberücksichtigt bleiben.

Initialisierung und Registration im ELAN

- **Initialisierungsphase**
Kommt ein neuer Client (LEC) zu einem bereits bestehendem emulierten LAN hinzu, müssen

ihm erst einige grundsätzlichen Parameter des LAN übermittelt werden. Diese sind in der Initialisierungsphase z.B. die Paketgröße und die ATM-Adresse des ihm zugeordneten Emulation Servers (LES). Die Informationen werden vom zentralen LECS verteilt. Da ein LEC aber keinerlei Informationen über das Netzwerk zur Verfügung hat, muß die erste Kommunikation über die sog. "Well Known Address" erfolgen. Sie ist keine Adresse im herkömmlichen Sinn, sondern sie steht für eine standardisierte Kombination von VPI und VCI (VPI=0, VCI=17) über die man immer den LECS erreicht.
Der LEC sendet nun ein **LE_Configure_Request** an den LECS. Als Antwort erhält er ein **LE_Configure_Response** vom LECS und wird mit seiner ATM- und MAC-Adresse registriert. Die beim Initialisieren aufgebaute SVC-Verbindung zwischen LEC und LECS bleibt während des gesamten Betriebes bestehen.

- **Join-Phase**
In der nun folgenden Join-Phase muß der LEC sich beim LES anmelden. D.h. er schickt dem LES ein **LE_Join_Request** und erhält im Gegenzug ein **LE_Join_Response** und eine LEC-ID. Für die Kommunikation zwischen LEC und LES wird ein festgelegter VCC verwendet. Es besteht außerdem die Möglichkeit, sich bei mehreren LESn, falls vorhanden anzumelden.

- **Registrations-Prozedur**
Der Join-Phase folgt nun die Registrations-Prozedur. Der LEC sendet seine ATM- und zugehörige MAC-Adresse mit einem **Registration_Request** an den LES, um diese registrieren zu lassen. Die Registration wird mit einem **Registration_Response** quittiert.

- **Anmeldung beim BUS**
Um die Anmeldung am LAN zu vollenden, ist noch die Registration beim BUS nötig. Da nun der LES dem LEC bekannt ist, wird zuerst die ATM-Adresse des BUS mit einer **LE_ARP_Request**-Nachricht an den LES erfragt. Von ihm erhält der LEC mit dem **LE_ARP_Reply** die Adresse des BUS. Nun sendet der LEC ein "Multicast send VCC" an den BUS. Dieser wiederum antwortet mit einem "Multicast forward VCC" und reiht die ATM-Adresse des LEC in seine Multicast-Tabelle ein.

Nachdem der LEC diese vier Stationen (Phasen) durchlaufen hat, ist er vollständig ins emulierte LAN integriert.

ARP im ELAN

Mit Hilfe des ARP im emulierten LAN, können Clients data direct VCCs zu anderen Clients aufbauen. Für die Adreßauflösung im ELAN ist der LES zuständig. Möchte ein Client eine Verbindung zu einem anderen aufbauen, dessen Adresse ihm unbekannt ist, so muß er einen LE_ARP_Request über seinen control point-to-point VCC an den LES schicken. Ist dem LES die Adresse bekannt, schickt er sie über den control VCC zurück an den Client mit einem LE_ARP_Reply. Findet der LES allerdings keinen Eintrag in seiner Tabelle, so schickt er über den control distribute VCC einen LE_ARP_Request an alle ihm bekannten Clients. Ein Client, der diese Adresse besitzt oder für diese Adresse ein Proxy ist, muß auf diese Anfrage des LES reagieren und mit einem ARP_Reply antworten. Diese Antwort wird vom LES an den Anfragenden Client geforwarded.

Datenübertragung im ELAN

- **Multicast Datenübertragung**
Unter einer Multicastübertragungen versteht man eine "Punkt-zu-Mehrpunkt" Verbindung,

bei der ein Host ein oder mehrere Pakete an mehrere Netzwerkteilnehmer gleichzeitig sendet. Für diese Übertragungsart ist im emulierten LAN der BUS zuständig. Pakete, die ein Client über seine Multicast Send VCC oder Multicast Forward VCC an den BUS schickt, werden von diesem an alle Multicastempfänger versandt. Damit ein LEC beim verschicken seiner Multicast-Nachricht nicht seine eigenen Pakete bekommt, muß er im LANE Header jedes Dataframes seine LECID eintragen, nach der er dann ankommende Pakete filtern kann.

• **Direkte Datenübertragung**
Möchte ein Client A Verbindung zu einem anderen Client B aufnehmen und ist A die ATM-Adresse bekannt, so kann er eine direkte Datenverbindung zu B aufbauen ohne Umwege über den LES oder BUS. Ist jedoch nur die MAC-Adresse bekannt, muß A eine ARP-Anfrage an den LES stellen, um die korrespondierende ATM-Adresse zu bekommen. Diese wird dann zum Aufbau einer "Punkt-zu-Punkt" Verbindung zu B verwandt und die Daten werden dann über den data-direct-VCC verschickt.

9.1.3 Ausblick auf LANE 2

In LANE 2.0 wurde mehr Rücksicht auf den redundanten Aufbau des LANE Services genommen. Fiel bei LANE 1 der LES oder BUS aus, so verabschiedete sich auch das gesamte emulierte LAN. Diesen Nachteil hat man durch die Spezifikation der LNNI Schnittstelle behoben. Dadurch ist es möglich geworden, mehrere LES und BUS einzusetzen und damit die Zuverlässigkeit des emulierten LANs zu steigern. Zusätzlich wurde in der Version 2 auch eine "Quality of Service" (QoS) Unterstützung hinzugenommen. Mit diesem ATM Feature lassen sich Bandbreiten im Netzwerk für eine Übertragung reservieren.

9.2 Classical IP over ATM

1994 hat die Internet Engineering Task Force (IETF) "Classical IP and ARP over ATM" entworfen. Es ist einer der möglichen Wege ein traditionelles LAN, das aus IP-Subnetzen besteht mit der ATM Technologie auszurüsten. Classical IP wurde speziell für das IP Protokoll ausgelegt und als Vermittler zwischen ihm und der ATM-Schicht entworfen. Mit ihm können sogenannte "Logical IP Subnetworks" (LIS) gebildet werden.
Abbildung 9.2 zeigt die Einreihung von "Classical IP over ATM" in das OSI-Schichtenmodell.

Die Charakteristiken des klassischen Modells sind:

- Die "Maximum Transmission Unit" (MTU) Größe ist für alle aufgebauten VCs standardmäßig gleich, nämlich 9180 Byte. Bei Punkt-zu-Punkt Verbindungen können allerdings über das "Path MTU Discovery" (RFC 1191) Verfahren andere MTUs ausgehandelt werden, um eine bessere (effizientere) Datenübertragung zu realisieren.
- IP-Pakete werden in LLC/SNAP-Pakete eingekapselt
- End-to-end IP Routingarchitektur bleibt bestehen
- IP-Adressen werden mit Hilfe eines ATMARP service in ATM-Adressen aufgelöst. Es gibt allerdings nur einen ATMARP service pro LIS und auch alle ATMARP-Anfragen bleiben innerhalb des LIS.

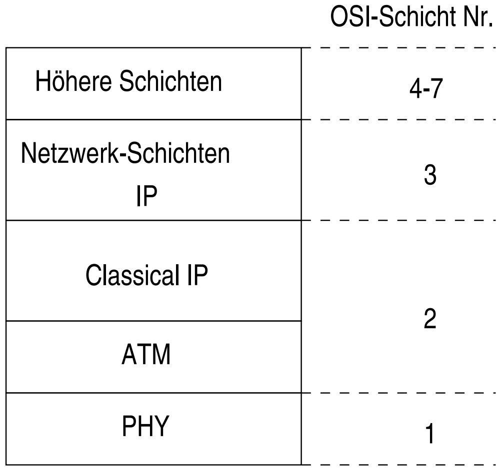


Abbildung 96: Classical-IP im OSI-Schichtenmodell

- Ein LIS wird aus Hosts und Routern geformt. Jeder VC verbindet direkt zwei Teilnehmer eines LIS.

9.2.1 Voraussetzungen für ein LIS

Damit in einem ATM-Netz ein LIS geformt werden kann, müssen folgende Voraussetzungen erfüllt werden:

- Jeder Teilnehmer eines LIS muß dieselbe IP Netzwerk/Subnetzwerknummer und Adreßmaske besitzen.
- Alle Teilnehmer müssen direkt mit dem ATM Netzwerk verbunden sein.
- Wenn SVCs benutzt werden, müssen alle Teilnehmer das ATMARP und InvATMARP-Protokoll unterstützen, das zur Adreßauflösung IP-Adresse -> ATM-Adresse und ATM-Adresse -> IP-Adresse benötigt wird.
- Wenn PVCs benutzt werden, müssen alle Teilnehmer VCs mit Hilfe des InvATMARP IP-Adressen zuordnen können.
- Alle Teilnehmer müssen in der Lage sein, direkt über ATM-Verbindungen mit anderen Teilnehmern im LIS zu kommunizieren.

Jeder IP-Client muß folgende Anforderungen erfüllen:

- Er muß eine eindeutige ATM-Adresse besitzen
- Er muß eine ATMARP Request Liste besitzen. In dieser ist die ATM-Adresse eines oder mehrerer ATMARP-Server im LIS eingetragen. In einer SVC Umgebung darf allerdings im Gegensatz zu einer PVC Umgebung nur eine ATM-Adresse zu einem ATMARP Server aufgelistet sein. Dieser wird für die Adreßauflösung IP -> ATM Adresse benötigt.

LISs können nur über Router kommunizieren. Abbildung 97 zeigt den prinzipiellen LIS-Aufbau.

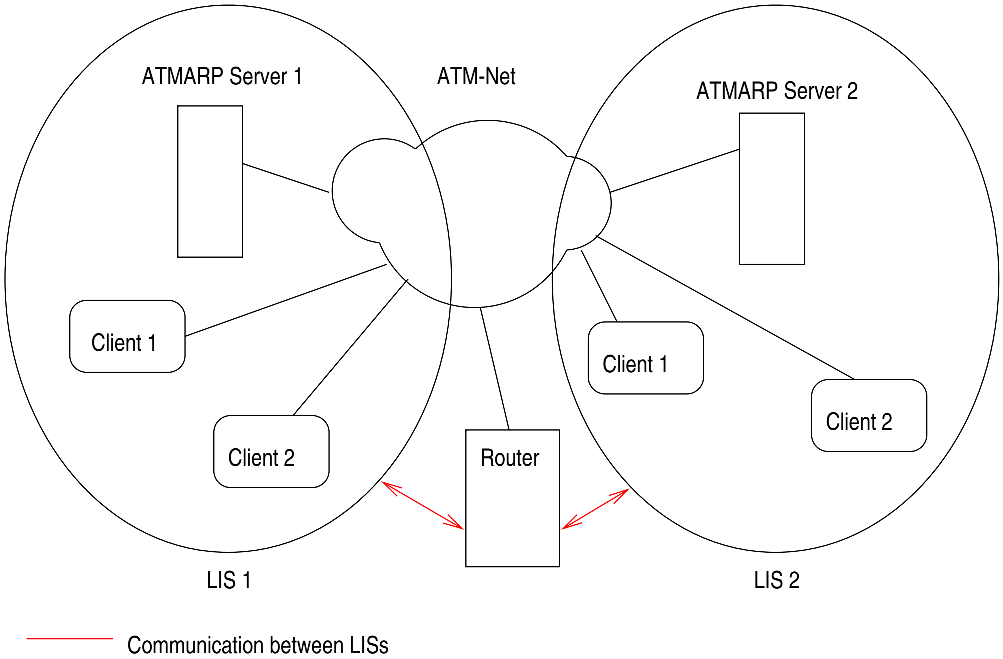


Abbildung 97: Mehrere LISs sind in einem ATM-Netz realisierbar

9.2.2 Packetencapsulation

Da Classical IP, wie schon der Name sagt, nur für das IP-Protokoll ausgelegt ist, stellt es, im Gegensatz zu LANE, eine wesentlich höhere Bandbreite zur Verfügung. Die Begründung liegt in der Art in der die IP-Pakete über ATM übertragen werden. Sie werden direkt in AAL-5 eingekapselt. AAL-5 hat im Vergleich zu AAL-3/4 einen kleineren Overhead und bessere Sicherungsmechanismen (CRC-32). Außerdem eignet sich AAL-5 sowohl für die verbindungslose als auch für die verbindungsorientierte Übertragung.
Die IETF hat zuerst ein Verfahren entwickelt, das es erlaubt, mehrere Pakettypen über eine bereits bestehende ATM-Verbindung zu senden, ohne den Paketverkehr auf dieser Leitung zu beeinflussen (z.B. können Pakete verschiedener Protokolle (z.B. TCP, UDP) über eine bereits bestehende Verbindung gesandt werden. Man erspart sich damit viel Zeit, die zum Aufbau der Verbindung nötig wäre). Um das zu erreichen, müssen die Pakete eindeutig identifiziert und dem jeweiligen Protokoll zugeordnet werden können. Dafür wird die sogenannte LLC/SNAP Encapsulation verwendet.
Bei dieser Methode wird jedem Paket ein "Packet Identifier" in einem LLC/SNAP-Header vorangestellt. Der Identifier ist jedem Protokoll eindeutig zugeteilt. Über ihn kann der Empfänger die Pakete wieder identifizieren. Abbildung 98 zeigt die Einkapselung vom IP-Paket bis zur ATM-Zelle.

Die maximale Paketgröße ist bei "IP über ATM" 9180 Byte. Theoretisch wären 64KByte möglich, jedoch ginge bei Verlust einer einzigen ATM-Zelle auch das gesamte AAL-5 Paket verloren. Dieses muß dann erneut übertragen werden.

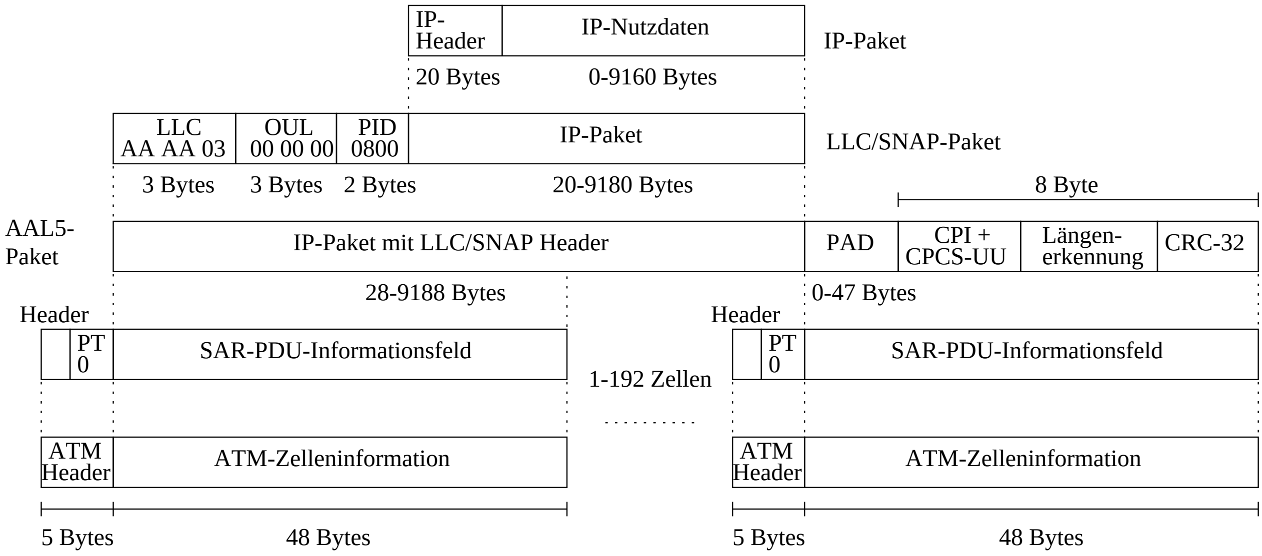


Abbildung 98: LLC/SNAP Encapsulation eines IP-Pakets

9.2.3 Anmeldung beim ATMARP Server, Adreßauflösung, Datenübertragung

Pro LIS gibt es einen ATMARP-Server, der für die Adreßauflösung IP- nach ATM-Adresse zuständig ist. Jeder Client innerhalb des LIS wird mit der ATM-Adresse des ATMARP-Servers konfiguriert, um über diese ARP Anfragen stellen zu können.
Demzufolge ist jeder Client mit seiner IP- und ATM-Adresse beim ATMARP Server registriert. Dies geschieht gleich nach dem Einschalten des Clients. Er schickt einen ARP_Request mit seiner eigenen IP-Adresse an den ATMARP Server. Dieser stellt fest, daß sich die zu ermittelnde Adresse mit der des Anfragenden Clients deckt und registriert ihn in seiner ARP Tabelle. Jeder Eintrag bekommt ein Verfallsdatum zugeteilt, nach dem der Tabelleneintrag verfällt und aus der Tabelle entfernt wird. Um diesen Eintrag zu erhalten, muß der Client sich in regelmäßigen Zeitabständen wieder beim ATMARP Server melden, um das Verfallsdatum zu verlängern.
Soll nun eine Datenübertragung stattfinden, so wird zuerst eine ARP-Anfrage an den ATMARP-Server gestellt. Dieser löst die IP-Adresse zu einer ATM-Adresse auf und schickt diese als Antwort zurück. Nun sind dem Client alle Parameter bekannt, um eine SVC Verbindung aufzubauen. Ist die Adresse dem ATMARP-Server unbekannt, so schickt er die Adresse des Standardrouters an den Client.

9.3 Multiprotocol over ATM (MPOA)

9.3.1 Next-Hop Resolution-Protocol (NHRP)

Das "Next-Hop Resolution Protocol" (NHRP) findet seinen Einsatz in sogenannten "Non-broadcast Multi-access" (NBMA) Netzwerken. Diese zeichnen sich durch die Eigenschaft aus, daß sie keine Broadcast Mechanismen, wie sie in herkömmlichen LANs vorhanden sind, zur Verfügung stellen. Dazu zählen zum Beispiel X.25, Frame Relay oder aber auch ATM.
NHRP verwaltet das NBMA Netz in sogenannten "Local Address Groups" (LAGs). Im Gegensatz zum LIS, in dem nur Hosts mit gleicher IP Netz/Subnetz Adresse direkt miteinander kommunizieren, können in einer LAG alle Hosts, die an einem NBMA Netz angeschlossen sind,

direkte ATM-Verbindungen untereinander aufbauen. Somit werden Umwege über Router, die zwei verschiedene LISs miteinander verbinden, vermieden.

Ein physikalisches NBMA Netz kann in mehrere disjunkte NBMA Subnetze aufgeteilt werden. Ein solches Subnetz besteht aus Hosts und Routern, die uneingeschränkt miteinander kommunizieren können. Zwischen den einzelnen NBMA Subnetzen können mit Hilfe des NHRP Firewalls realisiert werden, so daß es geschützte Bereiche innerhalb des NBMA Netzes geben kann.

Das NHRP ist in erster Linie ein Protokoll zur Adreßauflösung. Bezogen auf IP und ATM bearbeitet es ARP-Anfragen zur Auflösung von IP- nach ATM-Adressen. Das besondere daran ist, daß es erlaubt Adressen über LIS-Grenzen hinaus aufzulösen, und so direkte Datenverbindungen zwischen LISs ermöglicht, ohne auf Router zurückgreifen zu müssen.
NHRP teilt sich in zwei Gruppen auf:

- Next Hop Server (NHS)
- Next Hop Client (NHC)

Der NHC ist ein Host, der eine ARP-Anfrage an den NHS stellt, um eine direkte Datenverbindung zu einem anderen Client aufzubauen. Jeder NHC ist mit der Adresse des ihm zugeordneten NHS konfiguriert, d.h. es gibt pro LIS nur einen NHS. Dieser stellt zwei Funktionen bereit:

- den Servermodus: Jeder NHS innerhalb eines NBMA-Netzes wird statisch über die IP-Adressen der Zielorte konfiguriert, die andere NHS zur Verfügung stellen. Das ist vor allem dann sinnvoll, wenn NHRP in einem kleinen Netz zum Einsatz kommt.
- den Strukturmodus: Hier werden die Zielorte mit Hilfe anderer NHS durch Intra- beziehungsweise Inter-Domain-Routing-Protokolle in Erfahrung gebracht. Außerdem lassen sich die Zielorte entlang eines gerouteten Pfades identifizieren.

Jeder NHS verwaltet einen "Next Hop Resolution Cache" (NHRC) mit IP-zu-ATM Adressenumsetzungen. Bekommt er eine ARP-Anfrage von einem Client, so prüft er seinen NHRC auf dieses Mapping. Ist ein Eintrag vorhanden, wird dieser an den Client zurückgeschickt. Enthält der NHRC jedoch keinen Eintrag zu der angefragten Adresse, so wertet der NHS seine Routing-Tabellen aus. Ist auch dort kein passender Eintrag, so kontaktiert der NHS den nächsten ihm bekannten NHS und stellt seinerseits eine ARP-Anfrage. Diese Prozedur wird solange fortgesetzt, bis ein NHS die gefragte IP-Adresse auflösen kann und auf dem gleichen Weg über alle anfragenden NHSs zurückschickt. Die NHSs auf diesem Weg tragen dieses Mapping nun jeweils in ihren NHRCs ein. Nachdem der NHC die ARP-Antwort erhalten hat, kann eine direkte Datenverbindung aufgebaut werden.
Das NHRP stellt noch andere Zusatzfunktionen bereit, die aber an dieser Stelle nicht behandelt werden sollen.

9.3.2 MPOA-Aufbau

Folgende Protokolle werden benötigt, um ein MPOA Netz aufzubauen:

1. ATM Signalisierung [UNI 3.0, UNI 3.1, oder UNI 4.0]

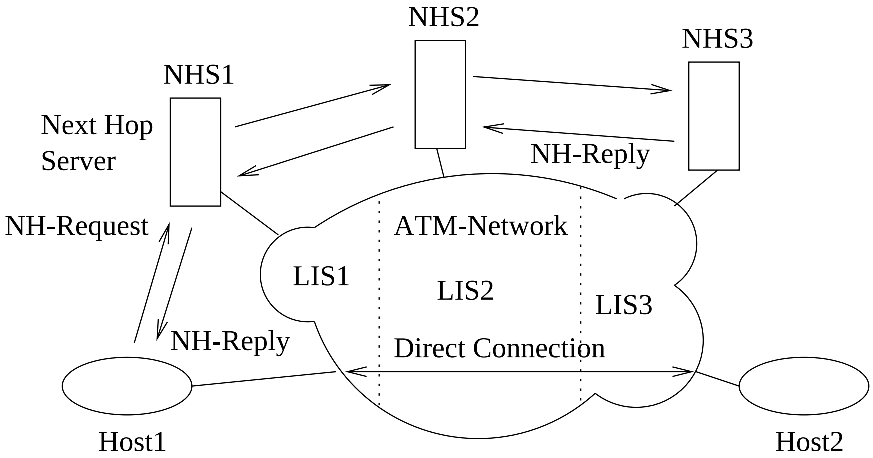


Abbildung 99: Arbeitsweise von NHRP

2. LANE 2.0 [LANE]

3. Next Hop Resolution Protocol [NHRP]

MPOA trennt die verschiedenen Dienste für Routing auf OSI-Schicht 3 und Data Forwarding auf OSI Schicht 2. Diese Methode wird auch Virtueller Router genannt. Dieses Konzept bringt folgende Vorteile:

- Effiziente Kommunikation zwischen Subnetzen
- Höhere Management-Fähigkeit (weniger Geräte zur Routing-Berechnung)
- Höhere Skalierbarkeit
- Edge Devices werden von Routing-Funktion befreit

Da MPOA als eine Client/Server Architektur entworfen wurde, besteht ein MPOA Netz aus zwei logischen Komponenten MPOA Clients (MPC) und deren MPOA Server (MPS). Sie kommunizieren miteinander über einen Satz von Protokollen. Als Transitdomänen kommen emulierte LANs (ELAN), Logische IP-Subnetze (LIS) oder Lokale Adreßgruppen (LAG) vor. Mittlerweile spricht man bei LIS und LAG von sogenannten ION-Subnets (Internetworking Over NBMA Networks). MPOA stellt außerdem die Zusammenarbeit mit herkömmlichen Router-basierenden Netzen sicher, indem die MPS auch an anderen Routing-Protokollen wie OSPF teilnehmen.

MPOA Client (MPC)

Die Hauptaufgabe eines MPCs besteht darin, Internetwork-Shortcuts zu finden und zu merken. Um diese Funktion bereit zu stellen, unterstützt der MPC Layer-3-Forwarding aber kein Routing.
In Eingangsrichtung des ATM-Netzes (Ingress) untersucht der MPC den Fluß von Datenpaketen, der über ein emuliertes LAN an einen Router mit MPS weitergeleitet wird. Erkennt der MPC, daß der Datenfluß einen Shortcut nutzen und so den Router umgehen kann, verwendet er das NHRP-Protokoll, um diesen Shortcut zum Zielgerät in Erfahrung zu bringen. Falls ein Shortcut existiert, speichert der MPC diese Information in seinem Ingress-Cache und leitet die

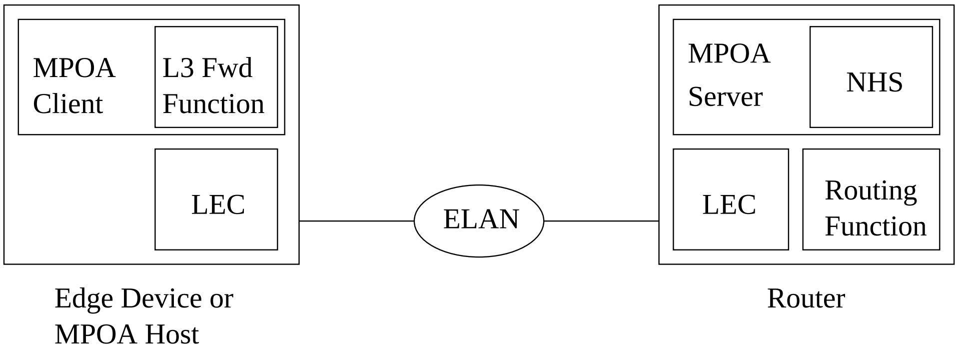


Abbildung 100: Die Komponenten eines MPOA Systems

Datenpakete, die für das Zielgerät bestimmt sind, über den Shortcut weiter.
Aus dem Netz heraus erhält der MPC Schicht-3-Datenpakete von anderen MPCs, die an seine lokalen Schnittstellen/Anwender weitergeleitet werden müssen. Falls Datenpakete über einen Shortcut empfangen werden, fügt der MPC die entsprechende DLL-Kapselung (beispielsweise Ethernet oder Token-Ring-Frame) hinzu und sendet sie weiter über die lokale LAN-Schnittstelle. Die entsprechende DLL-Kapselung wird vom MPS zur Verfügung gestellt und vom MPC in seinem Egress-Cache gespeichert.
Ein MPC kann einen oder mehrere LECs bedienen (die auch aus verschiedenen ELANs oder ELAN-Type stammen können) und aus Skalierbarkeits- und Redundanzgründen mit einem oder mehreren MPS kommunizieren.

MPOA Server (MPS)

Ein MPS ist als logische Komponente Bestandteil eines Routers und stellt den MPCs Internetworking-Informationen für die Datenweiterleitung (Forwarding) auf OSI-Schicht 3 zur Verfügung. Der MPOA Server integriert alle Funktionen eines Next Hop Servers (NHS) und interagiert mit diesem und dem Router, um MPOA-Anfragen von Ingress-MPCs zu beantworten. Für Egress-MPCs hält der MPS die passende Schicht-2-Kapselungen bereit. Der MPS konvertiert MPOA-Anfragen und -Antworten sowie NHRP-Requests und Replies stellvertretend für MPCs.

9.3.3 Beziehungen zwischen LECs, MPOA Komponenten und MPOA Geräten

Ein Edge Device, also ein physisches MPOA Gerät, kann mehrere MPOA-Clients integrieren, die wiederum mit mehreren LECs verbunden sein können. Umgekehrt ist ein bestimmter LEC nur mit genau einem MPC assoziiert. Abbildung 100 zeigt den prinzipiellen Aufbau eines Edge Device und Routers.

Entsprechendes gilt auch für Router. Auch sie können einen oder mehrere MPS in sich vereinen, von denen wiederum jeder einen oder mehrere LECs zugeordnet bekommt. Es kann aber jedem LEC jeweils nur ein MPS zugeordnet sein. (Abbildung 102)

Control- und Data Flows

MPOA definiert eine Vielzahl an Informationsströmen (information flows), die in zwei große Kategorien eingeteilt werden können:

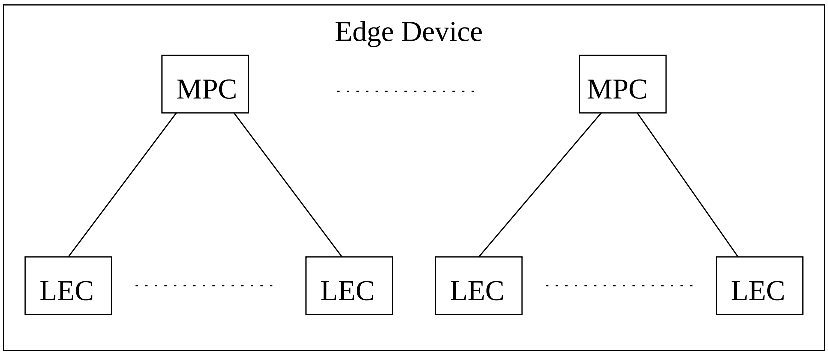


Abbildung 101: Beziehung zwischen MPC und LEC

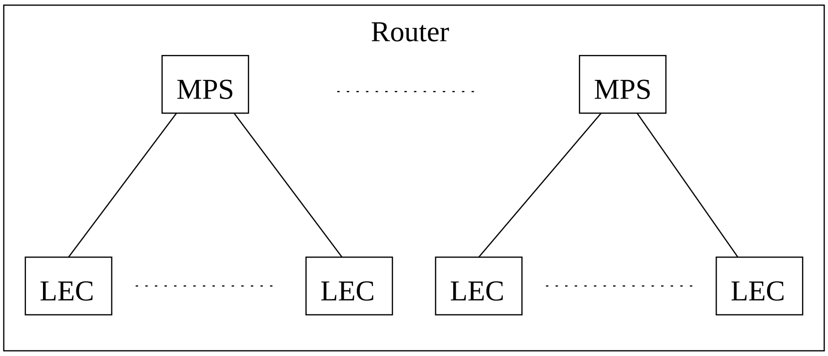


Abbildung 102: Beziehung zwischen MPS und LEC

- Kontrollströme (Control Flows)
- Datenströme (Data Flows)

Standardmäßig werden alle Control und Data Flows über ATM VCCs übertragen. Dabei wird die, in vorherigen Kapiteln bereits besprochene LLC/SNAP Encapsulation, verwendet. Configuration Flows benutzen das LANE-Format.

9.3.4 MPOA Operationen/Funktionen

Configuration

Sowohl MPCs als auch MPSs müssen konfiguriert werden. Standardmäßig müssen alle MPOA-Komponenten in der Lage sein, alle benötigten Konfigurationsdaten vom LECS empfangen zu können. Sie können allerdings auch andere Konfigurationsmöglichkeiten unterstützen. Wie diese allerdings aussehen, bleibt dem Hersteller überlassen.

Discovery

Damit MPOA-Komponenten die Topologie des Netzwerkes langsam kennenlernen können, gibt es die "Discovery-Funktion". Um nicht unnötig viel Netzlast zu erzeugen, sind diese Informationen im erweiterten LE_ARP Protokoll enthalten. In diesem sind nicht nur die ATM-Adresse des jeweiligen Gerätes enthalten, sondern auch sein Typ (MPC oder MPS). Diese Informationen

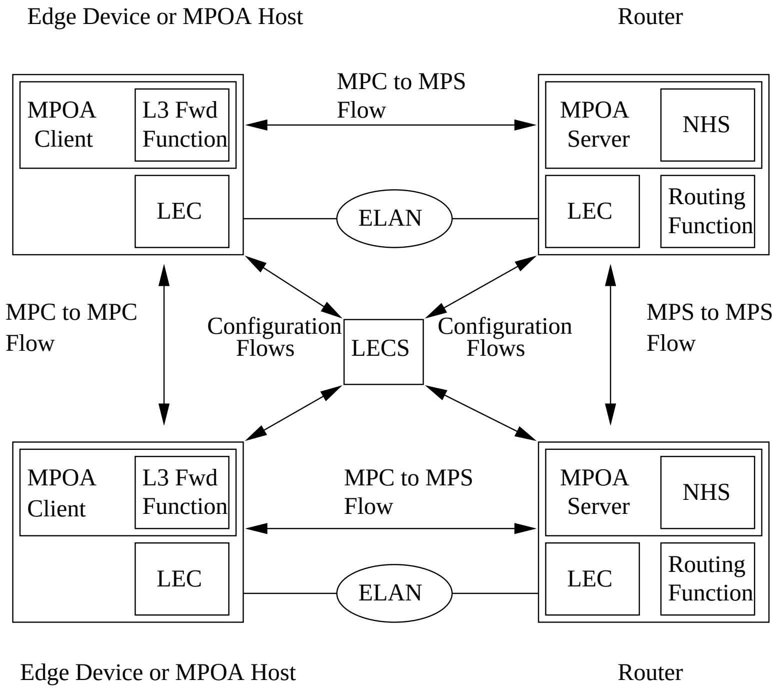


Abbildung 103: Information Flows in einem MPOA System

werden dynamisch erfragt und nach Bedarf genutzt. Sie müssen allerdings nach Ablauf einer bestimmten Zeit wieder aktualisiert werden.

Target Resolution

Für die Adreßauflösung wird ein erweitertes NHRP (s.o.) benutzt. Mit seiner Hilfe können MPCs die ATM-Adresse des Gerätes am Ende eines Shortcuts erfragen.

Connection Management

MPOA Komponenten bauen untereinander VCCs auf, um Datenpakete oder Kontrollnachrichten zu versenden. Sie bedienen sich dafür der Discovery-Prozedur und der Adreßauflösung (s.o.)

Data Transfer

Bei MPOA gibt es zwei Wege, über die Daten übertragen werden können. Das ist zum einen der "Default Path" und zum anderen der "Shortcut". Der Default Path ist die geroutete Verbindung, die über eine LANE 2 Verbindung zustande kommt. Wird ein sog. "Flow" (Datenstrom) vom MPC erkannt, so versucht er einen "Shortcut" durch das ATM-Netz zu finden und aufzubauen. Auf diesen wird dann der Datenstrom umgeleitet. Die Erkennung eines "Flows" ist nicht in der MPOA-Spezifikation enthalten. Sie kann von den verschiedenen Herstellern von MPOA-Komponenten individuell implementiert werden.

9.3.5 Beispiel einer Datenübertragung

Kommt ein Paket über die "Eingangsschnittstelle des ATM-Netzes" (ingress MPC), so wird es standardmäßig zum LANE Router weitergeleitet und somit über die LANE Verbindung weitergereicht. Stellt der MPC fest, daß es sich um ein Paket handelt, das zu einem Flow gehört, für

das bereits ein Shortcut existiert, so wird es nach entfernen der DLL Kapselung über diesen Shortcut versandt. Am "Ausgang des ATM Netzes" (egress MPC) angekommen, wird das Paket auf die Empfängeradresse geprüft. Existiert ein Eintrag der Senderadresse im Cache des egress MPCs für dieses Paket, so wird es der nächst höheren Protokollschicht übergeben. Gibt es einen entsprechenden Eintrag nicht, so wird es verworfen.
Ist kein Shortcut für den Flow vorhanden, so wird über NHRP ein neuer Shortcut aufgebaut. Abbildung 104 zeigt die möglichen Übertragungswege.

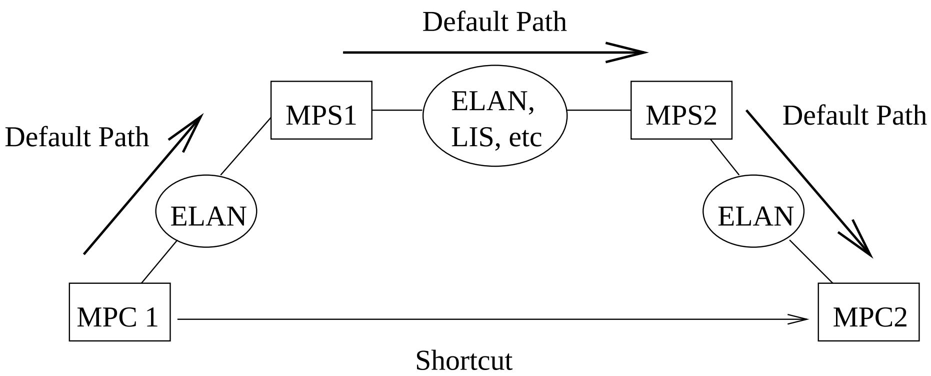


Abbildung 104: Beispiel einer Datenübertragung

9.3.6 Vergleich LANE, Classical IP, MPOA

9.3.7 Vorteile

LAN-Emulation

- universelle Implementierung (MAC-Schicht wird emuliert)
- hohe Funktionalität
- Migration bisheriger LANs auf ATM
- zusätzliche Nutzung von Appletalk und IPX sowie anderen Netzwerkprotokollen aus der LAN-Welt
- kurzer Verbindungsaufbau
- Nutzung der Anwendungsschicht ohne ATM-Konfigurierung
- Unterstützung von PVC- und SVC-Verbindungen
- IP-Multicasting wird unterstützt

Classical IP over ATM

- stabiler Standard
- direkte Nutzung von IP über ATM

- Unterstützung von PVC- und SVC-Verbindungen
- höhere Paketbegrenzung ermöglicht bessere Effizienz

MPOA

- Einbindung von Multiprotokollen in ATM-Netze wird ermöglicht
- Integration traditioneller LANs in ATM
- bessere Ausnutzung der ATM-Eigenschaften (QoS wird unterstützt)
- Nutzung virtueller LANs
- Vermeidung der durch Router verursachten Engpässe
- ATM-Clients können Routing-Informationen nutzen
- Verwaltung großer Netze möglich
- kleinerer Overhead nötig
- geringere Verzögerungszeiten beim Verbindungsaufbau
- IP-Multicasting und später auch ATM-Multicasting werden unterstützt

9.3.8 Nachteile

LAN-Emulation

- Nutzdatenrate zu niedrig
- Paketbegrenzung auf 1500 Byte; dadurch größerer Overhead
- gleiche MTU muß bei allen beteiligten Netzen vorhanden sein
- BUS als Einschränkung bei Multimediaanwendungen.
- LAN-Adapter herkömmlicher LANs begrenzen die Übertragung
- schlechte Skalierbarkeit (ungeeignet für große Netze)
- keine QoS Unterstützung bei V1.0 (bei V2.0 schon)
- Brückenproblematik zwischen Token-Ring und Ethernet wird nicht gelöst

Classical IP over ATM

- bisher keine Broadcast/Multicast-Unterstützung
- Unterstützung anderer Protokolle fehlt
- Verbindungsaufbau kann sehr lange dauern

- schlechte Skalierbarkeit (ungeeignet für große Netze)
- keine direkte Verbindung über verschiedene Subnetze möglich
- keine QoS Unterstützung

MPOA

- komplexe Protokollstruktur
- Stabilitätsprobleme

10 IPnG über ATM/RSVP und Aufbau von Virtual Private Networks

10.1 Übersicht

Netzwerktechniken wie ATM werden in Zukunft sicher eine bedeutende Stellung in IP-Basierten Netzen haben. Dieser Teil wird sich mit neuen Techniken auseinandersetzen, die eine bessere Integration der IP-Welt, insbesondere von IP-basierten Echtzeitanwendungen, in ATM-Netzwerke versprechen.

10.2 Vorstellung von IPnG

10.2.1 Entstehungsgeschichte von IPnG

Durch den enormen Zuwachs des Internets in den letzten Jahren sind einige Probleme entstanden. Bis vor kurzem wurde es hauptsächlich von Universitäten, der High Tech Industrie und Regierungen genutzt. Das alte Internetprotokoll IP Version 4 stößt zunehmend an seine Grenzen. Die neu hinzugekommenen Nutzer stellen neue Anforderungen an das Netz. Die größten Probleme von IPv4 sind die langsam ausgehenden Adressen und die mangelhafte Unterstützung von Dienstarten. Deshalb begann die IETF 1990 mit der Entwicklung eines neuen Protokolls. Dieses neue Protokoll sollte im wesentlichen folgenden Anforderungen genügen:

- Einen Adreßraum zu Verfügung stellen, bei dem die Adressen voraussichtlich nie ausgehen werden
- Reduzierung des Umfangs der Routingtabellen
- Vereinfachung des Protokolls, damit Router Pakete schneller abwickeln können
- Höhere Sicherheit als das heutige IP (Authentifikation und Datenschutz)
- Mehr Gewicht auf Dienstarten, insbesondere für Echtzeitanwendungen
- Flexible Unterstützung von Multicasting
- Möglichkeit für Hosts, ohne Adreßänderung auf Reisen zu gehen
- Möglichkeit für das Protokoll, sich künftig weiterzuentwickeln
- Unterstützung der alten und neuen Protokolle in Koexistenz für Jahre

Dazu veröffentlichte die IETF eine Ausschreibung zur Einreichung von Angeboten und Diskussion in RFC 1550. Insgesamt sind 21 Vorschläge eingegangen, die nicht alle vollständig waren. Im Dezember 1992 lagen 7 brauchbare Protokolle auf dem Tisch. Im Endeffekt wurde eine kombinierte Version der drei besten Vorschläge unter dem Namen SIPP (*Simple Internet Protocol Plus*) gewählt und später in IPv6 oder IP next Generation umbenannt. (IPv5 wurde als experimentelles Protokoll bereits benutzt.)

10.2.2 Änderungen von IPv6 zu IPv4

Das vorgeschlagene IPv6 enthält viel von der Funktionalität, die IPv4 so erfolgreich gemacht hat. So wurden alle Basiskonzepte mit ein paar Veränderungen übernommen. IPv6 unterstützt immer noch verbindungslosen Verkehr, der Sender kann immer noch die Größe der Datagramme bestimmen und muß die maximale Anzahl der Hops angeben. Weiterhin enthält IPv6 die meisten Fähigkeiten der IPv4 Optionen, wie z.B. Fragmentierung und Source Routing, wenn auch in einer anderen Form.

Trotz der konzeptuellen Gemeinsamkeiten, ändern sich in IPv6 viele Details. Nach der Vorstellung der wichtigsten Änderungen und dem Hintergrund dazu, werden die Details näher erläutert.

Die Änderungen von IPv6 gegenüber IPv4 können in 4 Kategorien aufgeteilt werden:

- *Größerer Adreßraum.* Die Vervierfachung der Adreßlänge von 32 bit auf 128 bit ist wohl die erwähnenswerteste Änderung. Dies ergibt ungefähr $3 * 10^{38}$ mögliche Adressen. Auf den Quadratmeter der Erdoberfläche bezogen, sind das immerhin noch $7 * 10^{23}$ mögliche Adressen. Man hat errechnet, das selbst bei sehr ineffizienter Adreßvergabe noch einige 1000 Adressen pro Quadratmeter zu Verfügung stehen.

- *Flexibler Header-Aufbau.* IPv6 benutzt einen neuen und inkompatiblen Aufbau der Datagramme. Im Gegensatz zu IPv4, bei dem alle Optionen im Header stehen müssen, führt IPv6 Erweiterungsheader ein, die optional sind.

- *Zusätzliche Optionen.* Wie in IPv4 können IPv6 Datagramme zusätzliche Kontrollinformationen beinhalten. Neben neuen Optionen, die jetzt schon unterstützt werden, haben die Entwickler durch das Erweiterungs-Header Konzept die Möglichkeit geschaffen, in Zukunft das Protokoll mit neuen Features auszustatten, an die man heute noch gar nicht denkt.

- *Unterstützung von Echtzeitanwendungen.* IPv6 enthält Mechanismen, um Ressourcen für Echtzeitanwendungen zu reservieren. Hierauf werden wir ausführlich im Bereich RSVP eingehen.

Im folgenden Teil werden die Änderungen im Detail besprechen. Abbildung 105 zeigt den Aufbau des IPv4-Headers, Abbildung 106 den Aufbau des IPv6-Basis-Header.

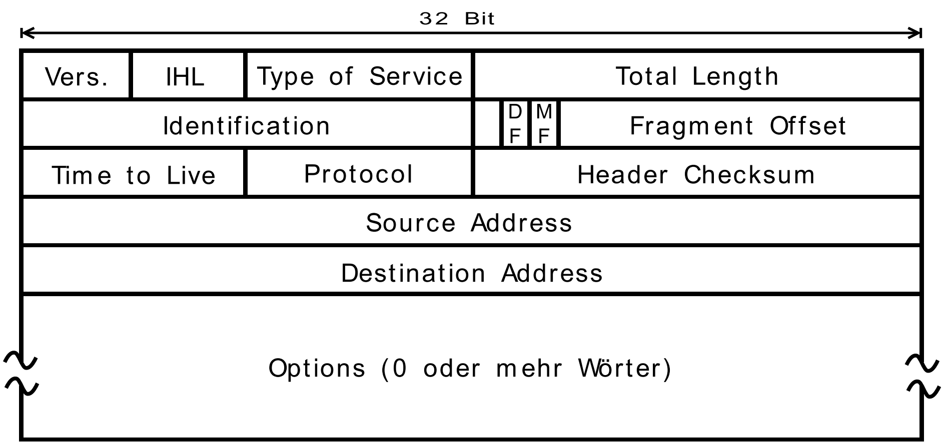


Abbildung 105: Der IPv4 Header

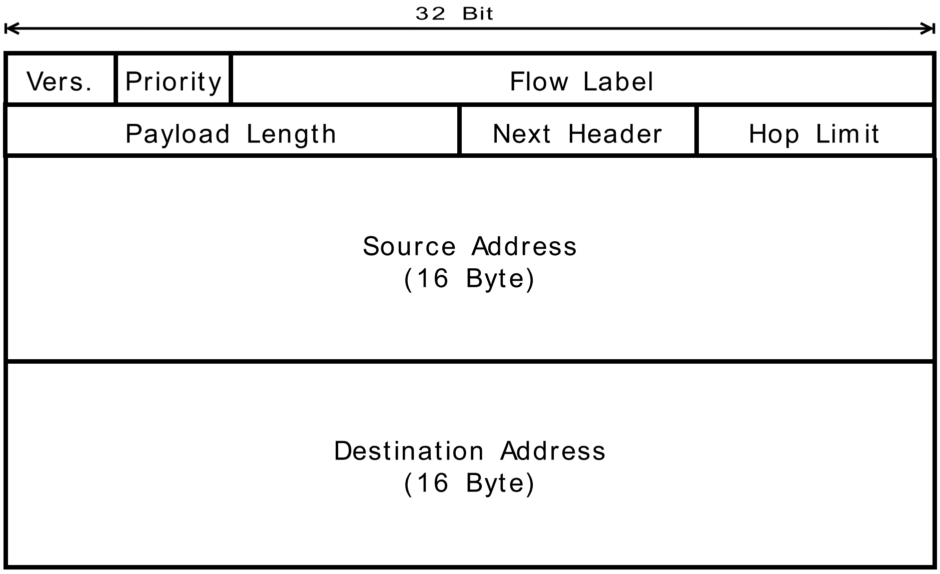


Abbildung 106: Der IPv6 Header

Wie man sieht, ist die Anzahl der Einzelfelder im Header reduziert worden. Das Feld *Version* ist identisch geblieben. Bei IPv4 enthält es immer den Wert 4, bei IPv6 immer 6, und gibt die Versionsnummer der Protokolls an.

Das Feld *IHL*, das die Länge des Headers mit Optionen als Anzahl von 32 bit Wörtern angibt, wurde weggelassen, da der IPv6-Basis-Header immer 40 Bytes lang ist. In diesem 4-Bit Feld muß mindestens der Wert 5 stehen (Header ohne Optionen = 20 Byte) und kann maximal den Wert 15 enthalten (Header mit 40 Byte Optionen = 60 Byte). Der Nachteil ist, daß die Optionen auf maximal 40 Byte beschränkt werden. Für manche Optionen, wie z.B. die Aufzeichnung der Route eines Pakets, ist dies viel zu wenig.

Mit *Type of Service* kann einem Teilnetz die gewünschte Dienstqualität mitgeteilt werden. In der Praxis wird dieses Feld von den heutigen Routern ignoriert. Es wurde durch *Priority* ersetzt. *Priority* kann Werte zwischen 0 und 15 annehmen, wobei 0-7 für Pakete stehen, die bei Überlast verzögert werden dürfen und 8-15 für Pakete von Echtzeitverkehr. So ist z.B. der Wert 1 für News, 4 für FTP und 6 für Telnet vorgesehen, da die Verzögerung eines Newspakets um ein paar Sekunden nicht feststellbar ist, die Verzögerung eines Telnetpakets aber sehr wohl.

Total Length wurde durch *Payload Length* ersetzt, wobei *Total Length* die Länge des kompletten Datagramms inklusive Header und Optionen angibt, und *Payload Length* nur die Länge des auf den Basisheader folgenden Teiles angibt.

Identification gibt an, zu welchem Datagramm ein Fragment gehört. Diese Funktionalität wurde bei IPv6 in einem eigenen Erweiterungsheader ausgegliedert.

DF und *MF* sind Steuerbits, die ebenfalls die Fragmentierung betreffen und nicht im IPv6 Basis-Header vorhanden sind, ebenso wie das Feld *Fragment Offset*, das die Position des Fragments im Datagramm angibt.

Time to Live, das ursprünglich die Zeit angeben sollte, die ein Paket leben kann, bis es verworfen wird, wurde durch *Hop Limit* ersetzt. In der Praxis hat man aber auch *Time to Live* wie *Hop Limit* als Feld eingesetzt, das bei jedem Hop um 1 reduziert und bei 0 verworfen wird.

Protocol gibt an, an welchen Transportprozeß (z.B. TCP, UDP ...) ein komplettes Datagramm weiterzuleiten ist. Bei IPv6 steht dies entweder im *Next Header*-Feld, oder, falls noch Erweiterungsheader folgen, im *Next Header*-Feld des letzten Erweiterungsheaders.

Die *Header Checksum*, die die Integrität des IPv4 Headers sicherstellt, wurde ersatzlos gestrichen, da man bei den heutigen Netzwerken die Fehlerwahrscheinlichkeit als zu gering betrachtet, als daß sich der enorme Rechenaufwand in den Routern bei der Integritätsprüfung noch lohnen würde. Außerdem kann die Integrität in höheren Schichten auch noch überprüft werden.

Source Address und *Destination Address* sind von der Bedeutung her gleich geblieben, abgesehen von der Vervierfachung der Adreßlänge.

Das *Options*-Feld wurde, wie schon erwähnt, durch die Erweiterungsheader abgelöst.

Neu ist das Feld *Flow Label*. Es kann dazu verwendet werden, ein Paket einem bestimmten Fluß zuzuordnen, der bestimmte Anforderungen an Durchsatz und Verzögerung stellt. Als Beispiel wäre hier die VPI/VCI Kombination eines virtuellen ATM-Channels, um einer Applikation (z.B. ein Video Stream) einen eigenen Kanal mit garantierter Güte zu Verfügung zu stellen. Darauf wird im Kapitel RSVP noch einmal eingegangen.

Das Feld *Next Header* verweist, wie schon erwähnt, auf den nächsten Erweiterungsheader oder den Transportprozeß, an den das Datagramm weitergereicht wird. Der Erweiterungsheader wird im nächsten Kapitel ausführlich behandelt.

10.2.3 Die IPv6 Erweiterungsheader

Da in manchen Situationen zusätzliche Felder notwendig sind, hat man sich bei IPv6 auf optionale *Erweiterungs-Header* geeinigt. Abbildung 107 zeigt den Aufbau eines IPv6-Paketes. Die

Abbildung 107: Aufbau eines IPv6-Datagramms

Erweiterungs-Header werden benutzt, um zusätzliche Informationen bereitzustellen. Da sie jeweils auf eine Aufgabe spezialisiert sind, können sie effizient kodiert werden. Derzeit sind sechs Erweiterungs-Header definiert (siehe Tabelle 9). Alle Erweiterungs-Header sind optional. Werden aber mehrere benutzt, so müssen sie aufeinanderfolgend nach dem festen Header und vorzugsweise in der in Tabelle 9 vorgegebenen Reihenfolge erscheinen.

Jeder Erweiterungs-Header hat eine eine festgelegte Nummer (sie steht im Hext Header Feld des vorherigen Header). Einige der Erweiterungs-Header haben ein festes Format, andere enthalten eine variable Anzahl von Feldern mit variabler Länge. Bei diesen Feldern wird jedes Element als Tupel (Typ, Länge, Wert) kodiert. *Typ* ist ein 1-Byte-Feld, das die Option bezeichnet. *Length* ist ebenfalls ein 1-Byte-Feld. Es gibt an, wie lang der Wert ist (0 bis 255 Byte). *Wert* kann beliebige Informationen mit einer Länge von bis zu 255 Byte enthalten. Im Anschluß folgt eine kurze Beschreibung der Erweiterungs-Header:

- Der *Header für Teilstrecken (Hop-by-Hop)* wird für Informationen benutzt, die alle Router auf der jeweiligen Strecke prüfen müssen. Bisher wurde eine Option definiert: Unterstützung von Datagrammen, die über 64 KByte hinausgehen (Auch Jumbogramme

Erweiterungs-Header	**Beschreibung**
Optionen für Teilstrecken (Hop by Hop)	Verschiedene Informationen für Router
Routing	Definition einer vollen oder teilweisen Route
Fragmentierung	Verwaltung von Datagrammfragmenten
Authentifikation	Echtheitsüberprüfung des Senders
Verschlüsselte Sicherheitsdaten	Informationen über den verschlüsselten Inhalt
Optionen für Ziele	Zusätzliche Informationen für das Ziel

Tabelle 9: Die IPv6 Erweiterungsheader

genannt). Damit soll es möglich sein, auch Datentransfer im Gigabyte-Bereich effizient zu tätigen. Selbstverständlich müssen alle Router auf der Strecke Jumbogramme unterstützen, ansonsten wird die Beförderung der Jumbogramme abgelehnt.

- Der *Routing-Header* führt bestimmte Routen auf, die das Pakete auf Weg zum Ziel durchlaufen muß. Sowohl striktes Routing (Paket muß von *Router 1* über *Router 2*... bis *Router n* laufen), als auch loses Routing (Paket muß über beliebigen Pfad zum *Router 1*, dann über beliebigen Pfad zum *Router 2*... bis *Router n* laufen) und Kombinationen sind möglich. Dieser Erweiterungsheader kann sinnvoll sein, wenn z.B. aus Sicherheits- oder Kostengründen das Paket möglichst lange in dem eigenen Teilnetz befördert werden soll, oder beim Multicasting eine Satellitenstrecke benutzt werden soll. Es können bis zu 24 einzelne Stationen angegeben werden.

- Der *Fragment-Header* betrifft die Fragmentierung in ähnlicher Weise wie IPv4. Der Header enthält den Datagrammbezeichner, die Fragmentnummer und ein Bit, das bezeichnet, ob weitere Fragmente folgen. Im Gegensatz zu IPv4 kann in IPv6 nur der Quellhost ein Paket fragmentieren. Router entlang der Strecke sind dazu nicht berechtigt (Ausnahme: IPv6 durch IPv6 Tunneling, bei dem der Router die Ursprungs-Pakete in neue fragmentiert verpacken kann, und ein anderer oder das Zielsystem wieder auspacken kann). Das ist zwar ein größerer Bruch mit der bisherigen Philosophie, vereinfacht aber die Arbeit der Router und beschleunigt das Routing insgesamt. Wird ein Router mit einem zu großen Paket konfrontiert, verwirft er es und sendet statt dessen ein ICMP-Paket an die Quelle zurück. Dadurch wird der Quellhost angewiesen, das Paket zu fragmentieren und es in kleineren Stücken erneut zu übertragen.

- Der *Authentifikations-Header* bietet einen Mechanismus, durch den der Empfänger eines Pakets sicher sein kann, wer es gesendet hat. Bei IPv4 gibt es diese Garantie nicht (oder höhere Schichten müssen dies übernehmen). Hier können die Daten zwar noch unverschlüsselt versendet werden, erhalten aber eine verschlüsselte Prüfsumme, die die Herkunft des Paketes sicherstellt.

- Der Header für *Verschlüsselte Sicherheitsdaten* ermöglicht die Verschlüsselung des Inhalts eines Pakets, so daß es nur der beabsichtigte Empfänger lesen kann. Man kann sowohl standardisierte als auch eigene Verfahren verwenden.

- Der Header für *Zieloptionen* kann für Felder benutzt werden, die nur vom Zielhost interpretiert werden müssen. Vorläufig sind nur Nulloptionen zum Auffüllen dieses Headers auf ein Vielfaches von 8 Byte definiert. Er wird also vorläufig nicht benutzt, kann aber künftig neue Optionen der Hostsoftware zu Verfügung stellen.

Der bedeutendste Vorteil an dem Erweiterungs-Header Konzept ist die Möglichkeit in Zukunft zusätzliche Features ins Protokoll zu implementieren. Mit RSVP werden zum Beispiel ein neuer Erweiterungs-Header definiert.

10.2.4 Aufbau der IPnG-Adressen

Wie bei IPv4 assoziiert IPv6 eine Adresse mit einer speziellen Netzwerkverbindung, und nicht mit einem einzelnen Host. Ein Router kann 2 oder mehr Adressen haben, ein IPv6-Host mit nur einer Netzwerkverbindung benötigt nur eine. In IPv6 hat ein physikalisches Netzwerk auch einen Adreß-Präfix. Um die Adreßzuweisung und -änderung leichter zu gestalten, ist es gestattet, einem Netzwerk mehrere Präfixe zuzuordnen, und auch mehrere Adressen einen Computer für verschiedene Schnittstellen zuzuordnen. Grundsätzlich lassen sich die Adressen in 3 Typen einteilen:

- ***Unicast***: Ein Paket mit einer Unicast-Adresse spezifiziert genau einen einzigen Computer (Host oder Router). Das Datagramm sollte auf dem kürzesten Weg zum Zielcomputer geroutet werden. Die Unicast-Verbindung ist die klassische Punkt zu Punkt Verbindung.
- ***Anycast***: Ein Paket mit einer Anycast-Adresse spricht genau einen Rechner einer Gruppe mit dem selben Adreß-Präfix an, typischerweise den am nähesten gelegenen. Ein Datagramm wird somit zu genau einem Rechner der Gruppe geroutet. Kann zum Beispiel für die Spiegelung eines HTTP-Servers genutzt werden, um die Netzlast zu verringern.
- ***Multicast***: Ein Paket mit einer Multicast-Adresse wird allen Mitgliedern der Multicastgruppe ausgeliefert, auch wenn diese in verschiedenen Netzwerken über die Welt verteilt sind.

Der IPv6-Adreßraum teilt sich, wie in Tabelle 10 gezeigt, auf. Momentan sind erst 28% des Adreßraumes vergeben. Die übrigen 72% sind für künftige Verwendungszwecke reserviert, die heute noch gar nicht vorstellbar sind.

Adressen, die mit 8 Nullen beginnen, sind für IPv4-Adressen reserviert. Die Verwendung von separaten Adressen für Provider und für geographische Bereiche ist ein Kompromiß zwischen den verschiedenen Vorstellungen von der Zukunft des Internets. Adressen für Provider sind sinnvoll, wenn man bedenkt, daß es in Zukunft viele Unternehmen geben wird, die ihren Kunden Internet-Dienste anbieten. Diese Unternehmen sollen einen Bruch eines Adreßraumes erhalten. Die ersten 5 Bit nach dem Präfix 010 geben an, in welchem Register der Provider ist. Derzeit gibt es 3 Register (Nordamerika, Europa und Asien). Später können bis zu 29 Register hinzugefügt werden.

Der Registerverwaltung steht es frei, die verbleibenden 15 Byte nach eigenem Ermessen aufzuteilen. Es wird erwartet, das viele 3 Byte lange Provider-Nummern benutzen werden, was etwa 16 Millionen Provider ergibt. So können Großunternehmen als ihre eigenen Provider fungieren. Eine andere Möglichkeit ist die Verwendung eines Bytes, um nationale Provider zu bezeichnen, und diesen die weitere Zuweisung zu überlassen. Auf diese Weise können nach Bedarf weitere Hierarchieebenen eingefügt werden.

Präfix (binär)	Verwendung	Bruch
0000 0000	Reserviert(einschließlich IPv4)	1/256
0000 0001	Nicht zugewiesen	1/256
0000 001	OSI-NSAP-Adressen	1/128
0000 010	Novell Netware IPX-Adressen	1/128
0000 011	Nicht zugewiesen	1/128
0000 1	Nicht zugewiesen	1/32
0001	Nicht zugewiesen	1/16
001	Nicht zugewiesen	1/8
010	Adressen für Service-Provider	1/8
011	Nicht zugewiesen	1/8
100	Adressen für geographische Bereiche	1/8
101	Nicht zugewiesen	1/8
110	Nicht zugewiesen	1/8
1110	Nicht zugewiesen	1/16
1111 0	Nicht zugewiesen	1/32
1111 10	Nicht zugewiesen	1/64
1111 110	Nicht zugewiesen	1/128
1111 1110 0	Nicht zugewiesen	1/512
1111 1110 10	Verbindungsspezifische lokale Adressen	1/1024
1111 1110 11	Standortspezifische lokale Adressen	1/1024
1111 1111	Multicast	1/256

Tabelle 10: Die IPv6 Adressen

Das geographische Modell entspricht dem im heutigen Internet, bei dem Provider keine große Rolle spielen. Auf diese Weise kann IPv6 beide Adreßarten handhaben.

Die verbindungs- und standortspezifischen Adressen haben nur lokale Bedeutung. Sie können in jedem Unternehmen ohne Konflikt wiederverwendet werden. Außerhalb des Unternehmens besitzen sie keine Gültigkeit. Somit eignen sie sich gut für Unternehmen, die momentan Firewalls einsetzen, um sich vor Zugriffen aus dem Internet zu schützen.

Multicast-Adressen haben nach dem Präfix ein 4-Bit großes Flagfeld, ein 4-Bit großes Umfangfeld und einen Gruppenbezeichner von 112 Bit. Eines der Flagbits unterscheidet permanente und vorübergehende Gruppen. Das Umfangfeld ermöglicht die Begrenzung eines Multicasts auf eine Verbindung, einen Standort, ein Unternehmen oder den Planeten. Diese vier Umfänge werden über die 16 Werte verteilt, so daß später neue Umfänge hinzugefügt werden können. Der planetarische Umfang ist z.B. 14, so daß Code 15 für künftige Erweiterungen des Internets auf andere Planeten, Sonnensysteme und Galaxien frei ist.

10.2.5 Migrationspfad von IPv4 nach IPv6 und abschließende Bemerkungen

Da das bestehende Internet nicht von heute auf morgen auf das neue Protokoll umgerüstet werden kann, muß es Migrationspfade geben, die ein langsames Umsteigen ermöglichen. So sollen IPv6-Inseln entstehen, die dann allmählich zusammenwachsen. Dazu werden folgende Techniken zu Verfügung gestellt:

- *Übersetzung von IPv6 nach IPv4 und umgekehrt.* Auf die neuen Features muß dabei natürlich verzichtet werden.
- *Tunneling von IPv6 durch das IPv4-Netz.* Die IPv6 Pakete werden von einem Dualstack-Router oder Host in IPv4 Pakete gekapselt und durch das IPv4-Teilnetz geschleust. Bei Ende des IPv4-Teilnetzes und Übergang in das IPv6-Netz werden die IPv6-Pakete wieder von einem Dualstack-Router oder Host ausgepackt (Dualstack-Router können sowohl IPv4 als auch IPv6).

Daß IPv6 trotz seiner Vorteile noch nicht sehr verbreitet ist, hat mehrere Ursachen. Einerseits wurde das Problem der ausgehenden Adressen in IPv4 durch Konzepte wie *Classless InterDomain Routing-CIDR* aufgeschoben und IPv4 durch teilweise propritäre Funktionen aufgewertet. Doch dies soll hier nicht das Thema sein.

10.3 Vorstellung von RSVP

10.3.1 Motivation und Einordnung des Protokolls

Eine der zukünftigen Aufgaben des Internet wird die Bereitstellung von Diensten mit garantierter Güte sein, um Echtzeitanwendungen zu unterstützen. Dazu zählen z.B. Datendurchsatz, Varianz und Verzögerung der Datagramme einer Verbindung. Dieses Problem ist nicht trivial, da sowohl IPv4 als auch IPv6 verbindungslose Protokolle sind, die nach dem Prinzip *best effort* arbeiten. Sie können sich ständig neue Wege zum Ziel suchen. Verbindungen werden durch höhere Schichten wie z.B. TCP aufgebaut. Die meisten Netzwerke, die unter TCP/IP arbeiten, sind aber verbindungsorientiert und könnten Garantien für eine Verbindung zur Verfügung stellen (z.B. ATM, FDDI ... etc). Genau hier liegt der Ansatz für RSVP (*Ressource reSerVation Protocol*). RSVP dient als Kontroll-Protokoll, das Reservierungsansprüche von der Applikation an die darunterliegenden Netzwerkschichten weiterleitet (siehe Abbildung 108).

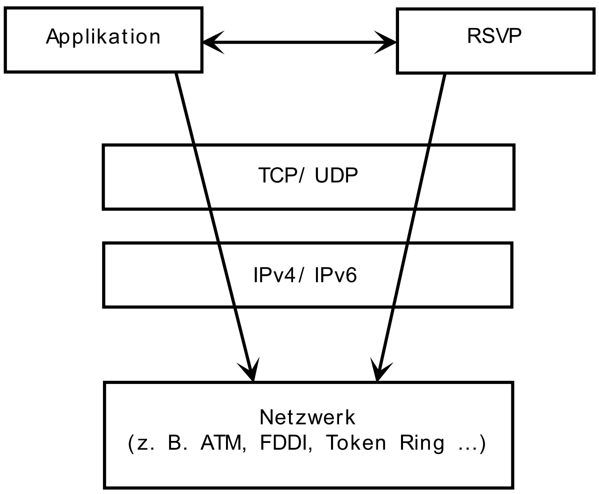


Abbildung 108: Einordnung von RSVP

10.3.2 Anwendungen von RSVP

Im folgenden Teil wird anhand einiger Beispiele die Funktionsweise von RSVP verdeutlicht.

Wenn das Netzwerk akzeptable Performance liefert, hängt der Erfolg oder Mißerfolg einer Echtzeit-Übertragung von IP ab. Angenommen ein Host hat eine ATM-Anbindung an das Internet von 51 Mbit/s. Ein Video-Datenstrom benötigt 31 Mbit/s. Zusätzlich läuft ein Filetransfer mit Bursts von bis zu 30 Mbit/s. Wenn der Router den Filetransfer auf 20 Mbit/s beschränken kann, würden beide Applikationen akzeptabel funktionieren. Dazu benötigt der Router aber die Kenntnis der Anforderungen der einzelnen Ströme.

RSVP beruht auf zwei Schlüsselkonzepten: *flows* und *reservations*.

Wenn RSVP Ressourcen reserviert, werden diese für Datenströme (*flows*) reserviert. Ein Datenstrom wird von RSVP durch die Zieladressen und optional durch den Zielport identifiziert. Anhand des Verkehrs, der den Datenstrom ausmacht, wird die benötigte *Quality of Service* bestimmt. RSVP versteht die Datenstromspezifikation (*flowspec*) nicht, leitet sie aber von der Applikation an die Hosts und Router weiter. Diese Systeme können dann überprüfen, ob sie die Reservierungsanforderungen akzeptieren können oder nicht. Einmal akzeptiert, benutzen sie die *flowspec*, um die benötigten Ressourcen zu reservieren.

Im Gegensatz zu anderen Reservierungsprotokollen erwartet RSVP vom Empfänger die Reservierungsanforderung. Das scheint rückschrittlich. Das System, das den Verkehr erzeugt, sollte am besten über die Anforderungen Bescheid wissen; das stimmt auch meistens in den einfachen Fällen. RSVP unterstützt aber nicht nur einfache Fälle. Die empfängerbasierte Reservierung ermöglicht einen viel flexibleren Umgang mit dynamischen Multicast-Strömen. So können Hosts, die zum Beispiel eine Verbindung mit geringerer Bandbreite besitzen oder über weniger Rechenkapazität verfügen, einen Teilstrom reservieren. Voraussetzung dazu ist natürlich, daß die Applikation dies erlaubt. Der untere Host in Abbildung 109 hat eine 100 Mbit/s Anbindung, während der rechte an einem Token Ring mit 16 Mbit/s hängt. Wenn der Video-Server einen MPEG-Strom mit 30 frames/s sendet (31 Mbit/s), kann der rechte Host zwar nicht alle Bilder empfangen, aber eventuell genügt ihm jedes zweite Bild des MPEG-Stromes. Er reserviert also einen Teilstrom mit 15,5 Mbit/s, während der untere Host die vollen 31 Mbit/s reserviert (siehe Abbildung 110). Zudem ist es einem Host jederzeit möglich, sich zum Videostrom hinzuzuschalten oder auszuklinken, ohne den Server damit zu belasten.

Ein anspruchsvolles Problem für den Empfänger war das Herausfinden des genauen Pfades des Datenstroms. Ein Host weiß z.B. von einem Videomulticast die Herkunftsadresse und will für diesen Strom Ressourcen reservieren. Jetzt könnte man meinen, daß er lediglich seine Reservierungsanforderung zum Videoserver schicken muß, und diese automatisch den umgekehrten Pfad des Multicast-Stromes zum Server folgt. Diese Annahme ist leider falsch! Abbildung 111 zeigt, warum es sinnvoll sein kann, einen Videostrom über eine andere Route zu leiten, als den normalen Verkehr. In diesem Beispiel existieren zwei verschiedene Weitverkehrs-Netzwerke: Alle Router sind sowohl über Satellit als auch über Frame Relay verbunden. Die Satellitenverbindung hat eine hohe Verzögerung und ist somit für den meisten Verkehr nicht ideal. Generell bevorzugen die Router deshalb die Frame Relay Verbindung. Jedoch ist ein Satellit prädestiniert für die kosteneffiziente Übermittlung von Broadcast-Verkehr. Für Multicast-Verkehr bevorzugen die Router also die Satellitenverbindung.

Um die Reservierungen auf dem korrekten Pfad einzurichten, schickt der Sender im Datenstrom spezielle RSVP *path messages*. Sobald ein Router eine *path message* erhält, merkt er sich die letzte Station der *path message* dieses Stroms, und trägt seine eigene Adresse für die nächste Station ein. Erhält nun ein Router eine Reservierungsanforderung von einem Empfänger für einen Datenstrom, so schickt er sie an die vorherige Station des Stromes weiter.

Ein Router muß in der Lage sein, verschiedene Reservierungen intelligent zu verschmelzen. In Abbildung 110 erhält der linke Router 2 Reservierungsanforderungen (Wie in Abbildung 109 dar-

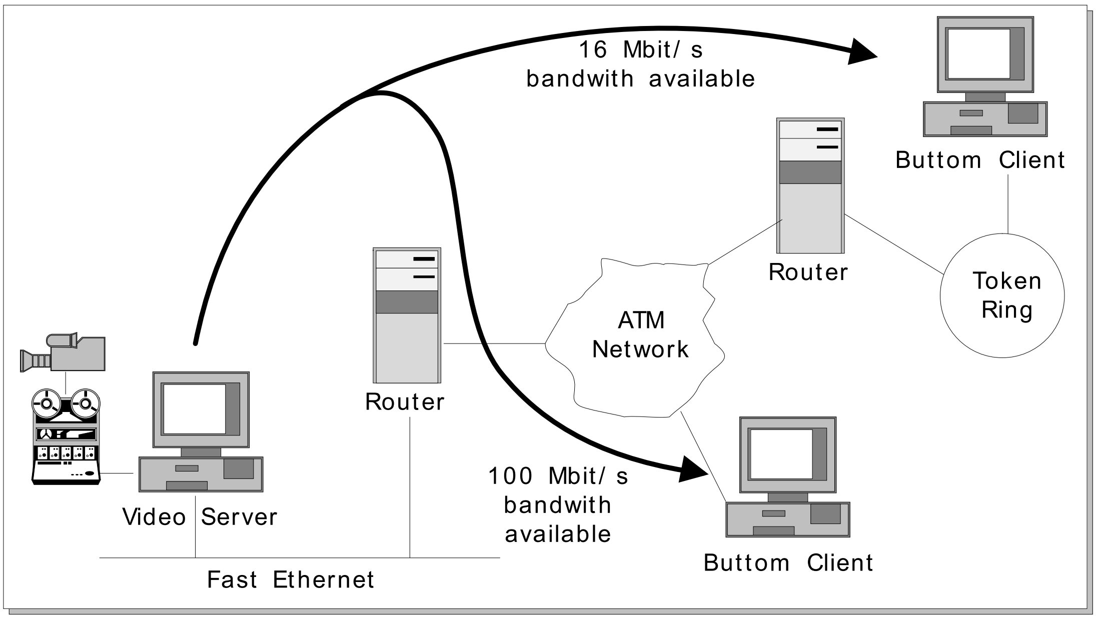


Abbildung 109: Multicast-Datenstrom mit verschiedenen Dienstgüten für verschiedene Ziele

gestellt, kann der rechte Klient nur einen Teilstrom empfangen). Die erste Anforderung benötigt nur 15,5 Mbit/s (hier: nur jedes zweite Bild), die zweite benötigt 31 Mbit/s (jedes Bild). Er muß nun beide Anforderungen kombinieren. In diesem Fall sendet er eine Reservierungsanforderung von 31 Mbit/s zum Videoserver. Verschiedene Ressourcen können verschiedene Behandlungen benötigen. Wenn bei einer Reservierung eine maximale Verzögerung angegeben ist und der Router mehrere verschiedene Anforderungen erhält, wird er den kleinsten Wert der Verzögerung weitersenden.

FF	Fixed Filter	Reserviert Ressourcen für einen Teilstrom
SE	Shared Explicit	Reserviert Ressourcen für mehrere spezifizierte Ströme, die sich die Ressourcen teilen dürfen
WF	Wildcard Filter	Reserviert Ressourcen für eine generellen Art von Strömen, ohne die einzelnen Ströme zu spezifizieren; alle Ströme dieser Art teilen sich die reservierten Ressourcen.

Tabelle 11: Die Reservierungsarten

Bis jetzt wurden nur effektive, aber etwas eingeschränkte Arten der Reservierung beschrieben, bei denen der Empfänger für jeden einzelnen Strom explizit die Reservierung angeben muß. RSVP stellt aber auch noch flexiblere *Reservierungsarten* zur Verfügung. Tabelle 11 zeigt die momentan definierten Reservierungsarten. In Zukunft können noch weitere definiert werden.

Das Beispiel von Abbildung 110 benutzte den Fixed Filter, bei dem jeder Strom seine eigene Reservierung benötigt. Abbildung 112 zeigt eine Anwendung, die einen anderen Reservierungs-Stil braucht. Hier nehmen 5 PC's an einer Audiokonferenz teil. Alle bis auf einen befinden sich im selben Ethernet-LAN. Der fünfte PC benötigt zwei Router und eine Punkt zu Punkt Verbin-

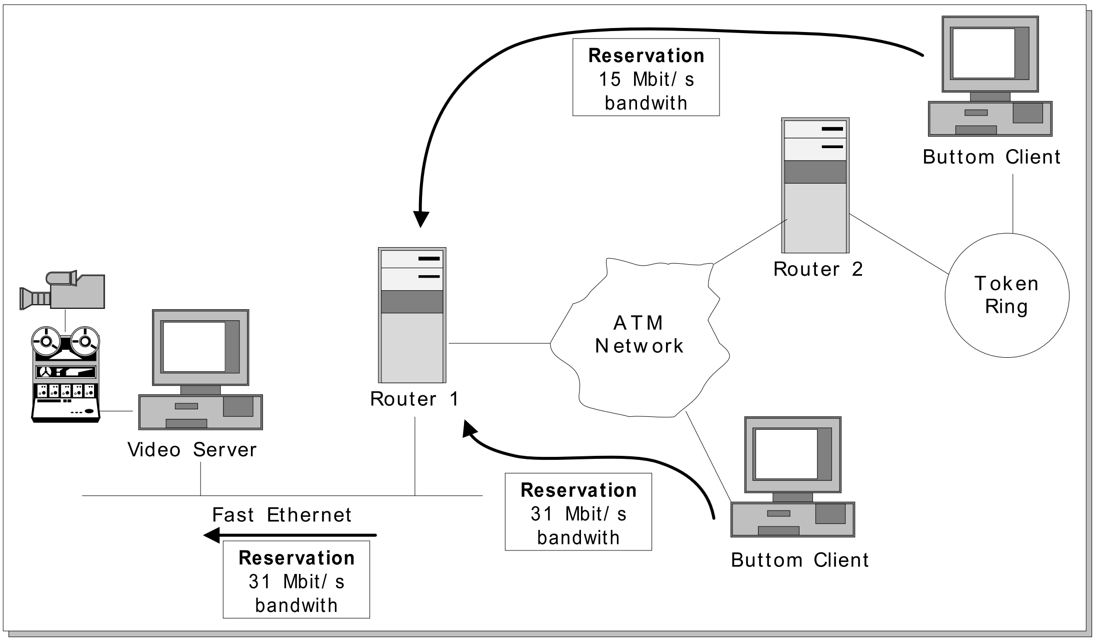


Abbildung 110: Verschmelzen von Reservierungsanforderungen

dung um mit den anderen verbunden zu sein. Das Beispiel konzentriert sich auf die Verbindung zwischen den zwei Routern und den Strom vom Ethernet zum Token Ring.

Weil sich auf der Ethernet-Seite vier PC's befinden, gibt es vier separate Ströme. Vorausgesetzt ein Audiodatenstrom benötigt eine Bandbreite von 16 Kbit/s, dann müßte der 5. PC bei dem *Fixed Filter*-Stil 4*16 Kbit/s = 64 Kbit/s reservieren.

Eine Audiokonferenz hat aber eine besondere Eigenschaft: nicht alle Teilnehmer sprechen gleichzeitig (wenn doch, versteht man sowieso nichts, also muß das Netzwerk diesen Fall nicht unterstützen). In freundlich geführten Konferenzen spricht eigentlich nur eine Person zur gleichen Zeit, also wird von der Applikation nur eine Bandbreite von 16 Kbit/s benötigt. Realistischer Weise sollte man 32 Kbit/s reservieren, um zeitweilige Bursts auch noch zu unterstützen.

Der Unterschied zwischen *Shared Explicit* und *Wildcard Filter* ist, daß der Empfänger bei *Shared Explicit* die Adressen von Teilnehmer 1 - 4 explizit mit angeben muß, während er bei dem *Wildcard Filter* nur z.B. "aller Verkehr mit Zielport 5004" angibt, ohne die einzelnen Sender kennen zu müssen. Diese Art der Reservierung setzt voraus, daß kein anderer Verkehr diesen Port benutzt.

In einigen Dingen scheint die Reservierung von Ressourcen im grundsätzlichen Widerspruch zu TCP/IP zu stehen. Die TCP/IP Protokolle, IP im besonderen, wurden für dynamische Netzwerke entwickelt. Wie gut kann eine Reservierung sein, wenn einer der Router, der sie unterstützt, ausfällt? Und wie geht RSVP vor, wenn plötzlich eine effizientere Route möglich ist?

RSVP trägt diesen Ereignissen mit der Etablierung einer *soft state* Verbindung Rechnung. Der Ausdruck *soft state* wird benutzt, weil sowohl der Pfad als auch die Reservierung ständig erneuert werden. Wenn der Router eine Reservierung akzeptiert, sagt er: "Ich reserviere die Ressourcen jetzt für dich, aber solange du sie brauchst, mußt du sie regelmäßig aktualisieren."

Der Sender sendet die Pfad-Messages in regelmäßigen Abständen, solange der Strom existiert.

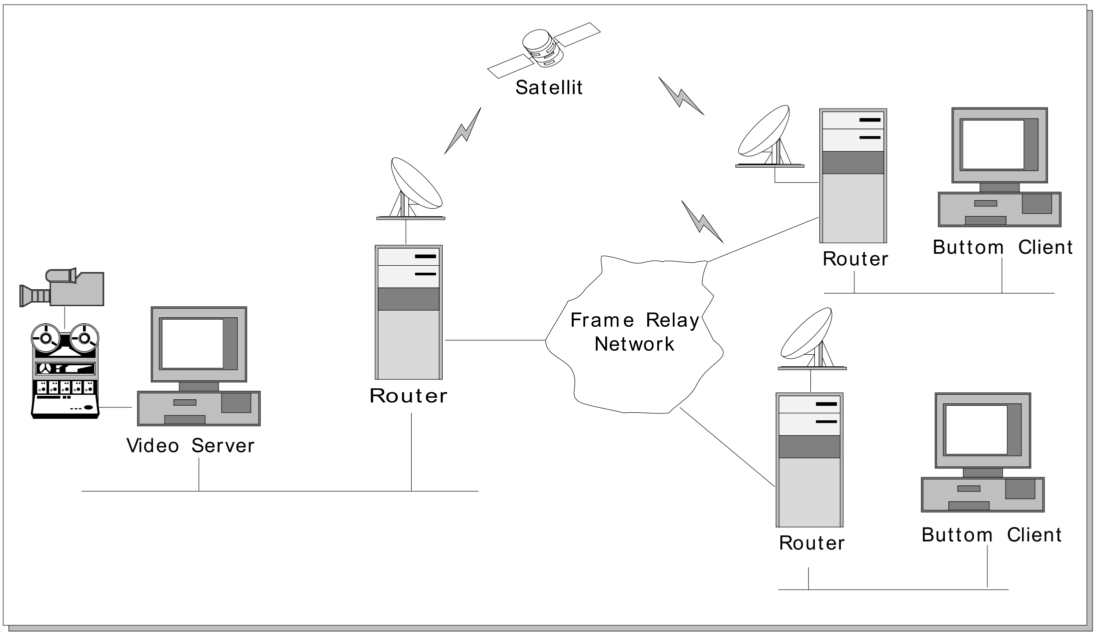


Abbildung 111: Multicast Video bevorzugt den Satelliten, während "normaler" Verkehr über Frame Relay transportiert wird.

Solange der Pfad sich nicht ändert, wird er nur aktualisiert (besser: aufrechterhalten). Wenn der Strom seine Route ändert, werden den neuen Routern im Pfad die Eigenschaften des Stroms mitgeteilt. Die Reservierung arbeitet auf dieselbe Art. Die Empfänger senden periodisch ihre Reservierungsanforderungen. Diese Meldungen aktualisieren die alten Reservierungen, oder sie teilen einem neuen Router die benötigten Anforderungen mit.

Um zu zeigen, wie ein Aufbau einer RSVP-Verbindung funktioniert, werden nun die einzelnen Schritte erläutert, die ein Host benötigt, um sich in einen bestehenden Strom mit einzuklinken. RSVP benutzt dabei mehrere verschiedene TCP/IP-Kontroll-Protokolle. Die Abbildungen 113 und 114 veranschaulichen die folgenden Schritte graphisch.

1. Der besagte Host beginnt mit der Eintragung in die Multicastgruppe des Stroms. Dazu sendet er ein *ICMP Group Membership Report Message* in sein lokales Ethernet.

2. Der Router empfängt diese Nachricht und aktualisiert seine eigenen Tabellen. Um den Rest des Netzwerkes zu aktualisieren, bildet er ein *OSPF Link State Advertisement* und gibt es ins Netz.

3. Sobald alle Router aktualisiert sind, strömt der Multicast-Verkehr zum neuen Klient.

4. Jetzt tritt RSVP in Aktion. Da nun der Multicastverkehr zum neuen Klient kommt, erhält er auch die regelmäßigen *path messages* des Senders.

5. Mit diesen *path messages* kann der Klient nun den Strom identifizieren und seine eigenen Reservierungen festlegen und senden.

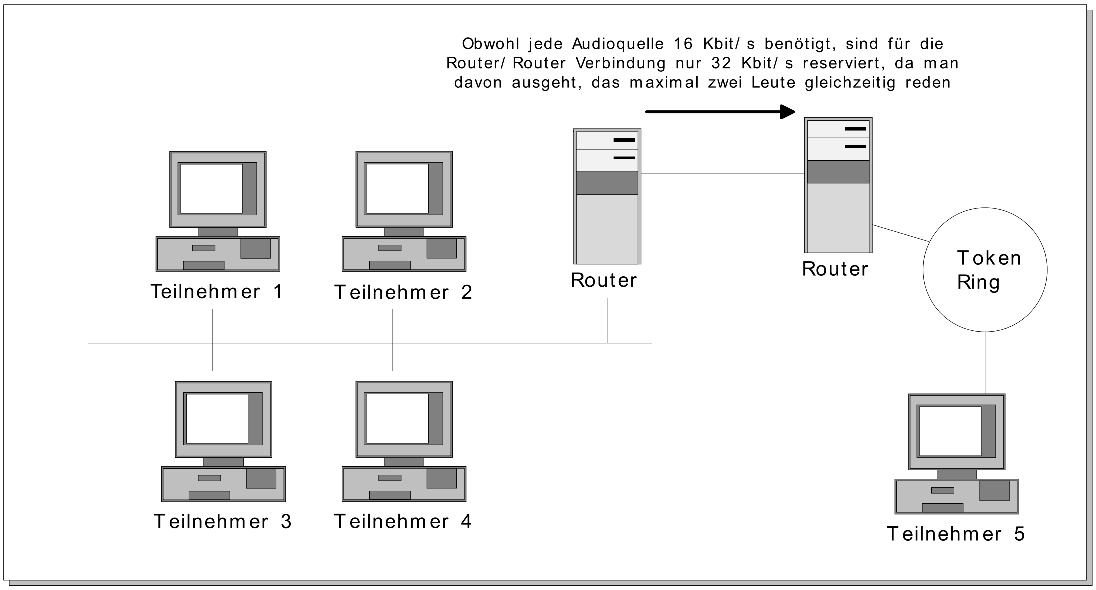


Abbildung 112: Eine Audiokonferenz mit RSVP

Wenn alle Schritte erfolgreich waren, steht nun eine garantierte Verbindung zu allen 3 Clients mit ihren jeweils eigenen Reservierungen. Der Videoserver schickt weiterhin seine *path messages* und die Clients aktualisieren regelmäßig ihre Reservierungen.

10.3.3 Aufbau und Integration von RSVP in IPv6

In dem vorangegangenen Teil wurde die Leistungsfähigkeit und Funktionsweise von RSVP dargelegt. Nun geht es um die technischen Details. Als Kontrollprotokoll wird RSVP in IP-Paketen transportiert. Es kann zwar auch in UDP-Datagrammen transportiert werden, aber nur um auch IPv4 zu unterstützen. Die Nummer des RSVP Erweiterungs-Header in IPv6 ist 46. Jede RSVP-Nachricht hat einen festen Header. Abbildung 115 zeigt zeigt den Aufbau einer RSVP-Nachricht.

Der RSVP-Header besteht aus 8 definierten und 2 reservierten Felder.

Version ist ein 4-Bit Feld, in dem die Version von RSVP steht (momentan: 2). Die nächsten 4-Bit sind *flags*. Sie dienen zukünftigen Erweiterungen des Protokolls und haben momentan keine Aufgabe.

RSVP type gibt die Art der RSVP-Nachricht an. *path messages* und *reservation messages* wurden schon vorgestellt. Zusätzlich gibt es noch 2 Fehler-Nachrichten als Antworten auf die ersten beiden Nachrichten und jeweils eine Nachricht für das explizite Beenden eines Pfades oder einer Reservierung (siehe Tabelle 12).

Das Feld *checksum* überprüft die Datenintegrität. Die *Message Length* gibt die Länge der RSVP-Nachricht an (RSVP-Header und Body).

Message Identifier, More Fragments und *Fragment Offset* ermöglichen und regeln die Fragmentierung einer RSVP-Nachricht. Dabei gibt der *Message Identifier* an, zu welcher Original-Nachricht das Fragment gehört, *More Fragments*, ob noch weitere Fragmente folgen und *Fragment Offset* die Position des Fragments in der Originalnachricht.

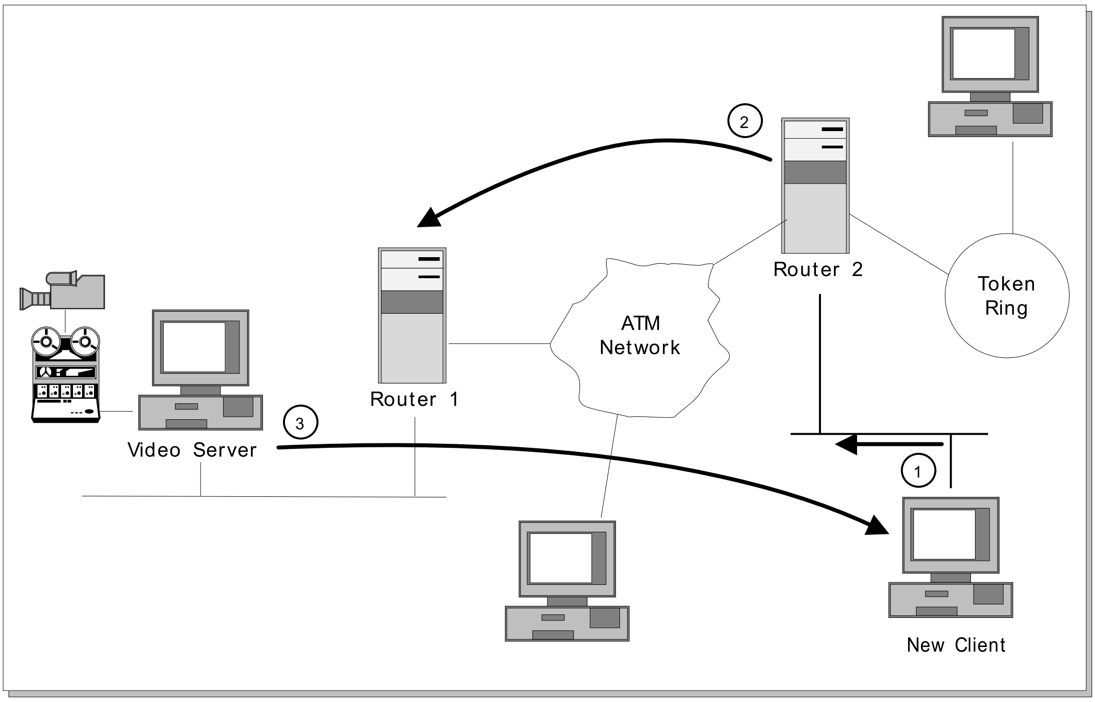


Abbildung 113: Ein neuer Klient tritt in die Multicastgruppe einer Videoübertragung ein

1	path message
2	reservation message
3	error indication in response to path message
4	error indication in response to reservation message
5	path teardown message
6	reservation message

Tabelle 12: Die *RSVP Message Types*

Der Inhalt einer RSVP-Nachricht besteht aus einer Serie von Objekten. Die einzelnen Objekte sind mit Nummer und Typ in Tabelle 13 kurz erklärt. Abbildung 116 zeigt den Aufbau eines RSVP-Objektes im Body der RSVP-Nachricht.

Eine typische *path message* enthält zum Beispiel die Objekte *SESSION*, *RSVP_HOP*, *TIME_VALUES* und *SENDER_TEMPLATE*.

Das *SESSION* Objekt enthält neben Zieladresse und Zielport noch einige Flags. Bis jetzt ist erst eines davon definiert. Es dient dem *traffic policing*.

In das *RSVP_HOP*-Objekt trägt jede Station ihre Adresse ein und merkt sich die der vorherigen (um den richtigen Pfad für die Reservierungen zu wissen).

TIME_VALUES enthält die maximale und normale Zeitspanne, in der die *path messages* aktualisiert werden.

SENDER_TEMPLATE beschreibt den Datenstrom.

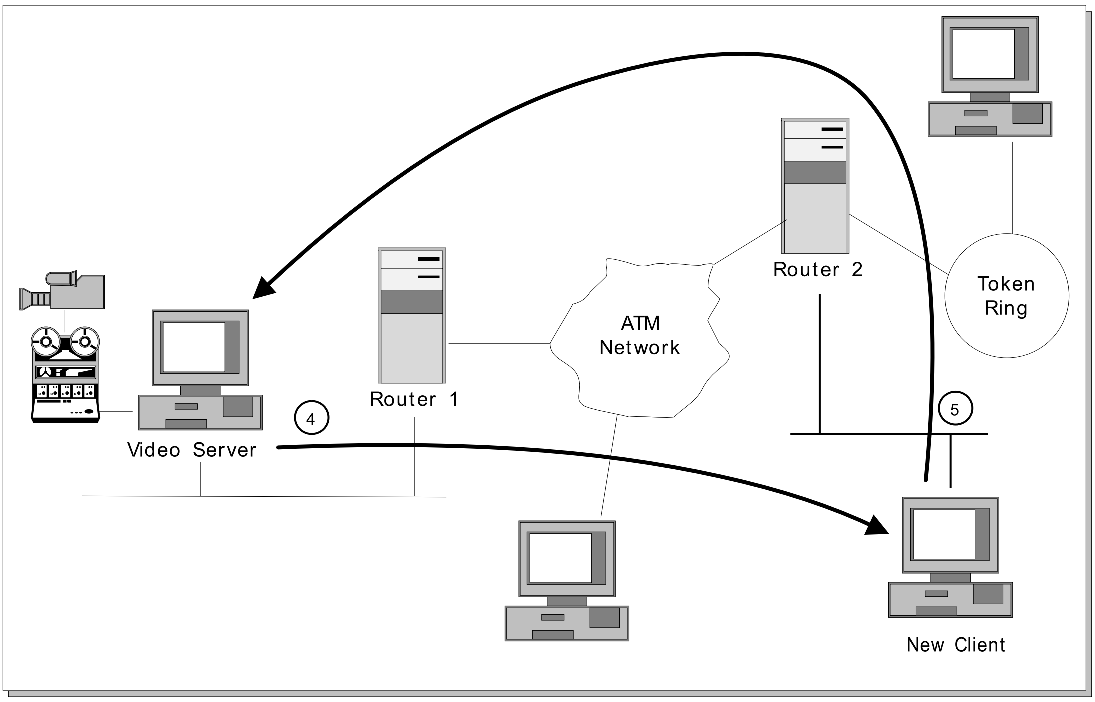


Abbildung 114: RSVP etabliert die Reservierung der Ressourcen für den neuen Klient

Ein typischer *reservation request* enthält zum Beispiel die Objekte *SESSION*, *RSVP_HOP*, *TIME_VALUES*, *STYLE*, *FLOW_SPEC* und *FILTER_SPEC*.

STYLE gibt die Reservierungsart an (siehe Tabelle 11).

Das *FLOW_SPEC*-Objekt ist das Herz der Reservierung. Hier definiert der Empfänger seine Reservierungsanforderungen. RSVP versteht dieses Objekt nur soweit, daß es die Verschmelzung von verschiedenen Reservierungsanforderungen realisieren muß. Normalerweise reicht RSVP den Wert dieses Objektes nur an den Teil des Routers weiter, der die Reservierungen tätigt.

Das letzte Objekt, *FILTER_SPEC* rundet die Beschreibung des Datenstromes ab. Es kann zum Beispiel die Herkunftsadresse und den *flow label* enthalten.

Sowohl bei den *path messages* als auch bei den *reservation requests* können Fehler auftreten. RSVP definiert für jede eine eigene Fehler-Nachricht. Tritt ein Fehler auf, so läuft die Fehler-Nachricht in die entgegengesetzte Richtung der Nachricht, die den Fehler verursacht hat. So können die vorherigen Systeme, die möglicherweise schon Ressourcen reserviert haben, diese wieder auflösen.

Eine *error message* enthält genug Objekte, um die Nachricht, die den Fehler erzeugt hat, zu identifizieren. Zusätzlich beinhaltet sie immer das *ERROR_SPEC*-Objekt, das neben dem Fehlercode und Wert auch die Adresse des Systems enthält, bei dem der Fehler aufgetreten ist. Tabelle 14 zeigt die möglichen Fehlercodes.

Mit der *reservation teardown message* kann ein Empfänger seine Reservierung von Ressourcen ausdrücklich wieder freigeben (wenn sie nicht in einer bestimmten Zeit aktualisiert werden, geben die Router die Reservierungen auch ohne eine *reservation teardown message* frei). Analog wird mit der *path teardown message* ein Pfad explizit beendet.

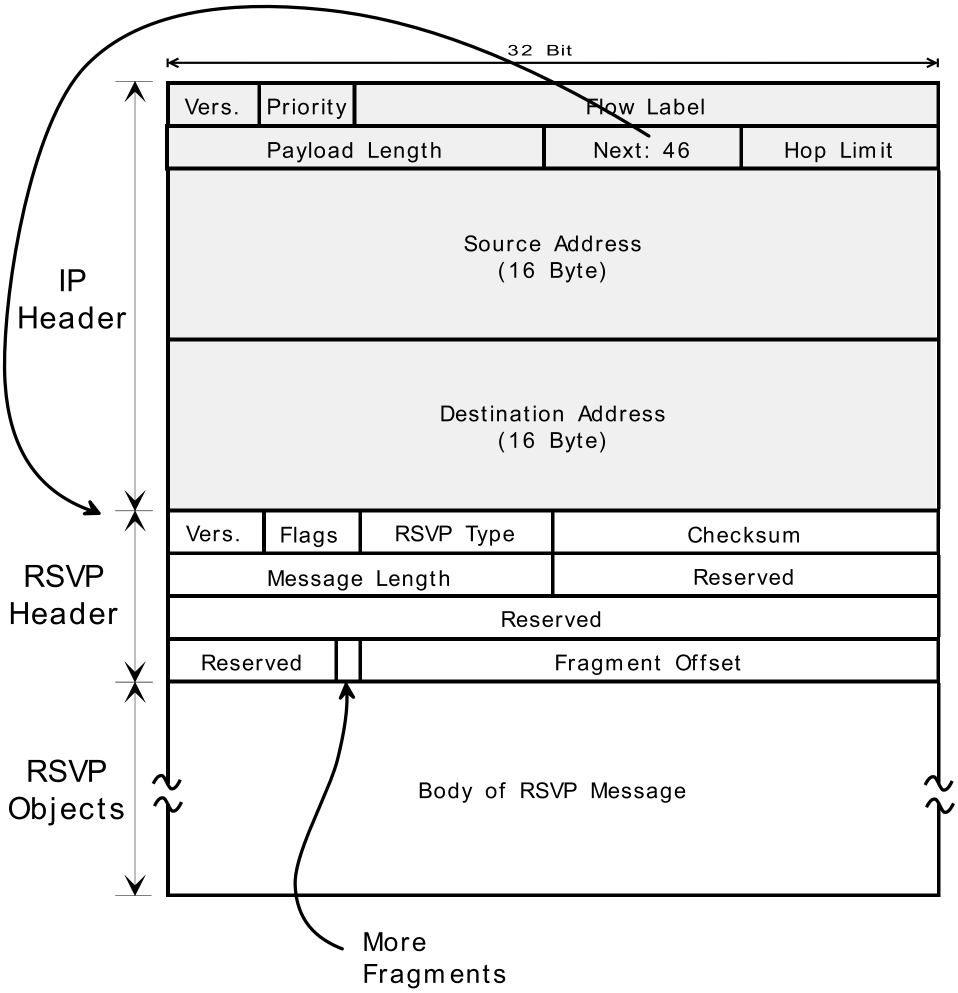


Abbildung 115: Der RSVP-Nachrichtenaufbau

Für weitere Details werden die *RFC - requests for comment* und *internet drafts* zu diesem Thema, sowie die im Anhang aufgeführte Literatur, empfohlen.

10.3.4 Mapping von RSVP auf ATM

Die Integration von RSVP in ein ATM-Netzwerk beinhaltet einige Probleme. Die Konzept der beiden Protokolle sind sehr unterschiedlich.

Im Gegensatz zu RSVP ist die Reservierung von Ressourcen bei ATM senderorientiert. Desweiteren benutzt RSVP *soft state*-Verbindungen, die auch bei bestehender Verbindung ihre Parameter ändern kann. ATM kennt dagegen nur *hard state*-Verbindungen bei denen vor Verbindungsaufbau bereits alle Parameter mitgeteilt werden müssen. Steht eine ATM-Verbindung einmal, kann kein Parameter mehr geändert werden. Zusätzlich unterstützt RSVP auch mehrere Sender in einer Multicast-Gruppe, ein Feature das bei ATM nicht vorhanden ist.

Um nun RSVP auf ATM abzubilden gibt es verschiedene Konzepte. Da RSVP Ressourcen nur in eine Richtung reserviert, wäre es vorstellbar, daß der Empfänger eine ATM-Verbindung zum

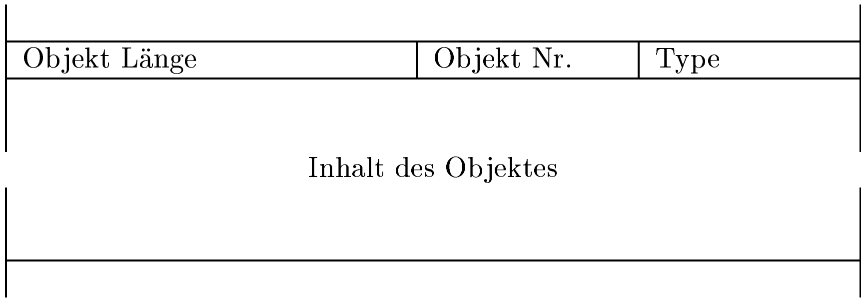


Abbildung 116: Format der RSVP-Objekte

Sender etabliert. Da ATM in beide Richtungen Ressourcen reservieren kann, scheint dies eine mögliche Emulation der Empfängerorientiertheit zu sein. Dies ist aber nur für Unicast Verkehr geeignet und steht im Widerspruch mit der ATM-Multicast-Umgebung.

Hier wird ein anderer, effizienterer Weg vorgestellt. Da die Entwicklung in diesem Bereich noch voranschreitet, können sich in Zukunft aber auch andere Realisierungen durchsetzen.

Wenn Router oder Host eine Reservierungsanforderung von einem flußabwärts gelegenen Host empfängt, baut er eine ATM-Verbindung zum nächsten Hop in Richtung Host auf. Dadurch können auch ATM Point-to-Multipoint Verbindungen aufgebaut werden. (Zumindest zu homogenen Empfängern, RSVP unterstützt dies auch bei inhomogenen.)

Auch die unterschiedlichen Spezifikationen der Dienstklassen bedürfen einer genauen Analyse. So wird die *quality of service* bei IP/RSVP durch die Verzögerung festgelegt, während ATM sie in Klassen von Bit-Raten einteilt (Constant Bit Rate: CBR, Available Bit Rate: ABR).

Übersetzung des Verkehrs und der QoS Parameter ist ein weiterer Dienst, der durch die Schicht-zu-Schicht Kommunikation während des Verbindungsaufbaus gelöst werden muß. Sobald die ATM *link layer* die Parameter in ihrem Format bekommen hat, muß sie die Verfügbarkeit überprüfen. Die Beschreibung des Datenstroms durch die Applikation muß es ermöglichen, daß die ATM-Schicht die Anforderung richtig umsetzen kann, um die Bandbreite optimal auszunutzen.

ATM unterstützt sowohl variable Bitraten mit Peaks als auch konstante Bitraten. Die Abbildung der Verkehrsparameter wird in Tabelle 15 dargestellt. Offensichtlich können alle IP-Ströme mit Ressourcen-Anforderungen über ATM-Kanäle mit konstanter Bitrate transportiert werden. Aber gerade bei burstartigen Verkehr werden die Ressourcen dann nicht optimal genutzt.

Zusammenfassend kann man sagen, daß eine vollständige Übersetzung der Service-Klassen nicht möglich ist. So kann zum Beispiel die CLR (*cell loss rate*) generell nicht von RSVP-Parametern hergeleitet werden. Die ATM-Schicht muß wegen dieser Schwierigkeiten immer mehr Ressourcen reservieren, als die Applikation eigentlich benötigt.

Nr.	Object	Type	Description
0	NULL		ignored by recipient
1	SESSION	1	IPv4 session (destination of flow)
		2	IPv6 session (destination of flow)
3	RSVP_HOP	1	IPv4 previous or next hop address
		2	IPv6 previous or next hop address
4	INTEGRITY		keyed MD5 authentification data
5	TIME_VALUES	1	frequency of path or reservation refreshes
6	ERROR_SPEC	1	error information from an IPv4 system
		2	error information from an IPv6 system
7	SCOPE	1	list of IPv4 hosts to which wildcard style reservation refresh apply
		2	list of IPv6 hosts to which wildcard style reservation refresh apply
8	STYLE	1	style of reservation
9	FLOW_SPEC	1	flow specification requiring controlled delay
		2	flow specification requiring predictiv quality of service
		3	flow specification requiring guaranteed quality of service
		254	flow specification containing several unmerged flows
10	FILTER_SPEC	1	IPv4-based filter to applyto flow
		2	IPv6-based filter using source port values
		3	IPv6-based filter using flow label values
11	SENDER_TEMPLATE	1	IPv4-based description of flow that sender is generating
		2	IPv6-based description of flow that sender is generating
12	SENDER_TSPEC	1	upper bound on traffic that sender will generate
13	ADSPEC		sender's advertised information for flow
14	POLICY_DATA	1	policy information for flow
		254	several unmerged policy data objects
20	TAG	1	collection of objects to be associated with a given name

Tabelle 13: Die RSVP Objekte

1	Administration failure, reservation could not be granted
2	Administrative rejection, reservation prohibited
3	No path information available for reservation
4	No sender for reservation
5	Ambiguous path
6	Ambiguous filter specification
7	Conflicting or unknown style
11	Missing required Object
12	Unknown object class
13	Unknown object type
14	Oject error
21	Traffic contrl error
22	RSVP system error

Tabelle 14: Die *RSVP Error Codes*

ATM		
QoS Class		
CBR	QoS	CLR
		CDV
		Max CDT
	Traffic	PCR
VBR-RT	QoS	CLR
		CDV
		Max CDT
	Traffic	PCR
		SCR
		MBS
VBR-NRT	QoS	CLR
		Mean CTD
	Traffic	PCR
		SCR
		MBS
ABR	QoS	CLR
	Traffic	PCR
		MCR
UBR	Traffic	PCR

RSVP	
Service Class	
Guaranteed Service	n/a
	n/a
	n/a
	Average Token Rate (RSpec)
	n/a
	n/a
	n/a
	n/a
	Average Token Rate (R of RSpec)
	Token Bucket Depth
Predictive Service	n/a
	Delay as part of RSpec
	n/a
	Average Token Rate (TSpec)
	Token Bucket Depth (TSpec)
Controlled Delay	n/a
	n/a
	Average Token Rate (TSpec)
Best Effort	No declarations are necessary

Tabelle 15: Mapping der ATM *service classes* auf RSVP

10.4 VLAN's

10.4.1 Einleitung

Netzwerke gewinnen zunehmend an Bedeutung. Deswegen steigt auch das Bedürfnis nach flexiblerem Aufbau und Konfiguration. Der Begriff VLAN steht für *Virtual Local Area Network*. Die Aufgabe eines VLAN's ist es, die logische Struktur von der physikalischen zu trennen. Man möchte Einheiten, die logisch zusammengehören, in eigene Teilnetze gliedern, um den netzbelastenden Broadcastverkehr zu minimieren. Auch Sicherheitskriterien spielen eine Rolle. So soll der Netzverkehr aus sicherheitsrelevanten Bereichen nicht von anderen mitgehört werden können. Gerade in größeren Firmen ist die Zugehörigkeit eines Arbeitsplatzes zu einem logischen Teilbereich nicht unbedingt durch die örtliche Struktur gegeben. Das bedeutet, daß zum Beispiel die Mitarbeiter eines Forschungsprojekts über den Campus verteilt sind. Auch ist es die Aufgabe eines VLAN's, den Konfigurationsaufwand bei Änderungen von Teilnetzen zu minimieren. Wenn sich neue Arbeitsgruppen bilden, sollten die Teilnehmer idealerweise per Drag und Drop von den Administratoren einem neuen VLAN zugeordnet werden können. Es gibt sehr unterschiedliche Ausbaustufen von VLAN's, die jeweils eigene Vor- und Nachteile haben. Im folgenden Teil werden die verschiedenen Typen von VLAN's vorgestellt.

10.4.2 Die VLAN-Typen

Der Begriff *VLAN* ist sehr weit auslegbar. *VLAN* ist nicht gleich *VLAN*. Auch gibt es keine Lösung, die alle Anforderungen erfüllt. Im folgenden Teil werden die einzelnen Ausbaustufen von *VLAN's* vorgestellt.

Layer-1-VLAN's: Die Zugehörigkeit eines Endgeräts oder Segments von Endgeräten zu einem VLAN wird durch den Port des Switches festgelegt, an den es angeschlossen ist.

Vorteile:

- Problemlose Integration in bestehende Netzwerke
- Kostengünstig
- Unterstützt alle Layer-3 Protokolle im Backbone

Nachteile:

- Große Sicherheitslücken
- Unflexibel bei häufigen Umzügen der Mitarbeiter
- Zuordnung eines Endgeräts zu mehreren VLANs nur durch mehrere Netzwerkkarten sinnvoll
- Hoher administrativer Aufwand bei Umzügen

Layer-2-VLAN's: Die Zugehörigkeit eines Endgeräts zu einem VLAN wird durch die MAC Adresse seiner Netzwerkkarte definiert.

Vorteile:

- Unterstützt alle Layer-3 Protokolle im Backbone

- Bei Umzug einer Mitarbeiters mit PC entsteht kein administrativer Aufwand
- Durch hardwarenahe Zuordnung können sichere Strukturen aufgebaut werden
- Konfigurationsaufwand nur bei Austausch oder Hinzukommen einer Netzwerkkarte

Nachteile:

- Zuordnung eines Endgeräts zu mehreren VLANs nur durch mehrere Netzwerkkarten sinnvoll
- Beim Austausch einer Netzwerkkarte muß umkonfiguriert werden

Layer-3-VLAN's: Die Zugehörigkeit eines Endgeräts zu einem VLAN wird durch die zugewiesene Netzwerkadresse definiert. (z.B. Die IP-Adresse)
Vorteile:

- Unabhängig von der eingesetzten Hardware
- Bei Umzug mit Endgerät sehr praktisch
- Sicher und flexibel
- Konfiguriert sich weitgehend automatisch
- Layer-3-Switches können Routingalgorithmen unterstützen
- Konfigurationsaufwand entsteht erst, wenn Endgeräte einem anderen VLAN zugewiesen werden

Nachteile:

- Nicht mehr protokolltransparent
- Netzwerkadressen lassen sich per Software ändern

10.4.3 LANE, ein VLAN über ATM

LANE (*LAN-Emulation*) ist eine von ATM-Forum standardisierte Möglichkeit, verbindungslose Protokolle wie IP in ATM-Netzen zu nutzen. Mit LANE können VLAN's sehr flexibel eingerichtet werden.

Das Prinzip der LAN-Emulation basiert auf der Abbildung der MAC-Adressen auf ATM-Adressen, wodurch dieses Verfahren für Schicht-3-Protokolle transparent wird. Für die Implementierung der LAN-Emulation im ATM-Netz müssen im wesentlichen vier Komponenten in das Netz integriert werden:

- LAN-Emulation-Client (LEC)
- LAN-Emulation-Server (LES)
- LAN-Emulation Configuration Server (LECS)

- Broadcast and Unknown Server (BUS)

Der Aufbau von LANE ist hierarchisch. An oberster Stelle steht der LECS. Er hat Kenntnis von allen LAN's, die ATM benutzen. Jedes Endgerät im ATM-Netz muß die ATM-Adresse des LECS kennen (*well known address*). Daher wird die Adresse beim Konfigurieren eines Endgeräts vom Administrator eingetragen. Die Aufgabe des LECS ist es, jedem Endgerät (LEC) mitzuteilen, zu welchem ELAN (*Emulated LAN*) es gehört. ELAN ist dabei der Teil des VLAN's, der sich auf ATM befindet. Ein gravierender Nachteil dieses LECS ist, daß er einen *single point of failure* darstellt. Für jedes ELAN muß es im ATM-Netzwerk einen LAN-Emulation-Server geben. Diese Server halten Tabellen vor, indem die ELAN-ATM-Adressen entsprechenden MAC-Adressen zugeordnet werden. Der LES ist also für die Adreßauflösung in seinem ELAN zuständig. Will ein Ethernet-Switch eine Verbindung zu einem anderen Ethernet-Switch über das ATM-Netzwerk aufbauen, schickt er eine *LAN-Emulation Address Resolution- (LE_ARP-)* Anfrage zum LES, mit der die Ziel-MAC-Adresse übermittelt wird. Der LES übersetzt die MAC-Adresse anhand seiner Tabelleneinträge in eine ATM-Adresse und schickt sie dem LEC auf dem Ethernet-Switch zurück. Dieser kann jetzt anhand der ATM-Adresse eine virtuelle Verbindung aufbauen.

Da das ATM-Netz verbindungsorientiert arbeitet, hat es grundsätzlich Schwierigkeiten, Broadcast- und Multicastdaten weiterzuleiten. Beides sind Adressierungsarten, die von LAN- und Schicht-3-Protokollen häufig genutzt werden. Um diese Adressierung im ATM-Netzwerk zur Verfügung zu stellen, gibt es einen Broadcast and Unknown Server (BUS). Dieser ist im ATM-Netz immer dann der Empfänger, wenn Daten an alle oder mehrere Adressaten im ELAN gesendet werden sollen. Seine zweite Aufgabe ist es, Daten weiterzuleiten, deren MAC-Adressen durch den LES aufgelöst werden konnten. Wie der LES muß der BUS für jedes ELAN einmal vorhanden sein.

Der LAN-Emulation-Client (LEC) ist der Vertreter (Proxy) im ATM-Netz für alle "hinter" ihm stehenden MAC-Adressen. Bei einem Ethernet-Switch mit ATM-Karte ist der LEC daher der Proxy für alle Ethernet-Anwender in einem VLAN, die direkt am Switch oder indirekt über einen Hub angeschlossen sind. In der Tabelle des zugehörigen LES sind somit für die eine ATM-Adresse mehrere MAC-Adressen eingetragen. Dies ist ein wesentlicher Unterschied zu IP-ARP-Tabellen, in denen für jede IP-Adresse nur eine MAC-Adresse eingetragen ist. Ein LEC kann immer nur zu einem ELAN und damit zu einem VLAN gehören. Ein Router, der über eine ATM-Verbindung den Datenverkehr zwischen verschiedenen VLAN's transportieren will (One trunk solution), muß natürlich allen VLAN's gleichzeitig angehören. Das setzt zwingend voraus, daß auf der ATM-Karte mehrere LEC's existieren. Das gleiche gilt für einen Ethernet-Switch, der seinen ATM-Port gleichzeitig für mehrere VLAN's nutzen möchte. Für jedes VLAN muß auf dem Switch ein LEC aktiviert und konfiguriert werden.

Das Zusammenwirken des LAN-Emulation-Clients mit den einzelnen Servern wird bei der Anmeldeprozedur deutlich. Möchte sich ein LEC im ATM-Netzwerk bekannt machen, schickt er zuerst eine Anfrage an den LAN-Emulation Configuration Server. Die Adresse dieses Servers hat der Administrator bei der Konfiguration im LEC eingetragen. Auf Anfrage des LEC antwortet der LECS mit der ATM-Adresse eines LES. Damit legt er fest, zu welchem ELAN und somit VLAN der LEC gehört. Die Zuweisung der LEC's in ein ELAN und somit auch VLAN kann über ein zentrales Managementsystem erfolgen.

Nachdem der LEC die ATM-Adresse eines LES kennt, meldet er sich am Netzwerk an. Dazu schickt der LEC den Userteil seiner ATM-Adresse zum LES. Der LES ergänzt den Userteil durch den Netzwerkteil, trägt MAC und ATM-Adresse in eine Tabelle ein und schickt die vollständige ATM-Adresse an den LEC zurück. Damit ist der Client im ATM-Netzwerk angemeldet.

Nach der Anmeldung ermittelt der Client die Broadcast-ATM-Adresse. Dazu schickt er einen LE_ARP Request für die Broadcast-MAC-Adresse "FFFFFFFFFFFF" an den LES. Dieser beantwortet die Anfrage mit der ATM-Adresse des Broadcast and Unknown Server.

Der Client baut anschließend eine Virtual Channel Connection zum BUS auf, die er für alle Broadcast- und Multicastdaten nutzt. Diese Verbindung wird *Multicast Send VCC* genannt. Der BUS baut in der Gegenrichtung auch eine VCC auf, die *Multicast Forward VCC* heißt.

Erhält der LEC aus den oberen Schichten Broadcastdaten, schickt er diese über den *Multicast Send VCC* zum BUS. Da der BUS zu allen angemeldeten Clients eine *Multicast Forward VCC* unterhält, kopiert er die Daten und schickt sie über diese Leitung zu allen LEC's.

Auch für MAC-Adressen, die der LES nicht in ATM-Adressen wandeln konnte, wird dieses Verfahren genutzt. Um eine unbekannte ATM-Adresse zu ermitteln, schickt der LEC die MAC-Adresse zum BUS. Dort wird sie dann an alle LEC's des VLAN weitergeleitet. Ist einem LEC die ATM-Adresse bekannt, schickt er dem BUS eine entsprechende Antwort inklusive seiner ATM-Adresse. Der BUS leitet die ATM-Adresse an den LEC, der die Anfrage initiiert hat, weiter. Dieser kann jetzt auf Grund der ATM-Adresse eine virtuelle Verbindung im ATM-Netz aufbauen.

Lesern, die sich mit dem Thema VLAN's intensiv auseinandersetzen wollen, wird das Buch [19] empfohlen.

Abbildungsverzeichnis

Tabellenverzeichnis

Literatur

[1] *sox - Universal sound converter, player, and recorder.* ftp://sunsite.unc.edu/pub/-Linux/apps/sound/convert/.

[2] ALLES, A. *ATM Internetworking.* Cisco Systems, Inc., Mai 1995.

[3] ATM-FORUM. *Interim Inter-switch Signaling Protocol (IISP) Specification v1.0.* ftp://ftp.atmforum.com/pub/approved-specs/af-pnni-0026.000.pdf, December 1994.

[4] ATM-FORUM. *LAN Emulation Over ATM Version 1.0*, January 1995.

[5] ATM-FORUM. *ATM User - Network Interface (UNI), Signalling Specification*, vol. 4.0. The ATM Forum, Juli 1996.

[6] ATM-FORUM. *Integrated Local Management Interface (ILMI) Specification Version 4.0.* ftp://ftp.atmforum.com/pub/approved-specs/af-ilmi-0065.000.pdf, September 1996.

[7] ATM-FORUM. *Private Network-Network Interface (P-NNI) Specification Version 1.0.* ftp://ftp.atmforum.com/pub/approved-specs/af-pnni-0055.000.pdf, March 1996.

[8] ATM-FORUM. *LAN Emulation Over ATM Version 2.0 - LUNI Specification*, July 1997.

[9] ATM-FORUM. *Multi-Protocol Over ATM Version 1.0*, July 1997.

[10] BONK, D. T. *RTB-Projekt: Migration einer ATM-nativen Videokonferenz-Anwendung in das B-WiN.* Leibnitz Rechenzentrum, München, 1997.

[11] CHEN, T. M., AND LIU, S. S. *ATM Switching Systems.* Artech House, 1995.

[12] CISCO SYSTEMS, I. *ATM Internetworking*, May 1995.

[13] COOMBS, C. F. *Communication Networks Test and Meassurement.* Mc Graw-Hill, 1 1998.

[14] DEMBOWSKI, K. *PC-gesteuerte Meßtechnik.* Markt-und-Technik-Verlag Haar bei München, 1991.

[15] DETKEN, K.-O. Dienste mit Qualität. *Gateway* (May 1997), 108 – 110.

[16] DETKEN, K.-O. Pakete huckepack. *Gateway* (March 1997), 112 – 117.

[17] DETKEN, K.-O. Virtueller Router. *Gateway* (April 1997), 112 – 118.

[18] DFN VEREIN. *ATM-Tutorium*, 1997.

[19] DIETRICH, J., AND VON THIENEN, U. *VLANs - Migration zu modernen Netzwerken.* International Thomson Publishing, 1997.

[20] GADEGAST, F. *TCP/IP-basierte Dienste zur Speicherung von Multimedia-Daten.* http://www.powerweb.de/phade/diplom/, Dezenber 1997.

[21] GN NETTEST INC. *ATM-Chart*, 1997.

[22] IETF. *Classical IP and ARP over ATM*, October 1997.

[23] ITU-T. *B-ISDN ATM Adaption Layer (AAL) Specification ITU-T Recommendation I.363*, 1993.

[24] KESIDIS, G. *ATM Network Performance.* Kluwer Academic Publishers, 1996.

[25] KIEFER, R. *Meßtechnik in digitalen Netzen: Troubleshooting für PDH, SDH, ISDN und ATM.* Hüthig, 6 1997.

[26] KÜSTERS, H. *Bilddatenkomprimierung mit JPEG und MPEG.* Franzis', 1995.

[27] KYAS, O. *ATM-Netzwerke: Aufbau-Funktion-Performance.* Datacom-Verlag, Bergheim, 1995.

[28] KYAS, O. *ATM-Netzwerke: Aufbau-Funktion-Performance.* Datacom-Verlag, Bergheim, 1996.

[29] MACDYSAN, DAVID E. SPOHN, D. L. *ATM.* McGraw-Hill, 1994.

[30] MIL3. *OPNET Modeler, User Manual 1-12.*

[31] NEUKAM, T. Zwei in einem. *Gateway* (November 1997), 122 – 126.

[32] RATHGEBER, E. P., AND WALLMEIER, E. *ATM - Infrastruktur für die Hochleistungskommunikation.* Springer, 12 1997.

[33] RIGGERT, W. Kontrollierbare Verbindungen. *Gateway*, 3 (März 1997), 118–123.

[34] RITTER, M., AND TRAN-GIA, P. Mechanismen zur Steuerung und Verwaltung von ATM-Netzen, Teil1: Grundlegende Prinzipien. *Informatik Spektrum 4* (1997).

[35] RITTER, M., AND TRAN-GIA, P. Mechanismen zur Steuerung und Verwaltung von ATM-Netzen, Teil1: Modellierung und Leistungsbewertung. *Informatik Spektrum 5* (1997).

[36] ROPPEL, C. *In-Betrieb-Messverfahren für Zellenlaufzeiten und Zellenlaufzeitschwankungen in ATM-Netzen*, vol. Fortschrittsbericht VDI Reihe 10 Nr.475. VDI-Verlag, 6 1997.

[37] SAITO, H. *Teletraffic Technologies in ATM Networks.* Artech House, 1993.

[38] SCHILL, A. *ATM-Netze in der Praxis, Einsatzszenarien, Produktkategorien, aktuelle Standards.* Addison-Wesley, 12 1997.

[39] SCHILL, A., HESS, R., KÜMMEL, S., HEGE, D., AND LIEB, H. *ATM-Netze in der Praxis.* Addison–Wesley, 1997.

[40] SCHILL, A., KÜHN, S., AND BREITER, F. Internetworking over ATM: Experiences with IP/IPnG and RSVP. Tech. rep., Technische Universität Dresden, Institut für Informatik, 1996.

[41] SCHOOL OF COMPUTING SCIENCE, SIMON FRASER UNIVERSITY. *Hypermedia/Multimedia.* `http://fas.sfu.ca/cs/undergrad/CourseMaterials/CMPT365/material/notes/-Chap1/Chap0.html`, 1996.

[42] SIEGMUND, G. *ATM - Die Technik*, 3 ed. Huethig, Heidelberg, 1997.

[43] SIEGMUND, G. *ATM - Die Technik: Grundlagen, Netze, Schnittstellen, Protokolle*, vol. 3, neubearb. und erw. Aufl. Hüthing Verlag, Heidelberg, Juni 1997.

[44] Tanenbaum, A. S. *Computer Networks.* Prentice-hall International, Inc., 1996. ISBN: 0-13-394248-1
`http://www.cs.vu.nl/~ast/ph/cn3.html`.

[45] Tanenbaum, A. S. *Computernetzwerke.* Prentice Hall, 1997.

[46] Tanenbaum, A. S. *Computernetzwerke.* Prentice Hall Verlag, 1997, ch. 5.5.10 IP Version 6 (IPnG).

[47] Watson, J. *Mastering Electronics*, vol. 3. MACMILLAN EDUCATION LTD, 1991.

[48] Widjaja, I., and Elwalid, A. I. Performance Issues in VC-Merge Capable Switches for IP Over ATM Networks. Tech. rep., Bell Laboratories, 1997.

[49] Wolf, A. *Meßtechnik für das B-ISDN.* VDE-Verlag, 6 1992.