

# Studienarbeit

erstellt von:

[Thomas Martin Knoll](#)

geb. am: 10.01.1973

**Studiengang:** Elektrotechnik  
**Studienrichtung:** Informationstechnik

**Thema:** „[Management von ATM-Netzen](#)“

**Betreuer der Aufgabe:** [Dipl.-Ing. Pätz](#)

**Tag der Ausgabe:** 02.05.1997

**Abgabetermin:** 31.07.1997

**Tag der Abgabe:** 15.08.1997

## Bibliographische Beschreibung

Management von ATM-Netzen

[Knoll, Thomas Martin](#) - 1997 - 54 Seiten, 24 Abbildungen, 4 Tabellen, 103 Literaturquellen

HTML-Referenz: <http://www.infotech.tu-chemnitz.de/~knoll/studienarbeit.html>

Chemnitz, Technische Universität Chemnitz,  
Fakultät für Elektrotechnik und Informationstechnik  
Lehrstuhl für Daten- und Kommunikationstechnik

Studienarbeit

## Kurzreferat

Inhalt der Studienarbeit ist die Untersuchung der vorhandenen Netzwerkmanagementprotokolle und -programme mit Blick auf deren Einsatz in ATM-Netzen.

Es erfolgt zuerst eine kurze Einführung in die Problematik mit anschließender Beschreibung des Standes der Standardisierung. Schwerpunktmäßig wird die Realisierung des Managements von ATM-Netzen betrachtet. Es wird ein Überblick über vorhandene Lösungsansätze gegeben. Sowohl das Anpassen der ATM-Netzwerkkomponenten an das bereits weitverbreitete Internet-Management als auch die Umsetzung der Spezifikationen seitens des ATM-Forums sind dabei zu berücksichtigen.

Neben der ausführlichen Betrachtung der Managementprotokolle wird auf den Bereich der Netzwerkmanagementapplikationen eingegangen. Dabei werden die kommerzielle Netzwerkmanagementplattform HP OpenView und das frei verfügbare Softwarepaket Scotty/Tkined vorgestellt.

Mit Hilfe der zur Studienarbeit gehörenden HTML-Referenz soll der Zugriff auf netzwerkmanagementrelevante Informationen im Internet erleichtert werden.

# Inhaltsverzeichnis

<b>1 Aufgabenstellung</b>	<b>4</b>
<b>2 Einführung in das Netzwerkmanagement</b>	<b>4</b>
<b>3 Grundlagen des Netzwerkmanagements</b>	<b>5</b>
<b>3.1 Allgemeines</b>	<b>5</b>
<b>3.2 Management bei ISO/OSI</b>	<b>7</b>
<b>3.3 Management im Internet mittels SNMP</b>	<b>9</b>
3.3.1 Die Management Information Base	9
3.3.2 SNMP Version 1	11
3.3.3 SNMP Version 2	14
<b>3.4 Management in ATM-Netzen</b>	<b>18</b>
3.4.1 Das allgemeine Managementmodell	19
3.4.2 Integrated Local Management Interface - ILMI	20
3.4.3 Die Managementschnittstelle M3	27
3.4.4 Die Managementschnittstelle M4	31
<b>3.5 Überlegungen zur Implementation eines Management-Agenten</b>	<b>36</b>
<b>3.6 Kurzreferenz zu ASN.1</b>	<b>39</b>
<b>4 Netzwerkmanagementapplikationen</b>	<b>41</b>
<b>5 Zusammenfassung</b>	<b>42</b>
<b>6 Abkürzungsverzeichnis</b>	<b>44</b>
<b>7 Literaturverzeichnis</b>	<b>46</b>
<b>8 Selbstständigkeitserklärung</b>	<b>54</b>

## 1 Aufgabenstellung

Um große Datennetze überwachen und managen zu können, werden geeignete Werkzeuge und Protokolle benötigt. Eine große Bedeutung hat das Netzwerkmanagementprotokoll SNMP, das in TCP/IP-basierten Netzen Verwendung findet.

Um ATM-Netze in gleicher Weise überwachen und steuern zu können, wurden neue und an die speziellen Erfordernisse eines ATM-Netzes angepaßte Werkzeuge und Protokolle entwickelt.

Die Vielfalt der in ATM-Netzen einzusetzenden Netzwerkmanagementprotokolle und Programme sind in dieser Studienarbeit zu sichten, zu klassifizieren und vor dem Hintergrund einer möglichen Implementierung zu bewerten. Dabei ergeben sich folgende Schwerpunkte:

- Aufwandabschätzung für die Implementierung einer Netzmanagementunterstützung für eine ATM-NIC
- Umfang und Funktionalität der vom ATM-Forum definierten Managementschnittstellen (M1...M5)
- Erweiterungen der SNMP-Welt zur Unterstützung von ATM
- Unterstützung von Standardvisualisierungstools wie HP-OpenView oder Scotty

## 2 Einführung in das Netzwerkmanagement

Der Datenaustausch zwischen elektronischen Geräten wird zunehmend umfangreicher, komplexer und schneller. Durch Multimediaanwendungen mit Video- und Telefonübertragungen, verstärktem Einsatz verteilter Anwendungen, zunehmend elektronischem Geldtransfer und vielem mehr ergibt sich die Notwendigkeit der Planung, Überwachung und Steuerung des Datenstromes, um eine wirtschaftliche, qualitätsgerechte und überprüfbare Dienstbereitstellung zu erreichen.

Bereits in der Zeit, da Netzwerke ausschließlich Komponenten eines Herstellers beinhalteten, wurde die Wichtigkeit des Netzwerkmanagements für die Effizienz und die Verfügbarkeit eines Rechnernetzes erkannt. Für solche homogenen Netze boten die Hersteller proprietäre Managementsysteme an. Beim Übergang zu den heterogenen Netzen bildeten sich damit entsprechende Managementinseln, wie sie auch heute teilweise noch bestehen. Ziel der Standardisierungsbestrebungen war es nun, eine Vereinheitlichung des Managements zu erreichen. Dabei entstand zum einen das Netzwerkmanagement-Rahmenwerk, das von der Internationalen Standardisierungsorganisation ISO für die Kommunikation offener Systeme (OSI) genormt wurde und zum anderen die Internet-Netzwerkmanagement-Architektur mit dem Managementprotokoll „Simple Network Management Protocol (SNMP)“.

Im Bereich der ATM-Netze wurde ein eigenes Managementmodell bestehend aus fünf definierten Schnittstellen (M1..M5) entworfen. Diese benutzen SNMP als Kommunikationsprotokoll. Bis zur endgültigen Standardisierung dieses Modells entwickelte man eine Übergangslösung genannt „Interim Local Management Interface - ILMI“. Im September 1996 wurde jedoch dieser Entwurf als zeitlich unbeschränkt bestehender Standard unter dem Namen „Integrated Local Management Interface - ILMI“ spezifiziert.

## 3 Grundlagen des Netzwerkmanagements

### 3.1 Allgemeines

Definition: [Seits 94]

Netzwerkmanagement hat zum Ziel, die Ressourcen eines Kommunikations- oder Rechnernetzes zu planen, zu überwachen und zu koordinieren, welche zur Kommunikation innerhalb dieses Netzes benötigt werden.

Klassifizierung der Netzwerkmanagementfunktionsbereiche nach ISO 7498:

#### 1. Fehlermanagement (Fault Management)

- Eingrenzung und Behebung abnormaler Schichten-, System- bzw. Netzfunktionen
- Tests zur Funktionsprüfung, Fehlerlokalisierung
- Fehlermeldungen austauschen, Fehlerstatistik führen („Logfiles“ anlegen/auswerten)

#### 2. Konfigurationsmanagement (Configuration Management)

- Identifikation (Namensvergabe) für die zu betrachteten Einheiten („Managed Objects“)
- Objekte starten (einfügen) und außer Betrieb nehmen
- Ermittlung des aktuellen Zustandes
- Modifikation des aktuellen Zustandes (Attribute setzen ...)

#### 3. Leistungsmanagement (Performance Management)

- Gewinnung statistischer Analysen über Leistungsparameter
- Leistungsoptimierung (z.B. IP-Paketgröße bei NFS)

#### 4. Abrechnungsmanagement (Accounting Management)

- Ermittlung des Ressourcenverbrauchs (z.B. Überwachung des Verkehrsvertrages mit entsprechenden mengen- oder zeitbezogenen Gebühren)
- Begrenzung des Ressourcenverbrauchs (z.B. Datenübertragungskontingente)

#### 5. Sicherheitsmanagement (Security Management)

- Autorisierung (auch für Management selbst)
- Zugriffskontrolle / -schutz
- Datenverschlüsselung
- Authentifizierung von Benutzern
- Sicherheits-Log

#### 6. Softwaremanagement

- Softwareverteilung
- Versionsmanagement

Man unterscheidet bei der Managementkommunikation die folgenden zwei Ansätze:

- „In-Band-Management“ - direkte Nutzung des Netzes für die Kommunikation.
- „Out-of-Band-Management“ - Nutzung spezieller Managementschnittstellen (z.B. serielle Terminalzugänge mit Point-to-Point-Protocol).

In den Netzwerkmanagementstandards sind zwei Bereiche betrachtet. Zum einen erfolgt eine Modellbildung für die Repräsentation der Netzkomponenten und zum anderen die Darlegung der Managementprotokolle als Dienstleistungselemente. Innerhalb der Modelle geschieht die Umsetzung von der gerätespezifischen Gewinnung und Darstellung der Managementinformation hin zu einer einheitlichen Schnittstelle zur Managementanwendung. Diese management-spezifische Darstellung einer Netzwerkkomponente bezeichnet man als „Managed Object“. Ein Managed Object besitzt folgende vier Eigenschaften:

1. Attribute

In einem oder mehreren Attributen wird die Managementinformation abgelegt, die mit den Vermerken privat / öffentlich bzw. lesbar / schreibbar näher bestimmt wird.

2. Operationen

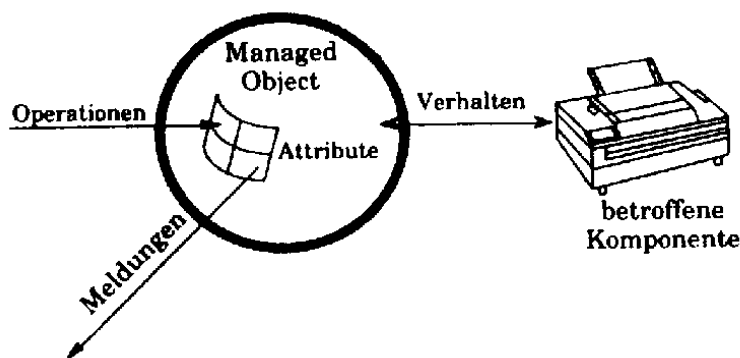
Die an der Schnittstelle gebotenen Operationen ermöglichen den Zugriff auf die Informationen der nachgebildeten Funktionalität. Es ist z.B. das Schreiben und Lesen der Attribute, sowie das Erzeugen und Löschen des gesamten Objektes möglich.

3. Meldungen

Managed Objects können Alarmmeldungen zugeteilt werden, die beim Eintreten eines Ausnahmezustandes zur Benachrichtigung der Managementanwendung dienen.

4. Verhalten

Das Verhalten ergibt sich aus der bidirektionalen Kopplung des Managed Objects mit der physischen Komponente. Änderungen innerhalb der Komponente bewirken Attributänderungen und umgekehrt.

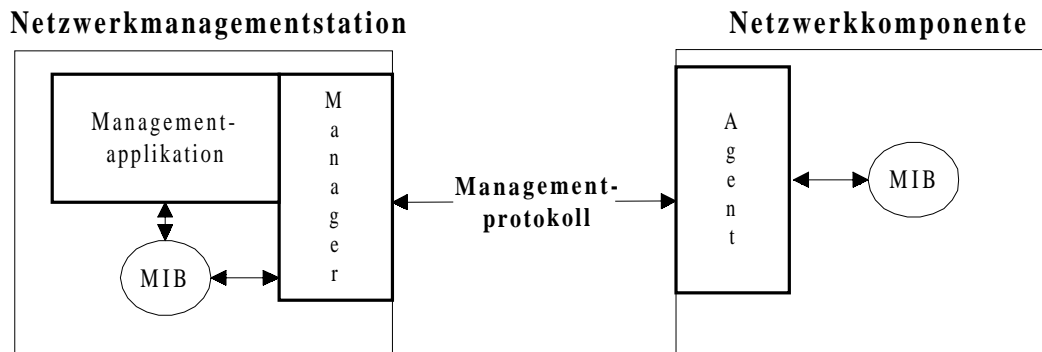


**Abbildung 1** Schematische Darstellung eines Managed Objects aus [Seits 94]

Die Managed Objects sind in der sogenannten „Management Information Base (MIB)“ angegeben, die somit die Gesamtheit der in einem Netzwerk vorhandenen Managementinformation darstellt.

Allen Managementmodellen ist gemeinsam, daß innerhalb der Netzkomponenten ein sogenannter „Agent“ die örtliche Verwaltung des oder der Managed Objects vornimmt und über ein offenes Managementprotokoll mit einem „Manager“ kommuniziert. Abbildung 2 zeigt

dazu eine Übersichtsgrafik. Das Netzwerkmanagement erstreckt sich somit beginnend beim Menschen über die Managementapplikation zum Manager, der mittels des Protokolls den Informationsaustausch mit den entfernten Agenten vornimmt. Zunehmend werden Managementapplikationen auch verteilt ausgeführt, was die Kommunikation der Manager untereinander voraussetzt.

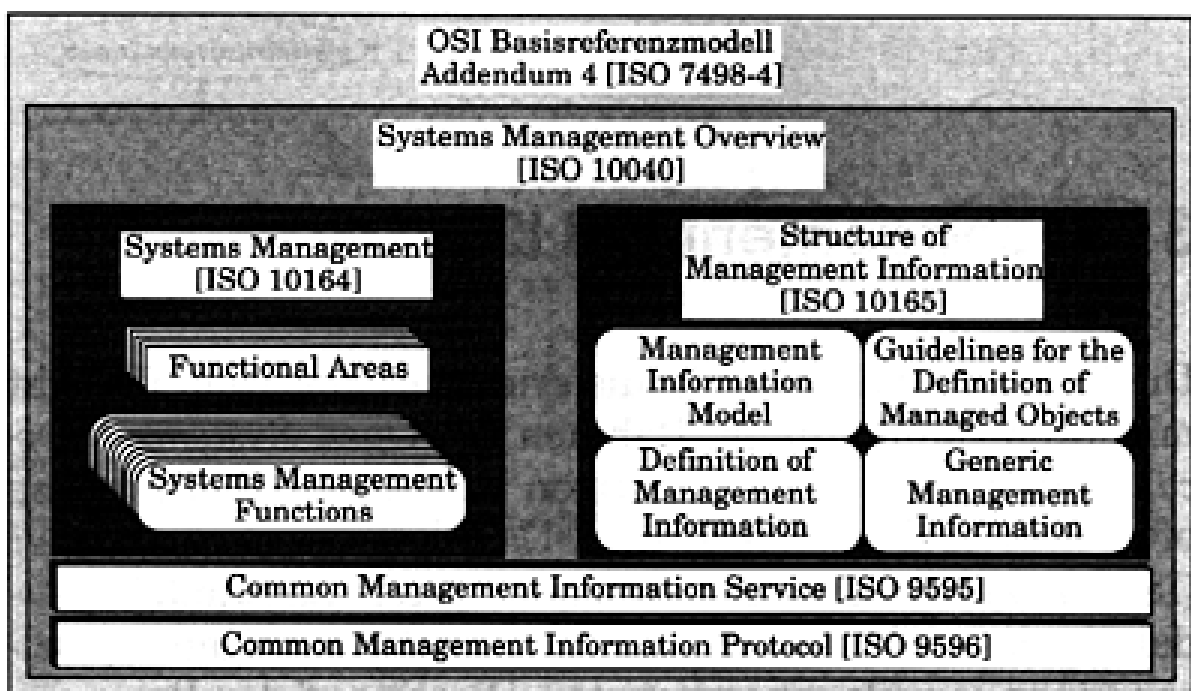


**Abbildung 2** Allgemeines Modell des Netzwerkmanagements

Das Management ist im OSI-Referenz-Modell schichtübergreifend einzuordnen. Dabei hat jede Schicht eine Art „Seitenzugang“, an dem schichtspezifische Information dem Management bereitgestellt, bzw. eine Steuerung des Schichtverhaltens ermöglicht wird.

### 3.2 Management bei ISO/OSI

Zu dem Open System Interconnection (OSI) - Referenzmodell [ISO 84] wurde 1989 ein Management Framework [ISO 89c] als Zusatz definiert. Abbildung 3 zeigt eine Übersicht zu dieser Zusatzdefinition.



**Abbildung 3** Netzwerkmanagement-Normenwerk der ISO/OSI [Seitz 94]

Managed Objects werden in der OSI-Welt als Instanzen von hierarchisch organisierten Managed Object Classes definiert. Unter Einhaltung bestimmter Vorschriften („Guidelines for the Definition of Managed Objects (GDMO)“ [ISO 91c]) werden diese Klassen aus allgemeinen Schablonen, sogenannten „Templates“, abgeleitet.

Als Managementprotokoll ist in [ISO 91b] das „Common Management Information Protocol (CMIP)“ genormt, welches den in [ISO 91a] angegebenen Dienst „Common Management Information Service“ erbringt. Das CMIP stützt sich dabei auf zwei Dienstelemente:

- Association Control Service Element (ACSE) [ISO 88a, ISO 88b]  
Dient zum Aufbau einer „Assoziation“, d.h. Verbindung auf unteren Schichten mit direktem Zugriff seitens der Applikation, zwischen zwei Managementinstanzen.
- Remote Operations Service Element  
Erbringung des Dienstes durch entferntes Ausführen der entsprechenden Objektoperationen.

Die CMIP-Dateneinheiten werden nach der „Abstract Syntax Notation One (ASN.1)“ kodiert. Die durch CMIP erbrachten Dienste „Common Management Information Services (CMIS)“ werden in drei Serviceklassen eingeteilt:

- Management Association  
Dienste zur Überwachung der Kommunikation zwischen CMIS-Systemen, sowie zum Auf- und Abbau von Verbindungen.
- Management Notification  
Dienste zum selbständigen Melden von Ausnahmesituationen seitens des Agenten.
- Management Operation  
Dienste zum Übermitteln von Informationen zwischen CMIS-Manager und CMIS-Agenten.

Im Hinblick auf eine mögliche Implementierung ist zu sagen, daß dieses Framework ein für alle Netzkomponenten anwendbares Management ermöglicht. In den Standards zu CMIP bzw. CMIS sind keine Festlegungen für den Umgang mit den Managementinformationen getroffen, so daß seitens der Managementsysteme eine freie Interpretierung erfolgt. Generell kann man sagen, daß dieses OSI-Werk die umfänglichste Standardisierung im Bereich des Netzwerkmanagements darstellt. Dies birgt jedoch folgende Nachteile in sich: [Hein94]

- CMIS/CMIP benötigen ungeheuer viel Speicher und CPU-Leistung.
- Es wird jede Menge Protokoll-Overhead generiert.
- Für einen Programmierer sind die Spezifikationen sehr schwer umzusetzen und deshalb nur mühsam zu implementieren.

Sowohl in der Fachliteratur als auch in namhaften Netzwerkmanagementapplikationen zeichnet sich jedoch der Trend ab, daß das OSI-Management zusätzlich zu anderen Standards angeboten wird. Es ist aber eher unwahrscheinlich, daß das OSI-Management zukünftig als alleiniges Modell den Markt beherrscht.

Der Versuch, mittels der Spezifikation „Common Management Information Services and Protocol over TCP/IP (CMOT)“ [RFC 1189] eine allmähliche Akzeptanz des OSI-Standards zu erreichen, ist nicht zuletzt daran gescheitert, daß viele Hersteller keine größeren Aufwendungen in eine Übergangslösung investieren wollten. Dieser Ansatz sah vor, daß die Struktur und die Dienste des OSI-Managements auf dem Internet-Transportprotokoll aufsetzen sollten.

### 3.3 Management im Internet mittels SNMP

Die erste Managementarchitektur im Internet war das 1987 entwickelte „Simple Gateway Monitoring Protocol (SGMP)“. 1988 wurden die in Tabelle 1 aufgeführten drei Grundsäulen des allgemeinen Internet-Managements veröffentlicht.

	<b>Titel</b>	<b>Anliegen</b>
RFC 1065	Structure and Identification of Management Information for TCP/IP-based Internets (SMI)	Protokollunabhängige Regeln zur Definition von Objekten
RFC 1066	Management Information Base (MIB) for TCP/IP-based Internets	Definition des Basissatzes von Managed Objects
RFC 1067	Simple Network Management Protocol (SNMP)	Definition des Managementprotokolls zum Austausch bzw. Ändern von Managementinformationen

**Tabelle 1** Grundstandards des allgemeinen Internet-Managements

Im Laufe der Entwicklung des SGMP wurden einige OSI-Merkmale in das Protokoll aufgenommen. So gibt es auch hier die „Management Information Base“ und die „Structure of Management Information“.

Im Mai 1990 wurde der von der SGMP-Arbeitsgruppe wesentlich vollständigte und leicht zu implementierende SNMP-Standard schließlich zum Internet-Standard erklärt. Diese endgültige Fassung von J.D. Case, M. Fedor, M.L. Schoffstall und C. Davin ist im [RFC 1157] nachzulesen.

Ebenfalls wurde 1990 eine erste MIB unter dem Namen „MIB I“ [RFC 1066] im Internet standardisiert. Im Frühjahr 1991 erfolgte jedoch die vollständige Übernahme und Erweiterung der MIB I in der sogenannten „MIB II“ [RFC 1213]. Neben diesem Standard existieren eine Vielzahl firmenspezifischer MIBs, die in eine hierarchische Organisation eingebettet sind. Zu dieser Struktur folgt ein kurzer Überblick.

#### 3.3.1 Die Management Information Base

In einer MIB werden „Schablonen“ für Managed Objects definiert, die bei den jeweiligen Agenten dann instanziiert werden. Es gibt eine globale MIB-Baumstruktur in der sich jede MIB eingliedert. Jede dieser MIBs ist wiederum als Baumstruktur definiert und stellt somit jeweils einen Subbaum in der globalen Einordnung dar. Die Endpunkte des Baumes sind dabei die einzelnen Managementobjekte. Ein Agent einer Netzwerkkomponente muß nun die der Komponentenfunktionalität entsprechenden Teilbäume des MIB-Baumes realisieren.

Die Informationen, die in den „Managed Objects“ enthalten sind, werden nach der „Structure of Management Information (SMI)“ [RFC 1155] beschrieben. Diese Struktur drückt sich in einem „Object-Type-Macro“ [RFC 1212] aus, das mittels der „Abstract Syntax Notation One (ASN.1)“ spezifiziert ist. Jede Objektdefinition erfolgt entsprechend dieser Makrodefinition, welche in Abbildung 4 dargestellt ist.

```

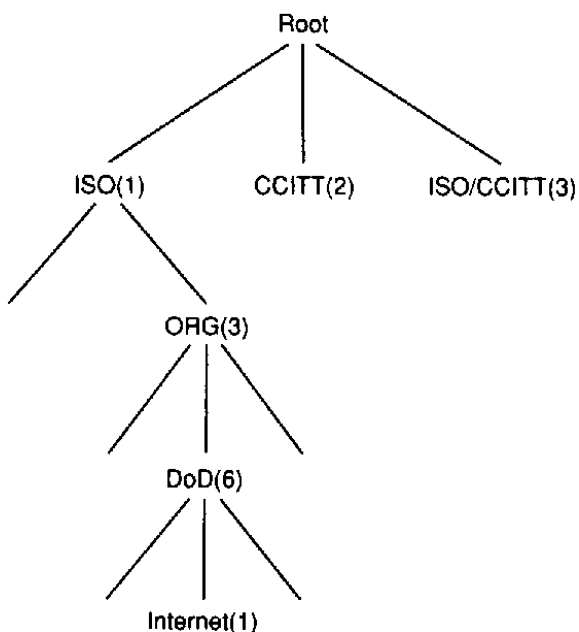
OBJECT-TYPE  MACRO ::=
BEGIN
    TYPE NOTATION ::= -- must conform to
                        -- RFC1155's ObjectSyntax
        "SYNTAX" type(ObjectSyntax)
        "ACCESS" Access
        "STATUS" Status
    VALUE NOTATION ::=
        value (VALUE ObjectName)

    Access ::= "read-only" | "read-write" | "write-only" |
               "not-accessible"
    Status ::= "mandatory" | "optional" | "obsolete"
END

```

**Abbildung 4** ASN.1-Makrodefinition für Managementobjekte

Die eindeutige Bezeichnung und zugleich die Einordnung der Objekte im Baum erfolgt über einen „Object Identifier“. Dabei gibt es sowohl einen symbolischen Bezeichner als auch die sogenannte „Doted Notation“ für ein Objekt. Jeder Hierarchyeintrag besitzt eine Nummer (in Klammern angegeben). Die Punktschreibweise des Objektnamens ergibt sich nun als eine durch Punkte getrennte Auflistung von diesen Nummern. Beginnend bei „Root (1)“ werden die abwärts durchlaufenen Hierarchieverzweigungen (deren Nummern) bis hin zum eigentlichen Objekt aufgeschrieben.



**Abbildung 5** Auszug aus dem globalen MIB-Baum

Abbildung 5 ist zu entnehmen, daß alle Objekte, die unterhalb des Internetzweiges angesiedelt sind, mit 1.1.3.6.1. beginnend bezeichnet werden. Bei manchen Objekttypen gibt es innerhalb eines Systems stets nur eine Instanz (z.B. das Objekt „sysDescr“, welches Auskunft über das Betriebssystem, den Namen des Gerätes und die Hardware liefert). Sie erhalten als Suffix der Punktnotation eine Null. Andere Objekttypen können mehrfach instanziiert sein (z.B. Tabelleneinträge). In solchen Fällen tritt eine Objektverschachtelung mittels der ASN.1-Typen SEQUENZE und SEQUENZE OF ein. Ein Objekt in der Tabelle wird also über seinen Namen und dem Zeilenindex als Suffix angesprochen. Solche Tabellen nennt man „Conceptual Table“.

Nachfolgend werden die eigentlichen Protokolle betrachtet.

Da die SNMP Version 2 im Grundprinzip der Version 1 gleicht, sind im Abschnitt SNMP Version 2 lediglich Änderungen zu den bereits gemachten bzw. im nächsten Abschnitt folgenden Angaben vermerkt.

### 3.3.2 SNMP Version 1

Diesem Standard liegt das asynchrone und symmetrische Request-Response-Modell mit einem Manager und den Agenten zu Grunde. Für die Kommunikation wird der UDP-Dienst benutzt. Applikationen müssen damit Sicherungsmechanismen für den Datentransport implementieren. Dem Manager wurde der „well-known“-Port 162 und dem Agenten der Port 161 zugeordnet. Jeder Datenaustausch erhält eine Kennung „Request Identifier“. Im Standard der Version 1 sind die in Abbildung 6 dargestellten fünf Operationen definiert.

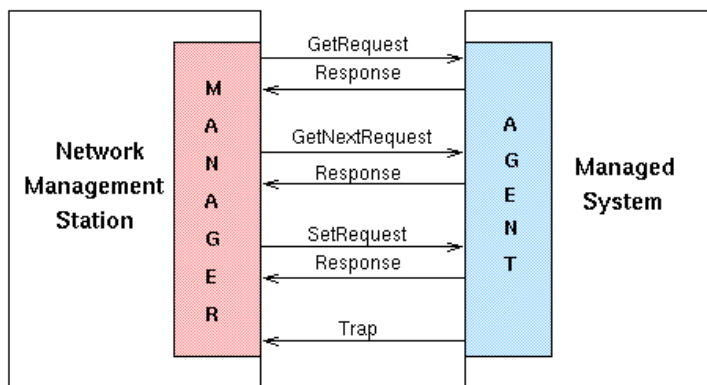
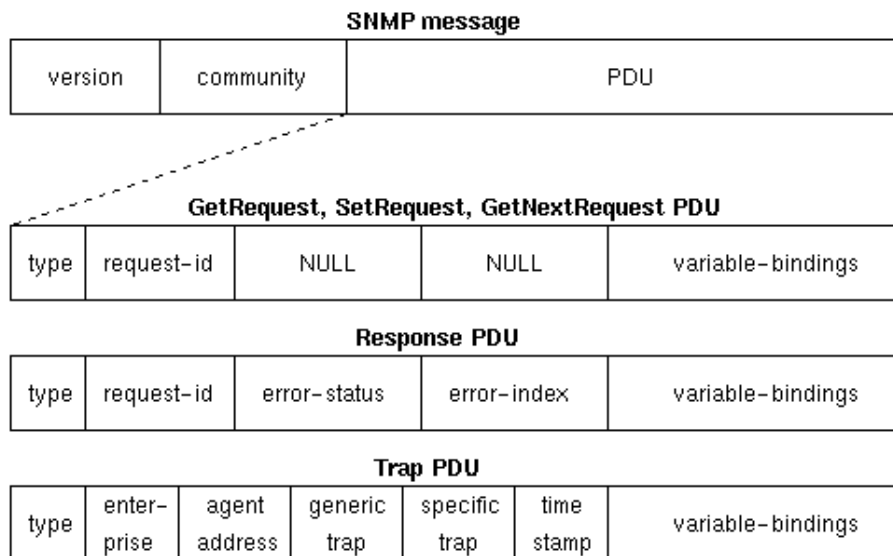


Abbildung 6 SNMPv1 - Nachrichten [Tautenhahn 97]

GetRequest und SetRequest sind Aktionen des Managers, um Werte zu lesen bzw. zu setzen. Der Agent reagiert jeweils mit Response. Eigene Aktivitäten entwickeln die Agenten nur bei Ausnahmesituationen, bei denen eine Mitteilung, genannt „Trap“, an die Manager gesendet wird. In [RFC 1215] sind folgende zu meldende Ausnahmezustände definiert: Kaltstart, Verbindungsverlust, Verbindungsaufbau, Authentifizierungsfehler, Verbindungsverlust zum nächsten Router und herstellenspezifische Festlegungen.

Der Nachrichtenaustausch erfolgt in ASN.1-kodierten PDUs (Abbildung 7). Jede Nachricht beginnt mit der Versionsnummer des Protokolls und einem „Community String“. Dieser unverschlüsselte String dient der Authentifizierung der Nachricht. In jedem Agenten existiert ein entsprechendes „Community Profile“ in dem zu jedem Community-String die Zugriffsrechte auf die verwalteten Objekte vermerkt sind. Ein Authentifizierungsdienst im Agenten prüft alle Anfragen daraufhin, ob der enthaltene Community-String ihm bekannt ist oder nicht. Erlauben sowohl das in ASN.1-Kodierung angegebene Zugriffsrecht auf das Objekt (not-accessible, read-only oder read-write) als auch die Rechte im entsprechenden Community-Profilen einen Zugriff, so wird die angeforderte Aktion durchgeführt.



**Abbildung 7** SNMPv1-PDUs [Tautenhahn 97]

Bei einer Implementierung muß in der Senderoutine folgendes ausgeführt werden:

- Konstruktion der entsprechenden Protocol Data Unit (PDU) als ASN.1-Objekt.
- Übergabe des ASN.1-PDU-Objektes zusammen mit dem Community String, der Quell- und der Zieladresse (IP-Adresse und UDP-Portnummer) an den lokalen Authentifizierungsdienst. Dieser kann nun das ASN.1-PDU-Objekt geeignet verschlüsseln. Da solche Verschlüsselungsverfahren für SNMP Version 1 nicht definiert sind, wird in der Regel das gleiche ASN.1-PDU-Objekt zurückgegeben.
- Konstruktion der SNMP-Nachricht als ASN.1-Objekt aus dem ASN.1-PDU-Objekt, dem Community String und der SNMP-Versionsnummer.
- Kodierung der SNMP-Nachricht nach den „Basic Encoding Rules“ und Senden des Paketes durch Übergabe an das Transportprotokoll.

Für die Empfangsroutine sind die nachfolgenden Punkte zu beachten:

- Dekodierung des empfangenen Paketes nach den „Basic Encoding Rules“. Dadurch wird das gesendete ASN.1-Nachrichten-Objekt wieder hergestellt.
- Überprüfen der SNMP-Versionsnummer. Stimmt sie nicht mit der eigenen Version überein, wird das Paket verworfen.
- Übergabe des ASN.1-PDU-Objektes, des Community-Strings, der Quell- und der Zieladresse an den lokalen Authentifizierungsdienst. Überprüfen der Echtheit des ASN.1-PDU-Objektes. Schlägt die Authentifizierung fehl, wird die Nachricht verworfen und ein SNMP Trap vom Typ „authenticationFailure“ erzeugt.
- Zerlegung des ASN.1-PDU-Objektes und Ausführen der enthaltenen Protokolloperation unter Berücksichtigung des Community-Profiles.

In der Diplomarbeit [Neuendorf 93] wurden einmal verschiedene ASN.1-Kodierungsprogramme untersucht. Mehrere Veröffentlichungen sprechen davon, daß diese Kodierungs- und Dekodierungsregeln einen großen Teil der Implementierung ausmachen, den ein Programmierer durch bereits vorhandene Routinen realisieren sollte.

Als Beispiel einer SNMPv1-Kommunikation sei in Tabelle 2 einmal die Abfrage des Objektes „sysDescr.0“ analysiert. Dieses Objekt ist stets nur einmal in den Geräten instanziiert und gibt Auskunft über das Betriebssystem, den Namen des Gerätes und die Hardware.

Gesendete Bytes (Hex)	Bedeutung	Protokoll
45	IP Version 4; 5x4 Bytes Kopflänge	IP-PDU
00	Type of Service = normal service	
00 47	Gesamtlänge = 71 Bytes	
5d 15	Identifikation	
00 00	Flag und Fragmentierungsoffset	
3c	Time to Live (TTL) = 60	
11	Diensttyp = UDP	
0c ac	CRC-Summe	
86 6d 04 0a	IP-Source = 134.109.4.10	
86 6d 04 01	IP-Destination = 134.109.4.1	
05 27	Source-Port = 1319	
00 a1	Destination-Port = 161	
00 33	UDP-Kopf & Last - Länge = 51	
00 00	Prüfsumme	
30 29	Anfrage-Kodierung	SNMP-PDU in ASN.1-Format
02 01 00	Int-ID; Länge 1; Wert 0 -> Version 1	
04 06	Octet-ID; Länge 6	
70 75 62 6c 69 63	Community = public	GetRequest-PDU in ASN.1-Format
a0	cont. spec.; primitive; getrequest	
02 04	Int-ID, Länge 4	
3c 3b 04 69	Request-ID	
02 01 00	Int-ID, Länge 1, Error-Status = 0	
02 01 00	Int-ID, Länge 1, Error-Index = 0	
30 0e	Beginn „Variable Bindings“	
30 0c		
06 08 2b	name	
06 01 02 01 01 01 00	sysDescr.0	
05 00	value simple empty	

**Tabelle 2** Analyse einer GetRequest-Nachricht

SNMPv1 ist einfach zu implementieren und wird daher von allen Internet-Managementapplikationen unterstützt.

Im Laufe der Zeit wurden jedoch auch einige Nachteile deutlich:

- Agenten erhalten nie Rückmeldung, ob ihre Mitteilungen erfolgreich waren.
- GetRequest-Nachrichten enthalten die genaue Bezeichnung des zu lesenden Objektes. Die zum Auslesen von Tabellen gedachte getNextRequest-PDU jedoch bezieht sich lediglich auf das nächste in der MIB-Struktur vorliegende Objekt, was eine aufwendige Variablensortierung im Agenten erfordert.
- Da zum Lesen einer Variablen stets zwei PDUs (GetRequest bzw. getNextRequest und Response) übertragen werden, entsteht eine hohe Netzlast.
- Da keine Manager-to-Manager-Kommunikation in der Version 1 vorgesehen ist, muß der Manager zumindest die gesamte Community überwachen. In großen Netzen mit mehreren Managern entstehen damit isolierte Informationsinseln.

- Außer der notwendigen Kenntnis des Community-Strings existieren keinerlei Sicherheitsmechanismen. So können z.B. Dateninhalte unbemerkt geändert oder infolge des unsicheren UDP-Dienstes die Reihenfolge verändert werden. Durch Neudefinierung des Community-Strings kann jeder Besitzer einer Managementstation freien Zugriff auf alle Agenten erreichen. Nicht zuletzt das Mitlesen der Pakete im Netz ist durch die unverschlüsselte Übertragung sehr ergiebig.
- SNMPv1 - Software unterliegt keiner Zertifizierung. Damit ist nicht eindeutig, was die Herstellerangabe 'SNMP-fähig' letztlich bedeutet.

Zum Beseitigen dieser Punkte entwickelte man die Version 2 des SNMP, die zukünftig die Version 1 vollständig ablösen soll.

Im nächsten Abschnitt wird auf die Neuheiten dieser Nachfolgeversion näher eingegangen.

### 3.3.3 SNMP Version 2

Die Probleme der Version 1 versuchten J. Case, S. Waldbusser, M.T. Rose und K. McCloghrie 1992 mit einer neuen Version des SNMP mit dem Namen „Simple Management Protocol (SMP)“ zu lösen. Im April 1993 wurde dieser eingereichte Entwurf von der „Internet Engineering Task Force (IETF)“ als „Simple Network Management Protocol Version 2 (SNMPv2)“ unter den RFCs 1441-1452 veröffentlicht.

Neuerungen :

- Einführung von sogenannten „Parties“, durch welche der Netzadministrator Sicherheits- und Authentifizierungsmechanismen konfigurieren kann. Jede Party hat ihre eigenen Zugriffsrechte und ihre Sichtweise der MIB (MIB-View). Damit brauchen Agenten nur noch die für sie nötigen Teile des MIB-Baumes konfigurieren.
- Manager-to-Manager-Kommunikation wird über eine spezielle MIB realisiert. Applikationen können als Manager oder als Agenten auftreten.
- Um die Sicherheit zu erhöhen, wird zur Authentifizierung der „Message Digestion 5 (MD5)“ - Algorithmus verwendet. Weiterhin sollen Zeitstempel das Wiederholen einer gültigen Nachricht verhindern. Letztlich gibt es auch noch die Möglichkeit alle Daten mittels des „Data Encryption Standard (DES)“ zu verschlüsseln („Privacy“).
- Mit der Einführung der GetBulkOperation können „Sammeltransporte“ realisiert werden, die die Netzlast erheblich verringern.
- Es wurden weitere Fehlerkennungen eingeführt.
- Durch die Festlegung verschiedener „Access-Points“ wird der Weg zum Multiprotokoll-Management frei. So kann SNMPv2 außer auf UDP auch z.B. auf Novell-IPX aufsetzen.

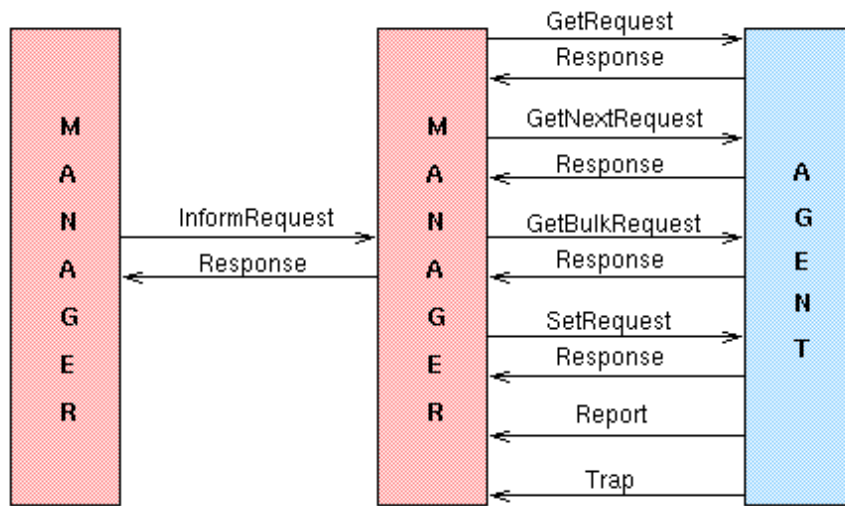
Neben der erheblich aufwendigeren Implementierung („Simple“ ist also nicht so streng zu sehen) ist der Einsatz des DES-Algorithmus hinderlich für die Verbreitung des Standards, da dieser Algorithmus nur innerhalb der USA benutzt werden darf (Exportbeschränkung).

Um trotzdem den neuen Standard verwenden zu können, wird in RFC 1901 die Umsetzung der Version 2 bei Beibehaltung der einfachen Community-Authentifizierung der Version 1

vorgeschlagen. Dies wird in der Literatur „Community-based SNMPv2 (SNMPv2C)“ genannt.

Ein weiterer Ausweg sind die RFC 1902-1908, die die Version 2 jedoch ohne Festlegung des Sicherheitsmodells angeben.

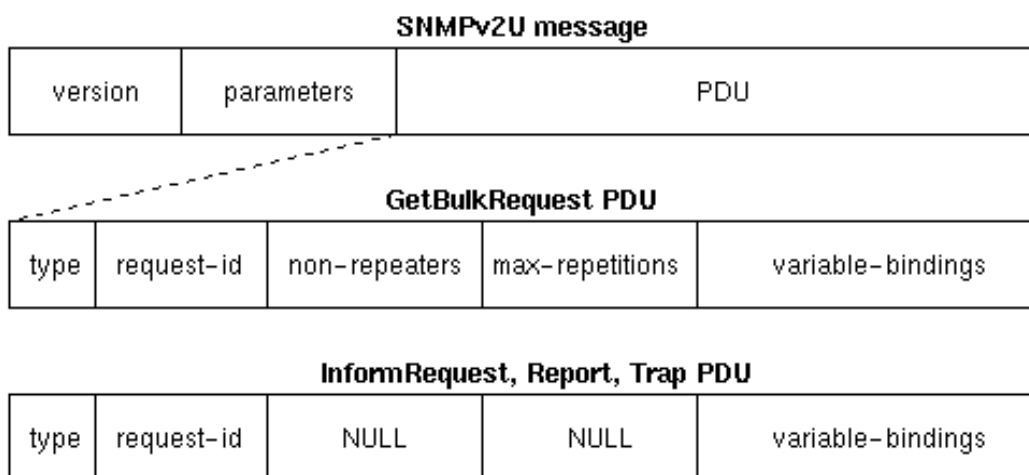
In den RFC 1909 „An Administrative Infrastructure for SNMPv2“ und RFC 1910 „User-based Security Modell for SNMPv2“ wird ein einfaches Sicherheitskonzept vorgeschlagen, was unter „SNMPv2U“ bekannt ist. Dieses Verfahren scheint die beste Lösung des Problems darzustellen. Nachfolgende Betrachtungen beziehen sich also auf SNMPv2U.



Wie in Abbildung 8 erkennbar, sind im Vergleich zur Version 1 drei neue Operationen definiert worden. Um das hohe Datenaufkommen der alten Version (pro Variable eine PDU) zu senken, wurde die **GetBulk**-Operation eingeführt. Sie entspricht einer mehrfachen Ausführung von **GetNextRequests**, deren Rückantworten in eine

**Abbildung 8** SNMPv2 - Nachrichten [Tautenhahn 97]

Response-PDU gebündelt werden. Die Manager-to-Manager-Kommunikation erfolgt mit den Operationen „**InformRequest / Response**“, welche den Datenaustausch nach einer speziellen MIB realisieren. Die **Report**-Operation steht den Agenten für den allgemeinen Austausch interner Protokollstackinformationen zur Verfügung.

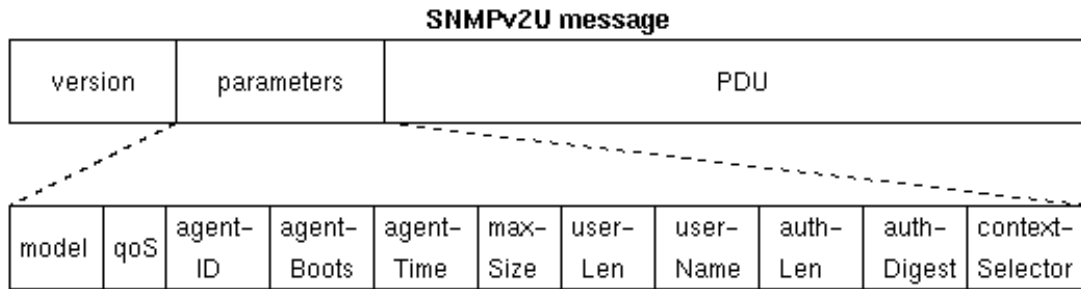


**Abbildung 9** SNMPv2-PDUs [Tautenhahn 97]

Wie bereits oben erwähnt, entspricht **GetBulk** einem mehrfachen **GetNextRequest**. Vor diesem Hintergrund sind dann auch die Felder „**non-repeaters**“ und „**max-repetitions**“ aus

Abbildung 9 zu verstehen. Das erste Feld gibt an, wieviele Objekte (beginnend mit dem in „variable-bindings“ angegebenen) vom Befehl unberücksichtigt bleiben sollen. Das zweite Feld, „max-repetitions“, legt die Anzahl von Folgeobjekten fest, für die ein GetNextRequest im Agenten auszuführen ist. Die dabei ermittelten Daten werden anschließend in einer Response-PDU dem Manager übergeben.

Im Vergleich zu den Nachrichten der Version 1 fällt auf, daß hier anstelle des Community-Strings das Feld „Parameters“ existiert. Diese Parameterangaben sind in Abbildung 10 gezeigt. Sie werden für das „user-based security model“ benötigt.



**Abbildung 10** Die Parameterangabe [Tautenhahn 97]

Dieses Sicherheitsmodell sieht sowohl ein Authentifizierungsverfahren als auch ein „Privacy“-Verfahren vor. Das Feld „QoS“ teilt nun mit, ob eines oder beide Verfahren auf die Nachricht angewendet worden sind.

Nachfolgend werden die Sicherheitsmechanismen näher erläutert:

- **Authentication** (digitale Unterschrift)

Im Manager und im Agenten werden Nutzer mit jeweiligem „Authentication-Key“ (Paßwort) konfiguriert. Dieser Schlüssel dient dem „Message Digest 5 (MD5)“ - Algorithmus zusammen mit der eigentlichen Nachricht zum Berechnen eines elektronischen „Fingerabdruckes“. Der Fingerabdruck wird zusammen mit dem Nutzernamen („user-name“) im Paket versendet. Eine entsprechende Fingerabdruckberechnung für den angegebenen Nutzer auf Empfängerseite führt bei Übereinstimmung zur Authentifizierung.

⇒ Verhinderung unerlaubter Modifikation der Nachricht und Schutz vor Auftreten unautorisierter Nutzer als Manager.

- **Privacy** (Verschlüsselung der Gesamtnachricht)

Zu jedem konfigurierten Nutzer wird ein „Privacy-Key“ festgelegt, der von einem symmetrischen Verfahren zur Verschlüsselung bzw. Entschlüsselung der Nachrichten-PDU genutzt wird.

⇒ Schutz vor Abhören der Nachrichten.

- **„agentID“, „agentBoots“, „agentTime“**

AgentID dient der Identifikation des Agenten. AgentBoots ist ein Zählerstand, wie oft der Agent „Reboots“ bzw. Initialisierungen durchgeführt hat. AgentTime gibt die Zeit in Sekunden seit der letzten Erhöhung von AgentBoots an.

⇒ Verhinderung von Verzögerungen oder Wiederholungen, sowie unerlaubter Umordnung von Nachrichten.

Bei einer Implementierung müssen in der Senderoutine folgende Schritte ausgeführt werden:

- Konstruktion der entsprechenden „Protocol Data Unit (PDU)“ als ASN.1-Objekt.
- Übergabe des ASN.1-PDU-Objektes zusammen mit dem gewünschten „QoS“, dem Nutzernamen, dem „Authentication-Key“, dem „Privacy-Key“, der Quell- und der Zieladresse (IP-Adresse und UDP-Portnummer) an den lokalen Authentifizierungsdienst. Dieser Dienst trägt den „Authentication-Key“ in das Feld „auth-Digest“ ein und ruft den MD5-Algorithmus auf. Danach wird das gleiche Feld mit dem berechneten Fingerabdruck gefüllt. Die restlichen Felder der „Parameters“ werden mit den aktuellen Werten belegt.
- Verschlüsselung der authentifizierten ASN.1-PDU mittels des „Privacy-Key“ und des symmetrischen Algorithmus.
- Konstruktion der SNMP-Nachricht als ASN.1-Objekt aus dem verschlüsselten authentifizierten ASN.1-PDU-Objekt, dem Parametern und der SNMP-Versionsnummer.
- Kodierung der SNMP-Nachricht nach den „Basic Encoding Rules“ und Senden des Paketes durch Übergabe an das Transportprotokoll.

Für die Empfangsroutine sind die nachfolgenden Punkte zu beachten:

- Dekodierung des empfangenen Paketes nach den „Basic Encoding Rules“. Dadurch wird das gesendete ASN.1-Nachrichten-Objekt wieder hergestellt.
- Überprüfen der SNMP-Versionsnummer. Stimmt sie nicht mit der eigenen Version überein, wird das Paket verworfen.
- Übergabe des ASN.1-PDU-Objektes, des Parameterfeldes, der Quell- und der Zieladresse an den lokalen Authentifizierungsdienst.
- Je nach „QoS“-Feld wird der zum Nutzer gehörende „Privacy-Key“ und „Authentication-Key“ ermittelt und die Dekodierung mit dem symmetrischen Algorithmus vorgenommen. Nachfolgend wird die authDigest-Angabe gespeichert und der „Authentication-Key“ des Nutzers in dieses Feld eingetragen. Nach Aufruf der MD5-Funktion werden die beiden Fingerabdrücke verglichen. Stimmen sie nicht überein, wird das Paket verworfen und ein SNMP Trap vom Typ „authentication-Failure“ erzeugt.
- Treten bei der Auswertung der Felder „agentID“, „agentBoots“ und „agentTime“ Fehler auf, so wird das Paket verworfen und ein SNMP Trap vom Typ „authenticationFailure“ erzeugt.
- Zerlegung des ASN.1-PDU-Objektes und Ausführen der enthaltenen Protokolloperation unter Berücksichtigung der objektspezifischen und nutzerspezifischen Zugriffsrechte.

### **3.4 Management in ATM-Netzen**

Asynchronous Transfer Mode ist ein Verfahren zur Mehrfachnutzung eines (physischen) Übertragungsmediums durch Adreßmultiplex und statistisches Multiplex [ATM 96].

Netze, die auf diesem Verfahren beruhen, sind derzeit im WAN- und MAN-Bereich zu finden. Die Verbreitung in den lokalen Netzen ist noch gering, jedoch wird dieses Anwendungsfeld von der Standardisierung vollständig überdeckt. Es ist also in Zukunft mit einer breiten Anwendung zu rechnen.

Die Standardisierung im Bereich des ATM ist noch nicht abgeschlossen und es werden gerade auf dem Gebiet des Netzwerkmanagements derzeit fehlende Standardisierungen vorgenommen. Eine entscheidende Rolle spielt dabei das „ATM-Forum“. Das ATM-Forum ist eine Organisation, die die industrielle Kooperation bei der ATM-Technik unterstützt. Sie vermittelt dabei zwischen der Industrie und der Standardisierung. Auf dem WWW-Server des ATM-Forums sind die bestehenden Spezifikationen direkt abrufbar. Einige dieser Dokumente wurden für die Erarbeitung der nachfolgenden Abschnitte entsprechend durchgearbeitet und aufbereitet.

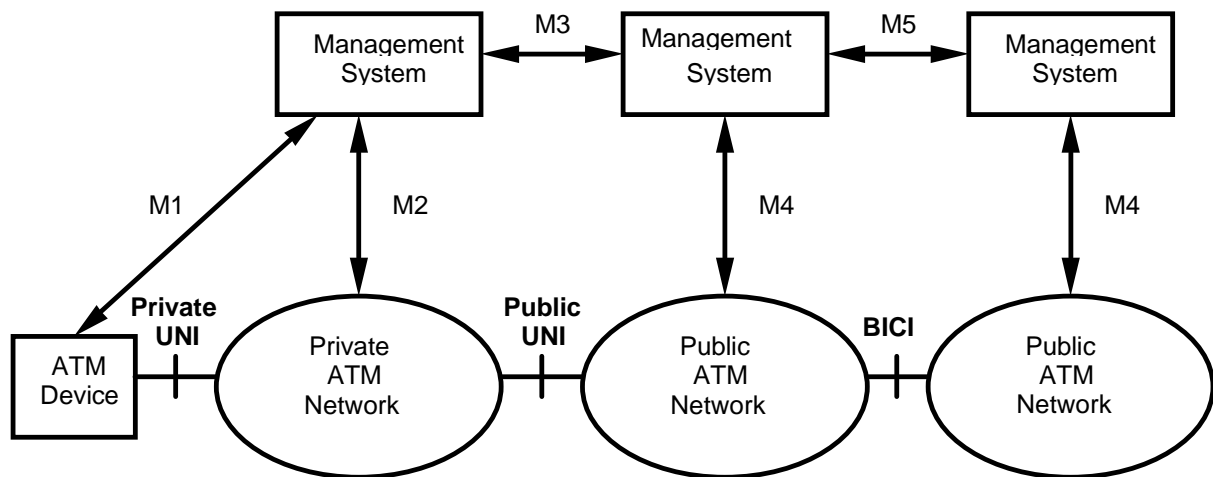
Da das Netzwerkmanagement bei ATM noch nicht vollständig festgelegt ist, existieren derzeit noch viele herstellereigene Lösungen. Dies bringt eine Vielzahl von genutzten Schnittstellen und Protokollen mit sich. Das Netzwerkmanagement umfaßt in diesen Netzen neben der Stabilität und Effizienz des Netzes auch den Bereich der Überwachung der vereinbarten und zu garantierenden Qualitätsparameter. Das Abrechnungsmanagement besitzt in ATM-Netzen ein anderes Niveau und eine größere Komplexität. So ist in der Abrechnung der Verbindungskosten die vereinbarte Verbindungsqualität (Übertragungsrates, max. Verzögerungszeiten ...), sowie entsprechende Strafmaßnahmen bei Bruch des Verbindungsvertrages zu beachten. Gerade die stichprobenhafte Überprüfung von Verbindungseigenschaften stellt dabei hohe Anforderungen an das Netzwerkmanagement und den dafür benötigten Ressourcen (Speicherplatz, Netzlast, CPU-Last).

Grundsätzlich wird das Management nach dem Kommunikationsweg in zwei Gruppen unterteilt. Im „Out-of-Band-Management“ gibt es zusätzlich zum ATM-Netz weitere Zugänge (z.B. serielle Schnittstelle) zur ATM-Komponente, über die die Managementinformation ausgetauscht wird. Bei dem „In-Band-Management“ benutzt man für diese Kommunikation das ATM-Netz selbst.

Für das Management in ATM-Netzen wurde vom ATM-Forum ein eigenes In-Band-Management-Modell entworfen. In den nächsten Abschnitten wird nun näher darauf eingegangen.

### 3.4.1 Das allgemeine Managementmodell

Um die verschiedenen Arten des Netzwerkmanagements für ATM-Geräte, private und öffentliche Netze und deren Interaktionen zu beschreiben, wurde vom ATM-Forum folgendes Netzwerkmanagement-Modell vorgestellt.



**Abbildung 11** ATM Network Management Model [M3 94]

Wie aus Abbildung 11 ersichtlich, existieren fünf Schnittstellendefinitionen (M1 .. M5), die entsprechend ihres Managementbereiches spezifische Managementaufgaben realisieren. Es ist dabei fest vorgegeben, welchen Managementeinfluß man mit der jeweiligen Schnittstelle ausüben kann.

M1, M2 und M4 definieren das Zusammenspiel zwischen Managementsystem und dem jeweiligen Netz bzw. Gerät. M3 und M5 beschreiben die Schnittstellen der Managementsysteme untereinander. Es wird die Trennung zwischen privat und öffentlich auch hier vorgenommen. Während M1 und M2 ausschließlich im privaten Bereich und M4 und M5 ausschließlich im öffentlichen Bereich genutzt werden, bildet M3 die Definition des Überganges zwischen beiden. Mittels M3 ist es daher möglich gerade auf den Teil des öffentlichen Netzes Einfluß zu nehmen, in dem sich das eigene private Netz befindet.

Da die Industrie nicht auf die Standardisierung von M1 und M2 warten konnte, wurde eine Übergangslösung, genannt „Interim Local Management Interface (ILMI)“, in der UNI 3.1-Spezifikation [UNI 94] veröffentlicht. Als im September 1996 der Standard „Integrated Local Management Interface (ILMI) Version 4.0“ [ILMI 96] vorlag, wurde ILMI als auf unbestimmte Zeit bestehender Standard für die Schnittstellen M1 und M2 festgeschrieben.

Zu ILMI, M3 und M4 sind derzeit beim ATM-Forum Dokumente verfügbar. Die Schnittstellendefinition M5, weitere Ausführungen zu M4, sowie Festlegungen zu Sicherheitsmechanismen der Schnittstellen sind für Ende des Jahres bzw. für Anfang 1998 angekündigt.

In den folgenden Abschnitten wird nun auf Grundlage der verfügbaren Dokumente auf die einzelnen Schnittstellen näher eingegangen. Innerhalb der Spezifikationen werden häufig Abkürzungen über den Status der Definitionen verwendet, deren Bedeutungen in Tabelle 3 aufgeführt sind.

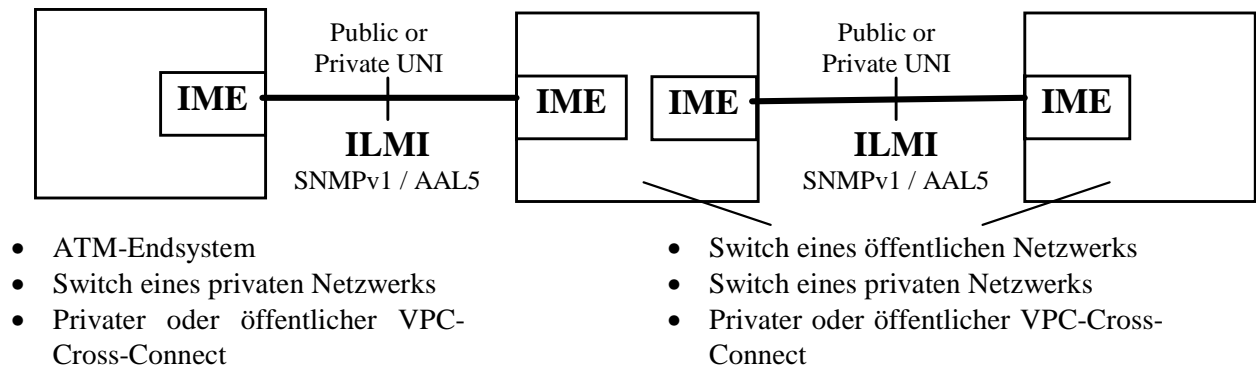
Abkürzung	Bezeichnung	Bedeutung
R	Required	Wenn eine Gruppe erforderlich ist, so ist auch jedes Element erforderlich.
CR	Conditionally Required	Ist eine Gruppe bedingt erforderlich, so ist jedes Element erforderlich, falls die Gruppe überhaupt implementiert wird.
O	Optional	Die so gekennzeichneten Definitionen sind nicht zwangsweise erforderlich, sondern den Bedürfnissen des Netzerkanbieters gemäß zu berücksichtigen.
D	Obsolete / Deprecated	Dieser Status gibt an, daß diese Definitionen zwar existent aber veraltet bzw. verworfen worden sind. Sie sollten bei Neuimplementationen hinterfragt werden. Man kann ebenfalls nicht voraussetzen, daß sie in anderen Geräten verfügbar sind.

**Tabelle 3** Gebräuchliche Abkürzungen in den ATM-Forum Dokumenten

### 3.4.2 Integrated Local Management Interface - ILMI

Im Bereich der privaten Netze wird mit der M1-Schnittstelle das Management einer einzelnen ATM-Station (Endsystem oder Switch) vorgenommen. M2 wird von einer privaten Netzwerkmanagementstation zum Management des privaten ATM-Netzes verwendet. Beide Managementschnittstellen waren für die privaten ATM-Netzbetreiber von großem Interesse. Seitens des ATM-Forums wurde deshalb 1994 eine Übergangslösung unter dem Namen „Interim Local Management Interface (ILMI)“, die den Funktionsumfang der M1 bzw. M2 Schnittstelle abdeckte veröffentlicht. Sie kann in der UNI 3.1 - Spezifikation [UNI 94] im Kapitel 4 nachgelesen werden. Eine Übergangslösung war es deshalb, weil man mit der endgültigen Spezifikation auf die Standards der ITU-T und ANSI Kommissionen warten wollte.

ILMI fügt sich in dem allgemeinen ATM-Managementmodell an den Stellen der M1- und M2-Schnittstellen, also dem Management von ATM-Geräten und privaten ATM-Netzen ein. Es werden dabei, wie in Abbildung 12 dargestellt, Punktverbindungen zwischen sogenannten „ATM Interface Management Entities (IME)“ für die ILMI-Kommunikation genutzt. Jedes ATM-Gerät (ATM-End-Gerät, privater oder öffentlicher ATM-Switch bzw. Cross-Connect) unterhält für jede ILMI-Verbindung eine solche IME. In der ILMI-Kommunikation werden nach AAL5 verpackte SNMPv1-PDUs übertragen.



**Abbildung 12** ILMI-Kommunikation zwischen ATM-Geräten

SNMP und eine „ATM UNI Management Information Base (MIB)“ stellen Status-, Konfigurations- und Steuerinformationen auf Verbindungs- und physischer Ebene an der UNI bereit. Das ILMI-Protokoll ist ein offenes Protokoll, das z.B. die Übertragung der SNMP-PDUs über AAL5 mit vorgegebenen VPI/VCI - Werten erlaubt und sich nicht fest auf UDP / IP festlegt. Im September 1996 wurde ein neues Dokument [ILMI 96] veröffentlicht, in dem ein Standard unter dem Namen „Integrated Local Management Interface (ILMI) Specification Version 4.0“ veröffentlicht wurde. Darin wird in Bezug auf die bisherige ILMI-Definition gesagt, daß auf unbestimmte Zeit ILMI für das Management im privaten Bereich angewendet wird. Dieses Dokument löst die bisherige Definition ab und stellt eine vollständige ILMI-Spezifikation dar. Da dieser Teil des allgemeinen Managementmodells bei der Implementierung einer Netzwerkmanagementunterstützung für eine ATM-NIC zu beachten ist, wird nachfolgend genauer auf den in [ILMI 96] vereinbarten Standard eingegangen.

In diesem Dokument ist nun beschrieben, wie SNMP und eine spezielle MIB (ATM Interface Management Information Base) genutzt werden, um ein ATM-Gerät mit Status- und Konfigurationsinformationen bezüglich virtueller Pfad- und Kanalverbindungen, registrierten ATM-Netzwerk-Prefixen, registrierten ATM-Adressen, registrierten Diensten und Funktionalitäten an dessen ATM-Interface zu versorgen. Die nachfolgend aufgeführten Prinzipien sind im Dokument [ILMI 96] nachzulesen.

Funktionsprinzipien und Optionen des „Integrated Local Management Interface“:

- Jedes ATM-Gerät (z.B. Switch, Endsystem, usw.) soll ein oder mehrere ATM-Interfaces unterstützen
- Die ILMI-Funktionen für ein ATM-Interface stellen Konfigurations-, Status- und Steuerinformationen über Parameter der physikalischen und der ATM-Ebene dieses Interfaces zur Verfügung.
- Pro ATM-Interface gibt es einen Satz von „Managed Objects“, die auch „ATM Interface ILMI attributes“ genannt werden, welcher für die Unterstützung der ILMI-Funktionen für dieses ATM-Interface ausreichend ist.
- Die ATM-Interface-ILMI-Attribute sind in einer Standard-MIB-Struktur organisiert. Für jedes Interface eines ATM-Gerätes gibt es dabei je eine Instanz dieser MIB-Struktur.
- Für jedes ATM-Interface eines ATM-Gerätes gibt es eine sogenannte „ATM Interface Management Entity (IME)“, welche die ILMI-Funktionen für dieses Interface unterstützt.

- Wenn zwei ATM-Geräte über ihre ATM-Interfaces verbunden sind, so gibt es zwei IMEs, die zu diesem ATM-Interface gehören - je eine IME pro ATM-Gerät. Die beiden IMEs bezeichnet man auch als „adjacent IMEs“.
- Die ILMI-Kommunikation findet jeweils zwischen diesen benachbarten IMEs über eine physische Verbindung oder eine virtuelle (z.B. virtuelle Pfadverbindung) statt.
- Das ILMI-Protokoll ist das offene Protokoll SNMP (z.B. SNMP/AAL5) ohne die UDP und IP Adressierung.
- Eine IME kann mittels des ILMI-Protokolls auf die ATM-MIB-Information der Nachbar-IME zugreifen.
- Ob zusätzlich zu den MIB-Informationen weitere Informationen mittels ILMI erreichbar sind, ist derzeit nicht festgelegt und obliegt jedem Hersteller selbst.
- Für die Adreßregistrierung über die UNI-Schnittstelle sind ebenfalls ILMI-Funktionen vorgesehen.
- ILMI-Funktionen für die LAN-Emulation ermöglichen die Autokonfiguration eines „LAN Emulation Clients (LEC)“. Mehr dazu ist in der „LAN Emulation Specification, Version 1.0“ des ATM-Forums nachzulesen.

An der Eingliederung von ILMI in das allgemeine ATM-Management-Modell hat sich in diesem neuen Standard im Vergleich zum alten nichts geändert. Ebenfalls gleich ist, daß ILMI in jedem ATM-Gerät pro Interface eine IME definiert. In jeder IME existiert dabei ein Agent, der mit der Managementapplikation in Verbindung steht. Diese Agenten operieren üblicherweise mit den gleichen MIBs. Dazu ist zu sagen, daß der Inhalt einzelner MIB-Objekte gerätespezifisch interpretiert wird. Überarbeitet wurden in dem neuen Standard die Anforderungen und Funktionsprinzipien von ILMI.

Bevor die verwalteten Objekte näher betrachtet werden, wird das eigentliche Protokoll beschrieben.

In dem ILMI-Standard ist als Protokoll SNMP in der Version 1 festgeschrieben. Eine Verwendung der Version 2 wird derzeit lediglich diskutiert.

Für das Übertragen der SNMP-PDUs (Requests, Responses, Traps) wird genau eine virtuelle Kanalverbindung belegt. Die SNMP-PDU wird dabei nach AAL 5 „verpackt“. Die dafür festgelegten „Common Part Convergence Sublayer (CPCS)“-Grundelemente und Parameter sind im Standard genau aufgeführt.

Die im allgemeinen für ILMI vorgesehene VPI/VCI-Kombination (VPI=0, VCI=16) sollte konfigurierbar sein. Die für ILMI gesendeten Zellen werden mit hoher Priorität (CLP=0) gesendet, um sicherzustellen, daß auch bei hoher Netzlast steuernde Eingriffe übermittelt werden. Um die Netzlast durch das Management und die Reaktionszeiten auf Anforderungen bzw. Ereignisse zu beschränken wurden folgende Randbedingungen vereinbart:

- Die ILMI-Kommunikation im Kanal (VPI=0, VCI=16) darf nicht mehr als 1% der gesamten Kanalbandbreite belegen.
- Der erlaubte Spitzenwert der ILMI-Last liegt bei 5% der gesamten Kanalbandbreite.
- Die maximale Burst-Länge ist 484 Oktetts.
- Ein Agent muß innerhalb einer Sekunde auf SNMP-Requests die entsprechende Antwort dem Nachbarsystem senden. Diese Antwortzeit bezieht sich auf ein MIB-Objekt und muß bei 95% aller Requests eingehalten werden.
- Für die Meldung eines Netzereignisses (Trap) ist eine Reaktionszeit von zwei Sekunden zulässig.

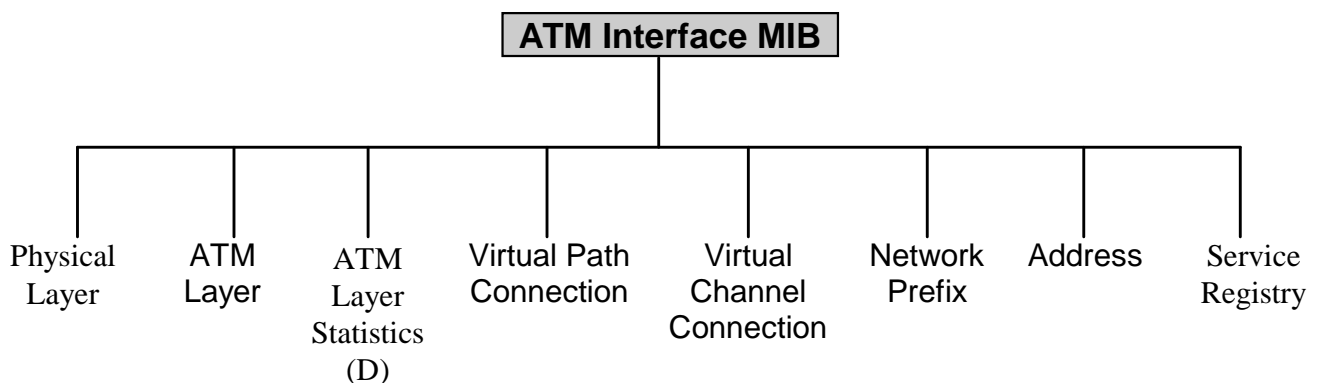
Zusätzlich zu diesen Festlegungen existieren weitere Unterschiede zum herkömmlichen SNMP. Diese sind:

- Als Standard-Community-Name wird vom ATM-Forum der String „ILMI“ mit dem Wert „494C4D49h“ vorgeschrieben.
- Alle ILMI-Implementationen akzeptieren PDU-Größen von 484 Oktetts. Größere PDUs sollten nicht gesendet werden, es sei denn nach speziellen Absprachen für das Out-Of-Band-Management.
- Das Agent-Address-Feld der Trap-PDUs enthält stets die IP-Adresse „0.0.0.0“.
- Das Time-Stamp-Feld der Trap-PDUs enthält den Wert des IME sysUpTime-Objektes.
- Das Enterprise-Feld der Trap-PDUs enthält den Wert des IME sysObjectID Objektes.

Die bisherigen Ausführungen beziehen sich auf Anfragen eines Netzwerkmanagementsystems, daß einen Agenten über ein bestehendes ATM-Netz anspricht. Für die Kommunikation mit dem Agenten, der auf der gleichen Maschine wie das Managementsystem arbeitet, wird die allgemein übliche Kommunikation über UDP gewählt. Dabei werden Nachrichten zum Agenten an den Port 161 gesendet und die entsprechenden Rückantworten am Port 162 von der Managementanwendung entgegengenommen. Die verwendeten MIB-Definitionen bleiben von dieser Änderung unberührt.

Zum besseren Verständnis des ILMI-Funktionsumfanges trägt die nachfolgende Betrachtung der MIB-Definitionen bei.

Die im ILMI-Standard vorgenommenen Objektdefinitionen sind in sieben Objektgruppen zusammengefaßt. Diese Gruppen sind in Abbildung 13 einmal dargestellt.



**Abbildung 13** Objektgruppen der ATM-Interface-MIB [ILMI 96]

Zusätzlich muß jede ILMI-Implementation laut Standard die Objekte der Systemgruppe der MIB II [RFC 1213] enthalten. Die Gruppe „ATM Layer Statistics“ stammt aus der ersten ILMI-Definition [UNI 94] und besitzt nun den Status „depricated“. Im Sinne der Abwärtskompatibilität ist sie jedoch in der aktuellen MIB-Definition mit enthalten.

Jede der Objektgruppen beinhaltet die sogenannten Objekt-Attribute. Die Objekte sind in Tabelle 4 aufgeführt.

Objektgruppe	Objekte / Attribute	MIB
System (R)	⇒ Systemdescriptor ⇒ Systemobjekt-ID ⇒ Systemzeit seit Reset ⇒ Systemkontaktperson bzw. -adresse ⇒ Systemname ⇒ Systemstandort ⇒ Systemdienste	MIB II (Standard SNMP-MIB)
Physical Layer (R)	⇒ Port-Index ⇒ Port-Adresse ⇒ Übertragungstyp ⇒ Mediumstyp ⇒ Betriebsstatus	ATM-Forum-MIB
ATM Layer (R)	⇒ Index ⇒ Maximale Anzahl von VPCs ⇒ Maximale Anzahl von VCCs ⇒ Anzahl der konfigurierten VPC ⇒ Anzahl der konfigurierten VCC ⇒ Maximale Anzahl VPI-Bits ⇒ Maximale Anzahl VCI-Bits ⇒ ATM UNI- und Gerätetyp ⇒ UNI-, ILMI- PNNI- und SNNI-Sign. - Version	
ATM Layer Statistics (D)	⇒ Index ⇒ empfangene ATM-Zellen ⇒ verworfene ATM-Zellen ⇒ gesendete ATM-Zellen	
Virtual Path Connection (R)	⇒ Index ⇒ VPI-Wert ⇒ Betriebsstatus ⇒ verschiedene Angaben zum Traffic Descriptor ⇒ QoS-Kategorien für Empfangen und Senden ⇒ Dienste-Kategorie ⇒ ABR-Attribute in Untergruppe „VPC ABR (O)“ pro VPC aufgeführt	
Virtual Channel Connection (R)	⇒ Index ⇒ VPI- / VCI- Wert ⇒ Betriebsstatus ⇒ verschiedene Angaben zum Traffic Descriptor ⇒ QoS-Kategorien für Empfangen und Senden ⇒ Frame-Discard für Empfangen und Senden ⇒ Dienste-Kategorie ⇒ ABR-Attribute in Untergruppe „VCC ABR (O)“ pro VCC aufgeführt	
Network Prefix (CR)	⇒ Port-Nummer ⇒ Network-Prefix ⇒ Status des Network-Prefix	ATM-Forum-Addr-Reg-MIB
Address (CR)	⇒ Port-Nummer	

	⇒ ATM-Adresse ⇒ Status der ATM-Adresse ⇒ Organisationsumfang ⇒ Registrierungsadministrierungs-Index ⇒ Registrierungsadministrierungs-Status	
Service Registry (O)	⇒ Port-Nummer ⇒ Dienste-ID ⇒ ATM-Adresse ⇒ Freier Stringparameter	ATM-Forum-Srvc-Reg-MIB

**Tabelle 4** Übersicht zu den Managementobjekten der ILMI-Spezifikation [ILMI 96]

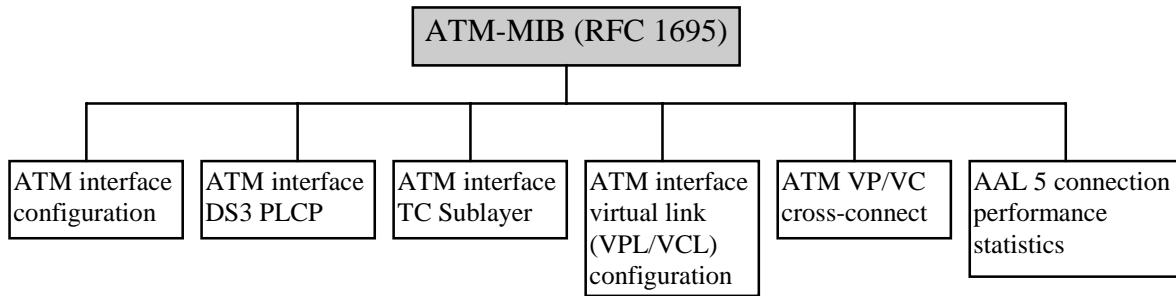
Wie aus der letzten Spalte von Tabelle 4 bereits ersichtlich, wurde die Gesamt-MIB-Definition auf vier ATM-Interface-MIB-Module aufgeteilt. Diese sind:

- **Textual Conventions MIB** (Ausgelagerte Deklarationen zu ATM-Management-spezifischen Typen und Objekt-Kennungen).
- **Link Management MIB** (Festlegungen für das allgemeingültige Verbindungsmanagement aller ATM-Interfaces - sogenannte „ATM-FORUM-MIB“).
- **Address Registration MIB** (Mechanismus zur ATM-Adress-Registrierung einer UNI mittels ILMI-Nachrichten).
- **Service Registry MIB** (Definitionen für eine allgemeine Dienstregistrierung zur Lokalisierung von ATM-Netzwerk-Diensten wie z.B. LAN Emulation Configuration Server (LECS) oder ATM Name Server (ANS) ).

Mit der ausführlichen Auflistung in Tabelle 4 erhält man einen Einblick in die durch ILMI abfragbaren und teilweise veränderbaren Objekte. Mit ihrer Hilfe kann man von jeder Komponente des ATM-Netzes ein detailliertes Abbild erstellen. Es obliegt daher dem Managementsystem, je nach Problemstellung die passenden Objekte abzufragen und in entsprechend aufbereiteter Form an der Managementstation anzuzeigen. Die ILMI-Definition bietet eine weit größere Funktionalität als die des allgemeinen SNMP-Modells.

### ATM-Management mittels SNMP

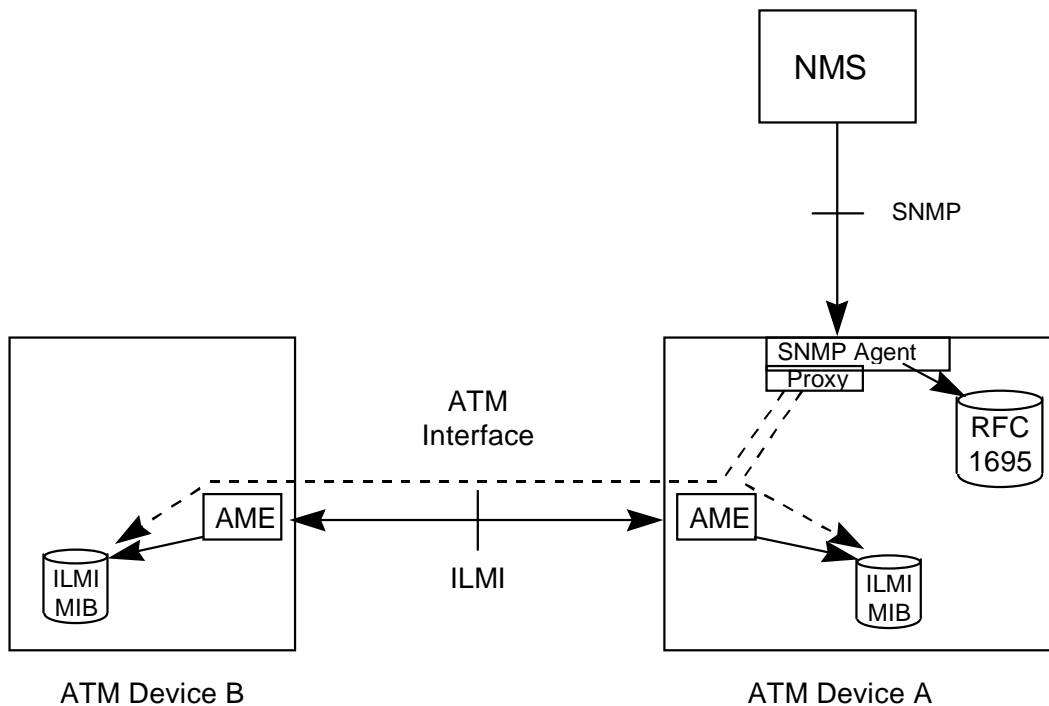
Parallel zu den ATM-Forum-Definitionen existiert ein RFC-Standard [RFC 1695] für das Management von ATM-Geräten mittels SNMP. Dieser Standard der IETF erschien 1994 und definiert eine MIB, die nach der „Structure of Management Information Version 2 (SMIV2)“ [RFC 1442] erstellt wurde. Ist ein Agent eines ATM-Gerätes auf dem UDP-Port 161 mit SNMP ansprechbar und unterstützt er diese MIB, so kann jede im Internet übliche Managementstation, die ebenfalls diese MIB kennt, dieses ATM-Gerät verwalten. Die mit der RFC1695-MIB erreichbare Funktionalität ist global gesehen kleiner als bei ILMI, jedoch gibt es einige zusätzliche Objekte. Die Managementobjekte dieser RFC-Definition sind in sechs Gruppen organisiert. Abbildung 14 gibt dazu einen Überblick.



**Abbildung 14** Objektgruppen der ATM-MIB aus [RFC 1695]

In dem Dokument [ILMI 96] wird im Anhang über den Standard-SNMP-Zugang zu ATM-Daten geschrieben. Es wird dabei ein Proxy-Mechanismus im Zusammenwirken mit der ILMI-MIB und der MIB aus [RFC 1695] vorgestellt.

Das Anliegen des Proxy-Mechanismus ist es, einer Netzwerkmanagementstation, die mittels SNMP bisher nur Agenten der RFC 1695 ansprechen konnte nun durch einen „Dolmetscher“ auch das ILMI-Management zugänglich zu machen. Die Situation mit dem proxy-Agenten zeigt Abbildung 15.



**Abbildung 15** Zugriff auf ILMI-MIB mittels Proxy-Agent [ILMI 96]

Dieser Proxy-Agent ist ein sehr komplexer Prozeß, der zum einen SNMPv1 bzw. SNMPv2 mit der MIB aus [RFC 1695] und zum anderen das ILMI-Protokoll mit der ATM-Forum-MIB beherrschen muß. Dabei arbeitet er bei einem Zugriff auf die Daten des Gerätes A wie ein Agent und beim Zugriff auf das Nachbargerät wie ein Manager im Auftrag des Managementsystems. Die einfachste Realisierung ist dabei die Übergabe der SNMP-Requests an den Proxy-Agenten, der entweder die Anfrage selbst ausführen muß bzw. diese seinem Nachbar einfach weitergibt. Die Hauptentscheidung des originalen SNMP-Agenten ist dabei das Herausfinden ob die Anfrage sich auf ein Objekt der lokalen RFC1695-MIB oder auf eine (welche) ILMI-MIB bezieht. Diese Entscheidung wird dabei aus dem verwendeten Community-

String abgeleitet. Wurde seitens der Managementstation mit SNMPv1 gearbeitet, so braucht bei der Übergabe in den ILMI-Bereich keine Protokollumsetzung vorgenommen werden. Bei Verwendung von SNMPv2 muß dies jedoch sowohl für die Anfrage als auch für die entsprechende Antwort geschehen. Näheres zu dieser Umsetzung ist in [RFC 1908] zu finden.

Aus der Beschreibung des Proxy-Agenten wird klar, welchen großen Aufwand aber auch Nutzen eine solche Managementrealisierung mit sich bringt. Es müssen beide „Protokoll-Welten“ beherrscht und zusätzlich die Übergabeprobleme gelöst werden. Der Lohn dieser Arbeit wäre dann das zentrale Management eines heterogenen Netzes ohne Managementverlust im ATM-Segment. Außerdem steht für die Geräte, die sowohl einen SNMP als auch einen ILMI-Zugang besitzen, die gebündelte Funktionalität beider Standards zur Verfügung.

Im folgenden Abschnitt wird nun der Managementübergang aus dem privaten ATM-Netz zum öffentlichen Netz betrachtet.

### 3.4.3 Die Managementschnittstelle M3

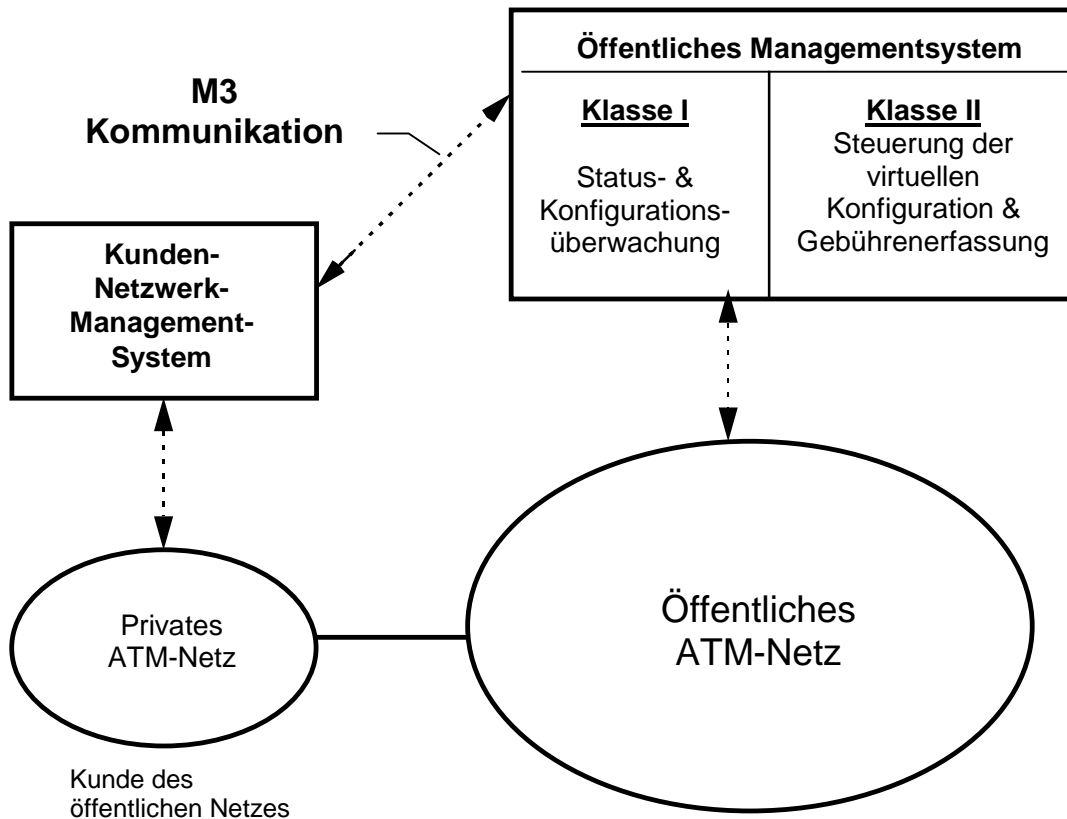
Die Definition der Schnittstelle M3 ist im allgemeinen Managementmodell des ATM-Forums für die Kommunikation eines privaten Managementsystems mit einem öffentlichen vorgesehen. Die Schnittstelle wird im Dokument [M3 94] beschrieben. Darin werden die möglichen Arten der ATM-Netzwerk-Management-Dienste, die ein Betreiber eines öffentlichen Netzes dem Nutzer dieses Netzes anbieten kann, formuliert. Es werden die Anforderungen an das Netzwerkmanagement der Nutzer, die Schnittstellenspezifikation und die Objektdefinitionen angegeben. Ziel dieser Spezifikation ist der Einsatz für das „Customer Network Management (CNM)“ von permanent virtuellen Verbindungen. Die bereitgestellten Informationen erlauben den Nutzern das Leistungs-, Fehler- und Konfigurationsmanagement mittels ihrer ATM-Service-Schnittstellen. Wie bereits erwähnt, kann der Nutzer mittels der M3-Management-Schnittstelle Einfluß auf die Nutzung seines Anteiles des öffentlichen Netzes nehmen.

M3 basiert im Gegensatz zu ILMI auf einer sogenannten „top-down“-Ansicht des Netzwerkes.

Um für die verschiedenen Niveaus der Kundenbedürfnisse seitens der öffentlichen Netzwerkanbieter einen modular erweiterbaren Dienstumfang bieten zu können, werden die Anforderungen für die M3-Schnittstelle in zwei Klassen eingeteilt:

- Die **erste Klasse** umfaßt die Anforderungen, die der öffentliche Netzwerkanbieter für Monitoring-Informationen bezüglich Konfigurations-, Fehler- und Leistungsmanagement des kundenspezifischen Anteils des öffentlichen Netzes erfüllen muß.
- Die **zweite Klasse** beschreibt die notwendigen Funktionalitäten zum Einrichten, Modifizieren und Löschen virtueller Verbindungen, sowie die Bereitstellung von Gebühreninformationen im öffentlichen ATM-Netz. Dabei bezieht man sich auf Verbindungen zwischen physischen Geräten eines privaten Netzes.

Abbildung 16 zeigt ein Übersichtsbild für diese Schnittstellendefinition.



**Abbildung 16** Kunden-Management für private und öffentliche Netze [M3 94]

Die M3 Spezifikation befaßt sich nicht mit dem Management der internen Aspekte des Netzwerkes (z.B. Switches, Steckkarten, Netzwerk-Routing-Tabellen ...).

Um die Anfragen der Kunden über die M3-Schnittstelle im öffentlichen Netz entgegenzunehmen, gibt es dort einen „Customer Network Management (CNM)“-Agenten, der Zugriff auf einen Teil der permanenten virtuellen Verbindungen besitzt und die MIBs der entsprechenden Klasse beherrscht. Jeder Agent ist für genau einen öffentlichen Netzwerkanbieter zuständig.

Nachdem auf die einzelnen Anforderungsgebiete kurz eingegangen wird, folgt die Betrachtung des Transportmechanismus.

### Generelle Voraussetzungen für das M3-Management :

Alle hier aufgeführten Anforderungen sind obligatorisch für einen Betreiber eines öffentlichen Netzes, der M3-Management anbieten möchte. Es werden nur die wichtigsten Punkte genannt, so daß bei einer Implementierung in der Spezifikation der volle Umfang nachzulesen ist. Die Angaben in Klammern entsprechen dem Definitionsstatus:

- Funktionsumfang der Klasse I ist zu bieten. (Erforderlich)
- Klasse II kann angeboten werden. (Optional)
- Für die Funktionen der Klassen I und II muß SNMP als M3-Protokoll verwendet werden. (Erforderlich)
- Die Verwendung von SNMPv1 bzw. SNMPv2 ist optional. (Optional)

- Die M3-MIBs sollen auf dem SNMP-Standard der IETF basieren. (Erforderlich)
- Der öffentliche Netzer hat dafür zu sorgen, daß mittels entsprechendem Sicherheitsmanagement einem Kunden ausschließlich dessen eigene Daten zugänglich gemacht werden. (Erforderlich)
- Der M3-Dienst umfaßt sowohl das Management von „point-to-point-PVCs“, als auch das von ATM-UNI-Schnittstellen. (Erforderlich)
- Die M3-Informationen dürfen nicht älter als die Netzer-spezifische Zeit „T1“ sein. (Erforderlich)
- Eine ähnliche, ebenfalls spezifische Zeit „T2“ darf zwischen Auftreten und Melden von bestimmten Netzwerkeignissen nicht überschritten werden. (Erforderlich)
- Der M3-Agent muß die in der Spezifikation vermerkten SNMP-Gruppen unterstützen und das Eintreffen von SNMP-Anfragen mitprotokollieren. (Erforderlich)

### **Voraussetzungen für die Klasse I des M3-Managements :**

Die folgenden Anforderungen sind für die Bereitstellung der Monitoring-Informationen für das Konfigurations-, Fehler- und Leistungsmanagement erforderlich:

- Klasse I besitzt den SNMP-Status „read-only“. (Erforderlich)
- Es werden die Leistungsparameter der ATM-Schicht und der Physikalischen Schicht bereitgestellt. (Erforderlich)
- Das gleiche gilt für Konfigurations- und Statusinformationen dieser Ebenen und des „ATM Level Virtual Path Link“. (Erforderlich)
- Der M3-Agent informiert den Kunden über jeden Auf- und Abbau einer Kunden-UNI. (Erforderlich)

### **Voraussetzungen für die Klasse II des M3-Managements :**

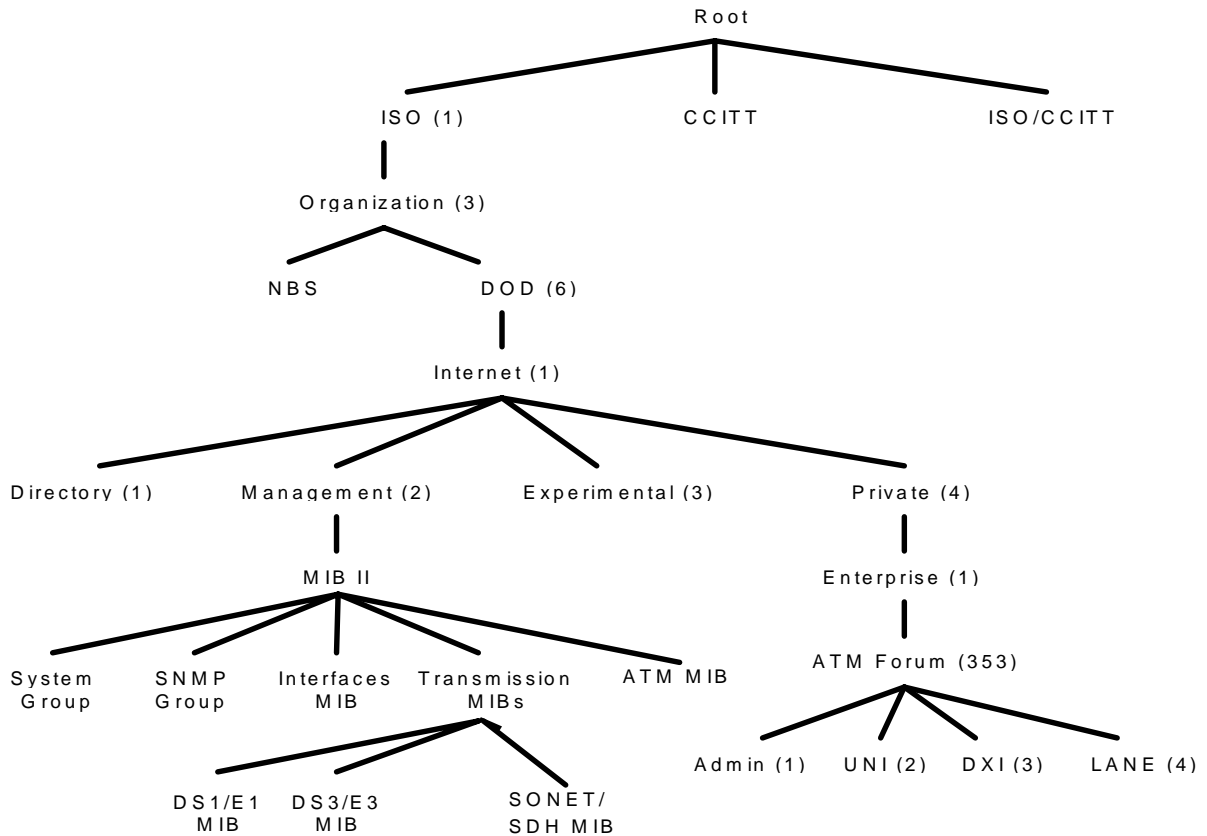
Die Anforderungen der Klasse II beziehen sich auf den Umgang mit virtuellen Verbindungen und den Gebühreninformationen.

Wenn die Klasse II implementiert wird, so muß bereits die Klasse I existieren. Die Funktionalität der zweiten Klasse wurde in drei Gruppen aufgeteilt. An dieser Stelle soll die Aufzählung dieser Gruppen zur Information genügen, da die jeweiligen Anforderungen wenig zum globalen Verständnis beitragen:

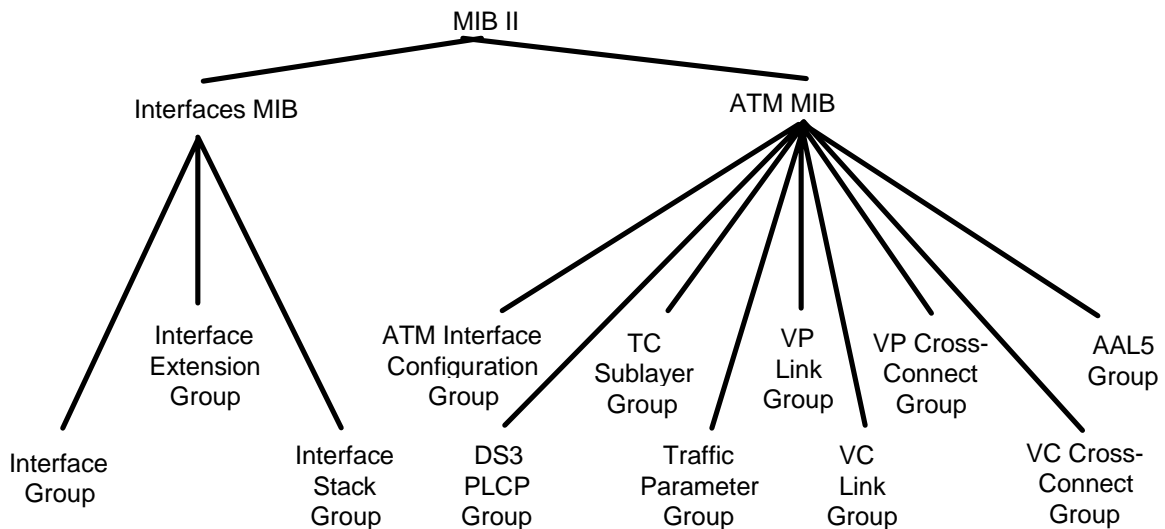
- ATM-Level-Gruppe.
- VPC / VCC-Gruppe.
- Verkehrsgruppe.

Jede der beiden Klassen des M3-Standards besitzt ihre eigene MIB. Diese sind jedoch sehr ähnlich und unterscheiden sich hauptsächlich darin, daß die Klasse II einen schreibenden Zugriff auf die Managementobjekte gewährt.

Um einen groben Überblick zu erhalten, zeigt das folgende Bild die M3-relevanten Gruppen von MIB-Objekten und deren Einordnung im MIB-Baum. Zusätzlich sind zwei wichtige Gruppen der MIB II etwas detaillierter aufgeführt.



**Abbildung 17** MIB-Baum mit M3-relevanten MIB-Gruppen [M3 94]

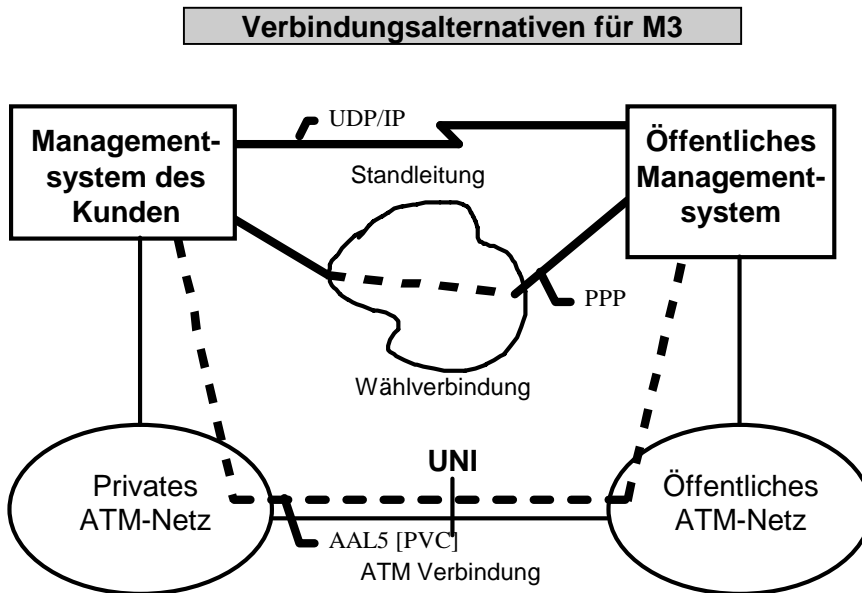


**Abbildung 18** Auszug aus der MIB II des IETF-RFC1213 [M3 94]

Anhand der in Abbildung 17 und Abbildung 18 dargestellten MIB-Gruppen bietet sich ein zweiter Zugang zur Funktionalität der M3-Schnittstelle. Es läßt sich daraus grob abschätzen, welche Managementobjekte an dem Übergang zwischen einem privaten und einem öffentlichen Netz abgefragt bzw. verändert werden können. Diese Darstellung ist damit eine Art Spiegelbild der gebotenen Schnittstellenfunktionalität.

Abschließend werden die möglichen Transportverbindungen für die Managementinformationen aufgezeigt. Es gibt typischerweise drei Arten von Verbindungen auf denen das M3-Protokoll aufsetzen kann. Abbildung 19 zeigt eine Übersichtsgrafik zu diesen Verbindungen.

Bei der eigenen Auswahl der günstigsten Variante spielen hauptsächlich die Punkte Kosten, Verkehrsaufkommen, Zuverlässigkeit, Sicherheit und Verfügbarkeit eine Rolle. Es gibt daher auch keine feste Vorschrift des zu wählenden Verbindungsverfahrens.



**Abbildung 19** M3 - Verbindungsalternativen mit typischen Protokollen [M3 94]

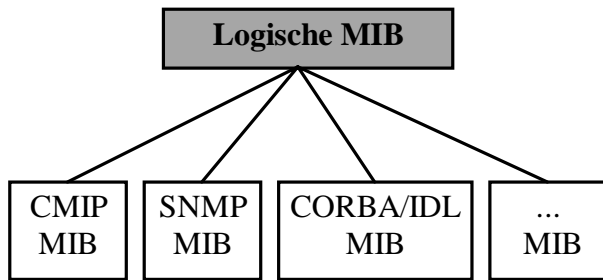
Für die Kommunikationswege außerhalb des ATM-Netzes spricht die Tatsache, daß das eigentliche Netz nicht belastet wird, das Management auch bei Netzausfall verfügbar bleibt und gerade bei Verwendung des UDP-Protokollstacks auf bestehende SNMP-Implementierungen teilweise zurückgegriffen werden könnte. Nachteilig ist der zusätzliche technische Aufwand.

Dem Management in öffentlichen ATM-Netzen wendet sich nun der folgende Abschnitt zu.

#### 3.4.4 Die Managementschnittstelle M4

Die Definition der Schnittstelle M4 ist im allgemeinen Managementmodell des ATM-Forums für die Kommunikation eines öffentlichen Managementsystems mit dem öffentlichen ATM-Netz vorgesehen. Die Schnittstelle wird in mehreren Dokumenten [M4 94], [M4 CMIP 95] und [M4 NV 96] beschrieben. Das Basisdokument ist [M4 94] mit dem Titel „M4 Interface Requirements and Logical MIB“. Das Anliegen dieses Standards ist die Beschreibung der funktionellen Erfordernisse für das Management von ATM-Netzen. Es werden dabei die Managementgebiete Konfigurations-, Fehler-, Leistungs- und Sicherheitsmanagement berührt.

M4 basiert wie M3 auf einer sogenannten „top-down“-Ansicht des Netzwerkes.



**Abbildung 20** Ableitung protokollspezifischer MIBs

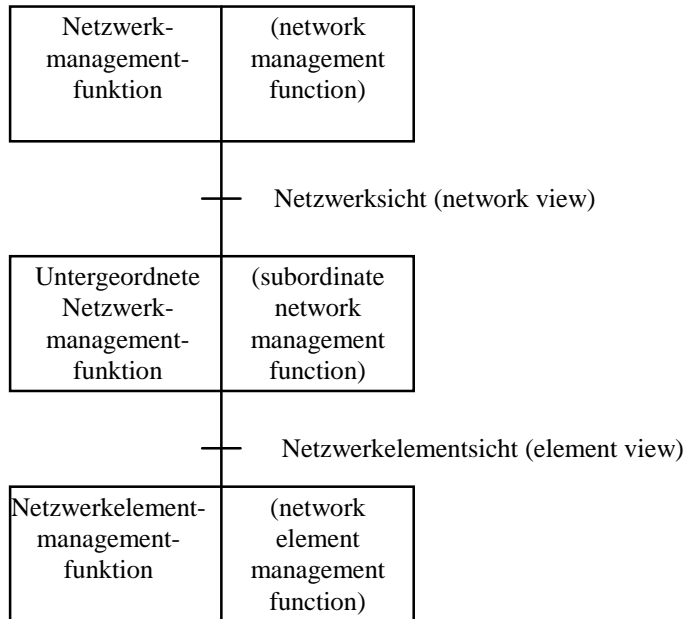
Wie der Titel bereits sagt, wird eine logische MIB im Standard angegeben. Die nähere Beschreibung „logisch“ hat dabei folgenden Hintergrund. Die für das Management benötigten Daten sind bei den verschiedenen Managementstandards (CMIP, SNMP,...) nahezu identisch. Trotzdem erfordert jeder dieser Standards eine andere Darstel-

lung und Verwaltung. Ziel dieser logischen MIB ist es daher, eine universelle Vorlage zur leichten Ableitung protokollspezifischer MIBs zu sein. Damit soll auch die funktionelle Gleichwertigkeit eines ATM-Managements mit CMIP, SNMP usw. an dieser Schnittstelle erreicht werden. Abbildung 20 stellt dieses Prinzip der logischen MIB vereinfacht dar.

Um den Grundumfang der Schnittstelle M4 zu beschreiben, beziehen sich die nachfolgenden Erläuterungen auf das Dokument [M4 94]. Die Spezifikation [M4 CMIP 95] stellt diese M4-Definitionen in aufbereiteter Form für das OSI-Management inclusive der GDMO-Beschreibung der Objekte vor.

Eine Erweiterung des M4-Standards wird im Dokument [M4 NV 96] vorgenommen. Mit der Definition einer speziellen strukturellen Ansicht der ATM-Netzwerk-Ressourcen, dem sogenannten „Network View“, soll ein flexibleres Management erreicht werden. Im Zuge dieser Festlegungen wird ein Satz von sogenannten „Network-View-Managed-Objects“ angegeben, auf dem ein hierarchisches Management (Subnetzaufteilungen...) aufsetzt. Auch hier erfolgt die Objektdefinition in Form der logischen MIB. Neben dieser abstrakteren Sicht auf das Netzwerk bleibt eine sogenannte „Netzwerkelementsicht“ weiterhin erhalten. Ein Managementeingriff kann somit in der Netzwerkhierarchie beginnen und im Subnetzbereich auf eine einzelne Komponente zugreifen. Auf diese Weise wird das Management großer (oft auch räumlich weit ausgedehnter) öffentlicher Netze erleichtert. Die Erweiterung des M4-Standards spricht die folgenden funktionellen Gebiete des ATM-Netzwerkmanagements an:

- Transport-Netzwerk-Konfigurations-Bereitstellung (einschließlich Subnetz- und Verbindungsunterstützung).
- Transport-Netzwerk-Verbindungsmanagement (einschließlich Einrichtung, Reservierung und Veränderung von Subnetzverbindungen, von Einzelverbindungen, von Pfadverbindungen und ganzer Segmente).
- Fehlermanagement (einschließlich Korrelation, Lokalisierung und Meldung der Fehler von Geräten und Verbindungen, sowie Loopback-Testung).
- Leistungsmanagement (einschließlich Stauüberwachung, sowie Verbindungs- und Segmentüberwachung).
- Abrechnungsmanagement.
- Sicherheitsmanagement.



In Abbildung 21 wird die oben angesprochene hierarchische Struktur gezeigt.

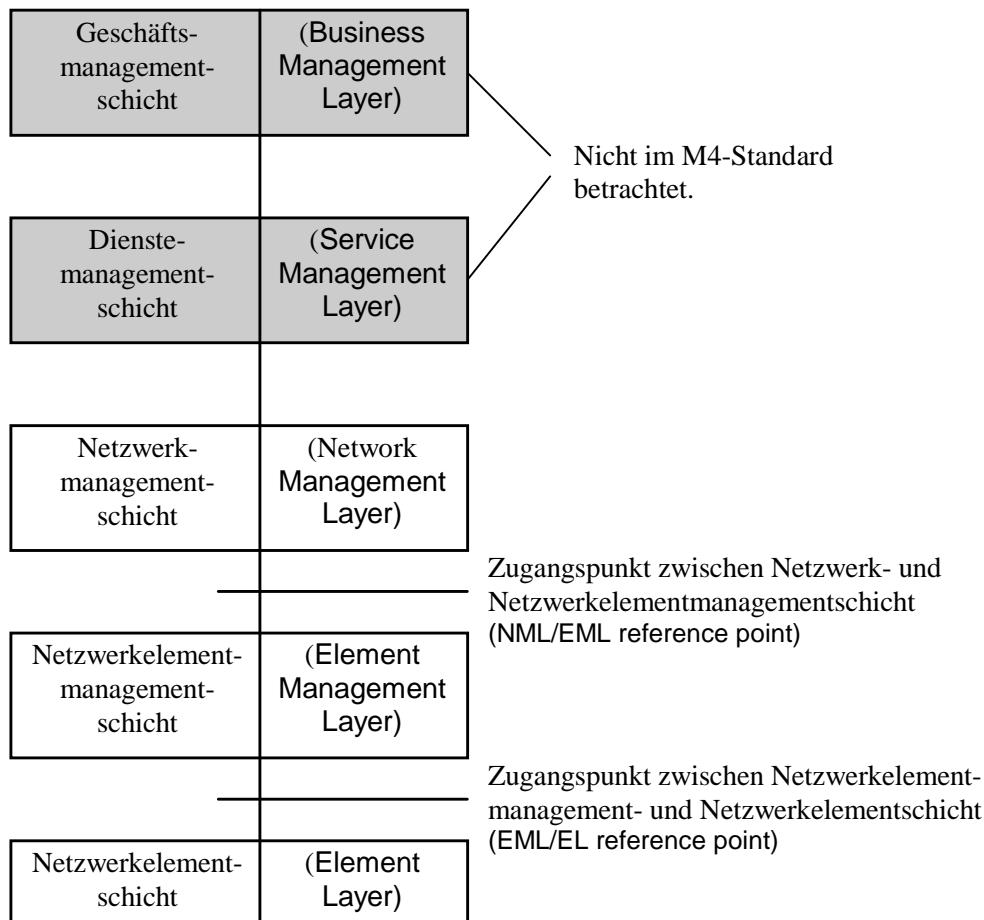
**Abbildung 21** M4 Network-View [M4 NV 96]

Nach diesen kurzen Ausführungen zur Erweiterung der M4-Spezifikation soll nun auf die Grunddefinition im Standard [M4 94] näher eingegangen werden.

Wie im Standard angeführt, werden die folgenden ATM-Netzwerkmanagement-Gebiete an der Schnittstelle M4 gehandhabt:

- Konfigurationsmanagement
  - ATM-Netzelement-Konfiguration, -Identifikation und Veränderungsmeldung.
  - UNI-, BICI-Konfiguration.
  - Durchschaltung von VPC und VCC, sowie Konfiguration von VPCs bzw. VCCs.
  - Konfiguration von VPC-/VCC-Segmentendpunkten.
  - Systemmanagementsteuerung (z.B. Filterung automatisch generierter Meldungen).
- Fehlermanagement
  - Automatisches Melden von ATM-Netzwerkelement-Fehlern, physikalischen Fehlern und ATM-Verbindungsfehlern.
  - Auslösen von ATM-OAM-Zell-Loopback-Tests.
- Leistungsmanagement
  - Leistungsüberwachung auf physischer Ebene (z.B. SONET, DS3, ...).
  - Leistungsüberwachung auf Übertragungs-Anpassungs-Ebene.
  - Protokollüberwachung auf ATM-Ebene und
  - Überwachung von UPC/NPC-Verletzungen.
- Sicherheitsmanagement ist prinzipiell geplant.

Die Gegebenheiten der M4-Schnittstelle sind derzeit nur für die Unterstützung permanenter virtueller Verbindungen vorgesehen. Die Anwendung auf geschaltete Verbindungen ist späteren Spezifikationen vorbehalten.



**Abbildung 22** Schichtarchitektur aus [ITU M.3010]

Die M4-Schnittstellendefinition nimmt Bezug auf ein fünfschichtiges Modell nach [ITU M.3010], welches in Abbildung 22 dargestellt ist. Von den darin aufgeführten fünf funktionalen Schichten werden nur die unteren drei einschließlich zweier funktioneller Schnittpunkte vom M4-Standard überdeckt.

Die gesamte M4-Definition befaßt sich mit den Interaktionen zwischen diesen drei Schichten, welche die nachfolgenden Bedeutungen besitzen:

#### **Network Management Layer (NML)**

Diese Schicht befaßt sich mit dem Management aller ATM-Netzwerk-Elemente, die jeweils auf Element-Management-Niveau dargestellt sind. Dabei wird das Innenleben der Elemente nicht berücksichtigt. Die Funktionen der Schicht ermöglichen das Management des „end-to-end telecommunications network“. Es können Netzwerkkapazitäten für die Dienstleistung zum Kunden durch steuernde und koordinierende Eingriffe bereitgestellt bzw. geändert werden. Zusätzlich werden den höheren Schichten Leistungs-, Verfügbarkeits- und Benutzungsinformationen bereitgestellt.

#### **Element Management Layer (EML)**

Auf diesem Niveau erfolgt das Management eines jeden Netzwerkelementes. Es werden die vom „Element Layer“ gebotenen Funktionen abstrakter zusammengefaßt.

## Element Layer (EL)

Diese Schicht bietet die Basiskommunikationsdienste je nach Technologie, Hersteller, Netzwerkressource und Netzwerkelement.

Entsprechend der oben aufgeführten ATM-Management-Gebiete erfolgt in der M4-Spezifikation eine ausführliche Beschreibung der für die jeweilige Managementart notwendigen Funktionen und Meldungen. Näheres dazu ist im Dokument [M4 94] nachzulesen.

Es folgt eine Betrachtung zur protokollunabhängigen „Management Information Base“ der M4 Spezifikation.

Eine Charakteristik dieser MIB ist die Untergliederung in sogenannte „Managed Entities“, welche abstrakte Abbilder von Ressourcen und Diensten eines ATM-Netzwerkelementes sind. Die erhöhte Abstraktion ist für die bereits beschriebene Anpassungsfähigkeit an bestehende Managementprotokolle notwendig. Die spätere Beschreibung dieser „Management-Eintrittspunkte“ umfaßt die Angaben: Zweck, Attribute, mögliche Operationen, generierte Meldungen und die Beschreibung der Beziehung zu anderen „Managed-Entities“.

Die gesamte MIB-Definition erstreckt sich nun darin, die in Abbildung 23 aufgelisteten „Managed-Entities“ in der eben genannten Form zu definieren.

- |   |  |
|---|--|
| • Alarm Record                              | • Multipoint Bridge                            |
| • Alarm Severity Assignment Profile         | • Physical Path Termination Point              |
| • ATM Cell Protocol Monitoring Current Data | • Plug-in Unit                                 |
| • ATM Cell Protocol Monitoring History Data | • Software                                     |
| • ATM Cell Protocol Monitoring Log Record   | • State Change Record                          |
| • ATM Cross Connection                      | • TC Adaptor                                   |
| • ATM Cross Connection Control              | • TC Adaptor Protocol Monitoring Current Data  |
| • ATM NE                                    | • TC Adaptor Protocol Monitoring History Data  |
| • Attribute Value Change Record             | • Threshold Data                               |
| • BICI                                      | • UNI  |
| • Equipment                                 | • UPC/NPC Disagreement Monitoring Current Data |
| • Equipment Holder                          | • UPC/NPC Disagreement Monitoring History Data |
| • Event Forwarding Discriminator            | • VCC Termination Point                        |
| • Latest Occurrence Log                     | • VCL Termination Point                        |
| • Managed Entity Creation Log Record        | • VPC Termination Point                        |
| • Managed Entity Deletion Log Record        | • VPL Termination Point                        |

**Abbildung 23** „Managed Entities“ der logischen MIB [M4 94]

Im Anschluß an dieses Kapitel müßte sich eine Betrachtung der noch verbleibenden Schnittstelle M5 anschließen. Wie aber bereits gesagt wurde, ist derzeit noch kein Standard zu M5 verfügbar. Prinzipiell beschreibt diese Schnittstelle die Managementkommunikation zwischen zwei öffentlichen Netzwerkmanagementsystemen. Mit dem Abschluß der Standardisierungsarbeiten ist laut ATM-Forum Anfang 1998 zu rechnen.

### 3.5 Überlegungen zur Implementation eines Management-Agenten

Bei der Implementation einer Netzwerkmanagementunterstützung für eine ATM-NIC besteht das Problem, die in dem ILMI-Standard definierten „Managed Objects“ der ILMI-MIBs mit den systemspezifischen Daten zu verbinden. Es werden dabei Zugriffe auf den Treiber der Einsteckkarte bzw. auf Systemvariablen notwendig. Der zu implementierende Agent muß diese Zugriffsmechanismen beherrschen und damit den Datenverkehr zwischen der „MIB-Datenbank“ und dem System realisieren. Aus vorhandenen Implementationen im Bereich des Internet-Managements ist dazu folgendes zu sagen. Die Beschreibung der Managementobjekte in den MIBs wird bei jeder Implementation in eine eigene Darstellung umgewandelt. Es bestehen daher sowohl für Agenten als auch für Manager entsprechende „Lademöglichkeiten“ für MIBs, die einer Übersetzung und Eingliederung in das implementationspezifische Format entsprechen. Diese interne Verwaltung und Repräsentation der Objekte sei hier als „MIB-Datenbank“ bezeichnet.

Für die Realisierung der bidirektionalen Bindung der Objekte an Systemvariable und Steckkartendaten gibt es keine Vorschriften seitens der Standards und sehr geringe Informationen seitens der einzelnen vorhandenen Implementationen. Der Zugriff auf diese „Kerndaten“ der verschiedenen Multiuser- / Multitasking-Betriebssysteme ist in den vorhandenen Agenten auch verschieden implementiert. Als Beispiele für diese Zugriffsmechanismen seien hier das „Proc-Filesystem“ bei dem Betriebssystem „LINUX“ und der Zugang über den „Kmem-Gerätetreiber“ bei dem Betriebssystem „SUN-OS“ genannt.

Für Abfragen bezüglich der Systemgruppe der MIB II aus [RFC 1213] ist diese Herangehensweise sicher zweckmäßig. Für die ATM-Treiber-spezifischen Zugriffe ist der Weg über sogenannte generische Systemrufe möglich. Diese Systemrufe, bei mehreren Betriebssystemen unter dem Begriff „ioctl-Funktionen“ bekannt, bieten einen äußeren Zugang zu Daten und Funktionen im Betriebssystemkern. Wegen ihrer vielseitigen Verwendung wird anhand eines Beispiels einmal näher auf diese Funktionen eingegangen.

Zu dem Betriebssystem LINUX gibt es eine Erweiterung (ATM on LINUX) von W. Almesberger. Darin wird eine Programmierschnittstelle („LINUX ATM API“) realisiert, die ATM-spezifische Systemrufe bereitstellt. Es werden an dieser Schnittstelle auch einige „ioctl-Funktionen“ geboten, die für eine Agentenimplementierung genutzt werden könnten. In der Schnittstellenbeschreibung „Linux ATM API Draft, version 0.4“ [Almesberger 96] werden ab Seite 31 diese Funktionen kurz erläutert. Unter der Überschrift „Administrierungsfunktionen“ werden einige Aufrufe zur Interface- und Verbindungskonfiguration, sowie für die physikalische Ebene mehr oder weniger genau aufgeführt. Der Datenaustausch erfolgt bei den „ioctl“-Funktionen über zwei untypisierte Puffer definierbarer Länge.

Folgende „ioctl“-Funktionen sind in der Applikationsschnittstellenbeschreibung aufgeführt:

#### **Interface Configuration**

- ATM\_GETNAMES (Liste verfügbarer ATM-Interfaces).
- ATM\_GETTYPE (Typenbezeichnung eines speziellen Interfaces).
- ATM\_GETESI (Endsystemkennung eines Interfaces).
- ATM\_GETCIRANGE (Lesen der verfügbaren Bits für VPI- und VCI-Werte).
- ATM\_SETCIRANGE (Schreiben der verfügbaren Bits für VPI- und VCI-Werte).
- ATM\_GETSTAT (Lesen der „AAL Level“-Statistik; für AAL 0, 3/4 und 5 werden korrekte und fehlerhafte Sende- bzw. Empfangsblöcke gezählt; Zusätzlich werden verworfene Blöcke angegeben (CRC-Fehler).
- ATM\_GETSTATZ (siehe GETSTAT jedoch mit Rücksetzen der Zählerstände nach dem Lesen).

### Address Configuration

- ATM\_GETADDR (Liste der ATM-Adressen eines Interfaces).
- ATM\_RSTADDR (Setzt diese Liste zurück).
- ATM\_ADDADDR (Fügt eine Adresse am Ende der Liste an).
- ATM\_DELADDR (Löscht eine Adresse aus der Liste).

### Connection Configuration

- SO\_AALTYPE (Abfrage des AAL-Typen eines Sockets).
- SO\_SETCLP (Setzen der Zell-Verlust-Priorität).
- SIOCGSTAMP (Eintreffzeit der zuletzt empfangenen SDU).

### Physical Layer

- SONET\_GETSTAT.
- SONET\_GETSTATZ.
- SONET\_SETDIAG.
- SONET\_CLRDIAG.
- SONET\_GETDIAG.
- SONET\_SETFRAMING.
- SONET\_GETFRAMING.
- SONET\_GETFFRSENSE.

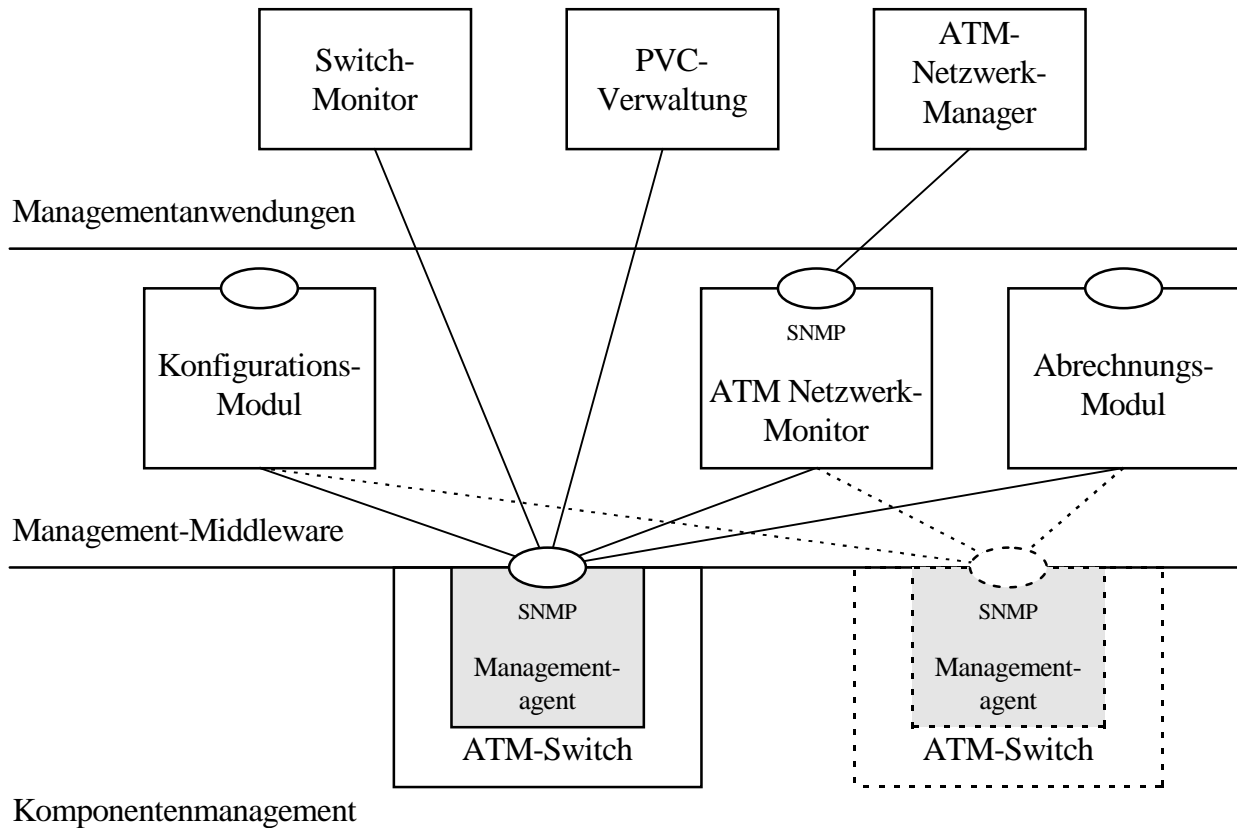
Zusätzlich zu diesen aufgelisteten Funktionen sind weitere Funktionsdefinitionen in den sogenannten „Header-Files“, die zum Quelltext der LINUX-Erweiterung gehören, enthalten. Diese können z.B. mittels der HTML-Referenz eingesehen werden.

Bei einer Implementation ist für jedes Managementobjekt die jeweils günstigste Zugriffsmethode auszuwählen und zu programmieren.

Einige grundlegende Betrachtungen zu Management-Agenten verbunden mit einem eigenen Management-Architektur-Modell werden in [Wiltfang 97] vorgestellt. Ausgangspunkt der Betrachtungen ist die Tatsache, daß heutige ATM-Geräte SNMP häufig im Out-of-Band-Management einsetzen und dabei sehr unterschiedliche Managementinformationen anbieten. Häufig sind die Managementzugänge über Ethernet oder serielle Schnittstellen realisiert. Neben SNMP mit herstellerspezifischen MIBs ist das Management mittels Telnet weit verbreitet. Bei diesen Telnetzugängen muß man sich jedoch bei jedem Gerät in das entsprechende textuelle Menüsystem einarbeiten.

In dem Artikel wird nun eine allgemeine Managementschnittstelle mittels eines Proxy-Agenten vorgeschlagen, der nach außen hin SNMP mit einer speziellen MIB bietet und sich zum Gerät hin um die jeweilige Ausführung der Operationen bemüht. In diesem Vorgehen, ist für jedes Gerät (hier wurde das Management von ATM-Switches betrachtet) ein solcher Agent zu implementieren. Die beschriebenen Agenten laufen als Daemon-Prozesse auf einem Rechner und realisieren aus ATM-Netz-Sicht durch ihre Zugriffe über die Out-of-Band-Schnittstellen ein scheinbares In-Band-Management. Nötigenfalls muß jedoch der Agent dabei den Ablauf einer Telnet-Sitzung zur Ausführung der SNMP-Requests durchführen.

In dem Artikel wird für das ATM-Management ein eigenes Schichtenmodell aufgestellt, das in Abbildung 24 dargestellt ist.



**Abbildung 24** Ausgewählte Instanzen der ATM-Managementarchitektur [Wiltfang 97]

Die drei enthaltenen Schichten haben folgende Bedeutung:

#### **Komponentenmanagement**

In dieser Schicht wird durch hersteller- und produktversionsspezifische Managementzugriffe den darüberliegenden Schichten ein komponentenunabhängiger Managementzugriff auf eine einzelne Netzwerkkomponente ermöglicht.

#### **Management-Middleware**

Hier werden komplexere Dienste für die obere Schicht angeboten, die sich auf eine oder mehrere ATM-Komponenten beziehen. Als Beispiele für solche Dienste sind das Abrechnungs-, Konfigurations-, Fehler- und Leistungsmanagement im Artikel angeführt.

#### **Managementanwendungen**

Diese Programme, die in grafischer oder textueller Form mit dem menschlichen Benutzer interagieren nutzen die Dienste der darunterliegenden Schichten für das Management der verschiedenen ATM-Geräte. Auch bereits vorhandene kommerzielle Managementsysteme können nun wegen der allgemeinen SNMP-Schnittstelle eingesetzt werden.

Nähere Informationen über dieses Modell und die im Bild angeführten Instanzen sind im Dokument [Wiltfang 97] zu finden.

Als Unterstützung der Implementierungsarbeiten folgt im nächsten Abschnitt eine kurze Erklärung zum ASN.1-Standard. Da diese Notation in jedem der beschriebenen Managementprotokolle verwendet wird, ist es zweckmäßig, sich eingehender damit zu beschäftigen.

### 3.6 Kurzreferenz zu ASN.1

Die „Abstract Syntax Notation One (ASN.1)“ ist ein OSI-Standard, der 1987 als „International Standard 8824“ festgeschrieben wurde.

Entwickelt wurde ASN.1 im Rahmen des OSI-Management-Frameworks für die Kodierung der Managementnachrichten. Nunmehr nutzen alle bekannten Managementprotokolle diese Notation, weshalb in diesem Abschnitt auch die darin festgelegten Kodierungsregeln, die sogenannten „Basic Encoding Rules (BER)“, kurz dargestellt werden.

Wie bereits erwähnt, sollte bei einer Implementierung auf vorhandene Hilfsroutinen zurück gegriffen werden, die die Datenumwandlung in ASN.1-Format vornehmen. Das Verständnis dieses Formates ist deswegen wichtig, da neben der übermittelten Management-PDUs auch die MIB-Definitionen entsprechend dieser Notation vorliegen.

Im ASN.1 Standard stehen eine Vielzahl von Typen zur Verfügung. Typen beginnen stets mit einem Großbuchstaben. Es können neue Typen deklariert werden, die entweder vorhandene Typen zusammenfassen oder völlig neu angegeben werden (genannt „Textual Convention“). Einige der Grundtypen sind nachfolgend aufgezählt:

BOOLEAN.	OBJECT IDENTIFIER.
BIT.	NULL.
INTEGER.	ANY.
REAL.	
BIT STRING.	SEQUENZE.
OCTET STRING.	SEQUENZE OF.
Printable String.	

Die Kodierungsregeln (Basic Encoding Rules) schreiben folgende Grundstruktur fest.

ID = TAG	Länge	Inhalt	
1	1	1	Byte

Diese Struktur trifft für alle einfachen Typen zu. Mittels SEQUENZE (Struktur) und SEQUENZE OF (FELD) werden oben erwähnte zusammenge-

faßte Typen gebildet. Ihr Kodierungsbild beinhaltet somit beliebig oft die gezeigte Grundstruktur.

ID des zusammengesetzten Datentypes	Länge des zusammengesetzten Datentypes	Grunddatentyp 1	Grunddatentyp 2	usw.
-------------------------------------	--	-----------------	-----------------	------

Die Unterscheidung der Datentypen erfolgt im sogenannten „ID-Feld“.

00	universal Tag (global)		
01	application wide Tag		
10	context specific Tag		
11	private use (enterprise specific)		
Bit 7	5	4	0
	Klasse	Primitive/Constructed	Tag

(Hier das Beispiel ‘universal Tag ⇒ 5 Bit - Tag’)

Integer-Zahlen besitzen z.B. einen Tag von 2 und Octet-Strings den Tag 4. Da beide sogenannte „universal Tags“ sind und Integer bzw. Octet String Grunddatentypen (also Status primitive) darstellen, sind hier ID-Wert und Tag-Wert identisch.

Die fortlaufende Tag-Nummerierung (unsigned integer) der ASN.1-Datentypen besitzt zwei Darstellungen.

$0 \leq \text{Tag} \leq 30 \Rightarrow$

Klasse	P/C	5 Bit - Tag
--------	-----	-------------

$\text{Tag} \geq 30 \Rightarrow$

Octet

	0	1	...	n
Klasse	P/C	11111	1xxxxxxx	... 0xxxxxxx

Das Längensfeld wird bis 128 Bytes als 

0	Länge (7 Bit)
---	---------------

 angegeben und bei größeren

Zahlen entsprechend nachfolgender Struktur.

Octet

	0	1	...	n
1	Länge der Länge	MSB	...	LSB

Die häufigsten Datentypen und ihre Kodierung sind :

- BOOLEAN 

ID (0)	Länge =1	true/false
--------	----------	------------

 .
- INTEGER 

ID (2)	Länge (1)	ZK-Zahlenwert
--------	-----------	---------------

 . [ Werte in () sind typische Werte ]
- BIT STRING 

ID	Länge	zu vollen Octets aufgefüllter Bitstring
----	-------	---

 .
- OCTET STRING 

ID (4)	Länge	Octets des Strings
--------	-------	--------------------

 .
- NULL 

ID	0
----	---

 .

Weitere Informationen zu diesem Thema sind in [Gora 92] und in [Neuendorf 93] enthalten.

## 4 Netzwerkmanagementapplikationen

Netzwerkmanagementsysteme bestehen häufig aus mehreren Programmen. Jedes dieser Programme bietet spezielle Möglichkeiten des Managements und deckt damit einen Teil des gesamten Netzwerkmanagemenauftrages ab. Der Grund für diesen Aufbau ist die kombinierte Anwendung der verschiedenen Netzwerkprotokolle (UDP, TCP, ICMP, CMIP, SNMP,...), um auf breiter Basis Zugang zu den Netzkomponenten zur Verfügung zu haben. Dies ist teilweise historisch so gewachsen aber auch durch die Vielgestaltigkeit des herstellerspezifischen Managements der Netzkomponenten bedingt. Es existieren deshalb sogenannte Managementplattformen, die häufig eine globale Verwaltung der Netzwerkstruktur realisieren und zum Management der einzelnen Komponente auf das jeweils angemessene Programm im Framework zurückgreifen. Das Standard-Netzwerkmanagement entspricht dabei dem konsistenten Zugriff auf Managed Objects bei den Agenten der Netzwerkkomponenten. Bekannte kommerzielle Managementplattformen sind NETVIEW von IBM, POLYCENTER von DEC und OpenView von HP.

Eine noch sehr junge Bestrebung ist die Strategie zum Management verteilter Systeme, die unter dem Namen „Distributed Management Environment“ bekannt ist. Es werden darin Dienste angeboten, die weit über den allgemein üblichen Netzwerkmanagementumfang hinaus gehen. Dies sind z.B.:

- Licence Management Service (Verwaltung von Software-Lizenzen in einem verteilten System).
- Software Management Service (Verteilung, Installation und Verwaltung von Software in einem verteilten System) und
- Printing Service (Einrichten und Verwalten eines verteilten Druckdienstes).

Im nichtkommerziellen Bereich gibt es mehrere kleinere Lösungsansätze für das Netzwerkmanagement. Die Palette reicht hier von „MIB-Browsern“, die in jedem Web-Browser lauffähig sind über die Scriptspracherweiterung Scotty mit der Oberfläche Tkined bis hin zu reinen C-Implementationen, die z.B. spezielle Shell-Kommandos für das Management bieten.

Im Rahmen dieser Studienarbeit wurde das sehr umfangreiche kommerzielle System „HP OpenView“ und die freie Implementierung „Scotty/Tkined“ näher betrachtet.

Das Managementframework der Firma Hewlett Packard heißt „HP OpenView“. Es ist das am meisten bekannte und verbreitete kommerzielle Managementsystem, welches auch innerhalb der Universität zum Verwalten des Universitätsnetzes eingesetzt wird. Eine ausführliche Vorstellung dieses Managementsystems ist der HTML-Referenz zu entnehmen.

Das Script-basierte Managementsystem „Scotty / Tkined“ wurde hauptsächlich von Herrn Schönwälder im Rahmen der Projektarbeiten an der TU Braunschweig erarbeitet. Scotty ist eine Erweiterung der „Tool Command Language (Tcl)“ und stellt netzwerkmanagementrelevante Funktionen zur Verfügung. Die darauf aufsetzende Managementoberfläche „Tkined“ benutzt das „Toolkit (Tk)“ für die Darstellung der Oberflächenfenster im X11-Windowssystem. Sowohl Scotty als auch Tkined sind in den Quellen vorhanden. Die dabei verwendete Programmierung in der Scriptsprache ist sehr übersichtlich und leicht nachvollziehbar. Eine Erweiterung von Tkined um eigene Managementscripts ist deshalb gut realisierbar.

Näheres zu diesem System bietet ebenfalls die [HTML-Referenz](#).

## 5 Zusammenfassung

Das Gebiet des Netzwerkmanagements ist in den letzten Jahren verstärkt beachtet worden. Es sind eine Vielzahl von Standards erstellt und viele Lösungsansätze diskutiert worden. Ergänzt durch die bereits vorhandenen Managementbemühungen der einzelnen Hersteller existiert derzeit ein unübersichtliches und kompliziertes System von Managementansätzen.

Da das Gebiet des Netzwerkmanagements in Zukunft immer größere Bedeutung erlangen wird, wird sich der Trend zu herstellerunabhängigen standardisierten Managementarchitekturen und -protokollen durchsetzen. Im Moment ist der SNMP-Standard als de-facto-Standard des Internets bereits etabliert und findet in anderen Netzwerken ebenfalls große Akzeptanz. Seitens des ATM-Forums existiert daher auch ein eigenes ATM-Management-Modell, welches in den Managementschnittstellen auf SNMP basiert. Leider ist dort bisher nur SNMPv1 mit seinen Sicherheitslücken vertreten. Derzeit laufen jedoch bereits Arbeiten des ATM-Forums zu einem entsprechenden Sicherheitsmodell. Für den Laborbetrieb könnte zum jetzigen Zeitpunkt folgende stark einschränkende aber leicht zu implementierende wirksame Übergangslösung für diese Problem eingesetzt werden. Bei fester Vorgabe der Adresse der Managementstation senden die Agenten der Netzkomponenten stets ihre Antwort-Nachrichten zu dieser Station und bearbeiten auch nur Anfragen, die von dieser Station gesendet wurden. Damit wird gewährleistet, daß durch Fremdzugriffe weder Aktionen ausgelöst, noch Informationen abgefragt werden können.

Für die Implementierung einer Netzwerkmanagementunterstützung einer ATM-NIC entsprechend des üblichen Internet-Managements müßte die Realisierung der MIB aus [RFC 1695] und die SNMPv1- bzw. SNMPv2-Kommunikation über den UDP-Port 161 bereitgestellt werden. Dieser Zugang wäre mittels der Festlegungen zu „IP over ATM“ [RFC 1483] und [RFC1577] zu realisieren. Um dies zu umgehen, wird der ILMI-Standard implementiert. Der Managementzugriff erfolgt dabei mit SNMPv1 über AAL5 (VPI=0, VCI=16) in Verbindung mit der MIB-Definition der ILMI-Spezifikation.

Die Verwendung eines Proxy-Agenten zur Unterstützung beider Managementzugänge wurde bei den Betrachtungen zur ILMI-Schnittstelle besprochen. Diese Variante stellt hohe Ansprüche an die Implementierung, bietet dafür aber eine universelle Ansprechbarkeit des Agenten seitens allgemein üblicher Netzwerkmanagementsysteme.

Die Hauptaufgaben einer Implementierung einer Netzwerkmanagementunterstützung für eine ATM-NIC umfassen folgende Bereiche:

- Beherrschung des Umganges mit den allgemeinen MIB-Definitionen.
- Umsetzung der darin definierten Managed Objects in handhabbare interne Strukturen.
- Bindung dieser Objekte an die system- und treiberinternen Daten (Systemvariablen, Kartenregister...).
- Beherrschung des SNMP-Protokolls mit Bindung an AAL5 bzw. UDP.
- Kodierung und Dekodierung der SNMP-Nachrichten nach dem ASN.1-Standard.

Für die beiden letzteren Punkte können bestehenden Implementationen als Vorlage dienen, wohingegen die anderen Punkte hauptsächlich selbstständig gelöst werden müssen.

Die Implementierung einer solchen Managementunterstützung ist ein sehr ehrgeiziges Vorhaben und wird in einem wünschenswerten vollen Umfang nur schwer umzusetzen sein. Da

SNMP keiner Zertifizierung unterliegt, wird vermutlich der Ausweg in der Beschränkung der Menge von realisierten Managementobjekten liegen.

Weil in den betrachteten Managementapplikationen jeweils die Möglichkeit zum „Nachladen“ eigener MIBs besteht, ist das Management der ATM-NIC mit diesen Applikationen prinzipiell denkbar. Voraussetzung hierfür ist jedoch die SNMP-Kommunikation auf Basis des UDP-Protokolls. Der in der Arbeit angesprochene Proxy-Agent aus dem Artikel [Wiltfang 97] erscheint dafür eine geeignete Lösung.

Durch die leichte Erweiterbarkeit von Scotty/Tkined scheint eine Möglichkeit zur Implementierung eines ILMI-Agenten auf Script-Basis gegeben. Dabei müßten die Probleme des Versendens der SNMP-Nachricht über AAL5, des Integrierens der ILMI-MIB in das Scotty-System und die Erstellung eigener Managementscripts, die in Tkined eingebunden werden können, gelöst werden.

Die Studienarbeit ist der Versuch die Vielfalt der vorhandenen Netzwerkmanagementlösungen und -probleme aufzuzeigen und einen Überblick zu den Teilgebieten des sehr umfangreichen Themas „Netzwerkmanagement“ zu bieten. Ein allgemeingültiges Konzept auch im Bereich der Netzwerkmanagementimplementierung kann im Zuge der Komplexität nicht angegeben werden. Je nach gestellten Problem und Vorhaben muß deshalb selbstständig zu einer Lösung gefunden werden. Die Erläuterungen zu den netzwerkmanagementrelevanten Gebieten und die beigefügte HTML-Referenz sollen eine Grundlage und Hilfe zur Projektplanung und -umsetzung bei zukünftigen Implementationen darstellen.

## 6 Abkürzungsverzeichnis

(CR) (D)	
(O) (R)	Abkürzungen des ATM-Forums, die im Kapitel 3.4.1 erläutert sind
AAL	ATM Adaption Layer (Anpassungsschicht, die die nötigen Anpassungsfunktionen beim Nutzer für die Anwendung der ATM-Dienste realisiert; Es erfolgt eine Einteilung in Klassen 1 bis 5)
ANSI	American National Standards Institute (Amerikanisches Standardisierungsinstitut)
ASN.1	Abstract Syntax Notation One (Standard zur Datenbeschreibung)
ATM	Asynchronous Transfer Mode (Verfahren zur Mehrfachnutzung eines (physischen) Übertragungsmediums durch Adreßmultiplex und statistisches Multiplex)
BER	Basic Encoding Rules (Grundregeln für das (De-)Kodieren nach dem ASN.1-Format)
BICI	Broadband Inter Carrier Interface (ATM-Forum-Spezifikation für die Schnittstelle innerhalb öffentlicher ATM-Netze)
CLP	Cell Loss Priority (Prioritätsbit im ATM-Zell-Header)
CMIP	Common Management Information Protocol (Managementprotokoll der OSI-Management-Architektur)
CMIS	Common Management Information Services (Management-Dienste, die mittels CMIP erbracht werden)
CMOT	Common Management Information Services and Protocol over TCP/IP (CMIP mit Transportprotokoll TCP/IP)
CNM	Customer Network Management (Von Kunden eines öffentlichen ATM-Netzes ausgeführtes Netzwerkmanagement)
CPCS	Common Part Convergence Sublayer (gerätespezifischer Teil der Konvergenzteilschicht der AAL-Typen AAL 3/4 bzw. AAL 5)
DES	Data Encryption Standard (Standard zur Datenverschlüsselung; urheberrechtlich vom Verteidigungsministerium der USA geschützt)
DS3	Übertragungsrate der dritten PDH-Multiplexhierarchie; Auch als Transmission Class-3 bezeichnet)
EL	Element Layer (Schicht 1 des Modells aus [ITU M.3010])
EML	Element Management Layer (Schicht 2 des Modells aus [ITU M.3010])
GDMO	Guidelines for the Definition of Managed Objects (Vorschriften zum Ableiten der Managed Objects aus sogenannten „Templates“)
HTML	HyperText Markup Language (Hierarchische Beschreibungssprache für Dokumente)
ICMP	Internet Control Message Protocol (Steuerprotokoll des Internets auf Ebene der Vermittlungsschicht (3); Existiert parallel zu IP)
IETF	Internet Engineering Task Force (Standardisierungsgremium des Internets)
ILMI	Interim [bzw. Integrated] Local Management Protocol (Managementprotokoll in ATM-Netzen)
IME	ATM Interface Management Entity (Schnittstellenbezeichnung für das Management eines Interfaces eines ATM-Gerätes)
IP	Internet Protocol (Internet-Protokoll / Schicht 3)
ISO	International Organization for Standardization (Internationales Standardisierungsgremium)
ITU-T	International Telecommunications Union Telecommunications

	(Internationales Gremium zum Entwurf und Verabschiedung von Telekommunikationsstandards)
M1..M5	Schnittstellenbezeichnungen des allgemeinen ATM-Management-Modelles
MAN	Medium Area Network
MD5	Message Digestion 5 (Algorithmus, der zur Berechnung einer elektronischen Unterschrift benutzt wird)
MIB	Management Information Base (Basissatz von Managementobjekten)
NIC	Network Interface Card (Einsteckkarte zur Netzwerkanbindung)
NML	Network Management Layer (Schicht 3 des Modells aus [ITU M.3010])
OAM	Operation and Maintenance (Informationen zum Management und zur Überwachung von B-ISDN-Netzwerken)
OSI	Open Systems Interconnection
PDU	Protocol Data Unit (Allgemeiner Begriff für die Dateneinheit eines Kommunikationsprotokolls)
PPP	Point-to-Point-Protocol (Kommunikationsprotokoll einer Stand- bzw. geschalteten Verbindung)
QoS	Quality of Service (Angabe der Dienstqualität; Diese Bezeichnung wird sowohl bei SNMPv2 als auch bei ATM-Verbindungen benutzt)
RFC	Request For Comments (Diskussionsvorschlag zur Erarbeitung eines INTERNET-Standards)
SGMP	Simple Gateway Monitoring Protocol (erste Managementarchitektur im Internet)
SMI	Structure of Management Information (Regeln zur Definition von Managementobjekten)
SNMP	Simple Network Management Protocol (offenes Managementprotokoll)
SNMPv1	SNMP Version 1 nach RFC 1155 und RFC 1157
SNMPv2C	Community Based SNMP Version 2 nach RFC 1901-1908
SNMPv2U	User Based SNMP Version 2 nach RFC 1909 und RFC 1910
SONET	Synchronous Optical Network (Von den amerikanischen Bell-Laboratories entwickeltes, auf Einstufen-Multiplexing basierendes Übertragungsverfahren für Weitverkehrsnetze, aus dem 1988 der CCITT-Standard für SDH hervorging)
Tcl	Tool Command Language (Interpretierte Script-Sprache zur der Steuerung und Erweiterung von Anwendungsprogrammen)
TCP	Transmission Control Protocol (Übertragungsprotokoll / Schicht 4 )
Tk	Toolkit (Tcl-Erweiterung für das X11-Windowssystem; Bietet Schnittstellen für ein komfortables Gestalten graphischer Oberflächen)
UDP	User Datagram Protocol (Verbindungsloses Transportprotokoll im Internet)
UME	UNI Management Entity (Zugangspunkt für das Netzwerkmanagement über die UNI)
UNI	User Network Interface (Nutzer-Netz-Schnittstelle)
VCC	Virtual Channel Connection (Virtuelle Kanalverbindung)
VCI	Virtual Channel Identifier (16 Bit breite Zahl im ATM-Header zur Identifizierung des virtuellen Verbindungskanales)
VPC	Virtual Path Connection (Virtuelle Pfad-Verbindung)
VPI	Virtual Path Identifier (8 Bit breite Zahl im ATM-Header zur Identifizierung der virtuellen Pfadverbindung)
WAN	Wide Area Network

## 7 Literaturverzeichnis

- [Almesberger 96] W. Almesberger; Linux ATM API Draft, version 0.4;  
<http://lrcwww.epfl.ch/linux-atm>, 1996
- [ASN.1 a] Open Systems Solutions Inc. (New Jersey USA); ASN.1 Glossary;  
<http://www.oss.com/rgloss.htm>
- [ASN.1 b] P. Hoschka; ASN.1 Homepage;  
<http://www.inria.fr/rodeo/personnel/hoschka/asn1.html>
- [ASN.1 c] University of Salford; Kapitel über ASN.1: Chapter 8: Writing it all down;  
<http://www.salford.ac.uk/docs/depts/iti/books/osi/chap8.html>
- [ATM 96] Prof. K. Franke; Asynchronous Transfer Mode; Vorlesungsscript zur Veranstaltung „Digitale Kommunikationsnetze“ des Lehrstuhls Daten- und Kommunikationstechnik der Fakultät für Elektrotechnik und Informationstechnik, 1996
- [ATM a] Northeast Parallel Architectures Center (NPAC) at Syracuse University; Asynchronous Transfer Mode (ATM) Technology Web Knowledgebase;  
[http://www.npac.syr.edu/users/dpk/ATM\\_Knowledgebase/ATM-technology.html](http://www.npac.syr.edu/users/dpk/ATM_Knowledgebase/ATM-technology.html)
- [ATM b] S. Dresler, J. Schiller (Universität Karlsruhe Institut für Telematik); Südwestdeutsche ATM Initiative;  
[http://www.telematik.informatik.uni-karlsruhe.de/forschung/atm-info/atm\\_index.html](http://www.telematik.informatik.uni-karlsruhe.de/forschung/atm-info/atm_index.html)
- [ATM c] S. Dresler, G. Schäfer, H. Wiltfang (Universität Karlsruhe Institut für Telematik); Bericht über Netzwerk-Management und Hochgeschwindigkeits-Kommunikation;  
<http://theseus.ubka.uni-karlsruhe.de/ira-techreport/1995/1995-35.html>
- [ATM d] European Advanced Networking Test Center (EANTC); ATM Standards Document Store;  
<http://eantc.prz.tu-berlin.de/Documents/Standards>
- [ATM e] A. Robel, C. Dent; ATM Dictionary;  
<http://cell-relay.indiana.edu/cell-relay/FAQ/dictionary/dictionary.html>
- [ATM-Forum a] ATM-Forum; Homepage; <http://www.atmforum.com>
- [ATM-Forum b] ATM-Forum; Approved Specifications;  
<http://www.atmforum.com/atmforum/approved-specs.html>
- [ATM-Forum c] ATM-Forum; Software Considerations;

- [http://www.atmforum.com/atmforum/rfi/rfi\\_sec5.html#network](http://www.atmforum.com/atmforum/rfi/rfi_sec5.html#network)
- [ATM-Forum d]** ATM-Forum; Glossary;  
<http://www.atmforum.com/atmforum/library/glossary/glosspage.html>
- [C&C]** Webstart Communications; Standards and Cross References;  
<http://www.cmpcmm.com/cc/standards.html/cc/standards.html>
- [CMU]** Carnegie Mellon University; CMU SNMP/SNMPv2 distribution;  
<ftp://ftp.net.cmu.edu/pub/snmp-dist>
- [Drafts]** ESnet Network Information Center; Internet Draft RFCs;  
<http://www.es.net/pub/internet-drafts>
- [GDMO-Browser]** M. Kernchen; WWW GDMO MIB Browser;  
<http://www.ibr.cs.tu-bs.de/cgi-bin/gbrowser.tcl>
- [Gora 92]** W. Gora; ASN.1 Abstract Syntax Notation One;  
DATACOM-Verlag Bergheim, 1992
- [Hein 94]** M. Hein; D. Griffiths; SNMP;  
International Thomason Publishing, 1994
- [HP a]** Hewlett Packard; Homepage; <http://www.hp.com/>
- [HP b]** Hewlett Packard; HP OpenView Homepage;  
<http://hpcc920.external.hp.com/openview/index.html>
- [HP c]** Hewlett Packard; Networking; <http://www.hp.com/ahp/Networking/>
- [HP d]** Hewlett Packard; Netzwerk-Management;  
<http://www-1.hewlett-packard.de/germany/network/net-mana/net-mana.htm>
- [IETF]** The Internet Engineering Task Force; Homepage;  
<http://www.ietf.cnri.reston.va.us/>
- [ILMI]** A. Robel, C. Dent; ILMI Reference;  
<http://cell-relay.indiana.edu/cell-relay/FAQ/dictionary/I/ILMI.html>
- [ILMI 96]** ATM-Forum; Integrated Local Management Interface (ILMI)  
Specification Version 4.0;  
ATM-Forum-Spezifikation af-ilmi-0065.000; 1996
- [ISO a]** International Organization for Standardization; Homepage;  
<http://www.iso.ch/>
- [ISO b]** International Organization for Standardization; International  
Standardizing Bodies; <http://www.iso.ch/infoe/stbodies.html>

- [ISO 84]** ISO/IEC IS 7498; Information Processing Systems - Open Systems Interconnection - Basic Reference Model; ISO-Standard; 1984
- [ISO 88a]** ISO/IEC IS 8649; Information Technology - Open Systems Interconnection - Service Definition for Association Control Service Element; ISO-Standard; 1988
- [ISO 88b]** ISO/IEC IS 8650; Information Technology - Open Systems Interconnection - Protocol Spezifikation for Association Control Service Element; ISO-Standard; 1988
- [ISO 89a]** ISO/IEC IS 9072-1; Information Technology - Open Systems Interconnection - Remote Operations - Part 1: Model, Notation and Service Definition; ISO-Standard; 1989
- [ISO 89b]** ISO/IEC IS 9072-2; Information Technology - Open Systems Interconnection - Remote Operations - Part 2: Protocol Specification; ISO-Standard; 1989
- [ISO 89c]** ISO/IEC IS 7498; Information Processing Systems - Open Systems Interconnection - Basic Reference Model - Part 4: Management Framework; ISO-Standard; 1989
- [ISO 91a]** ISO/IEC IS 9595; Information Technology - Open Systems Interconnection - Common Management Information Service Definition; ISO-Standard; 1991
- [ISO 91b]** ISO/IEC IS 9596; Information Technology - Open Systems Interconnection - Common Management Information Protocol Definition; ISO-Standard; 1991
- [ISO 91c]** ISO/IEC IS 10165-4; Information Technology - Open Systems Interconnection - Structure of Management Information - Part 4: Guidelines for the Definition of Managed Objects; ISO-Standard; 1991
- [ITU]** International Telecommunication Union; Homepage; <http://www.itu.ch>
- [ITU M.3010]** ITU-T, Principles of Telecommunications Management Network (TMN), ITU-T-Empfehlung M.3010, 1991
- [Kauffels 92]** F.-J. Kauffels; Netzwerk-Management: Probleme, Standards, Strategien; DATACOM-Verlag Bergheim, 1992
- [Kyas 93]** O. Kyas; ATM-Netzwerke: Aufbau, Funktion, Performance, DATACOM-Verlag Bergheim, 1993
- [Lange 96]** M. Lange; Management von ATM-Endsystemen; Diplomarbeit

am Lehrstuhl für Daten- und Kommunikationstechnik der Fakultät für Elektrotechnik und Informationstechnik, 1996

- [M3 94]** ATM-Forum; Customer Network Management (CNM) for ATM Public Network Service (M3 Specification);  
ATM-Forum-Spezifikation af-nm-0019.000; 1994
- [M4 94]** ATM-Forum; M4 Interface Requirements and Logical MIB;  
ATM-Forum-Spezifikation af-nm-0020.000; 1994
- [M4 CMIP 95]** ATM-Forum; CMIP Specification for the M4 Interface;  
ATM-Forum-Spezifikation af-nm-0027.000; 1995
- [M4 NV 96]** ATM-Forum; M4 Network-View Interface Requirements and Logical MIB; ATM-Forum-Spezifikation af-nm-0058.000; 1996
- [MIB-Browser]** J. Schönwälder; WWW SNMP MIB Browser;  
<http://www.ibr.cs.tu-bs.de/cgi-bin/sbrowser.cgi>
- [MIB-Master]** Equivalence Pty Limited; MibMaster;  
<http://www.ozemail.com.au/~equival/mibmaster/index.html>
- [Neuendorf 93]** B. Neuendorf; Symbolische Bearbeitung von ASN.1-Datenstrukturen; Diplomarbeit am Lehrstuhl Rechnernetze und verteilte Systeme der Fakultät für Informatik, 1993
- [NNM a]** Hewlett Packard; HP OpenView Documentation Network Node Manager 5.0; [http://ovweb.external.hp.com/lpe/cgi-bin/doc\\_serv/prod\\_req.pl?nnm5.0](http://ovweb.external.hp.com/lpe/cgi-bin/doc_serv/prod_req.pl?nnm5.0)
- [NNM b]** HP Network Node Manager Dokumentation; Using Network Node Manager; <http://www.infotech.tu-chemnitz.de/~knoll/HP/Dokumente/j1136-90002.pdf>; 1997
- [NNM c]** HP Network Node Manager Dokumentation; OpenView Windows Developer's Guide; <http://www.infotech.tu-chemnitz.de/~knoll/HP/Dokumente/j1150-90003.pdf>; 1997
- [NNM d]** HP Network Node Manager Dokumentation; SNMP Developer's Guide; <http://www.infotech.tu-chemnitz.de/~knoll/HP/Dokumente/j1150-90008.pdf>; 1997
- [NNM e]** HP Network Node Manager Dokumentation; A Guide to Scalability and Distribution for Network Node Manager;  
<http://www.infotech.tu-chemnitz.de/~knoll/HP/Dokumente/j1136-90001.pdf>; 1997

- [NNM f] HP Network Node Manager Dokumentation; SNMP Agent Administrator's Guide;  
<http://www.infotech.tu-chemnitz.de/~knoll/HP/Dokumente/j1136-90005.pdf>; 1997
- [RAD] RAD Data Communications; Networking Terminology;  
<http://www.rad.com/networks/netterms.htm>
- [Rechnernetze 96] Prof. U. Hübner; Protokolle und Management von Rechnernetzen; Vorlesungsscript des Lehrstuhls Rechnernetze und verteilte Systeme der Fakultät für Informatik, 1996
- [RFC 1065] M. Rose, K. McCloghrie; Structure and Identification of Management Information for TCP/IP-based Internets; IETF; 1988
- [RFC 1066] M. Rose, K. McCloghrie; Management Information Base for Network Management of TCP/IP-based Internets; IETF; 1988
- [RFC 1067] J. Case, M. Fedor, M. Schoffstall, J. Davin; A Simple Network Management Protocol; IETF; 1988
- [RFC 1155] M. Rose, K. McCloghrie; Structure and Identification of Management Information for TCP/IP-based Internets; IETF; 1990
- [RFC 1157] J. Case, M. Fedor, M. Schoffstall, J. Davin; A Simple Network Management Protocol (SNMP); IETF; 1990
- [RFC 1189] U. Warrior, L. Besaw, L. LaBarre, B. Handspicker; The Common Management Information Services and Protocols for the Internet (CMOT and CMIP); IETF; 1990
- [RFC 1212] M. Rose, K. McCloghrie; Concise MIB Definitions; IETF; 1991
- [RFC 1213] M. Rose, K. McCloghrie; Management Information Base for Network Management of TCP/IP-based internets: MIB-II; IETF; 1991
- [RFC 1441] J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Introduction to version 2 of the Internet-standard Network Management Framework; IETF; 1993
- [RFC 1442] J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Structure of Management Information for version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1993
- [RFC 1443] J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Textual Conventions for version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1993
- [RFC 1444] J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Conformance Statements for version 2 of the Simple Network Management Protocol

(SNMPv2); IETF; 1993

- [RFC 1445]** J. Galvin, K. McCloghrie; Administrative Model for version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1993
- [RFC 1446]** J. Galvin, K. McCloghrie; Security Protocols for version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1993
- [RFC 1447]** J. Galvin, K. McCloghrie; Party MIB for version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1993
- [RFC 1448]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Protocol Operations for version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1993
- [RFC 1449]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Transport Mappings for version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1993
- [RFC 1450]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Management Information Base for version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1993
- [RFC 1451]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Manager-to-Manager Management Information Base; IETF; 1993
- [RFC 1452]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Coexistence between Version 1 and version 2 of the Internet-standard Network Management Framework; IETF; 1993
- [RFC 1483]** Juha Heinanen; Multiprotocol Encapsulation over ATM Adaptation Layer 5; IETF 1993
- [RFC 1577]** M. Laubach; Classical IP and ARP over ATM; IETF; 1994
- [RFC 1695]** M. Ahmed, K. Tesink; Definitions of Managed Objects for ATM Management Version 8.0 using SMIV2; IETF; 1994
- [RFC 1901]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Introduction to Community-based SNMPv2; IETF; 1996
- [RFC 1902]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1996
- [RFC 1903]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1996
- [RFC 1904]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Conformance

Statements for Version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1996

- [RFC 1905]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1996
- [RFC 1906]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1996
- [RFC 1907]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2); IETF; 1996
- [RFC 1908]** J. Case, K. McCloghrie, M. Rose, S. Waldbusser; Coexistence between Version 1 and Version 2 of the Internet-standard Network Management Framework; IETF; 1996
- [RFC 1909]** K. McCloghrie; An Administrative Infrastructure for SNMPv2; IETF; 1996
- [RFC 1910]** G. Waters; User-based Security Model for SNMPv2; IETF; 1996
- [Scotty a]** J. Schönwälder; Scotty - Tcl Extensions for Network Management Applications; <http://wwwsnmp.cs.utwente.nl/~schoenw/scotty/>
- [Scotty b]** J.Schönwälder, H. Langendörfer; Scotty-Overview; <ftp://ftp.ibr.cs.tu-bs.de/pub/local/papers/tcltk-95.ps.gz>; 1995
- [Scotty c]** J.Schönwälder; Scotty-Workshop-Folien; PS-File im Scotty-Paket <ftp://ftp.ibr.cs.tu-bs.de/pub/local/papers/tcltk-95.slides.ps.gz>; 1995
- [Scotty d]** J. Schönwälder; Scotty man pages; <http://wwwsnmp.cs.utwente.nl/~schoenw/scotty/man/scotty.html>
- [Scotty e]** M. Newnham; Getting Started with tkined; <http://www.eng.auburn.edu/users/doug/tkined.html>; 1996
- [Seits 94]** J. Seits; Netzwerkmanagement; International Thomason Publishing, 1994
- [Tautenhahn 97]** M. Tautenhahn; Ein Praktikum Netzwerk-Management; Studienarbeit am Lehrstuhl Rechnernetze und verteilte Systeme der Fakultät für Informatik, 1997
- [TclTk]** SunScript; Tcl/Tk; <http://sunscript.sun.com>
- [TU-Braunschweig]** TU-Braunschweig; Network Management; <http://www.ibr.cs.tu-bs.de/projects/nm/>

- [Werkzeuge 96]** Prof. U. Hübner; Softwarewerkzeuge; Vorlesungsscript des Lehrstuhls Rechnernetze und verteilte Systeme der Fakultät für Informatik, 1996
- [Wiltfang 97]** H. R. Wiltfang; Management und Monitoring von ATM-Netzen; Zeitschrift: Praxis der Informationsverarbeitung und Kommunikation 2/1997, S. 68-75
- [UNI 94]** ATM-Forum; ATM User-Network Interface Specification Version 3.1; ATM-Forum-Spezifikation af-uni-0010.002; 1994

## **8 Selbstständigkeitserklärung**

Hiermit erkläre ich, daß ich diese Arbeit selbständig unter Verwendung der angegebenen Literatur angefertigt habe.

Chemnitz den, 15.08.1997

gez. Thomas Martin Knoll