Handbuch Netzwerk-Technologien

CISCO Documentation

Übersetzung:
Cosmos Consulting

Handbuch Netzwerk-Technologien

Markt&Technik Buch-und Software-Verlag GmbH

Die Deutsche Bibliothek – CIP-Einheitsaufnahme

Handbuch Netzwerk-Technologien : komplettes Grundwissen
für Internetworking und Networking / Merilee Ford ... –
Haar bei München : Markt und Technik, Buch- und Software-Verl.
Einheitssacht.: Internetworking technologies handbook <dt.>

ISBN 3-8272-2034-3

Buch. 1998
Gb.

Die Informationen in diesem Produkt werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht.
Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt.
Bei der Zusammenstellung von Texten und Abbildungen wurde mit größter Sorgfalt vorgegangen.
Trotzdem können Fehler nicht vollständig ausgeschlossen werden.
Verlag, Herausgeber und Autoren können für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.
Für Verbesserungsvorschläge und Hinweise auf Fehler sind Verlag und Herausgeber dankbar.

Autorisierte Übersetzung der amerikanischen Originalausgabe:
Internetworking Technologies Handbook © 1998 Macmillan Technical Publishing

Alle Rechte vorbehalten, auch die der fotomechanischen Wiedergabe und der Speicherung in elektronischen Medien.
Die gewerbliche Nutzung der in diesem Produkt gezeigten Modelle und Arbeiten ist nicht zulässig.

Fast alle Hardware- und Softwarebezeichnungen, die in diesem Buch erwähnt werden, sind gleichzeitig auch eingetragene Warenzeichen oder sollten als solche betrachtet werden.

Das Logo Cisco Press ist ein eingetragenes Warenzeichen von Cisco Systems, Inc., USA.

10 9 8 7 6 5 4 3 2 1

02 01 00 99 98

ISBN 3-8272-2034-3

© 1998 by Markt&Technik Buch- und Software-Verlag GmbH, Hans-Pinsel-Straße 9b, D-85540 Haar bei München/Germany
Alle Rechte vorbehalten
Einbandgestaltung: Helfer Grafik Design, München
Programmleitung: Erik Franz, efranz@mut.de
Übersetzung und Lokalisierung: Cosmos Consulting GmbH/ Systemhaus/ISP/Redaktion, Cisco@cosmosnet.de
Fachlektorat: Ralf Kothe, Cisco Systems GmbH
Herstellung: Claudia Bäurle, cbaeurle@mut.de
Satz: text&form, Fürstenfeldbruck
Druck: Media-Print, Paderborn
Dieses Produkt wurde mit Desktop-Publishing-Programmen erstellt und auf chlorfrei gebleichtem Papier gedruckt
Printed in Germany

Inhaltsverzeichnis

Vorwort 19

Teil 1 Einführung in das Internetworking 25

1 Grundlagen des Internetworking 27
1.1 Was ist ein Internetwork? 27
1.1.1 Die Geschichte des Internetworking 28
1.1.2 Die Herausforderungen des Internetworking 29
1.2 Das OSI-Referenzmodell 30
1.2.1 Eigenschaften der OSI-Schichten 31
1.2.2 Protokolle 32
1.2.3 Das OSI-Modell und die Kommunikation zwischen Systemen 33
1.2.4 Interaktion zwischen den Schichten des OSI-Modells 33
1.2.5 Dienste der OSI-Schichten 34
1.2.6 Schichten des OSI-Modells und der Datenaustausch 35
1.2.7 Die physische Schicht des OSI-Modells 37
1.2.8 Die Verbindungsschicht des OSI-Modells 38
1.2.9 Die Vermittlungsschicht des OSI-Modells 39
1.2.10 Die Transportschicht des OSI-Modells 39
1.2.11 Die Kommunikationsschicht des OSI-Modells 40
1.2.12 Die Darstellungsschicht des OSI-Modells 40
1.2.13 Die Anwendungsschicht des OSI-Modells 41
1.3 Datenformate 42
1.4 Die ISO-Hierarchie von Netzwerken 44
1.5 Verbindungsorientierte und verbindungslose Netzwerk-Dienste 45
1.6 Adressierung im Internetwork 47
1.6.1 Verbindungsschicht 47
1.6.2 MAC-Adressen 48
1.6.3 Adressen der Vermittlungsschicht 51

1.6.4	Hierarchischer oder ebener Adreßraum	52
1.6.5	Adreßzuordnung	53
1.6.6	Adressen oder Namen	53
1.7	Grundlagen der Flußsteuerung	54
1.8	Grundlagen der Fehlerprüfung	55
1.9	Grundlagen des Multiplexing	55
1.10	Organisationen für Standardisierung	57
2	**Einführung in die LAN-Protokolle**	59
2.1	Was ist ein LAN?	60
2.2	LAN-Protokolle und das OSI-Referenzmodell	60
2.3	Medium-Zugriffsmethoden im LAN	61
2.4	Übertragungsverfahren im LAN	61
2.5	LAN-Topologien	62
2.6	LAN-Geräte	64
3	**Einführung in die WAN-Technologien**	67
3.1	Was ist ein WAN?	67
3.2	Punkt-zu-Punkt-Verbindungen	68
3.3	Leitungsvermittlung	69
3.4	Paketvermittlung	70
3.5	Virtuelle Verbindungen im WAN	70
3.6	WAN-Einwahldienste	71
3.7	WAN-Geräte	72
3.7.1	WAN-Switch	72
3.7.2	Zugriffsserver	72
3.7.3	Modem	73
3.7.4	CSU/DSU	73
3.7.5	ISDN-Terminal-Adapter	74
4	**Grundlagen des Bridging und Switching**	75
4.1	Was sind Bridges und Switches?	75
4.2	Überblick zu den Geräten der Verbindungsschicht	76
4.3	Bridge-Typen	78
4.4	Switch-Typen	80
4.4.1	ATM-Switch	80
4.4.2	LAN-Switch	81
5	**Grundlagen des Routing**	83
5.1	Was ist Routing?	83
5.2	Komponenten des Routing	84
5.2.1	Pfadermittlung	84
5.2.2	Switching	86
5.3	Routing-Algorithmen	87
5.3.1	Entwicklungsziele	88
5.3.2	Algorithmusarten	90
5.3.3	Routing-Meßparameter	93
5.4	Netzwerk-Protokolle	95

6 Grundlagen des Netzwerk-Managements 97
6.1 Was ist Netzwerk-Management? 97
6.1.1 Ein geschichtlicher Rückblick 98
6.2 Netzwerk-Management-Architektur 98
6.3 ISO Netzwerk-Management-Modell 99
6.3.1 Performance-Management 100
6.3.2 Konfigurations-Management 101
6.3.3 Accounting-Management 101
6.3.4 Fehler-Management 102
6.3.5 Sicherheits-Management 102

Teil 2 LAN-Protokolle 105

7 Ethernet-Technologien 107
7.1 Hintergrund 107
7.2 Ethernet und IEEE 802.3 108
7.2.1 Ethernet- und IEEE-802.3-Betrieb 109
7.2.2 Ethernet und IEEE-802.3 – Service-Unterschiede 110
7.2.3 Ethernet- und IEEE-802.3-Frame-Formate 111
7.3 100-Mbit/s Ethernet 113
7.3.1 100BaseT im Überblick 114
7.3.2 100BaseT-Signalisierung 114
7.3.3 100BaseT-Hardware 115
7.3.4 100BaseT-Betrieb 117
7.3.5 100BaseT-Mediumtypen 118
7.4 100VG-AnyLAN 121
7.4.1 100VG-AnyLAN-Betrieb 123
7.5 Gigabit Ethernet 123
7.5.1 Gigabit-Ethernet-Spezifikation 124
7.5.2 Migrieren zum Gigabit Ethernet 125

8 Fiber Distributed Data Interface (FDDI) 127
8.1 Hintergrund 127
8.1.1 Standards 128
8.2 FDDI-Übertragungsmedium 128
8.3 FDDI-Spezifikationen 130
8.4 FDDI-Station-Attachment-Typen 131
8.5 FDDI-Fehlertoleranz 133
8.5.1 Doppelring 133
8.5.2 Optischer Bypass-Switch 134
8.5.3 Dual-Homing 135
8.6 FDDI-Frame-Format 136
8.6.1 FDDI-Frame-Felder 136
8.7 Copper-Distributed Data Interface (CDDI) 137

9 Token Ring/IEEE 802.5 139
9.1 Hintergrund 139
9.2 Physische Verbindungen 140
9.3 Betrieb eines Token Ring 141

9.4 Prioritäten-System 142
9.5 Mechanismen des Ausfall-Managements 143
9.6 Frame-Format 143
9.6.1 Felder des Token-Frames 144
9.6.2 Felder des Daten-/Befehls-Frames 145

Teil 3 WAN-Technologien 147

10 Frame Relay 149
10.1 Hintergrund 149
10.1.1 Frame-Relay-Standardisierung 150
10.2 Frame-Relay-Geräte 151
10.3 Frame Relay Virtual Circuits 152
10.3.1 Switched Virtual Circuits (SVC/GVV) 153
10.3.2 Permanent Virtual Circuits (PVC/FVV) 153
10.3.3 Data-Link Connection Identifier (DLCI) 154
10.4 Congestion-Control-Mechanismen 154
10.4.1 Frame Relay Discard Eligibility (DE) 155
10.4.2 Frame-Relay-Fehlererkennung 156
10.5 Frame Relay Local Management Interface (LMI) 156
10.6 Frame-Relay-Netzwerk-Implementation 157
10.6.1 Öffentliche Netzwerke 158
10.6.2 Private Unternehmensnetze 159
10.7 Frame-Formate des Frame-Relay 159
10.7.1 Standard-Frame des Frame Relay 159
10.7.2 LMI-Frame-Format 161

11 High-Speed Serial Interface 163
11.1 Hintergrund 163
11.2 Grundlagen zum HSSI 163
11.3 HSSI-Betrieb 164
11.3.1 Rückkopplungstests 165

12 Integrated Services Digital Network (ISDN) 167
12.1 Hintergrund 167
12.2 ISDN-Komponenten 167
12.3 Dienste 169
12.4 Schicht 1 170
12.5 Schicht 2 171
12.6 Schicht 3 172

13 Point-to-Point Protocol 175
13.1 Background 175
13.2 PPP-Komponenten 175
13.3 Das Verfahren 176
13.4 Anforderungen der physischen Schicht 176
13.5 PPP-Verbindungsschicht 177
13.5.1 PPP-Verbindungssteuerungs-Protokoll 178

14 Switched Multimegabit Data Service (SMDS) 181
14.1 Hintergrund 181
14.2 SMDS-Netzwerk-Komponenten 181
14.3 SMDS Interface Protokoll (SIP) 182
14.3.1 SIP-Stufen 183
14.4 Distributed Queue Dual Bus (DQDB) 184
14.5 SMDS-Zugriffsklassen 186
14.6 Überblick zur SMDS-Adressierung 186
14.7 SMDS-Referenz: SIP-Stufe-3-PDU-Format 187
14.8 SMDS-Referenz: SIP-Stufe-2-Zellformat 189

15 ADSL – Asymmetric Digital Subscriber Line 193
15.1 Hintergrund 193
15.1.1 ADSL-Standardisierung 193
15.2 Überblick zur ADSL-Technologie 194
15.3 ADSL-Funktionen 195
15.4 ADSL-Referenzmodell 197

16 Synchronous Data-Link Control und Derivate 201
16.1 Hintergrund 201
16.2 SDLC-Typen und Topologien 202
16.3 SDLC-Frame-Format 203
16.4 Abgeleitete Protokolle 205
16.4.1 High-Level Data-Link Control (HDLC) 205
16.4.2 Link-Access Procedure, Balanced (LAPB) 206
16.4.3 IEEE 802.2 207
16.4.4 Qualified Logical-Link Control (QLLC) 208

17 X.25 209
17.1 Hintergrund 209
17.2 X.25-Geräte und -Protokollfunktion 210
17.2.1 Packet Assembler/Disassembler (PAD) 210
17.2.2 X.25-Sitzung einrichten 211
17.2.3 Virtuelle Verbindungen bei X.25 211
17.3 X.25-Protokolle 213
17.3.1 Packet-Layer Protocol (PLP) 214
17.3.2 Link-Access Procedure, Balanced (LAPB) 215
17.3.3 X.21bis-Protokoll 216
17.4 LAPB-Frame-Format 217
17.5 X.121-Adreß-Format 218

Teil 4 Bridging und Switching 221

18 Asynchronous Transfer Mode (ATM) 223
18.1 Grundlagen 223
18.1.1 Standards 223
18.2 ATM-Geräte und Netzwerkumgebungen 224
18.2.1 ATM-Zellen-Basisformat 224

18.2.2	ATM-Geräte	225
18.2.3	ATM-Netzwerkschnittstellen	226
18.3	ATM-Zellenkopf-Format	227
18.3.1	Felder im ATM-Zellenkopf	227
18.4	ATM-Dienste	228
18.4.1	ATM Virtual Connections	229
18.5	ATM-Switch-Betrieb	229
18.6	ATM-Referenzmodell	230
18.6.1	ATM, Physikalische Schicht	231
18.6.2	ATM-Anpassungsschichten: AAL1	232
18.6.3	ATM-Anpassungsschichten: AAL3/4	233
18.6.4	ATM-Anpassungsschicht: AAL5	234
18.7	ATM-Adressierung	234
18.7.1	Sub-Netzwerk-Modell der Adressierung	235
18.7.2	NSAP-Format-ATM-Adresse	235
18.7.3	ATM-Adreßfelder	236
18.8	ATM-Verbindungen	237
18.9	ATM und Multicasting	238
18.10	ATM Quality of Service (QoS)	239
18.11	ATM-Signalisierung und Verbindungsaufbau	240
18.11.1	ATM-Verbindungsaufbau	241
18.11.2	Routing und Verhandlung der Verbindungsanforderung	241
18.12	ATM-Meldungen für die Verbindungsverwaltung	242
18.13	LAN-Emulation (LANE)	242
18.13.1	LANE-Protokoll-Architektur	244
18.13.2	Bestandteile des LANE	245
18.13.3	Verbindungsarten der LAN-Emulation	246
18.13.4	LANE im Betrieb	248
19	**Data-Link Switching**	251
19.1	Grundlagen	251
19.2	DLSw im Vergleich mit Source-Route-Bridging	252
19.3	DLSw-SNA-Unterstützung	254
19.4	DLSw-Switch-to-Switch Protocol (SSP)	255
19.5	DLSw-Betrieb	256
19.5.1	DLSw-Prozesse	256
19.6	DLSw-Meldungsformate	260
20	**LAN Switching**	267
20.1	Grundlagen	267
20.1.1	Zur Geschichte	268
20.2	Einsatz von LAN-Switches	268
20.2.1	Vermittlung beim LAN-Switching	269
20.2.2	Bandbreite des LAN-Switching	270
20.3	LAN-Switch und das OSI-Modell	270

21	**Tag Switching**	273
21.1	Grundlagen	273
21.2	Die Tag-Switching-Architektur	274
21.2.1	Die Forwarding-Komponente	274
21.2.2	Steuerungskomponenten	276
21.3	Destination-Based Routing (zielbasiertes Routing)	276
21.3.1	Downstream-Tag-Zuweisung	277
21.3.2	Downstream-Tag-Zuweisung auf Anforderung	278
21.3.3	Upstream-Tag-Zuweisung	278
21.4	Hierarchical-Routing	280
21.5	Flexibles Routing durch explizite Routen	281
21.6	Multicast-Routing	282
21.7	Tag-Switching mit ATM	283
21.8	Quality of Service	284
21.9	IP-Switching	285
22	**Mixed-Media-Bridging**	287
22.1	Grundlagen	287
22.2	Übertragungsanforderungen	288
22.3	Translational-Bridging	290
22.4	Source-Route-Transparent-Bridging	293
23	**Source-Route Bridging (SRB)**	295
23.1	Grundlagen	295
23.2	SRB-Algorithmus	295
23.3	Frame-Format	297
23.3.1	Routing-Control-Feld	298
23.3.2	Routing-Designator-Felder	299
24	**Transparent-Bridging**	301
24.1	Grundlagen	301
24.2	Transparent-Bridging-Betrieb	301
24.2.1	Bridging-Loops	302
24.2.2	Spanning-Tree-Algorithmus (STA)	304
24.3	Frames-Format	307
Teil 5	Netzwerk-Protokolle	311
25	**AppleTalk**	313
25.1	Background	313
25.2	AppleTalk-Netzwerk-Komponenten	314
25.2.1	Sockets	315
25.2.2	Knoten	315
25.2.3	Netzwerke	316
25.2.4	Zonen	317
25.3	Bitübertragungs- und Sicherungsschichten von AppleTalk	318
25.3.1	EtherTalk	319
25.3.2	LocalTalk	320

25.3.3	TokenTalk	322
25.3.4	FDDITalk	323
25.4	Netzwerk-Adressen	324
25.4.1	Zuweisung von Netzwerk-Adressen	324
25.5	AppleTalk Address-Resolution Protocol (AARP)	325
25.5.1	Address-Mapping Table	326
25.5.2	Address Gleaning	326
25.5.3	AARP-Operation	327
25.6	Datagram-Delivery Protocol (DDP) im Überblick	327
25.6.1	DDP-Übertragungsverfahren	328
25.7	AppleTalk-Transportschicht	329
25.7.1	Routing-Table Maintenance Protocol (RTMP) im Überblick	329
25.7.2	Name-Binding Protocol (NBP) im Überblick	330
25.7.3	AppleTalk Update-Based Routing Protocol (AURP)	332
25.7.4	AppleTalk Transaction Protocol (ATP)	334
25.7.5	AppleTalk Echo Protocol (AEP)	334
25.8	AppleTalk-Protokolle der oberen Schichten	335
25.8.1	AppleTalk Data-Stream Protocol (ADSP)	336
25.8.2	Zone Information Protocol (ZIP)	336
25.8.3	AppleTalk Session Protocol (ASP)	337
25.8.4	Printer-Access Protocol (PAP) im Überblick	338
25.8.5	AppleTalk Filing Protocol (AFP)	338
25.9	AppleTalk-Protokollreihe	338
25.9.1	Format von DDP-Paketen	339
26	**DECnet**	341
26.1	Background	341
26.2	DECnet Phase IV Digital Network Architecture (DNA)	342
26.2.1	Die Schichten von Phase-IV-DNA	343
26.2.2	Phase-IV-DECnet-Adressierung	344
26.3	DECnet/OSI Digital Network Architecture (DNA)	345
26.3.1	DECnet/OSI-DNA-Implementierungen	345
26.4	DECnet-Medienzugriff	346
26.5	DECnet-Routing	347
26.6	DECnet-Endkommunikationsschicht	348
26.6.1	Network-Services Protocol	348
26.7	DECnet/OSI-Transportschicht	348
26.8	Die oberen Schichten von DECnet Phase IV	349
26.8.1	Benutzerschicht	349
26.8.2	Netzwerk-Managementschicht	350
26.8.3	Netzwerk-Anwendungsschicht	350
26.8.4	Verbindungskontrollschicht	351
26.9	Die oberen Schichten von DECnet/OSI	351
26.9.1	Anwendungsschicht	351
26.9.2	Darstellungsschicht	352
26.9.3	Verbindungskontrollschicht	352

27	**IBM Systems Network Architecture (SNA) Protokolle**	355
27.1	Background	355
27.2	Traditionelle SNA-Umgebungen	356
27.2.1	IBM-SNA-Architektur	356
27.2.2	Physische Entitäten von IBM SNA	357
27.2.3	Datenübermittlungssteuerung von IBM SNA	358
27.2.4	IBM Network-Addressable Units (NAUs)	360
27.2.5	IBM SNA-Knoten	361
27.3	IBM Peer-to-Peer-Netzwerke	362
27.3.1	APPN-Komponenten	362
27.3.2	Knotenarten von IBM APPN	363
27.3.3	IBM APPN-Dienste	364
27.4	Das Format der Basic Information Unit (BIU)	369
27.4.1	Die Felder der BIU	369
27.5	Das Format der Path Information Unit (PIU)	371
27.5.1	Die Felder der PIU	371
28	**Internet-Protokolle**	375
28.1	Background	375
28.2	Internet-Protokoll (IP)	377
28.2.1	Das Format des IP-Pakets	377
28.2.2	IP-Adressierung	379
28.2.3	IP-Adreßklassen	380
28.3	Address-Resolution Protocol (ARP) im Überblick	386
28.4	Internet-Routing	386
28.4.1	IP-Routing	387
28.5	Internet Control-Message Protocol (ICMP)	388
28.5.1	ICMP-Nachrichten	388
28.5.2	ICMP Router-Discovery Protocol (IDRP)	389
28.6	Transmission-Control Protocol (TCP)	390
28.6.1	TCP-Verbindungsaufbau	391
28.6.2	Positive Acknowledgment and Retransmission (PAR)	392
28.6.3	TCP Sliding Window	392
28.6.4	TCP-Paketformat	393
28.6.5	Beschreibung der TCP-Paketfelder	393
28.7	User Datagram Protocol (UDP)	394
28.8	Die Anwendungsschichtprotokolle der Internet-Protokolle	395
29	**NetWare-Protokolle**	397
29.1	Hintergrund	397
29.2	NetWare Medienzugriff	398
29.3	Internetwork Packet Exchange (IPX) im Überblick	399
29.4	IPX-Kapselungsarten	400
29.5	Service-Advertisement Protocol (SAP)	401
29.5.1	SAP-Filter	402
29.6	NetWare-Transportschicht	402

29.7	NetWare-Protokolle und Dienste der oberen Schichten	402
29.7.1	NetWare-Dienste der Anwendungsschicht	403
29.8	IPX-Paketformat	404
30	**Protokolle der Open System Interconnection (OSI)**	407
30.1	Hintergrund	407
30.2	OSI-Netzwerkprotokolle	407
30.2.1	Physikalische Bitübertragungsschicht und Datensicherungsschicht von OSI	408
30.2.2	OSI-Netzwerkschicht	409
30.2.3	OSI-Protokolle für die Transportschicht	412
30.2.4	OSI-Protokolle für die Kommunikationsschicht	414
30.2.5	OSI-Protokolle für die Darstellungsschicht	415
30.2.6	OSI-Protokolle für die Anwendungsschicht	416
31	**Banyan VINES**	419
31.1	Hintergrund	419
31.2	Medienzugriff	420
31.3	Netzwerkschicht	420
31.3.1	VINES Internetwork Protocol (VIP)	420
31.3.2	Routing Table Protocol (RTP)	425
31.3.3	Address Resolution Protocol (ARP)	426
31.3.4	Internet Control Protocol (ICP)	427
31.4	Transportschicht	427
31.5	Protokolle übergeordneter Schichten	428
32	**Xerox Network Systems (XNS)**	429
32.1	Hintergrund	429
32.2	Übersicht über die XNS-Hierarchie	430
32.3	Medienzugriff	431
32.4	Netzwerkschicht	431
32.5	Transportschicht	433
32.6	Protokolle übergeordneter Schichten	434
Teil 6	Routing-Protokolle	437
33	**Border Gateway Protocol (BGP)**	439
33.1	Hintergrund	439
33.2	BGP-Operationen	440
33.3	BGP-Routing	442
33.4	Nachrichtentypen von BGP	443
33.5	BGP-Paketformate	444
33.5.1	Header-Format	444
33.5.2	Format der Open-Nachricht	445
33.5.3	Format der Update-Nachricht	446
33.5.4	Format der Notification-Nachricht	448
34	**Enhanced IGRP**	451
34.1	Hintergrund	451
34.2	Fähigkeiten und Attribute von Enhanced IGRP	452

34.3	Zugrundeliegende Prozesse und Techniken	453
34.4	Begriffe zum Routing	454
34.4.1	Nachbartabellen	455
34.4.2	Topologietabellen	455
34.4.3	Routenzustände	456
34.4.4	Routenkennzeichnung	456
34.5	Pakettypen von Enhanced IGRP	457
35	**IBM Systems Network Architecture (SNA) Routing**	459
35.1	Hintergrund	459
35.2	IBM-SNA-Sitzungsverbindungen	460
35.3	IBM-SNA-Übertragungsgruppen	460
35.4	IBM-SNA – explizite und virtuelle Routen	461
35.5	IBM-SNA-Dienstklassen	462
35.5.1	Dienstklassen beim Subarea Routing	462
35.5.2	Dienstklassen beim APPN Routing	463
35.6	IBM SNA – Subarea Routing	465
35.7	IBM Advanced Peer-to-Peer Networking (APPN) Routing	466
35.7.1	IBM APPN – Node Type 2.1 Routing	467
35.7.2	IBM APPN – DLUR/S Routing	470
35.7.3	IBM-APPN-Verbindungsnetzwerk	470
35.7.4	IBM APPN – Übergangsknoten	471
36	**Interior Gateway Routing Protocol (IGRP)**	473
36.1	Hintergrund	473
36.2	Eigenschaften von IGRP	474
36.2.1	Stabilitätsmerkmale	475
36.2.2	Timer	477
37	**Internet Protocol (IP) Multicast**	479
37.1	Hintergrund	479
37.2	Internet Group Membership Protocol (IGMP)	479
37.3	Protokolle für das IP Multicast-Routing	480
37.3.1	Protocol-Independent Multicast (PIM)	481
37.3.2	Distance-Vector Multicast Routing Protocol (DVMRP)	482
37.3.3	Multicast Open Shortest Path First (MOSPF)	482
38	**NetWare Link Services Protocol (NLSP)**	485
38.1	Hintergrund	485
38.2	NLSP – hierarchisches Routing	486
38.2.1	Leistungen des hierarchischen Routing	487
38.2.2	NLSP – angrenzende Umgebung	487
38.2.3	Hello-Pakete im LAN verschicken	489
38.3	NLSP – Vorgehen	489
38.4	NLSP – hierarchische Adressierung	490
38.5	NLSP – Hello-Pakete	491
38.5.1	Hello-Pakete für WANs	492
38.5.2	Hello-Pakete für LANs	494

39	**Open Systems Interconnection (OSI) Routing-Protokoll**	497
39.1	Hintergrund	497
39.1.1	OSI-Netzwerk-Terminologie	498
39.2	Endsystem-zu-Zwischensystem (ES-IS)	499
39.2.1	ES-IS-Konfiguration	499
39.2.2	ES-IS-Adressierung	500
39.3	Zwischensystem-zu-Zwischensystem (IS-IS)	500
39.3.1	OSI – der Routing-Vorgang	501
39.3.2	IS-IS – Metriken	502
39.4	Integrated IS-IS	504
39.5	Interdomain Routing Protocol (IDRP)	505
39.5.1	IDRP – Terminologie	505
39.5.2	IDRP-Routing	506
40	**Open Shortest Path First (OSPF)**	507
40.1	Hintergrund	507
40.2	Routing-Hierarchie	508
40.3	SPF-Algorithmus	511
40.4	Paketformat	512
40.5	Weitere Eigenschaften von OSPF	513
41	**Resource Reservation Protocol (RSVP)**	515
41.1	Hintergrund	515
41.2	RSVP – Datenströme	516
41.2.1	RSVP – Bearbeitung von Datenströmen	518
41.3	RSVP – Dienstqualität	519
41.4	RSVP – Hochfahren der Sitzung	519
41.5	RSVP – Reservierungsmethode	519
41.5.1	Wildcard-Filter-Methode (WF)	520
41.5.2	Fixed-Filter-Methode (FF)	520
41.5.3	Shared-Explicit-Methode (SE)	521
41.5.4	Folgen der Reservierungsmethoden	521
41.6	RSVP – Soft-State-Implementierung	521
41.7	RSVP – Modell des Ablaufs	523
41.7.1	Allgemeiner Protokollablauf von RSVP	523
41.7.2	RSVP – Tunneln	524
41.8	RSVP – Nachrichten	526
41.8.1	Reservation-Request-Nachrichten	526
41.8.2	Pfadnachrichten	526
41.8.3	Fehler- und Acknowledgment-Nachrichten	526
41.8.4	Abbaunachrichten	527
41.9	RSVP – Paketformat	528
41.9.1	Felder des Nachrichten-Headers für RSVP	528
41.9.2	Objektfelder für RSVP	529
42	**Routing Information Protocol (RIP)**	533
42.1	Hintergrund	533
42.2	Routing-Updates	534

42.3 RIP – Routing-Metrik 534
42.4 RIP – Stabilitätsmerkmale 535
42.5 RIP – Timer 535
42.6 Paketformate 535
42.6.1 Paketformat von RIP 536
42.6.2 Paketformat von RIP 2 537

43 Simple Multicast Routing Protocol (SMRP) 539
43.1 Hintergrund 539
43.2 SMRP – Multicast-Transportdienste 540
43.2.1 SMRP – Verwaltung von Multicast-Adressen 542
43.2.2 SMRP – Multicast-Transaktionsprotokoll (MTP) 544
43.2.3 SMRP – Knotenverwaltung 545
43.2.4 SMRP – Multicast-Routen 547
43.2.5 SMRP – Verwaltung von Multicast-Gruppen 548
43.2.6 Weiterleiten von Multicast-Datagrammen 549
43.2.7 Umgang mit Topologieänderungen unter SMRP 550
43.3 SMRP – Beispiel für eine Transaktion 551
43.4 SMRP – Paketformat 552

Teil 7 Netzwerk-Verwaltung 555

44 IBM Netzwerk-Verwaltung 557
44.1 Hintergrund 557
44.2 IBM – Funktionale Bereiche der Netzwerk-Verwaltung 558
44.2.1 IBM – Konfigurationsverwaltung 558
44.2.2 IBM – Performance- und Accounting-Verwaltung 559
44.2.3 IBM – Problemverwaltung 559
44.2.4 IBM – Betriebsverwaltung 560
44.2.5 IBM – Änderungsverwaltung 560
44.3 IBM-Architekturen zur Netzwerk-Verwaltung 561
44.3.1 Open Network Architecture (ONA) 561
44.3.2 SystemView 563
44.4 IBM – Plattformen zur Netzwerk-Verwaltung 563
44.4.1 NetView 563
44.4.2 LAN Network Manager (LNM) 564
44.4.3 Simple Network Management Protocol (SNMP) 564

45 Remote Monitoring (RMON) 565
45.1 Hintergrund 565
45.2 RMON-Gruppen 566

46 Simple Network Management Protocol (SNMP) 569
46.1 Hintergrund 569
46.2 SNMP – Grundlegende Komponenten 570
46.3 SNMP – Grundlegende Befehle 571
46.4 SNMP – Management Information Base (MIB) 572
46.5 SNMP und Datendarstellung 573

46.6	SNMP Version 1 (SNMPv1)	574
46.6.1	SNMPv1 und Structure of Management Information (SMI)	574
46.6.2	SNMPv1 – Protokolloperationen	576
46.7	SNMP Version 2 (SNMPv2)	576
46.7.1	SNMPv2 und Structure of Management Information (SMI)	576
46.8	SNMP – Verwaltung	578
46.9	SNMP – Sicherheit	578
46.10	SNMP – Zusammenarbeit	579
46.10.1	Proxy-Agenten	579
46.10.2	»Zweisprachige« Netzwerk-Verwaltungssysteme	579
46.11	SNMP-Referenz: SNMPv1 Nachrichtenformate	580
46.11.1	SNMPv1 – Nachrichten-Header	580
46.11.2	SNMPv1 – Protokolldateneinheit (PDU)	580
46.11.3	Format von Trap PDU	581
46.12	SNMP-Referenz: SNMPv2 Nachrichtenformate	582
46.12.1	SNMPv2 – Nachrichten-Header	582
46.12.2	SNMPv2 – Protokolldateneinheit (PDU)	583
Glossar		585
Stichwortverzeichnis		683

Vorwort

Die Technologien der Datenkommunikation entwickeln sich mit enormer Geschwindigkeit weiter. Die steigende Nachfrage nach Internet-Zugriff und Internet-Dienstleistungen fördert den raschen technischen Wandel bei Anwendern und Herstellern. Unglücklicherweise führt dies bei der Erstellung einer Informationsquelle wie diesem vorliegenden Handbuch dazu, daß einige der Informationen bei Drucklegung bereits veraltet sind.

Der Ansatz der Autoren dieses Buches war es, den Lesern zu helfen, auf der Grundlage der gebotenen Informationen technologische Entscheidungen zu treffen und sich über das genannte Dilemma im klaren zu sein. Wir hoffen, daß diese erste Ausgabe ein Schritt in die richtige Richtung ist und daß Sie zusammen mit den anderen Büchern der Cisco-Press-Reihe in der Lage sein werden, auch bei sich ändernden Anforderungen diejenigen Technologien auszuwählen, die funktionierende Netzwerklösungen bereitstellen.

Ziele

Diese Publikation bietet technische Informationen zu Internetworking-Technologien auf Cisco-Basis. Es kann in Verbindung mit anderen Cisco-Handbüchern oder als eigenständige Referenz genutzt werden. Das Handbuch Netzwerktechnologien (amerikanischer Originaltitel: Internetworking Technologies Handbook) kann nicht alle Informationen zu den erwähnten Technologien liefern. Eines der Hauptziele dieser

Publikation ist es, Netzwerk-Administratoren bei der Konfiguration von Cisco-Produkten zu unterstützen. Daher legt das Buch einen Schwerpunkt auf Cisco-unterstützte Systeme; die erwähnten Technologien setzen eine Unterstützung durch Cisco jedoch nicht voraus.

Zielgruppe

Dieser Band wurde für all diejenigen geschrieben, die die Funktionsweise von Netzwerken verstehen wollen. Wir vermuten, daß die meisten Leser die Informationen aus diesem Buch nutzen werden, um die Anwendbarkeit bestimmter Technologien für ihre Netzwerkumgebungen zu bewerten.

Aufbau

Dieses Buch besteht aus sieben Teilen. Jeder Teil befaßt sich mit Grundlagen oder zentralen Bereichen der Netzwerk- und Internetworking-Technologien. Sie bestehen aus Kapiteln, die entsprechende Aufgaben und Funktionen beschreiben.

Teil 1 – »Einführung in das Internetworking« – stellt die Konzepte dar, die grundlegend für das Verständnis des Internetworking und des Netzwerk-Managements sind.

Teil 2 – »LAN-Protokolle« – beschreibt die Standardprotokolle, die für den Zugriff auf die physikalischen Medien des Netzwerks eingesetzt werden.

Teil 3 – »WAN-Technologien« – beschreibt die Standardprotokolle, die für die Einrichtung von WANs genutzt werden.

Teil 4 – »Bridging und Switching« – erläutert die Protokolle und Technologien, die für die Connectivity zwischen Subnetzwerken auf Schicht 2 genutzt werden.

Teil 5 – »Netzwerkprotokolle« – beschreibt die Standardnetzwerkprotokollstapel, die in einem Netzwerk geroutet werden können.

Teil 6 – »Routing-Protokolle« – erläutert die Protokolle, die für das Routing von Informationen in einem Internetwork genutzt werden.

Teil 7 – »Netzwerk-Verwaltung« – beschreibt die Architektur und den Betrieb von weit verbreiteten Implementierungen für das Netzwerk-Management.

Danksagung

Dieses Buch wurde im Team geschrieben. Es ist das Ergebnis mehrerer Jahre der Informationssammlung und integriert diverse Informationsquellen, die von den Entwicklern von Cisco Knowledge Products erstellt wurden. Die Hauptautoren dieser Publikation sind Merilee Ford, H. Kim Lew, Steve Spanier und Tim Stevenson. In der letzten Überarbeitungsphase haben Margaret Young und Rick Fairweather bei der Integration des Materials in dieses Buch einen wertvollen Beitrag geleistet.

Die Autoren möchten all den Cisco-Experten danken, die das Material überarbeitet und darüber hinaus mit ihrem fundierten Wissen zu den dargestellten Technologien beigetragen haben. Einen Beitrag leisteten Priscilla Oppenheimer, Aviva Garrett, Steve Lin, Manoj Leelanivas, Kent Leung, Dave Stine, Ronnie Kon, Dino Farinacci, Fred Baker, Kris Thompson, Jeffrey Johnson, George Abe, Yakov Rekhter, Abbas Masnavi, Alan Marcus, Laura Fay, Anthony Alles, David Benham, Debra Gotelli, Ed Chapman, Bill Erdman, Tom Keenan, Soni Jiandani, Derek Yeung und viele mehr. Die Autoren möchten allen danken, die ihre Zeit geopfert und wichtige Beiträge geleistet haben, um dieses Buch zu einer wertvollen Informationsquelle zu machen.

Diese Publikation bedient sich großzügig anderer Publikationen und Schulungsprodukte, die von Cisco entwickelt wurden. Insbesondere die Publikation *Internetworking Technology Overview* und die Multimedia-CD-ROM *Cisco Connection Training* bildeten die Grundlage für diese Informationssammlung.

Vorwort zur deutschen Übersetzung

Netzwerk-Planer und -Betreiber werden permanent mit den immer komplizierter werdenden Anforderungen der eingesetzten Applikationen hinsichtlich deren Bedarf an Bandbreite und Services konfrontiert. Welche dieser Anforderungen stellt die Weichen, mit welchen Technologien ein Backbone geplant bzw. betrieben wird? Was sind die Entscheidungsgrundlagen bei der Auswahl der Kommunikations- bzw. Routing-Protokolle? Welche Sicherheitsaspekte sollten berücksichtigt werden? Und wie integriert man Mainframes und ihre Großrechnerwelten in die Datenkommunikation von Enterprise-Netzen? Gerade die Skalierbarkeit von großen unternehmensweiten Backbones stellt heute eines der größten Probleme im modernen Netzdesign dar. Speziell in diesem Bereich machen die heutigen Kommunikationsformen, Applikationen und Serviceanforderungen den Einsatz völlig neuer Technologien notwendig, um diesen Anforderungen gerecht zu werden.

Bisher war Cisco's Internetworking-Know-how hauptsächlich unseren Kunden zugänglich. Nun möchten wir mit der Gründung des Cisco Presse Forums neue Wege beschreiten, um unser Expertenwissen mit Ihnen zu teilen.

Unser Ziel ist es dabei, eine komplette Fachbibliothek von Publikationen zum Thema Internetworking zu erstellen. In diesen Veröffentlichungen sollen praxisorientierte, nützliche Tips über Design und Implementation von Routern, Switches, Access-Servern und netzübergreifenden Software-Lösungen im Vordergrund stehen.

»Netzwerk-Technologien« ist das zweite Buch dieser Reihe. Darin möchten wir Ihnen einen tiefen Einblick in die Gesetze und Grundlagen des Internetworking, die Anforderungen und Einsatzmöglichkeiten von Netzwerk-Protokollen und die Umsetzung von umfassenden Netzkonzepten geben. Es beschreibt ausführlich die Grundlagen und Problematiken, mit denen sich jeder Netzdesigner auseinandersetzen muß, der Netzwerke plant oder realisiert. Dabei fließen sowohl unsere langjährige Erfahrung als auch nützliche Tips aus der Praxis bei Design- und Implementationsproblematiken von Netzstrukturen ein.

»Netzwerk-Technologien« ist ein Buch, das Ihnen leicht verständlich die wichtigsten Technologie-, Design- und Implementationsrichtlinien des Internetworking erläutert. Es kann durch seine übersichtliche Strukturierung jederzeit als Nachschlagewerk für technologische Fachbegriffe und netzwerkspezifische Fachfragen genutzt werden.

Wir hoffen, daß diese Publikation auch eine Bereicherung für Ihre Netzwerk-Bibliothek ist.

Ralf Kothe

Product Marketing Manager
Cisco Systems GmbH

Kapitel 1: Grundlagen des Internetworking
Kapitel 2: Einführung in die LAN-Protokolle
Kapitel 3: Einführung in die WAN-Technologien
Kapitel 4: Grundlagen des Bridging und Switching
Kapitel 5: Grundlagen des Routing
Kapitel 6: Grundlagen des Netzwerk-Managements

TEIL 1

Einführung in das Internetworking

Teil 1, »Einführung in das Internetworking«, gibt einen groben Überblick zu den Technologien des Internetworking und zur Terminologie. In den einzelnen Kapiteln werden folgende Themen besprochen:

Grundlagen des Internetworking – Im ersten Kapitel werden grundlegende Dinge des Internetworking, einschließlich des OSI-Modells, der Adressierung, Netzwerk-Services behandelt.

Einführung zu den LAN-Protokollen – Dieses Kapitel bietet einen Überblick zu den gängigen LAN-Protokollen der Sicherungsschicht (auch Verbindungsschicht) und der physischen Schicht.

Einführung in die WAN-Technologien – In diesem Kapitel werden WAN-Protokolle, -Geräte und -Implementationen im Überblick dargestellt.

Grundlagen des Bridging und Switching – In diesem Kapitel wird ein Überblick zu den Bridging- und Switching-Technologien gegeben.

Grundlagen des Routing – Eine Einführung in die Routing-Protokolle.

Grundlagen des Netzwerk-Management – Dieses Kapitel bietet einen groben Überblick über das Netzwerk-Management und OSI-Netzwerk-Management-Modell.

KAPITEL 1

Grundlagen des Internetworking

Dieses Kapitel stellt die grundlegenden Konzepte und Begriffe aus dem Bereich des Internetworking vor. So wie das Buch als Ganzes dem Verständnis der modernen Netzwerk-Technik dient, werden in diesem Kapitel einige Themen, die im allgemeinen bekannt sind, nur kurz behandelt. Dazu gehören die Flußsteuerung, die Fehlerprüfung und das Multiplexing, jedoch wird der Schwerpunkt auf die Darstellung des OSI-Modells (*Open System Interconnect*) im Zusammenhang mit Netzwerk-/Internetworking-Funktionen gelegt. Es wird ein Überblick zu Adressierungsverfahren in Hinblick auf das OSI-Modell gegeben.

1.1 Was ist ein Internetwork?

Ein Internetwork besteht aus mehreren einzelnen Netzwerken, die über dazwischengeschaltete Netzwerk-Geräte miteinander verbunden sind, so daß ein großes Netzwerk entsteht. Internetworking bezieht sich auf die Industrie, Produkte und Verfahren, die es ermöglichen, ein Internetwork aufzubauen und zu administrieren. Bild 1.1 zeigt verschiedene Netzwerk-Technologien, die über Router oder andere Netzwerk-Geräte miteinander zu einem Internetwork verbunden werden können:

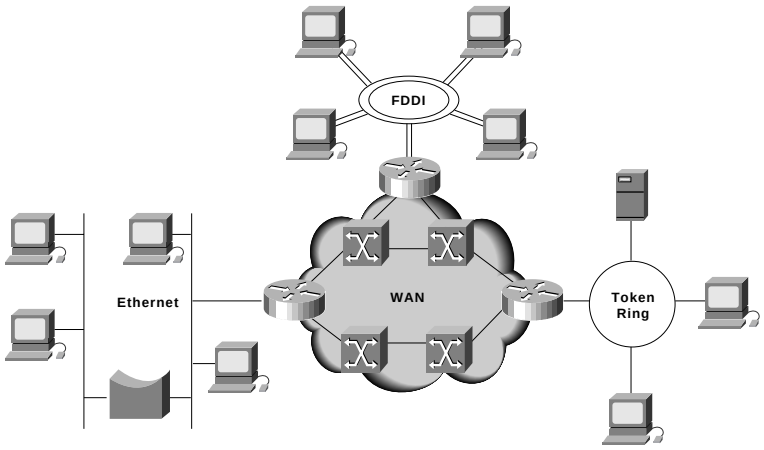


Bild 1.1: Verschiedene Netzwerk-Technologien können zu einem Internetwork verbunden werden

1.1.1 Die Geschichte des Internetworking

Die ersten Netzwerke waren Netzwerke im Teilnehmerbetrieb, bei denen Großrechner (Mainframes) mit angeschlossenen Terminals zum Einsatz kamen. Solche Umgebungen wurden mit der System Network Architecture (SNA) von IBM und der Netzwerk-Architektur von Digital eingerichtet.

Anmerkung des Übersetzers: Hier scheint man sich auf die nordamerikanische Welt zu beschränken, denn Siemens hat/hatte mit seinem Großrechner-Betriebssystem BS2000 und dem zugehörigen Transdata-Netz gleiches zu bieten.

Lokale Netzwerke (Local Area Network – LAN) entstanden während der PC-Revolution. In einem LAN können viele Benutzer, die räumlich nicht zu weit voneinander entfernt waren, Dateien und Nachrichten austauschen und auf Ressourcen gemeinsam zugreifen, z.B. auf Datei-Server.

Weitverkehrsnetze (Wide Area Network – WAN) verbinden LANs miteinander über normale Telefonleitungen (oder andere Medien), wobei die Benutzer (resp. die LANs) räumlich weiter voneinander entfernt sind.

Heute sind Hochgeschwindigkeits-LANs und vermittelte Internetworks weitverbreitet und häufig im Einsatz, weil die Übertragungsgeschwindigkeiten sehr hoch sind und sehr Band-

breite-intensive Anwendungen wie Telefon- und Video-Konferenzen unterstützt werden.

Das Internetworking entwickelte sich als Lösung der folgenden drei Schlüsselprobleme: voneinander isolierte LANs, doppelte Haltung von Ressourcen und fehlende Netzwerk-Verwaltung. Aufgrund der abgeschlossenen LANs war es unmöglich, daß verschiedene Büros oder Abteilungen elektronisch miteinander kommunizierten. Bei der doppelten Haltung von Ressourcen mußte die gleiche Hard- bzw. Software in jedem Büro und in jeder Abteilung vorhanden sein und vom eigenen Support eingerichtet und verwaltet werden. Für die Netzwerk-Verwaltung gab es keine zentralisierte Verwaltung und Verfahren für die Fehlerbehebung.

1.1.2 Die Herausforderungen des Internetworking

Ein funktionierendes Internetwork zu implementieren, ist keine leichte Aufgabe. Dabei sind eine Vielzahl von Anforderungen zu berücksichtigen, insbesondere was die Connectivity, die Zuverlässigkeit, das Netzwerk-Management und die Flexibilität betrifft. Jeder dieser Bereiche ist ein Schlüssel für den Aufbau eines effizienten und effektiven Internetwork.

Eine der Herausforderungen beim Verbinden verschiedener Systeme stellt die Unterstützung der Kommunikation zwischen unterschiedlichen Technologien dar. So können in verschiedenen Niederlassungen z.B. verschiedene Medien verwendet werden, oder die Datenübertragung erfolgt in verschiedenen Raten.

Eine weitere wichtige Überlegung betrifft die Betriebszuverlässigkeit eines Internetwork. Sowohl einzelne Benutzer als auch ein Unternehmen benötigen konsistenten, zuverlässigen Zugriff auf die Netzwerk-Ressourcen.

Des weiteren muß das Netzwerk-Management über einen zentralen Support und Möglichkeiten der Fehlerbeseitigung im Internetwork verfügen. Die Konfiguration, Sicherheit, Performance und weitere Anforderungen müssen entsprechend bedient werden, um die Funktionalität eines Internetwork gewährleisten zu können.

Flexibilität, als abschließende Überlegung, ist für den Fall der Netzwerk-Erweiterung, neuer Anwendungen und Dienste unabdingbar.

1.2 Das OSI-Referenzmodell

Das OSI-Referenzmodell (Open System Interconnection – OSI) beschreibt den Weg, den Daten von einer Software-Anwendung auf einem Computer über das Netzwerk-Medium bis zu einer Anwendung auf einem anderen Rechner nehmen. Das OSI-Referenzmodell ist ein Konzept, das diesen Weg in sieben Schichten unterteilt, die jeweils eine einzelne Netzwerkfunktion spezifizieren. Dieses Modell wurde 1984 von der International Standardization Organization (ISO) entwickelt und wird heute als das primäre Architekturmodell für die Kommunikation zwischen Computern betrachtet. Das OSI-Modell unterteilt die Aufgaben, die beim Transport von Daten über ein Netzwerk anfallen, in sieben kleinere, überschaubare Gruppen. Dabei werden jeder der sieben OSI-Schichten eine Aufgabe oder eine Gruppe von Aufgaben zugeordnet. Jede Schicht ist sinnvollerweise in sich abgeschlossen, so daß die der Schicht zugeordnete Aufgabe unabhängig von anderen implementiert werden kann. So kann die Implementation für eine Schicht aktualisiert werden, ohne daß andere Schichten davon betroffen sind. Die folgende Liste nennt die sieben Schichten des OSI-Referenzmodells:

Schicht 7	application layer	Anwendungsschicht (auch: Applikationsschicht)
Schicht 6	presentation layer	Darstellungsschicht (auch: Datendarstellungsschicht, Präsentationsschicht)
Schicht 5	session layer	Kommunikationsschicht (auch: Kommunikationssteuerungsschicht, Sitzungsschicht)
Schicht 4	transport layer	Transportschicht
Schicht 3	network layer	Vermittlungsschicht (auch: Netzwerkschicht)
Schicht 2	data link layer	Sicherungsschicht (auch: Verbindungs-(sicherungs-)schicht)
Schicht 1	physical layer	Physikalische Schicht (oft auch: physische Schicht)

Bild 1.2 stellt das siebenschichtige OSI-Referenzmodell grafisch dar.

Bild 1.2: Das OSI-Referenzmodell besteht aus sieben voneinander unabhängigen Schichten

1.2.1 Eigenschaften der OSI-Schichten

Die sieben Schichten des OSI-Referenzmodells können in zwei Gruppen unterteilt werden: höhere bzw. *obere Schichten* und niedrige bzw. *untere Schichten*.

Die *oberen Schichten* des OSI-Modells betreffen Anwendungen und sind im allgemeinen nur als Software implementiert. Die höchste Schicht, die Anwendungsschicht, ist die dem Benutzer nächste Schicht. Sowohl die Benutzerschicht als auch die Anwendungsschicht interagieren mit Software-Anwendungen, die eine Kommunikationskomponente enthalten. Wenn von höherer Schicht die Rede ist, dann wird damit manchmal auch nur die nächsthöhere Schicht im OSI-Modell bezeichnet.

Die *unteren Schichten* eines OSI-Modells betreffen den Datentransport. Die physische Schicht und die Verbindungsschicht sind als Hard- und Software implementiert. Die restlichen unteren Schichten sind nur als Software implementiert. Die unterste Schicht, die physische, ist dem physischen Netzwerk-Medium (z.B. der Netzwerk-Verkabelung) am nächsten und ist für das Absetzen der Daten auf das Medium zuständig.

Bild 1.3 zeigt die Unterteilung in untere und obere OSI-Schichten.

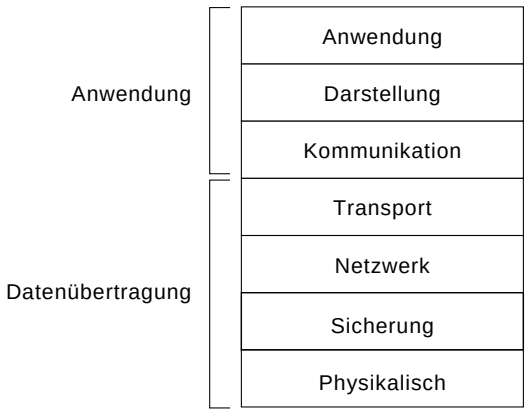


Bild 1.3: Die OSI-Schichten lassen sich in zwei Gruppen unterteilen

1.2.2 Protokolle

Das OSI-Modell bietet einen konzeptionellen Rahmen für die Kommunikation zwischen Computern, wobei das Modell an sich keine Methode für die Kommunikation ist. Die tatsächliche Kommunikation wird erst durch den Einsatz von Protokollen möglich. Im Kontext von Datennetzwerken ist ein *Protokoll* eine formale Zusammenstellung von Regeln und Konventionen, mit denen der Austausch von Daten zwischen Computern über ein Netzwerk-Medium geregelt wird. Mit einem Protokoll werden die Funktionen einer oder mehrerer OSI-Schichten implementiert. Es gibt eine Vielzahl an Kommunikationsprotokollen, die sich jedoch alle in eine der folgenden Gruppen einordnen lassen: *LAN-Protokolle*, *WAN-Protokolle*, *Netzwerk-Protokolle* und *Routing-Protokolle*. *LAN-Protokolle* arbeiten auf der Ebene der physischen und der Verbindungsschicht des OSI-Modells und definieren die Kommunikation über verschiedene LAN-Medien. *WAN-Protokolle* arbeiten auf der Ebene der drei untersten Schichten des OSI-Modells und definieren die Kommunikation über die verschiedenen Weitverkehrsmedien. *Routing-Protokolle* sind Protokolle der Vermittlungsschicht, die die Pfadfestlegung und das Verkehrs-Switching regeln. Die *Netzwerk-Protokolle* schließlich sind verschiedene Protokolle der oberen Schichten, die zu einer bestimmten Protokollfamilie gehören.

1.2.3 Das OSI-Modell und die Kommunikation zwischen Systemen

Daten, die von einer Software-Anwendung des einen Computers zur Software-Anwendung eines anderen Computers übertragen werden sollen, müssen jede OSI-Schicht passieren. Wenn z.B. eine Anwendung auf System A Daten an eine Anwendung auf System B übertragen will, übergibt das Anwendungsprogramm auf System A diese Daten an die Anwendungsschicht (Schicht 7) des Systems A. Dann reicht die Anwendungsschicht die Daten an die Darstellungsschicht (Schicht 6) weiter, die die Daten der Kommunikationsschicht (Schicht 5) übergibt usw. bis hinunter zur physischen Schicht (Schicht 1). Von der physischen Schicht werden die Daten in das physische Netzwerk-Medium eingespeist und darüber zum System B übertragen. Die physische Schicht des Systems B entfernt vom physischen Medium hinzugefügte Daten und gibt seine Daten an die Verbindungsschicht (Schicht 2) weiter, von wo sie an die Vermittlungsschicht (Schicht 3) übergeben werden usw. bis hinauf zur Anwendungsschicht (Schicht 7) des Systems B. Die Anwendungsschicht des Systems B schließlich übergibt die Daten an das Programm, das als Empfänger bestimmt ist, womit der Kommunikationsprozeß abgeschlossen ist.

1.2.4 Interaktion zwischen den Schichten des OSI-Modells

Jede Schicht des OSI-Modells kommuniziert mit drei anderen OSI-Schichten: mit der nächsthöheren Schicht, mit der nächstniedrigeren und mit der gleichen Schicht auf dem anderen, vernetzten Computer. So kommuniziert z.B. die Verbindungsschicht des Systems A mit der Vermittlungs- und der physischen Schicht seines Systems und mit der Verbindungsschicht des Systems B. Bild 1.4 veranschaulicht dieses Beispiel.

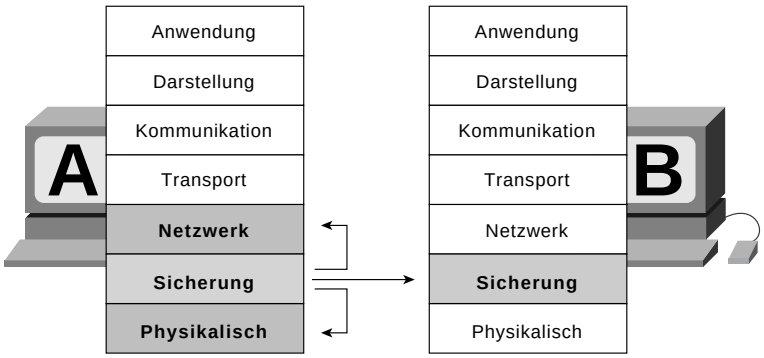


Bild 1.4: Die Schichten des OSI-Modells kommunizieren mit anderen Schichten

1.2.5 Dienste der OSI-Schichten

Die aufeinanderfolgenden OSI-Schichten kommunizieren miteinander, um ihre Dienste gegenseitig zu nutzen. Die Dienste der benachbarten Schichten dienen zur Kommunikation zwischen einer OSI-Schicht und der gleichen Schicht auf einem anderen Computer. Zu den Diensten der Schichten zählen drei grundlegende Elemente: der Dienstbenutzer, der Dienstanbieter und der Dienstzugriffspunkt (Service Access Point – SAP).

In diesem Zusammenhang ist der Dienst*benutzer* die OSI-Schicht, die von der benachbarten Schicht einen Dienst anfordert. Der Dienst*anbieter* ist die Schicht, die diesen Dienst dem Dienstbenutzer anbietet. OSI-Schichten können Dienste mehreren Dienstbenutzern gleichzeitig anbieten. Beim *SAP* handelt es sich um einen konzeptionell bestimmten Ort, an dem eine OSI-Schicht den Dienst einer anderen Schicht anfordern kann.

Bild 1.5 stellt dar, wie diese drei Elemente auf der Vermittlungs- und Verbindungsschicht interagieren.

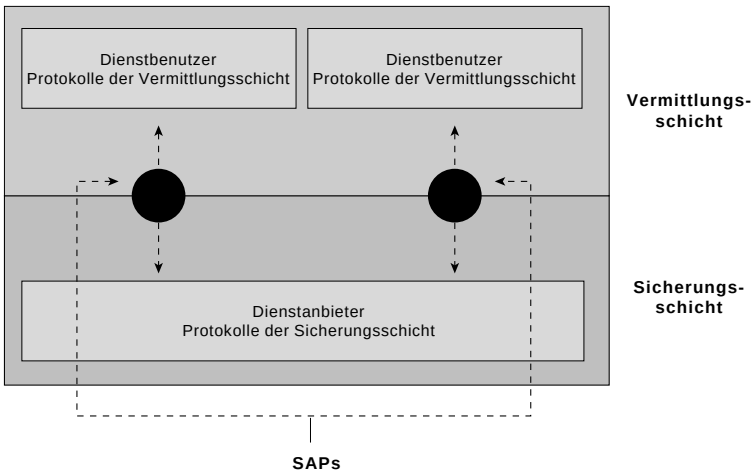


Bild 1.5: Dienstbenutzer, -anbieter und SAPs interagieren auf Ebene der Vermittlungs- und Verbindungsschicht

1.2.6 Schichten des OSI-Modells und der Datenaustausch

Die sieben OSI-Schichten setzen verschiedene Formen von Steuerdaten ein, um mit der gleichen Schicht auf einem anderen Computer zu kommunizieren. Diese *Steuerdaten* bestehen aus bestimmten Anforderungen und Anweisungen, die zwischen Partner-Schichten des OSI-Modells ausgetauscht werden.

Für Steuerdaten gibt es zwei Formen: den Header (Kopf) oder Trailer (Anhang). Der Header wird den von der höheren Schicht heruntergereichten Daten vorangestellt. Ein Trailer wird diesen Daten angehängt. Es ist allerdings nicht unbedingt erforderlich, daß eine OSI-Schicht den heruntergereichten Daten einen Header oder Trailer hinzufügt.

Die Bezeichnung Header, Trailer und Daten ist relativ zu verstehen, in Abhängigkeit von der Schicht, die die Informationseinheit analysiert. Für die Vermittlungsschicht besteht die Informationseinheit z.B. aus dem Header der Schicht 3 und den Daten. Auf Ebene der Verbindungsschicht wird jedoch alles, was von der Vermittlungsschicht heruntergereicht wird, als Daten behandelt (also der Header der Schicht 3 und die Daten).

Mit anderen Worten: Der Datenteil einer Informationseinheit auf Ebene einer bestimmten OSI-Schicht kann die Header, Trailer und Daten sämtlicher darüberliegenden Schichten enthalten. Genau dies wird als *Kapselung* bezeichnet. Bild 1.6 zeigt, wie Header und Daten einer Schicht vom Header der darunterliegenden Schicht gekapselt werden.

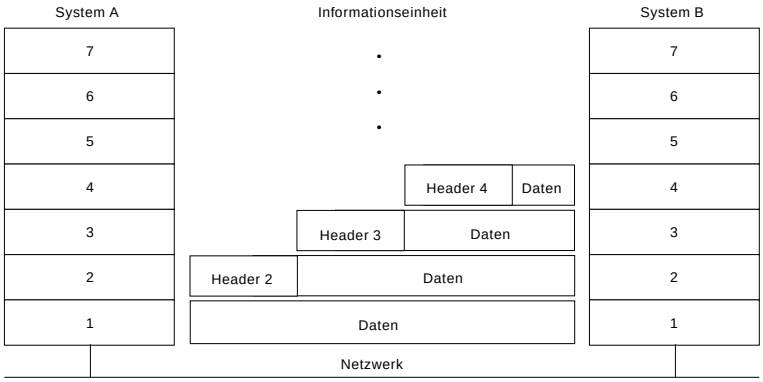


Bild 1.6: Header und Daten können während des Datenaustauschs gekapselt werden

Verfahren des Datenaustauschs

Der Datenaustausch erfolgt zwischen den Partnerschichten des OSI-Modells. Jede Schicht des Quellsystems fügt den eigentlichen Daten seine Steuerdaten hinzu. Jede Schicht des Zielsystems analysiert die Steuerdaten und entfernt diese wieder.

Wenn System A Daten einer Anwendung an System B senden will, müssen die Daten an die Anwendungsschicht übergeben werden. Die Anwendungsschicht des Systems A transportiert dann alle von der Anwendungsschicht des Systems B angeforderten Steuerdaten, indem sie den Daten einen Header voranstellt. Die sich daraus ergebende Informationseinheit (Header + Daten) wird an die Darstellungsschicht weitergegeben, die wiederum ihren eigenen Header voranstellt, der Steuerinformationen für die Darstellungsschicht des Systems B enthält. Die Informationseinheit wächst von Schicht zu Schicht mit Steuerdaten für die Partnerschichten im System B an, da jede Schicht ihren Header anfügt (oder ihren Trailer). Von der physischen Schicht wird die gesamte Informationseinheit in das Netzwerk-Medium eingespeist.

Die physische Schicht des Systems B empfängt die Informationseinheit und reicht sie an die Verbindungsschicht. Diese

liest die Steuerdaten aus dem Header, der von der Verbindungsschicht des Systems A hinzugefügt wurde. Der Header wird entfernt und der Rest der Informationseinheit wird an die Vermittlungsschicht übergeben. Jede Schicht führt die gleichen Aktionen aus: Sie liest den Header der Partnerschicht, entfernt diesen und übergibt den Rest der Informationseinheit an die nächsthöherliegende Schicht. Nachdem die Anwendungsschicht diese Aktionen ausgeführt hat, werden die Daten an die Anwendung des Systems B übergeben, für die die Daten bestimmt sind, und zwar genau in der Form, wie sie von der Anwendung des Systems A übertragen wurden.

1.2.7 Die physische Schicht des OSI-Modells

Die physische Schicht definiert die elektrische, mechanische, prozedurale und funktionale Spezifikation für die Aktivierung, Aufrechterhaltung und Deaktivierung der physischen Verbindung zwischen kommunizierenden Netzwerk-Systemen. Die Spezifikationen der physischen Schicht betreffen Eigenschaften wie die Spannung, zeitliche Vorgaben für Spannungsänderungen, physische Übertragungsraten, die maximale Übertragungsstrecke und die Stecker und Buchsen. Die Implementationen der physischen Schicht können unterteilt werden in LAN- oder WAN-Spezifikationen. Bild 1.7 zeigt einige der gängigsten LAN- und WAN-Implementationen der physischen Schicht.

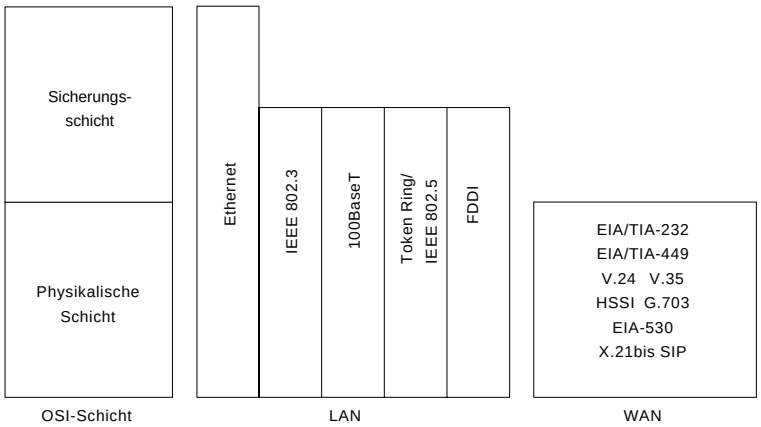


Bild 1.7: Implementationen der physischen Schicht können LAN- oder WAN-Spezifikationen sein

1.2.8 Die Verbindungsschicht des OSI-Modells

Die Verbindungsschicht sorgt für die zuverlässige Übertragung der Daten über eine physische Netzwerk-Verbindung. Die verschiedenen Spezifikationen der Verbindungsschicht definieren unterschiedliche Netzwerk- und Protokolleigenschaften, einschließlich der physischen Adressierung, Netzwerk-Topologie, Fehlererkennung, Frame-Abfolge und Flußsteuerung. Bei der physischen Adressierung (im Gegensatz zur Netzwerk-Adressierung) wird definiert, wie Geräte auf Ebene der Verbindungsschicht adressiert werden. Die Netzwerk-Topologie wird von Spezifikationen der Verbindungsschicht bestimmt, die definiert, wie Geräte physisch miteinander verbunden werden, z.B. über einen Bus oder einen Ring. Die Fehlererkennung alarmiert die Protokolle der oberen Schichten, wenn ein Übertragungsfehler auftritt, und die Sequenzierung der Daten-Frames sortiert Frames, falls diese nicht in der richtigen Reihenfolge eingehen. Die Flußsteuerung schließlich regelt die Übertragung der Daten, so daß das empfangende Gerät nicht mit Daten überlastet wird.

Das Institute of Electrical and Electronics Engineers (IEEE) hat die Verbindungsschicht noch in zwei Subschichten aufgeteilt: die Logical Link Control (LLC – logische Verbindungssteuerung) und die Media Access Control (MAC – Medium-Zugriffssteuerung). Bild 1.8 zeigt die IEEE-Subschichten der Verbindungsschicht.

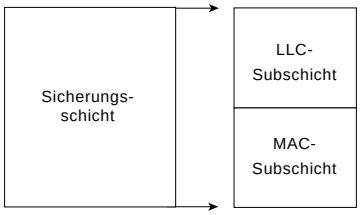


Bild 1.8: Die Verbindungsschicht besteht aus zwei Subschichten

Die Subschicht Logical Link Control (LLC) verwaltet die Kommunikation zwischen Geräten eines Netzwerks, die über eine einzige Leitung läuft. LLC ist in der Spezifikation IEEE 802.2 definiert und unterstützt sowohl die verbindungslosen als auch die verbindungsorientierten Dienste der Protokolle höherer Schichten. IEEE 802.2 definiert mehrere Felder in Frames der Verbindungsschicht, so daß es mehreren Protokol-

len der höheren Schichten möglich ist, eine einzige physische Datenverbindung gemeinsam zu nutzen. Die Subschicht Media Access Control (MAC) verwaltet den Protokollzugriff auf das physische Netzwerk-Medium. Die IEEE MAC-Spezifikation definiert MAC-Adressen, anhand derer mehrere Geräte auf Ebene der Verbindungsschicht eindeutig identifizierbar sind.

1.2.9 Die Vermittlungsschicht des OSI-Modells

Die Vermittlungsschicht bietet Routing- und ähnliche Funktionen, mit denen es möglich ist, mehrere Datenverbindungen in einem Internetwork zu kombinieren. Dies wird durch die logische Adressierung der Geräte (im Gegensatz zur physischen Adressierung) erreicht. Die Vermittlungsschicht unterstützt sowohl verbindungslose als auch verbindungsorientierte Dienste der Protokolle höherer Schichten. Die Protokolle der Vermittlungsschicht sind Routing-Protokolle; es werden aber auch andere Protokolle in dieser Schicht implementiert.

Einige der gängigsten Routing-Protokolle enthalten auch das Border Gateway Protocol (BGP), das ein Internet-Interdomain-Routing-Protokoll ist; Open Shortest Path First (OSPF) ist ein verbindungsorientiertes inneres Gateway-Protokoll, das für den Einsatz in TCP/IP-Netzen entwickelt wurde; und das Routing Information Protocol (RIP) ist ein Internet-Routing-Protokoll, das Sprungzähler (hop count) verwendet.

1.2.10 Die Transportschicht des OSI-Modells

Die Transportschicht implementiert Dienste für den zuverlässigen Datentransport im Internetwork, die für die höheren Schichten transparent sind. Zu den Funktionen der Transport-Schicht gehören die Flußsteuerung, das Multiplexing, die Verwaltung der virtuellen Verbindungen und die Fehlerprüfung und -behebung.

Die Flußsteuerung verwaltet die Datenübertragung zwischen Geräten, so daß das sendende Gerät nicht mehr Daten übermittelt als das empfangende Gerät verarbeiten kann. Mit dem Multiplexing können mehrere Anwendungen ihre Daten über eine einzelne physische Verbindung übertragen. Virtuelle Verbindungen werden von der Transportschicht aufgebaut, aufrechterhalten und abgebaut. Zur Fehlerprüfung gehört das

Aufsetzen von Mechanismen zur Fehlererkennung in der Datenübertragung, während es Aufgabe der Fehlerbehebung ist, daß fehlerhafte Daten ggf. erneut angefordert werden.

Manche Implementationen der Transportschicht umfassen auch das Transmission Control Protocol, Name Binding Protocol und OSI-Transport-Protokolle. Das Transmission Control Protocol (TCP) ist ein Protokoll der TCP/IP-Familie, die für zuverlässige Datenübertragung sorgt. Das Name Binding Protocol (NBP) ist ein Protokoll, das AppleTalk-Namen mit Adressen verknüpft. OSI-Transport-Protokolle gehören zu einer Familie von Transport-Protokollen aus der OSI-Protokoll-Familie.

1.2.11 Die Kommunikationsschicht des OSI-Modells

Diese baut Sitzungen zwischen Einheiten der Darstellungsschicht Kommunikationsverbindungen auf, verwaltet und beendet diese. Kommunikationsverbindungen bestehen aus Dienstanforderungen und Dienstantworten, die zwischen Anwendungen in verschiedenen Netzwerk-Geräten ausgetauscht werden. Diese Anforderungen und Antworten werden von Protokollen koordiniert, die in der Kommunikationsschicht implementiert sind. Einige Implementationen der Kommunikationsschicht enthalten das Zone Information Protocol (ZIP), das AppleTalk-Protokoll, das den Name-Binding-Prozeß koordiniert, und das Session Control Protocol (SCP), das Vermittlungsschicht-Protokoll der DECnet Phase IV.

1.2.12 Die Darstellungsschicht des OSI-Modells

Die Darstellungsschicht bietet eine Vielzahl von Kodier- und Konvertier-Funktionen, die auf Daten der Anwendungsschicht angewandt werden. Diese Funktionen stellen sicher, daß die Daten, die von der Anwendungsschicht des einen Systems von der Anwendungsschicht eines anderen Systems gelesen werden können. Zu den Kodier- und Konvertier-Verfahren der Darstellungsschicht gehören z.B. die gängigen Datendarstellungsformate, die Konvertierung von Zeichendarstellungsformaten, gängige Datenkompressionsverfahren und übliche Datenverschlüsselungen.

Übliche Datendarstellungsformate oder der Einsatz von standardisierten Bild-, Klang- und Videoformaten ermöglichen den Austausch von Anwendungsdaten zwischen verschiedenartigen Computersystemen. Konvertierungsverfahren werden dazu verwendet, Daten zwischen Systemen auszutauschen, die mit unterschiedlichen Text- und Datendarstellungen arbeiten, z.B. mit EBCDIC und ASCII. Mit Standard-Datenkompressionen können Daten, die an der Quelle komprimiert wurden, problemlos vom Zielgerät dekomprimiert werden. Gleiches gilt für den Einsatz von Verschlüsselungen.

Die Implementationen der Darstellungsschicht sind normalerweise nicht mit einem bestimmten Protokoll-Stack verbunden. Zu den weitverbreiteten Standards für Video gehören QuickTime und Motion (MPEG). Bei QuickTime handelt es sich um eine Spezifikation für Video- und Audiodaten am Apple Computer, während MPEG ein Standard für die Komprimierung und Kodierung von Videodaten ist.

Zu den bekanntesten Grafikformaten zählen Graphics Interchange Format (GIF), Joint Photographic Experts Group (JPEG) und Tagged Image File Format (TIFF). GIF und JPEG sind ein Standard für die Komprimierung und Kodierung von Grafikdaten, TIFF ist ein Standard-Kodierformat für Grafikdaten.

1.2.13 Die Anwendungsschicht des OSI-Modells

Die Anwendungsschicht des OSI-Modells ist dem Benutzer am nähesten, d.h., daß die Anwendungsschicht und der Benutzer direkt mit der Software interagieren.

Diese Schicht interagiert mit Software-Anwendungen, die eine kommunizierende Komponente implementieren. Solche Anwendungen gehören nicht mehr in den Rahmen des OSI-Modells. Zu den Funktionen der Anwendungsschicht gehören die Identifizierung des Kommunikationspartners, das Ermitteln der Ressourcen-Verfügbarkeit und die Synchronisierung der Kommunikation.

Beim Identifizieren der Kommunikationspartner ermittelt die Anwendungsschicht für die Anwendung, die Daten übertragen will, die Identität und Verfügbarkeit des Kommunikationspartners. Bei der Ermittlung der Ressourcen-Verfügbarkeit

muß die Anwendungsschicht entscheiden, ob ausreichende Netzwerk-Ressourcen für die angeforderte Kommunikation vorhanden sind. Von der Synchronisierung ist jegliche Kommunikation zwischen Anwendungen betroffen, die von der Anwendungsschicht koordiniert werden muß.

Zwei Arten von Implementierungen der Anwendungsschicht sind TCP/IP-Anwendungen und OSI-Anwendungen. TCP/IP-Anwendungen sind Protokolle (z.B. Telnet, File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP)), die zur Internet-Protokoll-Familie zählen. OSI-Anwendungen sind Protokolle (z.B. File Transfer, Access und Management (FTAM), Virtual Terminal Protocol (VTP) und Common Management Information Protocol (CMIP)), die zur OSI-Familie zählen.

1.3 Datenformate

Die Daten und Steuerdaten, die in Internetworks übertragen werden, können in den unterschiedlichsten Formaten vorliegen. Die Begriffe, mit denen diese Formate bezeichnet werden, sind nicht immer konsistent und können gegeneinander ausgetauscht werden. Zu den gängigen Datenformaten gehören Frame-, Paket-, Datagramm-, Segment-, Nachrichten-, Zell- und Dateneinheiten. Ein Frame ist eine Informationseinheit, deren Quelle und Ziel Entitäten der Verbindungsschicht sind. Ein Frame setzt sich aus dem Header der Verbindungsschicht (und ggf. einem Trailer) und den Daten der höheren Schicht zusammen. Header und Trailer enthalten Steuerdaten, die für die Verbindungsschicht des Zielsystems bestimmt sind. Die Daten der höheren Schicht werden vom Header und Trailer der Verbindungsschicht gekapselt. Bild 1.9 zeigt die grundlegenden Komponenten eines Frame der Verbindungsschicht.

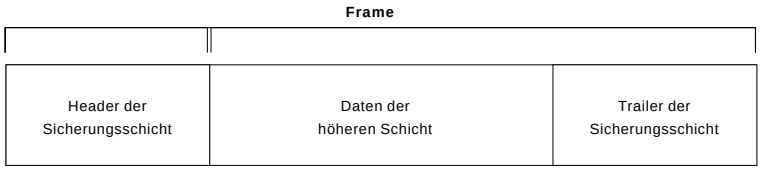


Bild 1.9: Die Daten der höheren Schichten bilden den Frame der Verbindungsschicht

Ein Paket ist eine Informationseinheit, deren Quelle und Ziel die Vermittlungsschicht ist. Ein Paket setzt sich aus dem Header der Vermittlungsschicht (und ggf. einem Trailer) und den Daten der höheren Schicht zusammen. Header und Trailer enthalten Steuerdaten, die für die Vermittlungsschicht des Zielsystems bestimmt sind. Die Daten der höheren Schichten sind vom Header und Trailer der Vermittlungsschicht gekapselt. Bild 1.10 zeigt die grundlegenden Komponenten eines Pakets der Vermittlungsschicht.

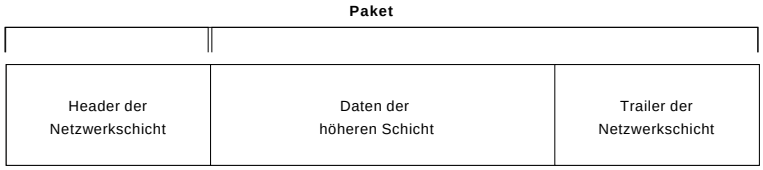


Bild 1.10: Drei grundlegende Komponenten bilden ein Paket der Vermittlungsschicht

Der Begriff *Segment* wird normalerweise auf Informationseinheiten bezogen, deren Quelle und Ziel Entitäten der Transportschicht sind.

Eine *Nachricht* ist eine Informationseinheit, deren Quelle und Ziel Entitäten sind, die oberhalb der Vermittlungsschicht liegen (oft die Anwendungsschicht).

Eine *Zelle* ist eine Informationseinheit mit fester Größe, deren Quelle und Ziel die Verbindungsschicht ist. Zellen werden in vermittelten Umgebungen verwendet, z.B. in Netzen mit Asynchronous Transfer Mode (ATM) und Switched Multimegabit Data Service (SDMS). Eine Zelle besteht aus einem Header und den Nutzdaten. Der Header enthält Steuerdaten, die für die Verbindungsschicht des Zielsystems bestimmt sind; der Header ist 5 Byte lang. Zu den Nutzdaten gehören die Daten der höheren Schicht (48 Byte), die vom Zell-Header gekapselt werden.

Die Länge des Header- und Nutzdatenfelds ist für alle Zellen immer gleich. Bild 1.11 zeigt die Komponenten einer typischen Zelle.

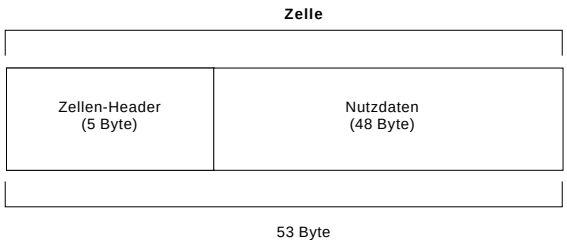


Bild 1.11: Eine typische Zelle besteht aus zwei Komponenten

Dateneinheit ist ein Oberbegriff, der eine Vielzahl von Informationseinheiten umfaßt. Zu den gängigen Dateneinheiten gehören Service Data Units (SDUs), Protocol Data Units und Bridge Protocol Data Units (BPDU). SDUs sind Informationseinheiten der Protokolle höherer Schichten, mit denen Dienstanforderungen an die niedrigeren Schichten gestellt werden. PDU ist der OSI-Begriff für ein Paket. BPDUs werden vom Spanning-Tree-Algorithmus als Hallo-Nachricht verwendet.

1.4 Die ISO-Hierarchie von Netzwerken

Große Netzwerke sind normalerweise hierarchisch organisiert. Ein solcher Aufbau bietet verschiedene Vorteile: beim Management, der Flexibilität und der Reduzierung überflüssigen Datenverkehrs. Deshalb hat die International Organization for Standardization (ISO) eine Vielzahl terminologischer Konventionen für die adressierenden Einrichtungen eines Netzwerks angepaßt. Schlüsselwörter in diesem Abschnitt sind *End-System* (ES), *intermediäres System* (IS), *Bereich* und *autonomes System* (AS).

Ein *ES* ist ein Netzwerk-Gerät, das keinerlei Routing- oder Weiterleitungsfunktion ausführt. Zu den ES gehören z.B. Terminals, Personal Computer und Drucker. Ein IS ist ein Netzwerk-Gerät, das Routing- oder Weiterleitungsfunktionen ausführt. Zu den IS gehören z.B. Router, Switches und Bridges. IS-Netzwerke können in zwei Arten unterteilt werden: Intradomain-IS und Interdomain-IS. Ein Intradomain-IS kommuniziert innerhalb eines einzigen autonomen Systems, während ein Interdomain-IS innerhalb und zwischen autonomen Systemen kommuniziert. Ein *Bereich* ist eine logische Gruppe von Netzwerk-Segmenten und den daran angeschlossenen Geräten. Bereiche sind unterteilt in autonome Systeme. Ein AS sind mehrere Netzwerke, die gemeinsam

administriert werden und eine gemeinsame Routing-Strategie haben. Autonome Systeme lassen sich in Bereiche unterteilen, wobei ein AS manchmal als eine *Domain* bezeichnet wird. Bild 1.12 zeigt ein hierarchisches Netzwerk mit seinen Komponenten.

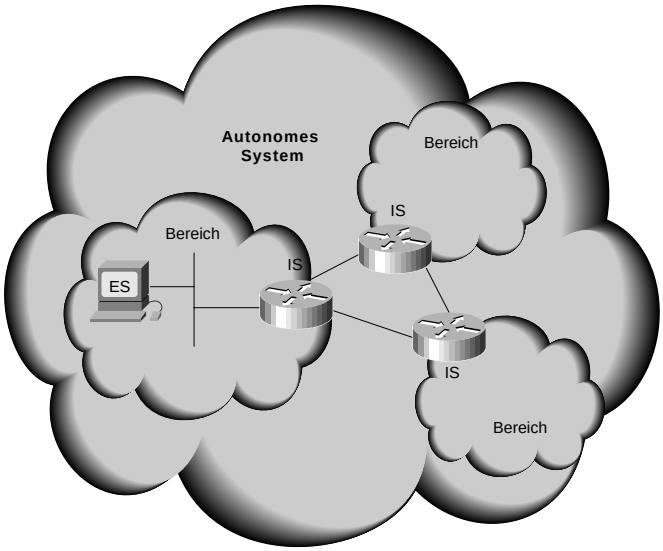


Bild 1.12: Ein hierarchisches Netzwerk besteht aus einer Vielzahl von Komponenten

1.5 Verbindungsorientierte und verbindungslose Netzwerk-Dienste

Allgemein kann festgestellt werden, daß es sich bei Netzwerk-Protokollen und dem von ihnen unterstützten Datenverkehr entweder um verbindungsorientierte oder verbindungslose handelt. Verbindungsorientierte Verfahren verwenden einen bestimmten Pfad (Leitungsweg), der für die Dauer einer (logischen) Verbindung besteht. Verbindungslose Verfahren übertragen die Daten über eine fest aufgebaute (logische) Verbindung.

Verbindungsorientierte Dienste durchlaufen drei Phasen: Verbindungsaufbau, Datenübertragung und Verbindungsbeendigung.

In der Phase des Verbindungsaufbaus wird ein bestimmter Pfad zwischen Ziel- und Quellsystem festgelegt. Netzwerk-Ressourcen werden zu diesem Zeitpunkt reserviert, um einen

konsistenten Dienst sicherzustellen, z.B. eine garantierte Durchsatzrate.

In der Phase der Datenübertragung werden die Daten sequentiell über den aufgebauten Pfad übertragen. Die Daten erreichen das Zielsystem in genau der Reihenfolge, in der sie gesendet wurden.

In der Phase der Verbindungsbeendigung wird die aufgebaute Verbindung, die nicht länger benötigt wird, abgebaut. Sollen zwischen Quell- und Zielsystem weitere Daten übertragen werden, muß eine neue Verbindung aufgebaut werden.

Verbindungsorientierte Netzwerk-Dienste haben zwei beträchtliche Nachteile gegenüber den verbindungslosen: den statischen Pfad und die statische Reservierung von Netzwerk-Ressourcen. Beim statischen Pfad kann es zu Schwierigkeiten kommen, weil alle Daten über den gleichen, festen Pfad übertragen werden müssen. Tritt irgendwo auf dieser Strecke ein Fehler auf, fällt die gesamte Verbindung aus. Die statische Reservierung von Netzwerk-Ressourcen führt zu Schwierigkeiten, da eine garantierte Übertragungsrate erforderlich ist, so daß andere Netzwerk-Nutzer diese Ressource nicht mitbenutzen können. Nur wenn die Verbindung unterbrechungsfrei Daten überträgt, wird die Bandbreite voll genutzt, ansonsten ist die Effizienz gering.

Für Anwendungen, bei denen in der Datenübertragung eine Verzögerung oder Paketneuordnung nicht in Frage kommt, ist der verbindungsorientierte Dienst das einzig sinnvolle Verfahren. Dazu gehören z.B. Anwendungen, die Sprach- und Video-Daten übertragen.

Ein Nachteil des verbindungslosen Netzwerk-Dienstes ist, daß kein Pfad von der Quelle zum Ziel festgelegt wird, keine Paketreihenfolge, keine Übertragungsrate und andere Netzwerk-Ressourcen garantiert sind. Jedes Paket muß vollständig adressiert sein, da für jedes Paket ein anderer Pfad durch das Netzwerk gewählt werden kann, abhängig von verschiedenen Faktoren. Jedes Paket wird vom Quellsystem einzeln gesendet und wird von den zwischengeschalteten Netzwerk-Geräten unabhängig von allen anderen Paketen behandelt.

Der verbindungslose Dienst bietet jedoch zwei deutliche Vorteile gegenüber dem verbindungsorientierten Dienst: dynamische Pfadwahl und dynamische Bandbreitenzuordnung. Mit der dynamischen Pfadwahl kann der Datenverkehr um ausgefallene Netzwerk-Ressourcen herum geleitet werden, da der Pfad für jedes Paket einzeln neu festgelegt wird. Bei der dynamischen Bandbreitenzuordnung kann die Bandbreite insgesamt effizienter genutzt werden, weil diese nicht unnötig belegt wird.

Für Anwendungen, die gegenüber Übertragungsverzögerungen und Paketneuordnung tolerant sind, ist der verbindungslose Dienst geeignet. Dazu gehören z.B. datenbasierte Anwendungen.

1.6 Adressierung im Internetwork

Adressen in einem Internetwork kennzeichnen ein einzelnes Gerät oder ein Gerät als ein Mitglied einer Gruppe. Die Adressierungsverfahren sind abhängig von der Protokollfamilie und der OSI-Schicht. Die folgenden drei Adreßarten werden am häufigsten verwendet: Adressen der Verbindungsschicht, Medium-Zugriffssteuerung-(MAC-)Adressen und Adressen der Vermittlungsschicht.

1.6.1 Verbindungsschicht

Die Adresse der Verbindungsschicht kennzeichnet jede physische Netzwerkverbindung von Netzwerk-Geräten eindeutig. Diese Adressen werden manchmal auch als *physische* oder *Hardware*-Adressen bezeichnet. Die Adressen der Verbindungsschicht liegen in einem ebenen Adreßraum und stehen in vordefinierter und fester Beziehung zu einem bestimmten Gerät.

Endsysteme sind im allgemeinen mit nur einer physischen Verbindung an das Netzwerk angeschlossen, weshalb sie nur eine Verbindungsschichtadresse benötigen. Router und andere Internetworking-Geräte verfügen normalerweise über mehrere physische Netzwerk-Verbindungen und haben deshalb auch mehrere Verbindungsadressen. Bild 1.13 zeigt, wie jede Schnittstelle eines Geräts anhand der Verbindungsschichtadresse eindeutig identifiziert ist.

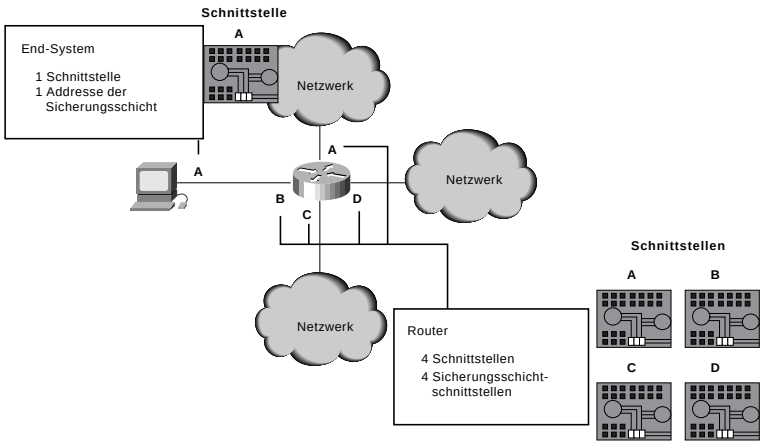


Bild 1.13: Jede Schnittstelle eines Geräts ist anhand der Verbindungsschichtadresse eindeutig gekennzeichnet

1.6.2 MAC-Adressen

Eine Media Access Control-Adresse (MAC – Medium-Zugriffssteuerung) bildet einen Teil der Verbindungsschichtadresse. MAC-Adressen kennzeichnen eine Netzwerk-Entität in einem LAN, das die IEEE-MAC-Adressen der Verbindungsschicht implementiert. Wie die meisten Adressen der Verbindungsschicht sind auch die MAC-Adressen für jede LAN-Schnittstelle eindeutig. Bild 1.14 zeigt den Zusammenhang zwischen MAC-Adresse, Verbindungsschichtadresse und IEEE-Subschichten der Verbindungsschicht.

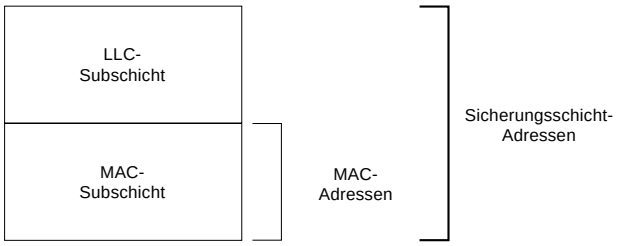


Bild 1.14: Beziehung zwischen MAC-Adresse, Verbindungsschichtadresse und den IEEE-Subschichten der Verbindungsschicht

MAC-Adressen sind 48 Bit lang und werden als zwölf hexadezimale Ziffern geschrieben. Die ersten sechs hexadezimalen Ziffern, die vom IEEE festgelegt sind, kennzeichnen den Hersteller. Dabei handelt es sich um den Organizational Unique Identifier (OUI). Die letzten sechs hexadezimalen Ziffern

geben die Seriennummer der Schnittstelle an oder einen anderen Wert, den der Hersteller festlegt. MAC-Adressen werden auch als Burned-in Adresses (BIAs – eingebrannte Adressen) bezeichnet, da sie in das ROM gebrannt sind. Beim Initialisieren der Schnittstellenkarte wird die Adresse ausgelesen und ins RAM kopiert. Bild 1.15 stellt das MAC-Adressenformat dar.

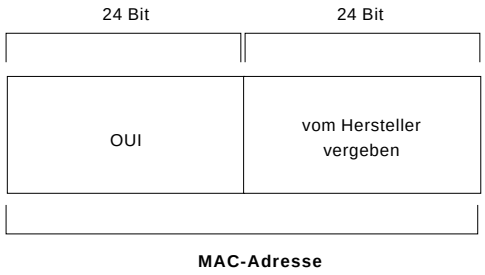


Bild 1.15: Die MAC-Adresse ist eine eindeutige Adresse aus hexadezimalen Ziffern

Die verschiedenen Protokollfamilien verwenden unterschiedliche Verfahren, um die MAC-Adresse eines Geräts zu ermitteln. Die folgenden drei Verfahren werden am häufigsten eingesetzt: Das Address Resolution Protocol (ARP – Adreßauflösungsprotokoll) bildet Netzwerk-Adressen auf MAC-Adressen ab. Das Hello-Protokoll ermöglicht es Netzwerk-Geräten, die MAC-Adressen anderer Netzwerk-Geräte zu ermitteln. MAC-Adressen sind entweder in die Adresse der Vermittlungsschicht eingebettet oder werden von einem Algorithmus generiert.

Bei der Adreßauflösung wird die Netzwerk-Adresse auf die MAC-Adresse abgebildet. Dazu wird das Address Resolution Protocoll (ARP) verwendet, das in vielen Protokollfamilien implementiert ist. Nachdem eine Netzwerk-Adresse erfolgreich einer MAC-Adresse zugeordnet werden konnte, speichert das Netzwerk-Gerät diese Information im ARP-Cache. Mit diesem ARP-Cache wird unnötiger ARP-Datenverkehr vermieden, da das sendende Gerät die MAC-Adresse bereits kennt und nicht erst ermitteln muß.

Wie die Adreßauflösung genau erfolgt, hängt von der jeweiligen Netzwerkumgebung ab. In einem einzelnen LAN beginnt die Adreßauflösung damit, daß ein End-System A eine ARP-

Anforderung an alle Geräte im LAN sendet (Broadcast), um die MAC-Adresse des Geräts B zu ermitteln. Die so gesendete Anforderung wird von allen Geräten im LAN empfangen und verarbeitet, wenngleich nur das End-System B die ARP-Anforderung beantwortet, indem es eine ARP-Antwort an das End-System A sendet, die seine MAC-Adresse enthält. End-System A empfängt diese Antwort und speichert die MAC-Adresse des Systems B in seinem ARP-Cache (der ARP-Cache ist eine Tabelle, in der die Netzwerk-Adressen den MAC-Adressen zugeordnet werden). Jedesmal, wenn nun das End-System A mit dem End-System B kommunizieren soll, prüft es den ARP-Cache, findet darin die MAC-Adresse des Systems B und sendet einen Frame, ohne zuvor eine ARP-Anforderung im LAN absetzen zu müssen.

Etwas anders läuft die Adreßauflösung ab, wenn Quell- und Zielgerät in verschiedenen LANs eingebunden sind, die über einen Router miteinander in Verbindung stehen. End-System Y sendet eine ARP-Anforderung ins LAN (Broadcast), um die MAC-Adresse des End-Systems Z zu ermitteln. Die so gesendete Anforderung wird von allen Geräten des LAN einschließlich des Routers X empfangen und verarbeitet. Dabei fungiert der Router X als Stellvertreter (Proxy) für das End-System Z. Der Router durchsucht seine Routing-Tabelle, um festzustellen, ob sich End-System Z in einem anderen LAN befindet. Er sendet dann eine ARP-Antwort an das End-System Y, die statt der MAC-Adresse des End-Systems Z *seine* MAC-Adresse enthält. End-System Y empfängt diese Antwort und trägt die MAC-Adresse des Routers X als MAC-Adresse des End-Systems Z ein. Wenn End-System Y mit dem End-System Z kommunizieren soll, prüft es seinen ARP-Cache und findet dort für das End-System Z die MAC-Adresse des Routers X, so daß es ohne weitere ARP-Anforderung einen Frame sofort senden kann. Der Router X empfängt die für End-System Z bestimmten Daten und leitet sie zum anderen LAN weiter.

Mit dem Hello-Protokoll (einem Protokoll der Vermittlungsschicht) können sich Netzwerk-Geräte gegenseitig identifizieren und feststellen, ob der Partner noch in Betrieb ist. Beim Einschalten eines Geräts sendet dieses Hello-Nachrichten an alle Geräte im Netzwerk (Broadcast). Die Netzwerk-Geräte

senden eine Hello-Antwort zurück. In bestimmten zeitlichen Abständen sendet jedes Gerät Hello-Nachrichten, um bekannt zu geben, daß es noch in Betrieb ist. Dabei können die Netzwerk-Geräte die MAC-Adressen anhand der Hello-Protokoll-Pakete ermitteln.

Von drei Protokollen werden berechenbare MAC-Adressen verwendet. In diesen Protokollfamilien sind die MAC-Adressen berechenbar, weil die Vermittlungsschicht entweder die MAC-Adresse einbettet oder einen Algorithmus zur Bestimmung verwendet. Diese drei Protokolle sind: Xerox Network Systems (XNS), Novell Internetwork Packet Exchange (IPX) und DECnet Phase IV.

1.6.3 Adressen der Vermittlungsschicht

Mit einer Adresse der Vermittlungsschicht wird eine Entität in dieser Schicht des OSI-Referenzmodells identifiziert. Vermittlungsschichtadressen liegen in einem hierarchischen Adreßraum und werden auch als *virtuelle* oder *logische Adressen* bezeichnet.

Die Beziehung zwischen einem Netzwerk-Gerät und einer Netzwerk-Adresse ist eine logische und keine feste. Diese Adresse basiert entweder auf den physischen Netzwerkeigenschaften (das Gerät ist an ein bestimmtes Segment angeschlossen) oder auf Gruppierungen, die ohne physische Grundlage sind (das Gerät ist Teil einer AppleTalk-Zone). Die End-Systeme benötigen für jedes von ihnen unterstützte Protokoll der Vermittlungsschicht eine eigene Adresse (wobei davon ausgegangen wird, daß jedes Gerät nur über eine physische Verbindung zum Netzwerk verfügt). Router und andere Internetworking-Geräte benötigen für jede physische Netzwerkverbindung eine Vermittlungsschichtadresse. Ein Router, der z.B. mit drei Schnittstellen ausgestattet ist, die jede mit AppleTalk, TCP/IP und OSI betrieben wird, muß für jede Schnittstelle drei Vermittlungsschichtadressen aufweisen. Daraus ergibt sich, daß dieser Router über neun Vermittlungsschichtadressen verfügt. Bild 1.16 illustriert, wie jeder Netzwerk-Schnittstelle eine Vermittlungsschichtadresse für jedes unterstützte Protokoll zugewiesen wird.

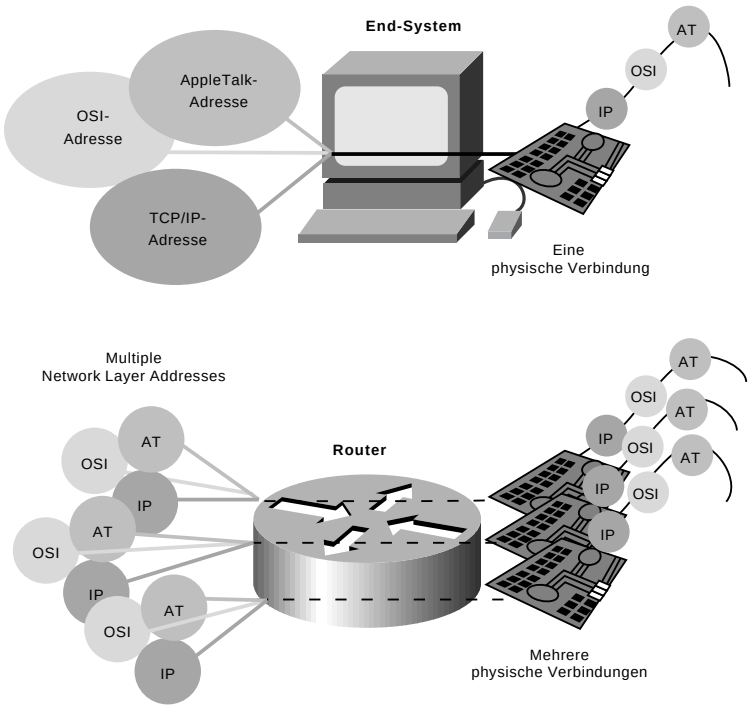


Bild 1.16: Jeder Netzwerk-Schnittstelle muß für jedes unterstützte Protokoll eine Vermittlungsschichtadresse zugewiesen sein

1.6.4 Hierarchischer oder ebener Adreßraum

Der Internetworking-Adreßraum ist in einer der beiden folgenden Formen abgebildet: hierarchisch oder eben.

Ein hierarchischer Adreßraum ist unterteilt in zahlreiche Untergruppen, mit denen die Adresse immer weiter angenähert wird, bis sie auf ein einziges Gerät weist (ähnlich einer Anschrift). Ein ebener Adreßraum besteht aus nur einer einzigen Gruppe (ähnlich der Sozialversicherungnummer in den USA).

Die hierarchische Adressierung bietet gewisse Vorteile gegenüber dem ebenen Adreßschema. So kann die Adressensortierung und -wiederholung relativ einfach durch Vergleichsoperationen erreicht werden. In Irland z.B. ist jede Straße eindeutig gekennzeichnet und kann in keinem anderen Land liegen. Bild 1.17 zeigt die Unterschiede zwischen hierarchischem und ebenem Adreßraum.

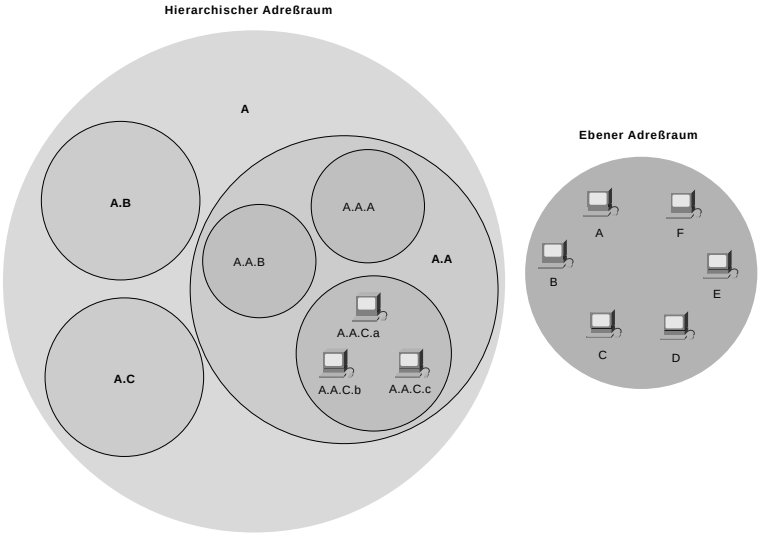


Bild 1.17: Unterschiede zwischen hierarchischem und ebenem Adreßraum zeigen sich bei Vergleichsoperationen

1.6.5 Adreßzuordnung

Adressen können Geräten auf drei verschiedene Weisen zugeordnet werden: *statisch*, *dynamisch* oder als *Server*-Adresse. *Statische Adressen* werden von einem Netzwerk-Administrator nach einem vorgefertigten Adressierungsplan vergeben. Eine statische Adresse ändert sich nicht, es sei denn, der Netzwerk-Administrator nimmt eine manuelle Änderung vor. *Dynamische Adressen* werden Geräten anhand protokollspezifischer Verfahren zugeordnet, wenn sie an das Netzwerk angeschlossen werden. Ein Gerät, das über eine dynamische Adresse angesprochen wird, erhält meistens bei jeder neuen Anbindung an das Netzwerk (z.B. beim Einschalten) eine andere Adresse. Die Adreßzuordnung durch einen Server erfolgt, wenn ein Gerät an das Netzwerk angeschlossen wird. Vom Server zugeordnete Adressen werden für andere Geräte weiter verwendet, wenn das Gerät die Verbindung zum Netzwerk beendet. So kann ein Gerät bei jedem Anschluß an das Netzwerk eine andere Adresse erhalten.

1.6.6 Adressen oder Namen

An Internetworking-Geräte werden normalerweise sowohl Adressen als auch Namen vergeben. Der Name ist dabei meistens ortsunabhängig und bleibt dem Gerät auch bei einem

Standortwechsel zugeordnet. Hingegen sind die Internetworking-Adressen ortsgebunden; sie ändern sich, wenn ein Gerät »umgezogen« ist (wobei natürlich die MAC-Adresse eine Ausnahme von dieser Regel bildet). Namen und Adressen stellen logische Kennzeichnungen dar, die von einem Systemadministrator oder einer Organisation, z.B. der Internet Assigned Numbers Authority (IANA), vergeben werden.

1.7 Grundlagen der Flußsteuerung

Die Flußsteuerung ist eine Funktion, die dem Datenstau im Netzwerk vorbeugt, indem sie sicherstellt, daß ein sendendes Gerät nicht mehr Daten übermittelt als das empfangende Gerät annehmen bzw. verarbeiten kann. Es gibt unzählige Ursachen für einen Datenstau. Wenn z.B. ein schneller Computer mehr Daten auf das Netzwerk überträgt als dieses überhaupt vermitteln kann, oder das empfangende Gerät mit diesem Datenschwall überschüttet wird, den es gar nicht abarbeiten kann. Drei Verfahren stehen zur Verfügung, um dem Datenstau vorzubeugen: Pufferung, Drosselung der Datenquelle und Windowing.

Gepuffert wird von Netzwerk-Geräten, um Daten mit besonders hoher Übertragungsrate zwischenzuspeichern, bis diese verarbeitet werden können. Gelegentlich hohe Übertragungsraten können durch die Pufferung problemlos abgefangen werden. Dauert dieser Zustand aber zu lange an, wird der Speicher zu stark beansprucht, und alle weiterhin eingehenden Datagramme werden abgewiesen.

Nachrichten zur Drosselung der Datenquelle werden vom empfangenden Gerät gesendet, damit dessen Puffer nicht überlaufen. Das empfangende Gerät sendet eine Drosselungsnachricht an das sendende Gerät, damit dieses seine Übertragungsrate reduziert. Zuerst verweigert das empfangende Gerät weitere Daten, wenn die Puffer überlaufen. Dann sendet es an das sendende Gerät für jedes verweigerte Paket eine Drosselungsnachricht. Das sendende Gerät reduziert daraufhin schrittweise die Übertragungsrate, bis es keine Drosselungsnachrichten mehr erhält. Schließlich steigert das sendende Gerät wieder nach und nach die Übertragungsrate, bis wieder Drosselungsnachrichten eingehen.

Windowing ist ein Verfahren zur Flußsteuerung, bei dem das Quellgerät vom Zielgerät nach einer bestimmten Anzahl gesendeter Pakete eine Quittung anfordert. Bei einer Window-Größe von 3 fordert das Quellgerät eine Quittung an, nachdem es drei Pakete gesendet hat. Zuerst sendet das Quellgerät drei Pakete an das Zielgerät. Nach Eingang der drei Pakete sendet das Zielgerät eine Quittung an das Quellgerät. Wenn das Quellgerät die Quittung empfängt, sendet es weitere drei Pakete. Erreichen das Zielgerät aus irgendeinem Grund nicht alle drei Pakete (z.B. Pufferüberlauf), sendet es keine Quittung. Vom Quellgerät werden dann die letzten drei Pakete mit einer niedrigeren Übertragungsrate erneut gesendet.

1.8 Grundlagen der Fehlerprüfung

Die Verfahren der Fehlerprüfung ermitteln, ob die Daten während der Übertragung zerstört wurden. Die Fehlerprüfung ist in vielen OSI-Schichten implementiert.

Eine der üblichen Fehlerprüfungen ist der Cyclic Redundancy Check (CRC), der zerstörte Daten erkennt und ausscheidet. Funktionen zur Fehlerbehebung (z.B. wiederholte Übertragung) bleiben Protokollen der höheren Schichten vorbehalten. Vom Quellgerät wird ein CRC-Wert aus den zu übertragenden Daten berechnet. Das Zielgerät berechnet auf die gleiche Weise einen solchen Wert und vergleicht ihn mit den übertragenen, um festzustellen, ob während der Übertragung ein Fehler auftrat. Sind die Werte gleich, wird das Paket als gültig betrachtet. Sollten die Werte unterschiedlich sein, ist ein Fehler während der Übertragung aufgetreten, und das Paket wird ausgeschieden.

1.9 Grundlagen des Multiplexing

Beim Multiplexing werden vom Quellgerät mehrere Datenkanäle auf einem Kanal oder auf einer physischen Leitung zusammen übertragen. Dieses Verfahren kann auf jeder Schicht des OSI-Referenzmodells implementiert sein. Auf der Gegenseite, beim Zielgerät, erfolgt das Demultiplexing, das die Datenkanäle wieder separiert. Multiplexing findet z.B. statt, wenn Daten mehrerer Anwendungen auf ein einzelnes Daten-

paket der unteren Schichten multiplext wird. Bild 1.18 zeigt dieses Beispiel.

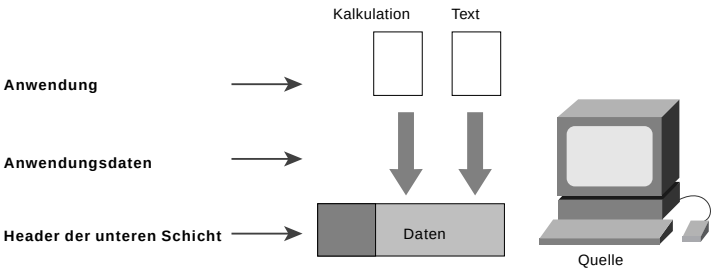


Bild 1.18: Die Daten mehrerer Anwendungen können in ein Paket der unteren Schichten multiplext werden

Ein weiteres Beispiel für Multiplexing ist die Übertragung der Daten mehrerer Geräte über einen physischen Kanal (wobei ein als Multiplexer bezeichnetes Gerät zum Einsatz kommt). Bild 1.19 veranschaulicht dieses Beispiel.

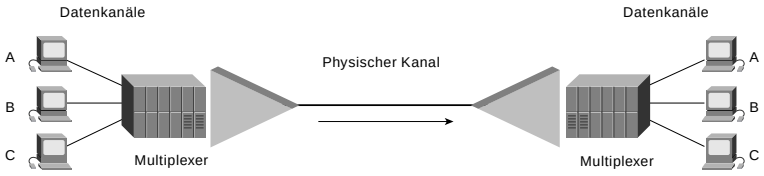


Bild 1.19: Die Daten mehrerer Geräte können auf einen physischen Kanal multiplext werden

Ein Multiplexer ist ein Gerät der physischen Schicht, das mehrere Datenströme auf einem oder mehreren Ausgangskanälen am Quellgerät zusammenfaßt. Auf der Gegenseite multiplext ein Multiplexer die Kanäle wieder und maximiert die Nutzung der Bandbreite des physischen Mediums, indem es für mehrere Datenquellen gleichzeitig zur Verfügung steht.

Es gibt verschiedene Multiplex-Verfahren: Zeitmultiplex (TDM – Time-Devision Multiplexing), asynchrones Zeitmultiplexing (ATDM – Asynchronous Time-Devision Multiplexing), Frequenz-Multiplex (FDM – Frequency Multiplexing) und das statistische Multiplexing.

Beim TDM wird jedem Datenkanal anhand einer Zeitscheibe die Bandbreite zugeordnet, unabhängig davon, ob zu dieser Zeit Daten zu übertragen sind oder nicht. Beim ATDM wird die Bandbreite nach Bedarf über eine dynamische Zeitscheibe zugewiesen. Beim FDM wird jedem Datenkanal in Abhängig-

keit seiner Signalfrequenz die Bandbreite zugeordnet. Beim statistischen Multiplexing wird die Bandbreite dynamisch jedem Kanal zugeordnet, der Daten zu übertragen hat.

1.10 Organisationen für Standardisierung

Eine Vielzahl von Organisationen trägt zu Internetworking-Standards bei, indem sie Diskussionsforen anbieten, aus diesen informellen Diskussionen formale Spezifikationen erstellen und für die Verbreitung dieser Spezifikationen sorgen, wenn diese zum Standard erklärt wurden.

Die meisten Standardisierungsorganisationen nutzen bestimmte Vorgehensweisen, um einen formalen Standard zu erstellen: Organisieren von Ideen, Diskussion der Umsetzung, Entwicklung eines Entwurfs, Abstimmung über den gesamten Standard oder über Teile und schließlich die formelle Bekanntgabe des vollständigen Standards.

Zu den bekanntesten Organisationen, die sich mit Internetworking-Standards befassen, zählen die folgenden:

- *International Organization for Standardization (ISO)* – ISO ist eine internationale Standardisierungsorganisation, die für einen weiten Bereich von Standards verantwortlich zeichnet, zu denen viele Standards gehören, die Netzwerke betreffen. Am bekanntesten sind das OSI-Referenzmodell und die OSI-Protkollfamilie.
- *American National Standards Institute (ANSI)* – ANSI ist ein Mitglied der ISO und koordiniert freiwillige Gruppen in den USA. ANSI entwickelte das Fiber Distributed Data Interface (FDDI) und weitere Kommunikationsstandards.
- *Electronic Industries Association (EIA)* – EIA spezifiziert elektrische Übertragungsstandards, zu denen auch solche aus dem Netzwerkbereich gehören. Von der EIA wurde der sehr häufig verwendete Standard EIA/TIA-232 (früher als RS-232 bekannt) entwickelt.
- *Institute of Electrical and Electronic Engineers (IEEE)* – IEEE ist eine professionelle Organisation, die Netzwerk-Standards und weitere darüber hinaus definiert. Vom IEEE

wurden die sehr häufig eingesetzten LAN-Standards IEEE 802.3 und IEEE 802.5 entwickelt.

- *International Telecommunication Union Telecommunication Standardization Sector (ITU-T)* – Früher trug diese Organisation die Bezeichnung Committee for International Telegraph and Telephone (CCITT). ITU-T ist heute eine internationale Organisation, die Kommunikationsstandards entwickelt. Unter anderem wurde von der ITU-T das X.25-Protokoll entwickelt.
- *Internet Activities Board (IAB)* – IAB ist eine Gruppe von Internetworking-Forschern, die neue Aspekte zum Internet diskutieren und Internet-Regeln verabschieden und Arbeitsgruppen einsetzen. Einige Dokumente mit dem Status Request For Comments (RFC) wurden vom IAB als Internet-Standards herausgegeben, wozu auch das Transmission Control Protocol/Internet Protocol (TCP/IP) und das Simple Network Management Protocol (SNMP) gehören.

KAPITEL 2

Einführung in die LAN-Protokolle

In diesem Kapitel werden die verschiedenen Medium-Zugriffsmethoden, Übertragungsmethoden, Topologien und Geräte vorgestellt, die in einem lokalen Netz (LAN – Local Area Network) zum Einsatz kommen. Dabei wird der Schwerpunkt auf Verfahren und Geräte gelegt, wie sie in Netzwerken mit Ethernet/IEEE 802.3, Token Ring/IEEE 802.5 und Fiber Distributed Data Interface (FDDI) vorkommen. Die Kapitel in Teil 2, »LAN-Protokolle«, gehen näher auf andere Protokolle ein. Bild 2.1 stellt das grundlegende Layout der drei genannten Protokolle dar.

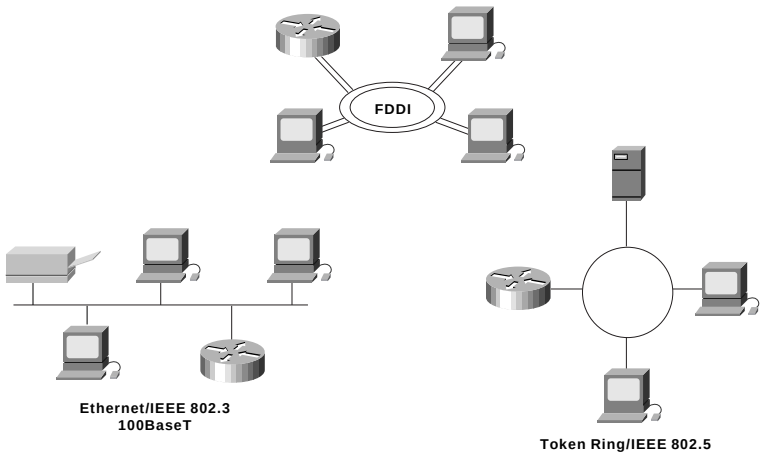


Bild 2.1: Diese drei LAN-Implementationen sind am häufigsten im Einsatz

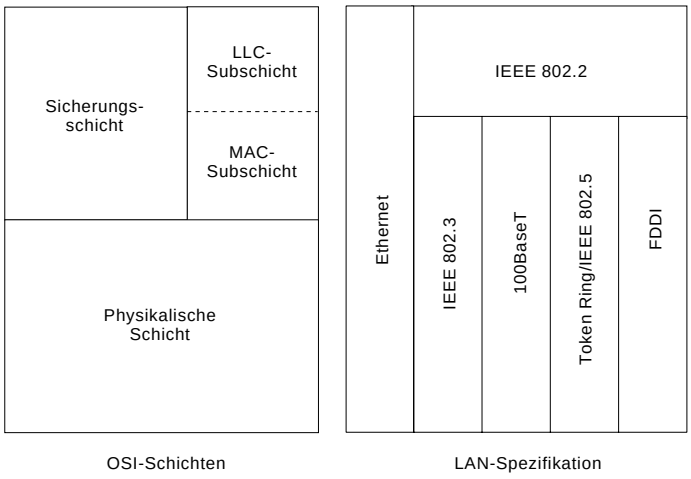
2.1 Was ist ein LAN?

Ein LAN ist ein räumlich kleines Netzwerk, in dem Daten mit hoher Geschwindigkeit und fehlertolerant übertragen werden. Es verbindet zumeist Workstations, Personalcomputer, Drucker und weitere Geräte. LANs bieten ihren Anwendern viele Vorteile, u.a. den gemeinsamen Zugriff auf Geräte und Anwendungen, die Möglichkeit des Datenaustauschs zwischen Anwendern und die Kommunikation per elektronischer Post (E-Mail) oder anderen Anwendungen.

2.2 LAN-Protokolle und das OSI-Referenzmodell

Die Funktion der LAN-Protokolle ist auf die untersten zwei Schichten des OSI-Referenzmodells (siehe Kapitel 1, »Grundlagen des Internetworking«) beschränkt: auf die physische Schicht und die Verbindungsschicht. Bild 2.2 veranschaulicht den Zusammenhang zwischen den gängigsten LAN-Protokollen und dem OSI-Referenzmodell.

Bild 2.2: Die gängigsten LAN-Protokolle und das OSI-Referenzmodell



2.3 Medium-Zugriffsmethoden im LAN

Die LAN-Protokolle verwenden eine der beiden folgenden Methoden, um auf das physische Netzwerk-Medium zuzugreifen: *Carrier Sense Multiple Access Collision Detect* (CSMA/CD) und *Token Passing*.

Beim Medium-Zugriff per CSMA/CD »streiten« die Netzwerk-Geräte um den Zugriff auf das physische Netzwerk-Medium. CSMA/CD wird deshalb auch als *Contention Access* bezeichnet. LANs, die den Medium-Zugriff per CSMA/CD regeln, sind z.B. Ethernet/IEEE-802.3-Netze, einschließlich 100BaseT.

Beim Medium-Zugriff nach dem Token-Passing-Verfahren greifen Netzwerk-Geräte auf das physische Medium zu, wenn sie ein Token erhalten. Netze, die den Medium-Zugriff per Token-Passing regeln, sind z.B. Token Ring/IEEE 802.5 und Fiber Distributed Data Interface (FDDI).

2.4 Übertragungsverfahren im LAN

Die Übertragung von Daten in einem LAN kann in drei Gruppen unterteilt werden: an einzelne Stationen (*Unicast*), an eine Gruppe von Stationen (*Multicast*) und an alle Stationen (*Broadcast*). Dabei wird immer nur ein Paket übertragen.

Bei der Unicast-Übertragung wird ein einzelnes Paket von einer Quelle zu einem Ziel im Netzwerk gesendet. Zuerst versieht der Quellknoten das Paket mit der Adresse des Zielknotens. Das Paket wird dann in das Netzwerk übertragen. Schließlich befördert das Netzwerk das Paket zum Ziel.

Bei der Multicast-Übertragung werden Kopien eines einzelnen Pakets an eine bestimmte Gruppe von Knoten im Netzwerk gesendet. Zuerst versieht der Quellknoten das Paket mit der Multicast-Adresse für die Gruppe. Das Paket wird dann in das Netzwerk übertragen, das eine entsprechende Anzahl Kopien anlegt und diese zu den entsprechenden Knoten im Netzwerk befördert.

Bei der Broadcast-Übertragung wird ein einzelnes Datenpaket an alle Knoten im Netzwerk gesendet. In diesem Fall versieht der Quellknoten das Paket mit der Broadcast-Adresse. Das Paket wird dann in das Netzwerk übertragen, das so viele Kopien des Pakets anlegt wie Knoten im Netzwerk vorhanden sind, und überträgt diese an alle Knoten im Netzwerk.

2.5 LAN-Topologien

Die LAN-Topologie definiert die Weise, wie Netzwerk-Geräte miteinander verbunden werden. Es gibt vier gängige LAN-Topologien: den Bus, den Ring, den Stern und den Baum. Dabei handelt es sich um logische Architekturen. Die Geräte müssen physisch nicht so angeordnet sein. Die logischen Topologien Bus und Ring z.B. sind physisch meistens in Sternform realisiert. Die Bus-Topologie ist eine lineare LAN-Architektur, in der sich die gesendeten Daten jeder Netzwerkstation über das gesamte Medium ausbreiten und von allen Stationen empfangen werden. Von den am häufigsten eingesetzten LAN-Implementationen werden Ethernet/IEEE 802.3, einschließlich 100BaseT, in Bus-Topologie realisiert, wie in Bild 2.3 dargestellt.

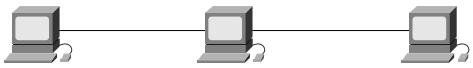


Bild 2.3: Bestimmte Netzwerke implementieren die lokale Bus-Topologie

Die Ring-Topologie ist eine LAN-Architektur mit mehreren Geräten, die untereinander durch uni-direktionale Verbindungen einen geschlossenen Kreis bilden. Sowohl Token Ring/ IEEE-802.5- als auch FDDI-Netzwerke implementieren diese Ring-Topologie. Bild 2.4 zeigt eine logische Ring-Topologie.

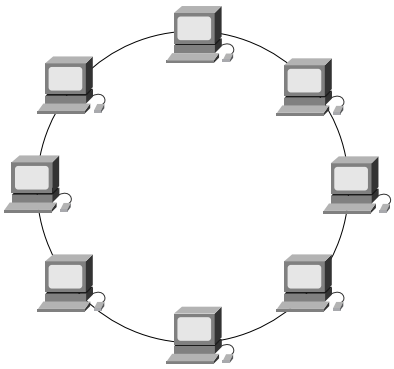


Bild 2.4: Andere Netzwerke implementieren eine logische Ring-Topologie

Die Stern-Topologie ist eine LAN-Architektur, in der die Endpunkte eines Netzwerks über einen normalen zentralen Hub oder Switch per dedizierten Verbindungen zusammenhängen. Die logischen Topologien Bus und Ring sind oft physisch als Stern-Topologien implementiert. Bild 2.5 veranschaulicht dies.

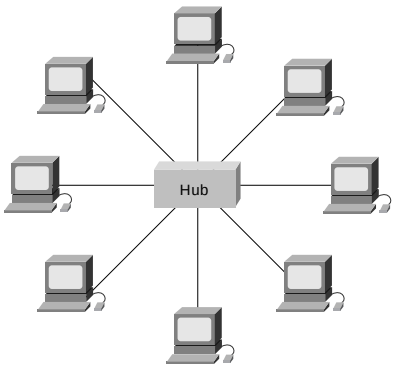


Bild 2.5: Stern-Topologie

Die Baum-Topologie ist eine LAN-Architektur, die der Bus-Topologie entspricht, außer daß Abzweigungen mit mehreren Knoten möglich sind. Bild 2.6 zeigt eine logische Baum-Topologie.

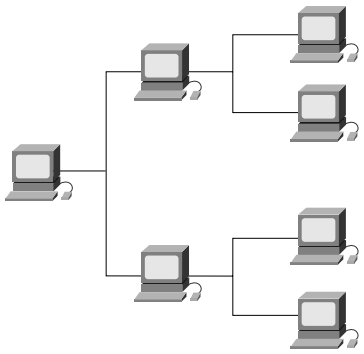


Bild 2.6: Eine logische Baum-Topologie kann mehrere Knoten enthalten

2.6 LAN-Geräte

Zu den in einem LAN eingesetzten Geräten gehören: Repeater, Hubs, LAN-Extender, Bridges, LAN-Switches und Router.

Auf Repeater, Hubs und LAN-Extender wird in diesem Abschnitt nur kurz eingegangen. Die Funktion und der Betrieb von Bridges, Switches und Router werden im Kapitel 4, »Grundlagen des Bridging und Switching«, und im Kapitel 5, »Grundlagen des Routing«, erläutert.

Ein *Repeater* ist ein Gerät der physischen Schicht, das Medium-Segmente eines erweiterten Netzwerks miteinander verbindet. Die Hauptfunktion eines Repeater liegt darin, daß mehrere Kabelsegmente wie ein einzelnes Kabel gehandhabt werden können. Repeater empfangen Signale von einem Netzwerk-Segment und senden sie verstärkt in ein anderes Netzwerk-Segment. Dieses Verfahren verhindert, daß sich die Signale aufgrund der Kabellänge und der vielen angeschlossenen Geräte zu sehr verschlechtern. Repeater können keine komplexe Filterung und andere Verarbeitung des Datenverkehrs vornehmen. Es werden alle elektrischen Signale wiederholt und verstärkt, auch elektrische Störungen und andere Fehler. Wie viele Repeater eingesetzt und Netzwerk-Segmente so verbunden werden können, hängt z.B. vom Zeitverhalten ab. Bild 2.7 zeigt einen Repeater, der zwei Netzwerk-Segmente miteinander verbindet.

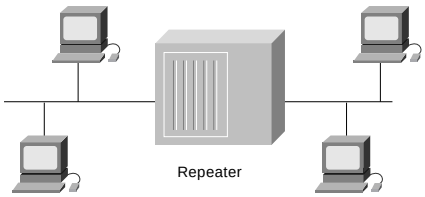


Bild 2.7: Ein Repeater verbindet zwei Netzwerk-Segmente

Ein *Hub* ist ein Gerät der physischen Schicht, das mehrere Benutzerstationen miteinander verbindet, und zwar jede über ein bestimmtes Kabel. Die elektrischen Verbindungen werden im Hub hergestellt. Mit Hubs werden physische Stern-Netzwerke realisiert, während es sich logisch um eine Bus- oder Ring-Konfiguration des LAN handelt. Unter bestimmten Bedingungen funktioniert ein Hub wie ein Multiport-Repeater.

Ein *LAN Extender* ist ein Remote-Access Multilayer Switch, der eine Verbindung zu einem Host-Router hat. LAN Extender leiten den Datenverkehr aller Standard-Protokolle der Vermittlungsschicht weiter (z.B. von IP, IPX und AppleTalk) und können Datenverkehr anhand der MAC-Adresse oder des Protokolltyps filtern. LAN Extender arbeiten sehr effektiv, da der Host-Router alle unnötigen Broadcasts und Multicasts herausfiltert. Mit LAN Extendern kann der Datenverkehr allerdings nicht segmentiert werden, sie haben auch keine Firewall-Funktion. Bild 2.8 zeigt mehrere LAN Extender, die über ein WAN mit einem Host-Router verbunden sind.

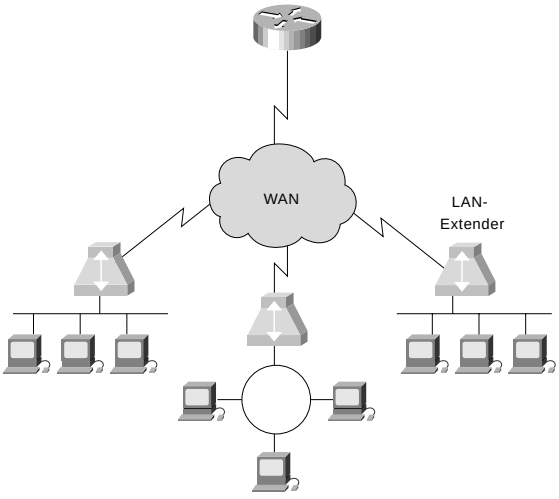


Bild 2.8: Mehrere LAN Extender können über ein WAN mit einem Host-Router verbunden sein

KAPITEL 3

Einführung in die WAN-Technologien

In diesem Kapitel werden die verschiedenen Protokolle und Technologien vorgestellt, die in Weitverkehrsnetzen (WAN – Wide Area Network) eingesetzt werden. Zu den angesprochenen Themen gehören Punkt-zu-Punkt-Verbindungen, Leitungsvermittlung, Paket-Switching, virtuelle Verbindungen, Einwahldienste und WAN-Geräte. Die Kapitel in Teil 3, »WAN-Technologien«, behandeln die einzelnen Technologien ausführlich.

3.1 Was ist ein WAN?

Ein WAN ist ein Datenkommunikationsnetzwerk, das sich über relativ große räumliche Entfernungen erstreckt und in dem Übertragungseinrichtungen der gängigen Anbieter (z.B. Telefongesellschaften) benutzt werden. Die Funktion der WAN-Technologien ist auf die drei unteren Schichten des OSI-Referenzmodells beschränkt: auf die physische Schicht, die Verbindungsschicht und die Vermittlungsschicht. Bild 3.1 zeigt den Zusammenhang zwischen den üblichen WAN-Technologien und dem OSI-Referenzmodell.

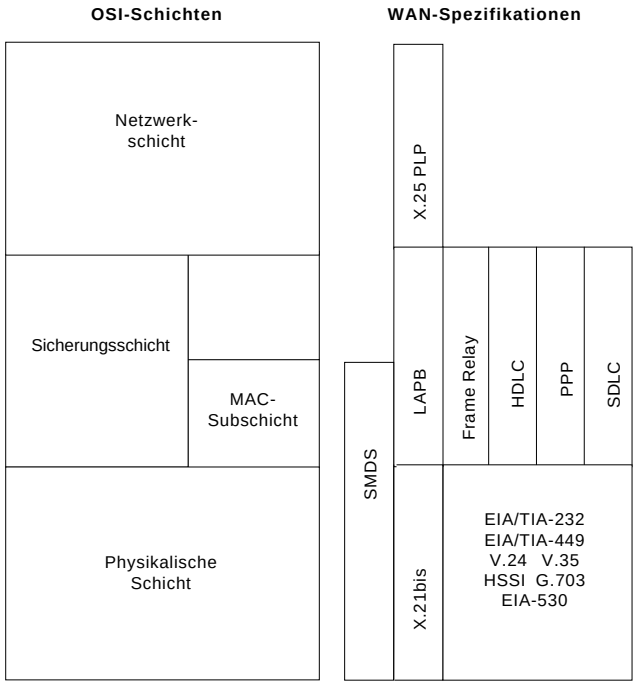


Bild 3.1: Die WAN-Technologien arbeiten auf den untersten Schichten des OSI-Referenzmodells

3.2 Punkt-zu-Punkt-Verbindungen

Eine Punkt-zu-Punkt-Verbindung stellt einen einzelnen, vorbereiteten WAN-Kommunikationspfad dar, der vom kundeneigenen Gerät über das Anbieternetzwerk (z.B. einer Telefongesellschaft) bis zum fernen Netzwerk reicht. Punkt-zu-Punkt-Verbindungen werden auch als »geleaste Leitungen« bezeichnet, da der eingerichtete Pfad über das Anbieternetzwerk zu jedem fernen Netzwerk permanent und festgelegt ist. Der Dienstanbieter reserviert eine Punkt-zu-Punkt-Verbindung für die private Nutzung durch den Kunden. Diese Verbindungen eignen sich für zwei Arten der Datenübertragung: für Datagramme, die sich aus einzeln adressierten Frames zusammensetzen, und für Datenströme, bei denen eine Adreßüberprüfung nur einmal stattfindet. Bild 3.2 zeigt eine typische Punkt-zu-Punkt-Verbindung über ein WAN.

Bild 3.2: Eine typische Punkt-zu-Punkt-Verbindung verläuft über ein WAN zu einem fernen Netzwerk

3.3 Leitungsvermittlung

Die Leitungsvermittlung ist ein WAN-Vermittlungsverfahren, bei dem eine dedizierte physische Verbindung für jede Kommunikationssitzung über das Anbieternetzwerk aufgebaut, aufrechterhalten und beendet wird. Die Leitungsvermittlung eignet sich für zwei Arten der Datenübertragung: für Datagramme, die sich aus einzeln adressierten Frames zusammensetzen, und für Datenströme, bei denen eine Adreßüberprüfung nur einmal stattfindet. Dieses Vermittlungsverfahren wird besonders häufig von Telefongesellschaften genutzt, da es dem normalen Telefonieren sehr ähnlich ist. So ist z.B. ISDN eine leitungsvermittelte WAN-Technologie, wie in Bild 3.3 dargestellt.

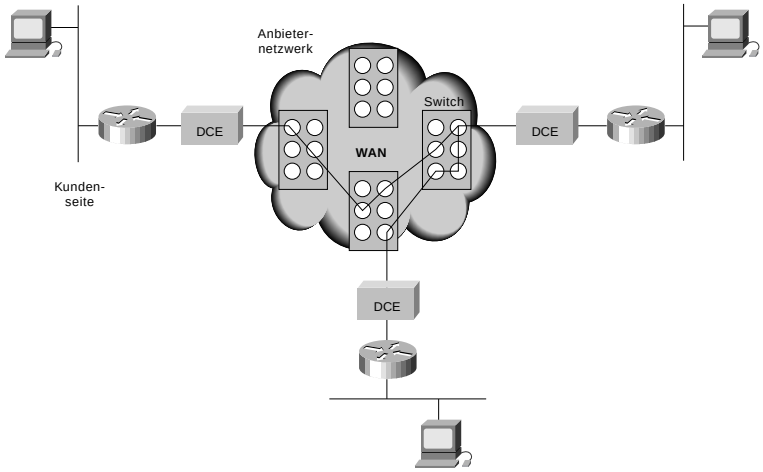


Bild 3.3: Ein leitungsvermitteltes WAN funktioniert so ähnlich wie ein normaler Telefonanruf

3.4 Paketvermittlung

Die Paketvermittlung ist eine WAN-Switching-Methode, bei der mehrere Netzwerk-Geräte eine einzige Punkt-zu-Punkt-Verbindung gemeinsam nutzen, um Pakete über ein Anbieternetzwerk von der Quelle zum Ziel zu übertragen. Um die gemeinsame Nutzung einer solchen Verbindung zu ermöglichen, wird das statistische Multiplexing verwendet. Zu den paketvermittelten WAN-Technologien gehören die Protokolle Asynchronous Transfer Mode (ATM), Frame Relay, Switched Multimegabit Data Service (SMDS) und X.25 (siehe Bild 3.4).

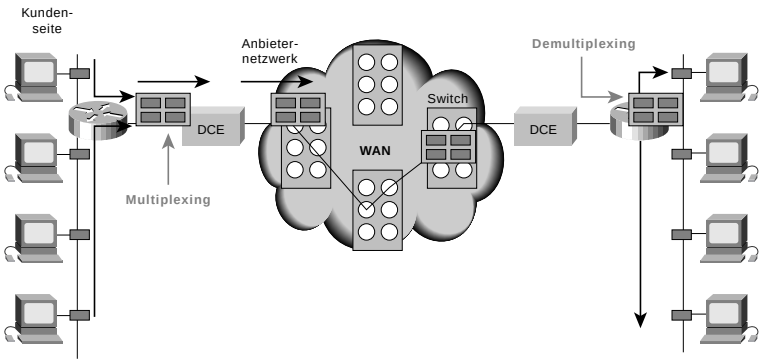


Bild 3.4: Bei der Paketvermittlung werden die Pakete über ein Anbieternetzwerk übertragen

3.5 Virtuelle Verbindungen im WAN

Eine virtuelle Verbindung ist eine logische Verbindung, die zur sicheren Kommunikation zwischen zwei Netzwerk-Geräten aufgebaut wird. Es gibt zwei Arten der virtuellen Verbindung: die gewählte virtuelle Verbindung (GVV – Switched Virtual Circuit [SVC]) und die feste virtuelle Verbindung (FVV – Permanent Virtual Circuit [PVC]).

Bei *SVC* handelt es sich um dynamische, bei Bedarf aufzubauende Verbindungen, die nach Beendigung der Übertragung abgebaut werden. Die Kommunikation über einen SVC durchläuft drei Zustände: Leitungsaufbau, Datenübertragung und Leitungsabbau. Zum Leitungsaufbau gehört, daß eine virtuelle Verbindung zwischen Quell- und Zielgerät hergestellt wird. Während der Datenübertragung tauschen die Geräte die Daten über die virtuelle Verbindung aus. Beim Leitungsabbau wird die virtuelle Verbindung zwischen Quell- und Zielgerät

beendet. SVCs werden oft dort eingesetzt, wo die Datenübertragung nur gelegentlich stattfindet, da von SVCs beim Verbindungsauf- und -abbau eine große Bandbreite beansprucht wird, jedoch die Kosten – bei immer verfügbarer virtueller Verbindung – gering sind.

Ein *PVC* ist eine ständig aufgebaute virtuelle Verbindung, die nur einen Zustand kennt: die Datenübertragung. PVCs kommen immer dann zum Einsatz, wenn zwischen Geräten ständig Daten ausgetauscht werden. Da bei PVCs der Verbindungsauf- und -abbau wegfällt, benötigen diese Verbindungen eine geringere Bandbreite, führen aber zu höheren Verbindungskosten, da die Leitung ständig geschaltet ist.

3.6 WAN-Einwahldienste

Einwahldienste stellen eine kostengünstige Variante der Verbindung über ein WAN dar. Die zwei bekanntesten Einwahlimplementationen sind Dial-on-Demand Routing (DDR) und Dial Backup.

DDR ist eine Technik, bei der ein Router leitungsvermittelte Sitzungen auf Anforderung von End-Stationen starten und beenden kann. Der Router wird dazu so konfiguriert, daß er den entsprechenden Datenverkehr (z.B. eines bestimmten Protokolls) erkennen kann. Wenn der Router den entsprechenden Datenverkehr empfängt, der zu einem fernen Netzwerk übertragen werden soll, baut der Router eine Verbindung auf und überträgt die Daten über diese Verbindung. Parallel läuft ein Timer, der immer wieder von neuem beginnt, wenn Daten über diese Verbindung übertragen werden. Empfängt der Router keine zu übertragenden Daten, bevor der Timer abläuft, wird die Verbindung beendet. Falls wieder Daten zum fernen Netzwerk übertragen werden sollen, baut der Router eine neue Verbindung auf. DDR kann anstelle von Punkt-zu-Punkt-Verbindungen und vermittelten Mehrfachzugriffs-WAN-Diensten eingesetzt werden.

Dial Backup ist ein Dienst, der unter bestimmten Bedingungen eine serielle Sicherungsleitung aktiviert. Eine sekundäre serielle Leitung kann als Sicherungsverbindung fungieren, die genutzt wird, wenn die primäre Verbindung ausfällt oder wenn zusätzliche Bandbreite erforderlich wird, weil die Last auf der

primären Leitung einen Grenzwert erreicht. Dial Backup bietet eine gewisse Vorsorge gegen Performance-Verluste im WAN und Ausfallzeiten.

3.7 WAN-Geräte

In einem WAN kommen eine Vielzahl verschiedener Geräte zum Einsatz, die typisch für eine WAN-Umgebung sind. Im folgenden Abschnitt werden diese Geräte vorgestellt, zu denen WAN-Switches, Zugriffs-Server, Modems, CSU/DSUs und ISDN-Terminaladapter gehören. Andere Geräte sind spezielle WAN-Geräte wie Router, ATM-Switches und Multiplexer.

3.7.1 WAN-Switch

Ein WAN-Switch ist ein Internetworking-Gerät mit mehreren Ports, das in Anbieternetzen zum Einsatz kommt. Von diesen Geräten wird zumeist Datenverkehr der Protokolle Frame Relay, X.25 und SMDS vermittelt. Diese Geräte gehören zur Verbindungsschicht des OSI-Referenzmodells. Bild 3.5 zeigt zwei Router an den Enden eines WAN, die über WAN-Switches miteinander verbunden sind.

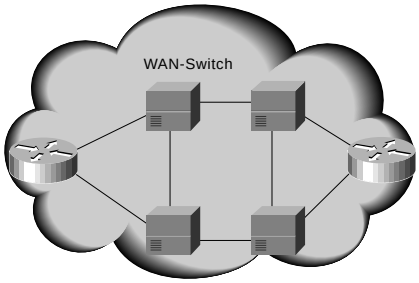


Bild 3.5: Zwei Router an den fernen Enden eines WAN können über WAN-Switches miteinander verbunden werden

3.7.2 Zugriffsserver

Ein Zugriffsserver fungiert als Konzentrator für sich ein- und auswählende Verbindungen. Bild 3.6 zeigt einen Zugriffsserver, der aus einem WAN sich auswählende Verbindungen konzentriert.

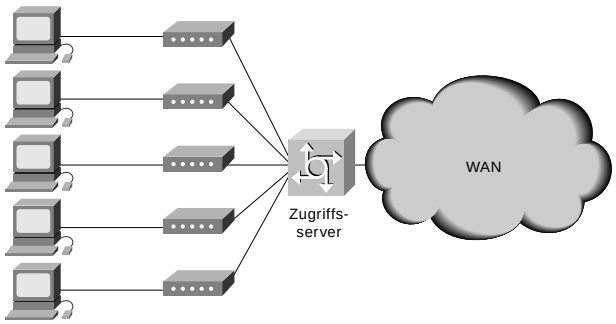


Bild 3.6: Ein Zugriffsserver konzentriert aus einem WAN sich auswählende Verbindungen

3.7.3 Modem

Ein Modem ist ein Gerät, das digitale und analoge Signale interpretiert und so Daten über normale Telefonleitungen senden kann. Auf der ausgehenden Seite werden die digitalen Signale so konvertiert, daß sie über analoge Kommunikationseinrichtungen übertragen werden können. Am Ziel, auf der eingehenden Seite, werden die analogen Signale wieder in digitale umgewandelt. Bild 3.7 zeigt eine einfache Verbindung zweier Modems über ein WAN.

Bild 3.7: Bei einer Modem-Verbindung über ein WAN werden analoge und digitale Signale verarbeitet

3.7.4 CSU/DSU

Eine Channel Service Unit/Data Service Unit (CSU/DSU – Kanal- und Datendiensteinheit) ist ein Gerät mit einer digitalen Schnittstelle (das manchmal aus zwei einzelnen digitalen Geräten besteht), über das die physische Schnittstelle eines DTE-Geräts (z.B. eines Terminals) an die Schnittstelle eines DCE-Geräts (z.B. eines Switches), das zum vermittelten Anbieter-Netz gehört, angeschlossen wird. Die CSU/DSU bietet für die Kommunikation zwischen diesen Geräten auch das Signal-Timing. Bild 3.8 zeigt, wo sich die CSU/DSU in einer WAN-Implementation befindet.

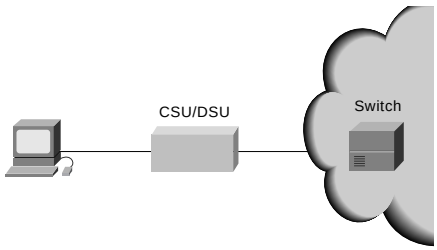


Bild 3.8: Die CSU/DSU befindet sich zwischen Switch und Terminal

3.7.5 ISDN-Terminal-Adapter

Der ISDN-Terminal-Adapter ist ein Gerät, mit dem an einen ISDN-Basisanschluß (BRI) weitere Geräte z.B. über die Schnittstelle EIA/TIA-232 angeschlossen werden können. Im Grunde handelt es sich beim Terminal-Adapter um ein ISDN-Modem. Bild 3.9 zeigt, wo sich ein Terminal-Adapter in einer ISDN-Umgebung befindet.

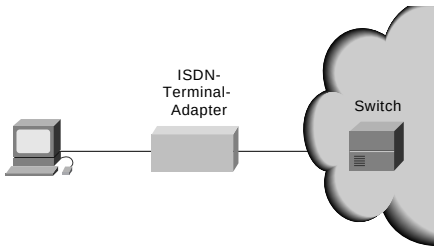


Bild 3.9: Über den Terminal-Adapter können weitere Geräte an das ISDN angeschlossen werden

KAPITEL 4

Grundlagen des Bridging und Switching

In diesem Kapitel werden die Technologien vorgestellt, die mit den als *Bridges* und *Switches* bezeichneten Geräten in Zusammenhang stehen. Zu den behandelten Themen gehören grundlegende Funktionen der Verbindungsschicht, das lokale und das ferne Bridging, ATM-Switching und LAN-Switching. Die Kapitel des Teils 4, »Bridging und Switching«, behandeln bestimmte Technologien ausführlich.

4.1 Was sind Bridges und Switches?

Bridges und Switches sind Datenkommunikationsgeräte, die in der Schicht 2 des OSI-Referenzmodells eingesetzt werden. Deshalb werden sie oft auch als Verbindungsschichtgeräte bezeichnet.

Bridges konnte man erstmals Anfang der 80er Jahre käuflich erwerben. Zu dieser Zeit dienten Bridges dazu, Pakete zwischen homogenen Netzen weiterzuleiten und diese so miteinander zu verbinden. In jüngerer Zeit wurde auch das Bridging zwischen unterschiedlichen Netzwerken definiert und standardisiert.

Viele verschiedene Arten des Bridging haben sich als wichtige Internetworking-Geräte bewährt. *Transparentes Bridging* findet man eher in Ethernet-Umgebungen, während *Source-Route-Bridging* in Token-Ring-Umgebungen beheimatet ist.

Übersetzendes Bridging bietet die Umformung von Formaten und Übertragungsprinzipien verschiedener Medien (normalerweise zwischen Ethernet und Token Ring). Das *transparente*

Source-Route-Bridging kombiniert die Algorithmen des transparenten Bridging und Source-Route-Bridging, um die Kommunikation in gemischten Ethernet/Token-Ring-Umgebungen zu ermöglichen.

Heute hat sich die Switchting-Technologie zu einem evolutionären Erbe für Bridging-basierte Internetworking-Lösungen entwickelt. Switching-Implementationen dominieren Anwendungen, in denen Bridging-Technologien bei früherem Netzwerk-Design vorgesehen waren. Die erstklassige Performance, größere Port-Dichte, niedrigere Kosten je Port und die höhere Flexibilität haben dazu beigetragen, daß Switches die Briding-Technologie ersetzen und ein Komplement zur Routing-Technologie bilden.

4.2 Überblick zu den Geräten der Verbindungsschicht

Bridging und Switching erfolgt auf Ebene der Verbindungsschicht, die den Datenfluß steuert, Übertragungsfehler bearbeitet, physische (im Gegensatz zur logischen) Adressierung bietet und den Zugriff auf das physische Medium verwaltet. Beim Einsatz von Bridges kamen dafür verschiedene Verbindungsschichtprotokolle zum Einsatz, die eine bestimmte Flußkontrolle, Fehlerbehandlung, Adressierung und Mediumzugriff-Algorithmen vorschrieben. Zu den bekannten Protokollen der Verbindungsschicht gehören das Ethernet, Token Ring und FDDI.

Bridges und Switches sind keine komplizierten Geräte. Sie analysieren eingehende Frames, entscheiden über die Weiterleitung anhand der in den Frames enthaltenen Informationen und übertragen die Frames zum Zielgerät. In bestimmten Fällen, z.B. beim Source-Route-Bridging, ist der gesamte Pfad zum Ziel im Frame enthalten. Beim transparenten Bridging z.B. werden die Frames immer nur schrittweise zum Ziel weitergeleitet.

Der zuerst zu nennende Vorteil, den Bridging und Switching bieten, ist die Transparenz für höhere Schichten. Da beide Gerätetypen auf der Ebene der Verbindungsschicht arbeiten, benötigen sie keine Informationen der höheren Schichten. D.h., daß von ihnen der Datenverkehr jedes beliebigen Ver-

mittlungsschichtprotokolls zügig weitergeleitet wird. Für eine Bridge ist es nicht ungewöhnlich, daß sie Datenverkehr der Protokolle AppleTalk, DECnet, TCP/IP, XNS usw. zwischen zwei oder mehr Netzwerken transportiert.

Bridges sind in der Lage, Frames anhand der Schicht-2-Felder zu filtern. So kann eine Bridge so programmiert werden, daß sie alle Frames eines bestimmten Netzwerks ablehnt (und nicht weiterleitet). Da die Informationen der Verbindungsschicht oft auch auf Protokolle höherer Schichten bezugnehmen, können Bridges auch anhand dieser Parameter filtern. Gerade bei der Behandlung unerwünschter Broadcast- und Multicast-Pakete ist diese Funktion hilfreich.

Auch bei der Unterteilung großer Netzwerke in selbständige Einheiten bieten Bridges und Switches verschiedene Vorteile. Weil nur ein bestimmter Prozentsatz des Datenverkehrs weitergeleitet wird, reduzieren Bridges und Switches das Verkehrsaufkommen in allen angeschlossenen Segmenten auf ein Minimum. Bei einigen Netzwerkfehlern können Bridges und Switches als Firewall dienen. Mit Bridges und Switches können wesentlich mehr Geräte verbunden werden, als dies beim Anschluß eines LAN an eine Bridge möglich ist. Bridges und Switches erweitern die effektive Länge eines LAN, indem weiter entfernte Stationen angeschlossen werden können.

Zwar haben Bridges und Switches viele Eigenschaften gemeinsam, es gibt jedoch auch entscheidende Unterschiede. So sind Switches wesentlich schneller, weil die Vermittlung auf der Hardware geschieht, während Bridges dazu Software verwenden. Switches sind in der Lage, LANs mit unterschiedlicher Bandbreite zu verbinden. So können z.B. ein 10-Mbit/s-Ethernet-LAN und ein 100-Mbit/s-Ethernet-LAN über einen Switch miteinander verbunden werden. Von Switches wird eine höhere Port-Dichte unterstützt als von Bridges. Einige Switches unterstützen das Cut-Through-Switching, was die Latenz und Verzögerungen in einem Netzwerk reduziert, während von Bridges nur ein Switching nach dem Speicher-und-Weiterleiten (Store and Forward) funktioniert. Schließlich helfen Switches auch bei der Vermeidung von Kollisionen in Netzwerk-Segmenten, da sie jedem Netzwerk-Segment eine bestimmte Bandbreite zuordnen.

4.3 Bridge-Typen

Bridges lassen sich in anhand verschiedener Eigenschaften kategorisieren. Eine der gängigsten Klassifikationen teilt Bridges in *lokale* oder *ferne (remote)* Bridges ein. *Lokale* Bridges verbinden mehrere, räumlich nahe LAN-Segmente. Remote Bridges verbinden mehrere, räumlich weit voneinander entfernte LAN-Segmente über Telefonleitungen. Bild 4.1 zeigt diese beiden Konfigurationen.

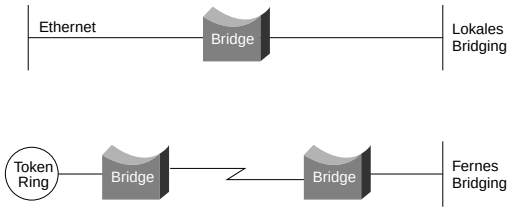


Bild 4.1: Lokale und ferne Bridges verbinden LAN-Segmente in unterschiedlicher räumlicher Entfernung

Remote Bridging sieht sich mehreren eindeutigen Internetworking-Anforderungen gegenüber, z.B. den unterschiedlichen Übertragungsgeschwindigkeiten im LAN und WAN. Auch wenn sich zur Zeit einige schnelle WAN-Technologien in räumlich verteilten Internetworks etablieren, bleibt doch die beträchtliche Differenz bei den Übertragungsgeschwindigkeiten bestehen. Der große Unterschied der LAN- und WAN-Geschwindigkeiten hält die Benutzer davon ab, verzögerungsempfindliche LAN-Anwendungen über ein WAN zu betreiben.

Remote Bridges können zwar die WAN-Geschwindigkeit nicht erhöhen, jedoch die Geschwindigkeitsunterschiede wenigstens bei ausreichender Pufferung kompensieren. So muß z.B. eine lokale Bridge den 3-Mbit/s-Datenstrom eines LAN-Geräts, das mit einem Gerät in einem fernen LAN kommunizieren will, so regeln, daß die serielle 64-Kbit/s-Leitung nicht vollkommen überlastet wird. Dazu puffert die Bridge die eingehenden Daten und sendet sie mit einer für die serielle Leitung verträglichen Geschwindigkeit. Diese Art der Pufferung funktioniert nur für kurzzeitig hohe Datenraten, die die Pufferkapazität der Bridge nicht überschreiten.

Das Institute of Electrical and Electronic Engineers (IEEE) unterteilt die Verbindungsschicht des OSI-Referenzmodells in

zwei einzelne Subschichten: die MAC-Subschicht (*Media Access Control*) und die LLC-Subschicht (*Logical Link Control*). Die MAC-Subschicht erlaubt und organisiert den Mediumzugriff, wie den Konkurrenzbetrieb und das Token-Passing, während die LLC-Subschicht das Framing, die Flußsteuerung, die Fehlerbehandlung und die MAC-Sublayer-Adressierung verwaltet.

Einige Bridges sind *MAC-Schicht-Bridges*, die eine Verbindung zwischen homogenen Netzwerken (z.B. IEEE 802.3 und IEEE 802.3) schaffen, andere Bridges können zwischen verschiedenen Protokollen der Verbindungsschicht (z.B. IEEE 802.3 und IEE-802.5) übersetzen. Wie eine solche Übersetzung funktioniert, veranschaulicht Bild 4.2.

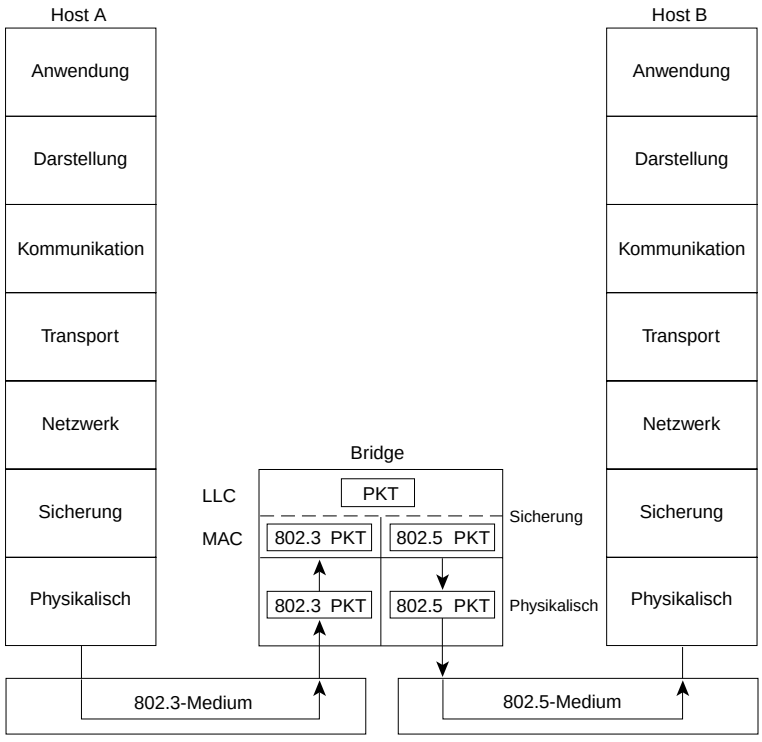


Bild 4.2: Eine Bridge der MAC-Schicht verbindet ein IEEE-802.3- und ein IEEE-802.5-Netzwerk

In Bild 4.2 wird ein IEEE-802.3-Host (Host A) gezeigt, der ein Paket mit Anwendungsdaten erstellt und es in einen IEEE-802.3-kompatiblen Frame kapselt, um es über ein IEEE-

802.3-Medium zu einer Bridge zu übertragen. Die Bridge entfernt den IEEE-802.3-Header auf Ebene der MAC-Subschicht und reicht den Frame zur Weiterverarbeitung an die darüberliegende LLC-Subschicht. Danach wird das Paket an die IEEE-802.5-Implementation heruntergereicht, von der das Paket mit einem IEEE-802.5-Header gekapselt wird, um es über ein IEEE-802.5-Netzwerk zum IEEE-802.5-Host (Host B) zu übertragen.

Die Übersetzung zwischen Netzwerken verschiedener Art durch eine Bridge kann nie vollständig geschehen, da einige Frame-Felder und Protokoll-Funktionen des einen Netzwerks nicht vom anderen unterstützt werden.

4.4 Switch-Typen

Switches sind Geräte der Verbindungsschicht, die wie Bridges mehrere physische LAN-Segmente zu einem großen Netzwerk verbinden. Ähnlich wie Bridges leiten Switches den Datenverkehr anhand von MAC-Adressen weiter. Da aber das Switching von der Hardware und nicht von einer Software ausgeführt wird, ist es bedeutend schneller. Switches setzen entweder die Technik des Zwischenspeicherns und Weiterleitens (store-and-forward) oder des Durchleitens (cut-through) ein. Switches gibt es in verschiedenen Ausführungen, als ATM-Switch, LAN-Switch und verschiedene WAN-Switches.

4.4.1 ATM-Switch

Ein *ATM*-Switch (*Asynchronous Transfer Mode*) bietet Hochgeschwindigkeits-Switching und skalierbare Bandbreiten für den Einsatz in Arbeitsgruppen, im unternehmensweiten Netzwerk-Backbone und in WANs. Von ATM-Switches werden Sprach-, Video- und Datenanwendungen unterstützt. Diese Switches sind so ausgelegt, daß sie die bei der ATM-Kommunikation verwendeten Dateneinheiten fester Länge switchen, die als *Zellen* bezeichnet werden. Bild 4.3 zeigt ein Unternehmensnetzwerk, das aus mehreren LANs besteht, die über einen ATM-Backbone miteinander verbunden sind.

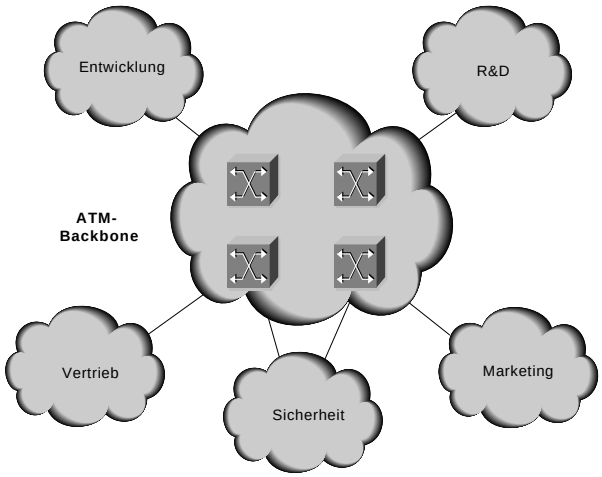


Bild 4.3: Netzwerke aus mehreren LANs können zum Vermitteln von Zellen einen ATM-basierten Backbone verwenden

4.4.2 LAN-Switch

LAN-Switches werden dazu eingesetzt, mehrere LAN-Segmente miteinander zu verbinden. Mit LAN-Switching werden die Voraussetzungen für eine dedizierte, kollisionsfreie Kommunikation zwischen Netzwerk-Geräten geschaffen, wobei mehrere Teilnehmer gleichzeitig aktiv sein können. LAN-Switches sind so ausgelegt, daß sie Daten-Frames mit hoher Geschwindigkeit vermitteln können. Bild 4.4 zeigt ein einfaches Netzwerk, in dem ein LAN-Switch ein 10-Mbit/s- und ein 100-Mbit/s-Ethernet-LAN verbindet.

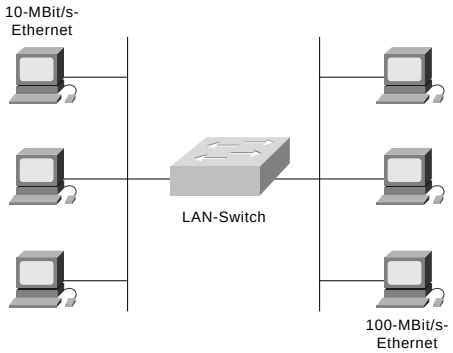


Bild 4.4: Ein LAN-Switch kann unterschiedlich schnelle Ethernet-Segmente verbinden

KAPITEL 5

Grundlagen des Routing

In diesem Kapitel werden die am häufigsten in Routing-Protokollen eingesetzten, grundlegenden Konzepte erläutert. Zu den behandelten Themen gehören die Komponenten und Algorithmen der Routing-Protokolle. Außerdem wird kurz auf die Unterschiede zwischen Routing-Protokollen und gerouteten, oder Netzwerk-, Protokollen eingegangen. Die Kapitel des Teils 6, »Routing-Protokolle«, behandeln bestimmte Routing-Protokolle ausführlich. Netzwerk-Protokolle, die Routing-Protokolle verwenden, werden in Teil 5, »Netzwerk-Protokolle« besprochen.

5.1 Was ist Routing?

Routing ist der Vorgang, bei dem Daten über ein Internetwork von einer Quelle zu einem Ziel übertragen werden. Auf diesem Weg befindet sich normalerweise mindestens ein dazwischen geschalteter Knoten. Routing wird oft dem Bridging gegenübergestellt, was für den normalen Betrachter zunächst kein Unterschied ist. Der primäre Unterschied ist, daß Bridging in der Schicht 2 (der Verbindungsschicht) erfolgt, während Routing in der Schicht 3 (der Vermittlungsschicht) stattfindet. Aufgrund dieses Unterschieds werden beim Routing und Bridging verschiedene Daten für die Übertragung von der Quelle zum Ziel verwendet, so daß diese beiden Funktionen ihre Aufgabe auf verschiedene Weise erfüllen.

Das Thema Routing wurde zwar seit mehr als zwei Jahrzehnten in der wissenschaftlichen Computerliteratur behandelt, zum kommerziellen Einsatz kam das Routing aber erst Mitte der 80er Jahre. Der primäre Grund für diese Verzögerung ist darin zu sehen, daß die Netzwerke der 70er Jahre einfache und homogene Umgebungen darstellten. Erst in jüngster Zeit wurde das Internetworking in großem Stil populär.

5.2 Komponenten des Routing

Zum Routing gehören zwei grundlegende Aktivitäten: das Ermitteln eines optimalen Routing-Pfads und die Übertragung von Datengruppen (auch als Pakete bezeichnet) über ein Internetwork. Dabei bezeichnet man letzteres als Switching. Selbst wenn das Switching relativ geradlinig zu sein scheint, kann die Pfadermittlung ein sehr komplexer Vorgang sein.

5.2.1 Pfadermittlung

Mit einem Meßparameter, z.B. der Pfadlänge, wird vom Routing-Algorithmus der optimale Pfad zum Ziel ermittelt. Außerdem werden von Routing-Algorithmen sog. *Routing-Tabellen* angelegt und gepflegt, um den Prozeß der Pfadermittlung zu unterstützen. Die darin enthaltenen Route-Informationen sind je nach Routing-Algorithmus verschieden.

Die Routing-Algorithmen füllen die Routing-Tabellen mit den verschiedensten Informationen. Die Kombinationen aus Ziel/nächster Hop signalisieren dem Router, daß ein bestimmtes Ziel am schnellsten erreicht wird, wenn das Paket zu einem bestimmten Router gesendet wird, der den »nächsten Hop« auf dem Weg zum Ziel darstellt. Nachdem ein Router ein Paket empfangen hat, prüft er die Zieladresse und versucht, diese einem nächsten Hop zuzuordnen. Bild 5.1 veranschaulicht eine Routing-Tabelle, die nach dem Prinzip Ziel/nächster Hop aufgebaut ist.

Zielnetzwerk:	Nächster Knoten:
27	Knoten A
57	Knoten B
17	Knoten C
24	Knoten A
52	Knoten A
16	Knoten B
26	Knoten A
. . .	. . .

Bild 5.1: Die Kombinationen aus Ziel/nächster Hop geben den optimalen Pfad für die Daten vor

Routing-Tabellen enthalten darüber hinaus Informationen, z.B. Daten über die Priorität eines Pfads. Router vergleichen die Meßparameter, um die optimalen Routen zu finden, wobei diese Parameter in Abhängigkeit vom Routing-Algorithmus variieren. Weiter unten in diesem Kapitel werden die verschiedenen, gebräuchlichen Meßparameter vorgestellt und beschrieben.

Router kommunizieren miteinander und pflegen ihre Routing-Tabellen mit Hilfe einer Vielzahl von *Nachrichten*. Die Nachricht *Routing-Aktualisierung* ist eine solche Nachricht, die die gesamte oder Teile der Routing-Tabelle enthält. Durch die Analyse der von anderen Routern gesendeten Routing-Aktualisierungen kann ein Router ein detailliertes Bild der Netzwerk-Topologie entwerfen. Ein weiteres Beispiel für Nachrichten, die zwischen Routern ausgetauscht werden, ist die Link-Status-Anzeige (Link-State Advertisement), die andere Router über den Status der Links des Senders informiert. Auch diese Link-Informationen können dazu genutzt werden, ein vollständiges Bild der Netzwerk-Topologie zu entwerfen, um es den Routern zu ermöglichen, die optimalen Routen zum Netzwerk-Ziel zu ermitteln.

5.2.2 Switching

Switching-Algorithmen sind relativ einfach aufgebaut und im Prinzip für die meisten Routing-Protokolle identisch. Meistens muß ein Host ein Paket an einen anderen senden. Verfügt der Quell-Host über die Router-Adresse, sendet er das Paket direkt an die physische (MAC-)Adresse des Router und versieht das Paket mit der Protokolladresse (Vermittlungsschicht) des Ziel-Hosts.

Der Router prüft die Ziel-Protokolladresse und stellt fest, ob bekannt ist, wie das Paket weiter gesendet werden kann. Wenn der Router nicht weiß, wie das Paket weiterzuleiten ist, wird das Paket »fallengelassen«. Andernfalls ändert der Router die physische Zieladresse auf die des nächsten Hop und sendet das Paket.

Der nächste Hop kann tatsächlich der Ziel-Host sein. Ist dies nicht der Fall, handelt es sich beim nächsten Hop um einen weiteren Router, der den gleichen Switching-Entscheidungsprozeß ausführt. Beim Weg des Pakets durch das Internetwork wird dessen physische Adresse mehrfach geändert, die Protokolladresse jedoch bleibt gleich (siehe Bild 5.2).

Damit ist der Weg von einem Quell- zu einem Ziel-Endsystem per Switching beschrieben. Die International Standardization Organization (ISO) hat eine hierarchische Terminologie entwickelt, die zur Beschreibung dieses Prozesses hilfreich ist. In dieser Terminologie werden Netzwerk-Geräte, die keine Pakete zwischen Sub-Netzwerken versenden können, als *End-Systeme* (ES) bezeichnet. Netzwerk-Geräte, die genau diese Fähigkeit aufweisen, werden als *intermediäre Systeme* (IS) bezeichnet. IS können unterteilt werden in Geräte, die innerhalb von Routing-Domänen kommunizieren können (*intradomäne IS*), und solchen, die sowohl innerhalb als auch über ihre Routing-Domäne hinaus kommunizieren können (*interdomäne IS*). Eine *Routing-Domäne* ist ein Teil eines Internetwork, der normal administriert wird, aber bestimmten Administrationsregeln unterworfen ist. Routing-Domänen werden auch als *autonome Systeme* bezeichnet. Mit Hilfe bestimmter Protokolle lassen sich Routing-Domänen in *Routing-Bereiche* unterteilen, wobei Intradomänen-Routing-Protokolle sowohl

für das Switching innerhalb als auch zwischen Bereichen verwendet wird.

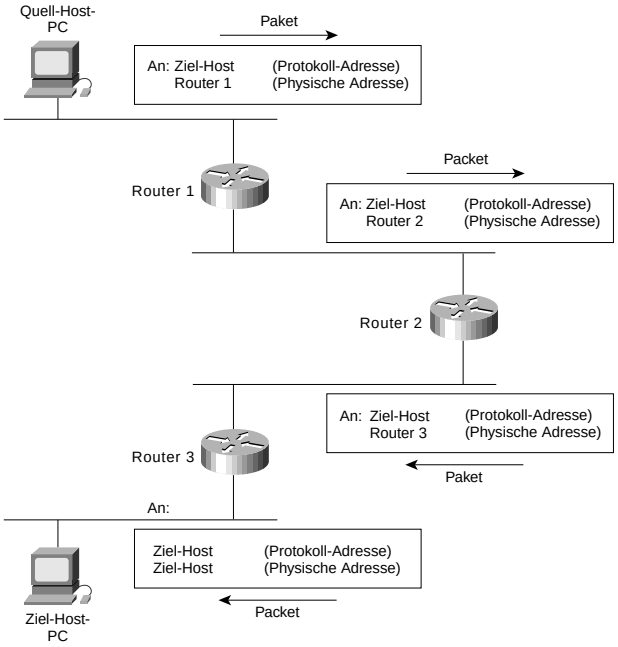


Bild 5.2: Während des Switching-Prozesses werden mehrere Router angesprochen

5.3 Routing-Algorithmen

Routing-Algorithmen können anhand bestimmter Eigenschaften unterschieden werden. Zuerst beeinflussen die Ziele desjenigen, der den Algorithmus entworfen hat, die Funktion des daraus entstandenen Routing-Protokolls. Zum zweiten stehen eine Vielzahl von Routing-Algorithmen zur Verfügung, wovon jeder die Netzwerk- und Router-Ressourcen auf andere Weise beeinflußt. Schließlich verwenden die Routing-Algorithmen die verschiedensten Meßparameter, die Einfluß auf die Berechnung der optimalen Route haben. In den folgenden Abschnitten werden diese Eigenschaften der Routing-Algorithmen analysiert.

5.3.1 Entwicklungsziele

Routing-Algorithmen werden mit einem oder mehreren der folgenden Ziele entwickelt:

- Optimierung
- Einfachheit und geringer Overhead
- Robustheit und Stabilität
- Schnelle Konvergenz
- Flexibilität

Optimierung bezieht sich auf die Eigenschaft des Routing-Algorithmus, die beste Route auszuwählen, was vom Meßparameter und seinen Gewichtungen bei der Berechnung abhängt. So kann z.B. ein Algorithmus eine große Anzahl von Hops und Verzögerungen ermitteln, jedoch bei der Berechnung der Route die Verzögerungen stärker berücksichtigen. Natürlich müssen die Routing-Protokolle die Berechnung in den Meßparametern der Algorithmen durchgängig definieren.

Routing-Algorithmen werden so einfach wie möglich entworfen. Das heißt, die Funktion des Algorithmus muß effektiv erreicht werden, mit einem Minimum an Software- und Bedienungs-Overhead. Die Effektivität spielt eine besondere Rolle, wenn die Software, mit der der Routing-Algorithmus implementiert wird, auf einem Computer mit beschränkten physischen Ressourcen läuft.

Bei der Robustheit eines Routing-Algorithmus kommt es darauf an, daß der Algorithmus auch in Ausnahmesituationen und unter unvorhersehbaren Bedingungen wie Hardware-Ausfällen, Überlastung und Fehlimplementationen korrekt abläuft. Da Router an den Knotenpunkten von Netzwerken eingesetzt werden, kann deren Ausfall zu schwerwiegenden Problemen führen. Die besten Routing-Algorithmen sind solche, die bereits länger im Einsatz sind und ihre Stabilität in einem Netzwerk unter den verschiedensten Bedingungen bewiesen haben.

Ein Routing-Algorithmus muß schnell konvergieren können. Unter Konvergenz versteht man den Prozeß der Abstimmung aller Router zu einer optimalen Route. Bevor ein Router auf-

grund eines Ereignisses in einem Netzwerk heruntergefahren wird oder nicht mehr verfügbar ist, sendet er Aktualisierungs-Nachrichten zum Routing, die nach und nach die Netzwerke erreichen, so daß eine Neuberechnung der optimalen Routen erfolgt, der eventuell alle Router zustimmen. Zu langsam konvergierende Routing-Algorithmen können zu Routing-Schleifen und Netzwerk-Ausfällen führen.

In Bild 5.3 ist eine Routing-Schleife dargestellt: Zu einem Zeitpunkt t1 geht ein Paket beim Router 1 ein. Router 1 wurde bereits aktualisiert und hat erkannt, daß die optimale Route zum Ziel über Router 2 als nächstem Hop führt. Also sendet Router 1 das Paket an Router 2 weiter. Da dieser Router jedoch noch nicht aktualisiert wurde, betrachtet er Router 1 als nächsten Hop auf dem optimalen Weg zum Ziel des Pakets. Deshalb sendet Router 2 das Paket wieder an Router 1. Nun wird das Paket zwischen diesen beiden Routern hin und her geschickt, und zwar so lange, bis auch Router 2 aktualisiert wurde oder das Paket die maximale Switch-Anzahl erreicht hat.

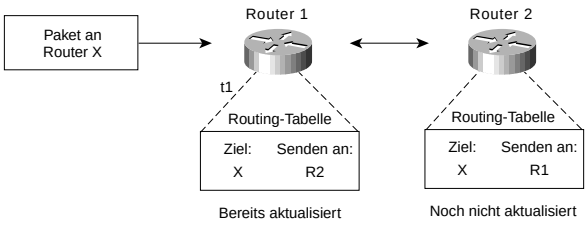


Bild 5.3: Langsame Konvergenz und Routing-Schleifen können die Weiterleitung behindern

Routing-Algorithmen sollten auch flexibel sein, d.h., sie sollten sich schnell und genau an eine Vielzahl von Netzwerk-Umständen anpassen können. Wenn z.B. ein Netzwerk-Segment ausfällt und dies von den Routern bemerkt wird, wählen die meisten Routing-Algorithmen den nächstbesten Pfad für alle Router, die sonst dieses ausgefallene Segment benutzen. Routing-Algorithmen können so programmiert werden, daß sie auf Änderungen in der Netzwerk-Bandbreite, den Umfang der Router-Warteschlange und Netzwerk-Verzögerungen etc. reagieren.

5.3.2 Algorithmusarten

Routing-Algorithmen können im Hinblick auf die folgenden Merkmale klassifiziert werden:

- Statisch oder dynamisch
- Einzel-Pfad oder Multi-Pfad
- Eben oder hierarchisch
- Host-Intelligenz oder Router-Intelligenz
- Intradomän oder interdomän
- Link-Status oder Entfernungsvektor

Statisch oder dynamisch

Statische Routing-Algorithmen sind eigentlich keine Algorithmen, sondern genaugenommen tabellarische Zuordnungen, die ein Netzwerk-Administrator zu den Anfangszeiten des Routing eingerichtet hat. An diesen Zuordnungen ändert sich nichts, es sei denn, der Administrator nimmt eine Änderung vor. Algorithmen, die statische Router verwenden, sind einfach zu entwerfen und eignen sich gut für Umgebungen, in denen der Datenverkehr im Netzwerk relativ vorhersehbar ist und die Netzwerk-Struktur sich einfach gestaltet.

Da statische Routing-Systeme nicht auf Netzwerk-Veränderungen reagieren können, sind sie im allgemeinen für heutige, große und sich ständig ändernde Netzwerke wenig geeignet. Die meisten der in den 90er Jahren vorherrschenden Algorithmen sind dynamische Routing-Algorithmen, die sich wechselnden Netzwerk-Bedingungen anpassen, indem sie eingehende Routing-Aktualisierungsnachrichten auswerten. Wenn eine solche Nachricht eine Änderung am Netzwerk mitteilt, werden die Routen durch die Routing-Software erneut berechnet, und es werden neue Routing-Aktualisierungsnachrichten versendet. Diese Meldungen durchdringen nach und nach das gesamte Netzwerk, so daß alle Router dazu veranlaßt werden, ihre Algorithmen erneut zu starten und die Routing-Tabellen entsprechend anzupassen.

Dynamische Routing-Algorithmen können dort, wo es geeignet erscheint, mit statischen Routen versehen werden. Der *Router des letzten Auswegs* (ein Router, an den alle nicht-routingfähige Pakete gesendtet werden) z.B. kann als Aufbewahrungsort für alle nicht-routingfähige Pakete eingerichtet werden, um sicherzustellen, daß alle Meldungen bearbeitet werden.

Einzel-Pfad oder Multi-Pfad

Einige ausgefeilte Routing-Protokolle unterstützen mehrere Pfade zu ein und demselben Ziel. Anders als Einzel-Pfad-Algorithmen lassen diese Multi-Pfad-Algorithmen ein Multiplexing über mehrere Leitungen zu. Die Vorteile der Multi-Pfad-Algorithmen liegen auf der Hand: Sie erreichen eindeutig höhere Durchsatzraten und sind zuverlässiger.

Eben oder hierarchisch

Einige Routing-Algorithmen arbeiten in einem ebenen Adreßraum, während andere einen hierarchischen Adreßraum verwenden. In einem ebenen Routing-System sind alle Router gleichberechtigt. In einem hierarchischen Routing-System bilden einige Router das, was man einen Routing-Backbone nennt. Dabei werden die Pakete von normalen Routern zu den Backbone-Routern gesendet, von wo aus sie über den Backbone bis zum Zielbereich geschickt werden. Erst ab diesem Punkt werden die Pakete von einem Backbone-Router zu einem oder mehreren normalen Routern bis zum Ziel gesendet.

In Routing-Systemen werden oft logische Gruppen von Knoten gebildet, die als Domänen, autonome Systeme oder Bereiche bezeichnet werden. In einem hierarchischen System können bestimmte Router einer Domäne mit Routern in anderen Domänen kommunizieren, während die anderen Router nur innerhalb ihrer Domäne kommunizieren können. In sehr umfangreichen Netzwerken können zusätzliche hierarchische Ebenen vorhanden sein, in denen die Router auf der höchsten Hierarchie-Ebene den Routing-Backbone bilden.

Der primäre Vorteil des hierarchischen Routings liegt darin, daß dabei meistens die Organisationsstruktur einer Firma nachgebildet wird und damit deren Verkehrsaufkommen am besten unterstützt wird. Der meiste Datenverkehr spielt sich

innerhalb kleiner Gruppen (Domänen) ab. Da interdomäne Router nur die anderen Router der eigenen Domäne kennen müssen, können deren Routing-Algorithmen vereinfacht werden. Abhängig vom verwendeten Routing-Algorithmus kann damit der Anteil der Routing-Aktualisierungsnachrichten am Datenverkehr reduziert werden.

Host-Intelligenz oder Router-Intelligenz

Einige Routing-Algorithmen gehen davon aus, daß der Quell-Endknoten die gesamte Route festlegt. In diesem Fall spricht man vom *Quell-Routing*. In Systemen mit Quell-Routing fungieren Router nur als Geräte, die zwischenspeichern und weiterleiten und Pakete einfach nur zum nächsten Router senden.

Andere Algorithmen arbeiten unter der Annahme, daß Hosts nichts von Routern wissen. In diesem Fall legen Router anhand ihrer eigenen Berechnungen den gesamten Pfad eines Pakets durch das Netzwerk fest. Im erstgenannten System liegt die Routing-Intelligenz bei den Hosts, im zweiten System bei den Routern.

Der wesentliche Unterschied zwischen Host-intelligentem und Router-intelligentem Routing besteht in der Frage nach dem optimalen Pfad und nach dem Verkehrsaufkommen. Host-intelligente Systeme wählen öfter die bessere Route, da sie alle möglichen Routen zu einem Ziel ermitteln, bevor sie das Paket tatsächlich absenden. Dabei wählen sie den besten Pfad in Abhängigkeit davon, wie auf dem einzelnen System der »optimale« Pfad definiert ist. Allerdings fällt beim Ermitteln aller Routen eines Pakets ein nicht zu vernachlässigender Datenverkehr an, abgesehen von der Zeit, die für dieses Unterfangen benötigt wird.

Intradomän oder interdomän

Einige Routing-Algorithmen betrachten nur die eigene Domäne. Andere schauen über die eigene Domäne hinaus. Die Natur dieser beiden Algorithmentypen ist verschieden. Es leuchtet ein, daß ein optimaler intradomäner Routing-Algorithmus nicht unbedingt auch als interdomäner Algorithmus geeignet ist.

Link-Status oder Distance-Vektor (Entfernungsvektor)

Link-Status-Algorithmen (auch bekannt als *Algorithmen des kürzesten Pfads*) reichen Routing-Informationen an alle Knoten des Internetwork. Dabei sendet jeder Router nur den Teil seiner Routing-Tabelle, die den Status seiner eigenen Links beschreibt. Entfernungsvektor-Algorithmen (auch bekannt als *Bellman-Ford-Algorithmen*) fordern jeden Router auf, einen Teil oder seine gesamte Routing-Tabelle zu senden, und zwar nur an seine direkten Nachbarn. Kurz gesagt, senden Link-Status-Algorithmen kleine Aktualisierungen an alle, wohingegen Entfernungsvektor-Algorithmen umfangreichere Aktualisierungen nur an die benachbarten Router senden.

Weil sie schneller konvergieren, neigen Link-Status-Algorithmen weniger zu Routing-Schleifen als Entfernungsvektor-Algorithmen. Andererseits beanspruchen Link-Status-Algorithmen die CPU und den Arbeitsspeicher stärker, als dies bei Entfernungsvektor-Algorithmen der Fall ist. Deshalb können Link-Status-Algorithmen aufwendiger zu implementieren und zu warten sein. Abgesehen von den genannten Unterschieden laufen beide Algorithmen unter den meisten Umständen einwandfrei.

5.3.3 Routing-Meßparameter

Die Routing-Tabellen enthalten Informationen, die von der Switching-Software verwendet werden, um die beste Route zu ermitteln. Ausgefeilte Routing-Algorithmen können ihre Routing-Auswahl nach mehreren Gesichtspunkten treffen, indem sie diese zu einem einzigen (hybriden) Meßparameter vereinigen. Alle folgenden Parameter werden verwendet:

- Pfadlänge
- Zuverlässigkeit
- Verzögerung
- Bandbreite
- Auslastung
- Kommunikationskosten

Die Pfadlänge ist der gebräuchlichste Routing-Meßparameter. Einige Routing-Protokolle ermöglichen es den Netzwerk-Administratoren, jedem Netzwerk-Link willkürliche Kosten zuzuordnen. Dann errechnet sich die Pfadlänge als Summe aller Kosten der verwendeten Links. Andere Routing-Protokolle definieren Hopcount (Sprungzähler), ein Meßparameter, der die Anzahl der passierten Internetworking-Produkte (z.B. Router) festlegt, die von einem Paket auf der Route zwischen Quelle und Ziel liegen müssen.

Von *Zuverlässigkeit* im Zusammenhang mit Routing-Algorithmen ist die Rede, wenn die Zuverlässigkeit jedes Netzwerk-Links (als Bit-Fehlerrate angegeben) betrachtet wird. Einige Netzwerk-Links fallen möglicherweise öfter aus als andere. Nach einem Netzwerk-Ausfall werden einige Netzwerk-Links einfacher oder schneller wieder instandgesetzt als andere. Alle Faktoren, die die Zuverlässigkeit beschreiben, können in die Zuordnung von Zuverlässigkeitsrängen eingehen. Dabei handelt es sich um willkürliche Zahlenwerte, die normalerweise vom Netzwerk-Administrator jedem Link zugewiesen werden.

Routing-Delay (Verzögerung) meint die Dauer einer Übertragung durch ein Internetwork von der Quelle bis zum Ziel. In die Verzögerung gehen viele Faktoren ein, zu denen auch die Bandbreite der zwischengeschalteten Netzwerk-Links, die Port-Warteschlange in jedem Router, Staus an den zwischengeschalteten Netzwerk-Links und die zu überbrückende physische Entfernung gehören. Da es sich bei der Verzögerung um ein Konglomerat aus den verschiedensten, wichtigen Variablen handelt, wird dieser Parameter oft verwendet.

Die *Bandbreite* bezieht sich auf die Kapazität hinsichtlich des Datenverkehrs auf einer Verbindung. Alles andere spielt keine Rolle, wenn ein 10-Mbit/s-Ethernet-Link an eine 64-Kbit/s-Leitung angeschlossen wird, die geleast wird. Auch wenn die Bandbreite die Reihenfolge des höchsten Durchsatzes eines Links bestimmt, müssen Routen über Links mit größerer Bandbreite nicht unbedingt die besseren Routen sein, als solche, die mit geringerer Bandbreite arbeiten. Wenn z.B. ein schneller Link stärker belastet wird, kann die tatsächlich benötigte Zeit für die Übertragung eines Pakets größer sein.

Auslastung bezieht sich darauf, wie stark eine Netzwerk-Ressource, z.B. ein Router, beansprucht wird. Die Auslastung kann auf verschiedene Weise berechnet werden. So können die CPU-Auslastung und die Anzahl der verarbeiteten Pakete pro Sekunde darin eingehen. Allein die kontinuierliche Überwachung dieser Parameter kann bereits sehr ressourcen-intensiv sein.

Telefonkosten sind ein weiterer wichtiger Parameter, zumal es Firmen gibt, denen die Kostenseite wichtiger ist als die Performance. Selbst wenn die Verzögerung auf den eigenen Leitungen größer ist als bei der Übertragung über öffentliche Telefonleitungen, werden die eigenen Leitungen bevorzugt, weil die öffentlichen Telefonleitungen Geld kosten.

5.4 Netzwerk-Protokolle

Geroutete Protokolle werden von Routing-Protokollen über das Netzwerk übertragen. In diesem Kontext werden geroutete Protokolle auch als *Netzwerk*-Protokolle bezeichnet. Von diesen Netzwerk-Protokollen werden eine Vielzahl an Funktionen ausgeführt, die für die Kommunikation zwischen Benutzer-Anwendungen in Quell- und Zielgeräten erforderlich sind. Dabei unterscheiden sich diese Funktionen zwischen den verschiedenen Protokollfamilien erheblich. Die Netzwerk-Protokolle benützen die oberen vier Schichten des OSI-Referenzmodells: in der Transport-, der Sitzungs-, der Darstellungs- und der Anwendungsschicht.

Daß es zwischen den Begriffen *geroutetes Protokoll* und *Routing-Protokoll* zu Verwechslungen kommt, ist normal. Geroutete Protokolle sind jene, die über ein Netzwerk geroutet werden. Zu diesen Protokollen gehören z.B. das Internet Protocol (IP), DECnet, AppleTalk, Novell NetWare, OSI, Banyan VINES und Xerox Network System (XNS). Routing-Protokolle sind jene, die die Routing-Algorithmen implementieren. Um es einfach zu sagen: Routing-Protokolle leiten Netzwerk-Protokolle durch ein Internetwork. Zu diesen Protokollen gehören z.B. das Interior Gateway Routing Protocol (IGRP), das Enhanced Interior Gateway Routing Protocol (Enhanced IGRP), das Open Shortest Path First (OSPF), das Exterior Gateway Protocol (EGP), das Border Gateway Protocol

(BGP), das Intermediate System to Intermediate System (IS-IS) und das Routing Information Protocol (RIP). Geroutete und Routing-Protokolle werden weiter unten ausführlicher besprochen.

KAPITEL 6

Grundlagen des Netzwerk-Managements

In diesem Kapitel werden die Funktionen beschrieben, die den meisten Netzwerk-Management-Architekturen und -Protokollen gemeinsam sind. Außerdem wird auf die fünf konzeptionellen Bereiche des Managements eingegangen, wie sie von der International Organization for Standardization (ISO) definiert wurden. In den nachfolgenden Kapiteln in Teil 7, »Netzwerkverwaltung«, werden die spezifischen Netzwerk-Management-Technologien, Protokolle und Plattformen detaillierter besprochen.

6.1 Was ist Netzwerk-Management?

Netzwerk-Management hat für verschiedene Personen unterschiedliche Bedeutung. In manchen Fällen meint man einen einzelnen Netzwerk-Consultant, der mit einem veralteten Protokoll-Analyzer die Netzwerk-Aktivitäten überwacht. In anderen Fällen gehören zum Netzwerk-Management eine verteilte Datenbank, selbständiges Pollen von Netzwerk-Geräten und High-End-Workstations, die mit Echtzeit-Grafiken Änderungen der Netzwerk-Topologie und den Datenverkehr darstellen. Ganz allgemein läßt sich sagen, daß Netzwerk-Management einen Service darstellt, der eine Vielzahl verschiedener Werkzeuge, Anwendungen und Geräte umfaßt, die den Netzwerk-Manager bei der Überwachung und Verwaltung eines Netzes unterstützen.

6.1.1 Ein geschichtlicher Rückblick

In den frühen 80er Jahren kam es zu einer erschreckenden Ausweitung des Personalbedarfs im Bereich Netzwerke. Zu dieser Zeit wurde erkannt, welche Kostenvorteile und Produktivitätsgewinne durch den Einsatz von Netzwerk-Technologie möglich waren. So wurden neue Netze aufgebaut und die vorhandenen erweitert, und zwar so schnell wie neue Netzwerk-Technologien und -Produkte am Markt eingeführt wurden. Mitte der 80er Jahre machten einige Firmen leidvolle Erfahrungen, weil sie viele verschiedene (und manchmal inkompatible) Netzwerk-Technologien im Einsatz hatten.

Die Probleme aus der Netzwerk-Erweiterung betrafen sowohl das tägliche Netzwerk-Management als auch die strategische Netzwerk-Planung. Jede neue Netzwerk-Technologie erforderte ihre eigenen Experten. In den frühen 80er Jahren führte allein der Personalbedarf für die Verwaltung großer, heterogener Netze zu einer Krise in vielen Firmen. So kam es zu einer dringenden Nachfrage nach automatisiertem Netzwerk-Management (einschließlich dessen, was man als *Kapazitätsplanung für Netzwerke* bezeichnet) über unterschiedliche Umgebungen hinweg.

6.2 Netzwerk-Management-Architektur

Die meisten Management-Architekturen für Netzwerke verwenden die gleiche grundlegende Struktur und die gleichen Beziehungen. Auf Computern und anderen Netzwerk-Geräten (*end stations* oder *managed devices*) läuft eine Software, die auftretende Probleme sofort meldet (z.B. wenn ein oder mehrere benutzerabhängige Grenzwerte erreicht werden). *Management-Einrichtungen* sind so programmiert, daß sie beim Empfang einer solchen Meldung eine, mehrere oder eine Gruppe von Aktionen ausführen. Dazu gehören die Benachrichtigung des Operators, die Ereignis-Protokollierung, die Systembeendigung und automatische Reparaturversuche.

Management-Einrichtungen können von Stationen im Netz (end station) Daten anfordern (poll), um die Werte bestimmter Variablen zu überprüfen. Das Polling kann automatisch erfolgen oder von einem Benutzer angestoßen werden. Die *Agenten* in den verwalteten Geräten reagieren jedoch immer auf alle

Polls. Agenten sind Software-Module, die zuerst die Daten des verwalteten Geräts kompilieren, diese dann in einer *Management-Datenbank* ablegen und schließlich den Management-Einrichtungen in *Netzwerk-Management-Systemen* (NMS) über ein *Netzwerk-Management-Protokoll* zur Verfügung stellen (pro-aktiv oder re-aktiv). Die bekannten Netzwerk-Management-Protokolle beinhalten das Simple Network Management Protocol (SNMP) und Common Management Information Protocol (CMIP). *Management-Proxies* sind Einrichtungen, die Management-Daten anstelle anderer Einrichtungen zur Verfügung stellen. Bild 6.1 zeigt eine typische Netzwerk-Management-Architektur.

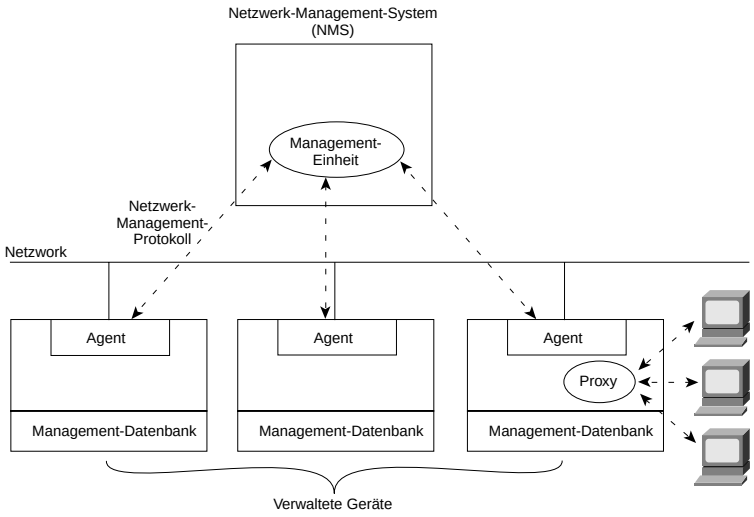


Bild 6.1: Eine typische Netzwerk-Management-Architektur verwaltet viele Beziehungen

6.3 ISO Netzwerk-Management-Modell

Die ISO hat einen großen Beitrag zur Standardisierung von Netzwerken geleistet. Das Netzwerk-Management-Modell der ISO ist das beste Hilfsmittel, um die Grundfunktionen eines Netzwerk-Management-Systems zu verstehen. Dieses Modell besteht aus fünf konzeptionellen Teilen:

– Performance-Management

– Konfiguration-Management

- Accounting-Management
- Fehler-Management
- Sicherheits-Management

6.3.1 Performance-Management

Das Ziel des *Performance-Managements* ist es, die Netzwerk-Performance unter verschiedenen Aspekten zu messen und verfügbar zu machen, damit die gesamte Netzwerk-Performance (Internetwork performance) auf einem akzeptablen Niveau verwaltet werden kann. Zu den Performance-Variablen, die bereitgestellt werden sollten, gehören z.B. der Netzwerk-Durchsatz, Benutzer-Antwortzeiten und Leitungsnutzung.

Das Performance-Management umfaßt drei Hauptschritte. Als erstes werden Performance-Daten aus Variablen gesammelt, die für den Netzwerk-Administrator von Bedeutung sind. Als zweites wird analysiert, ob sich die Daten im normalen Bereich bewegen. Schließlich werden geeignete Performance-Grenzwerte für jede wichtige Variable festgelegt, so daß bei Überschreiten eines solchen Werts ein Netzwerk-Problem angezeigt wird.

Management-Einrichtungen überwachen die Performance-Variablen ständig. Wenn ein Grenzwert erreicht wird, wird eine Meldung generiert und an das Netzwerk-Management-System gesendet.

Jeder der soeben beschriebenen Schritte ist Teil des Prozesses, der ein reaktives System ausmacht. Wenn die Performance zu stark absinkt, weil ein benutzerdefinierter Grenzwert erreicht wird, reagiert das System, indem es eine Meldung sendet. Das Performance-Management läßt auch pro-aktive Methoden zu: So kann z.B. eine Netzwerk-Simulation dazu verwendet werden, herauszufinden, wie sich die Vergrößerung eines Netzwerks auf die Performance-Parameter auswirkt. Solche Simulationen können dem Administrator in Kürze auftretende Probleme melden, so daß er Gegenmaßnahmen ergreifen kann.

6.3.2 Konfigurations-Management

Ziel des *Konfigurations-Managements* ist es, die Daten der Netzwerk- und Systemkonfiguration zu überwachen, so daß die Auswirkungen verschiedenster Hard- und Software auf den Netzwerkbetrieb verfolgt und verwaltet werden können.

Zu jedem Netzwerk-Gerät gibt es unterschiedliche Versionsinformationen. Eine Engineering-Workstation könnte z.B. wie folgt konfiguriert sein:

- Betriebssystem, Version 3.2
- Ethernet-Interface, Version 5.4
- TCP/IP-Software, Version 2.0
- NetWare-Software, Version 4.1
- NFS-Software, Version 5.1
- Controller für serielle Kommunikation, Version 1.1
- X.25-Software, Version 1.0
- SNMP-Software, Version 3.1

Subsysteme des Konfigurations-Managements speichern diese Informationen in einer Datenbank, damit darauf einfach zugegriffen werden kann. Bei auftretenden Problemen kann diese Datenbank nach Hinweisen zur Problemlösung durchsucht werden.

6.3.3 Accounting-Management

Ziel des *Accounting-Managements* ist es, die Netzwerk-Nutzung zu messen, um so die Belastung des Netzwerks durch einzelne Anwender oder Gruppen entsprechend zu regulieren. Dies führt zu weniger Netzwerk-Problemen (da die Ressourcen entsprechend ihrer Kapazitäten aufgeteilt werden können) und zu angemessener Verfügbarkeit des Netzwerks für alle Benutzer.

Wie beim Performance-Management ist der erste Schritt zu einem entsprechenden Accounting-Management, die Nutzung aller wichtigen Netzwerk-Ressourcen zu messen. Mit der Analyse dieser Ergebnisse erhalten Sie einen Einblick in aktuelle Nutzungsmuster, so daß bereits Nutzungsquoten einge-

stellt werden können. Im Laufe der Zeit werden Korrekturen nötig sein, um einen optimalen Zugriff zu erreichen. Ausgehend von diesen Überlegungen, führen fortgesetzte Ressourcen-Messungen zu Abrechnungsdaten, mit denen außerdem eine faire und optimale Ressourcen-Nutzung zu verwirklichen ist.

6.3.4 Fehler-Management

Ziel des *Fehler-Managements* ist es, Netzwerk-Fehler zu erkennen, zu protokollieren, Benutzer darüber zu unterrichten und (so weit möglich) die Fehler automatisch zu beheben, damit das Netz optimal nutzbar ist. Da Fehler zu Ausfallzeiten oder inakzeptablen Antwortzeiten führen können, gehört das Fehler-Management wahrscheinlich zu den am weitesten verbreiteten Elementen des ISO-Netzwerk-Managements.

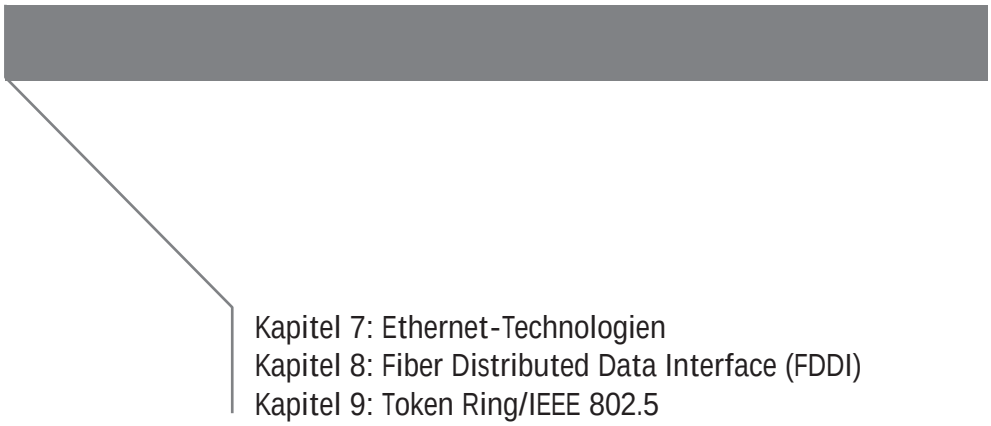
Zum Fehler-Management zählen in erster Linie das Erkennen von Symptomen und die Isolierung von Problemen. Dann kann das Problem behoben und die Lösung auf allen wichtigen Subsystemen getestet werden. Zuletzt müssen der gefundene Fehler und seine Beseitigung aufgezeichnet werden.

6.3.5 Sicherheits-Management

Ziel des *Sicherheits-Managements* ist es, den Zugriff auf Netzwerk-Ressourcen entsprechend den lokalen Regeln zu steuern, so daß das Netzwerk nicht sabotiert werden kann (versehentlich oder absichtlich) und sensitive Daten nicht von Unbefugten einsehbar sind. So kann z.B. mit einem Subsystem des Sicherheits-Managements das Anmelden der Benutzer an eine Netzwerk-Ressource überwacht werden. Und es können die Benutzer abgewiesen werden, die einen falschen Zugriffscode eingeben.

Subsysteme des Sicherheits-Managements dienen der Partitionierung der Netzwerk-Ressourcen in autorisierte und nichtautorisierte Bereiche. Bei bestimmten Anwendern, z.B. firmenfremde Personen, ist der Zugriff auf Netzwerk-Ressourcen unerwünscht. Für andere (interne) Netzwerk-Benutzer sollte es keine Zugriffsmöglichkeit auf Daten einer anderen Abteilung, z.B. der Personalverwaltung, geben.

Die Subsysteme des Sicherheits-Managements haben mehrere Funktionen. Sie legen sensitive Netzwerk-Ressourcen fest (einschließlich Systemen, Dateien und anderen Einrichtungen) und nehmen die Zuordnung von sensitiven Netzwerk-Ressourcen zu Benutzergruppen vor. Des weiteren überwachen sie Zugriffspunkte sensitiver Netzwerk-Ressourcen und protokollieren bereits den Versuch eines unerlaubten Zugriffs.



Kapitel 7: Ethernet-Technologien
Kapitel 8: Fiber Distributed Data Interface (FDDI)
Kapitel 9: Token Ring/IEEE 802.5

TEIL 2

LAN-Protokolle

Teil 2, »LAN-Protokolle«, bietet die Spezifikationen und operationale Informationen zu den heute wichtigsten Bestandteilen von lokalen Netzwerken (LAN – local area networking). In den einzelnen Kapiteln werden folgende Themen besprochen:

Ethernet-Technologien – In diesem Kapitel werden die Eigenschaften, Komponenten und der Einsatz der Ethernet-Technologien beschrieben, einschließlich Ethernet und IEEE 802.3, 100BaseT und 100VG-AnyLAN und Gigabit Ethernet.

Fiber Distributed Data Interface (FDDI) – Dieses Kapitel informiert über die FDDI-Architektur, Spezifikationen, Übertragungsmedien, Geräte, fehlertolerante Funktionen und das Frame-Format.

Token Ring/IEEE 802.5 – Dieses Kapitel nennt die betriebsnotwendigen Komponenten von Token-Ring- und IEEE-802.5-Netzwerken. Den Abschluß bildet eine Zusammenfassung der grundlegenden Netzwerk-Bedienung.

KAPITEL 7

Ethernet-Technologien

7.1 Hintergrund

Mit dem Begriff *Ethernet* wird eine Familie lokaler Netzwerk-Implementationen zusammengefaßt, zu der drei Kategorien gehören:

– Ethernet und IEEE 802.3 – LAN-Spezifikationen für eine Übertragungsgeschwindigkeit von 10 Mbit/s auf Coaxial-Kabel.

– 100-Mbit/s Ethernet – Eine einzelne LAN-Spezifikation, die auch als Fast Ethernet bekannt ist, für eine Übertragungsgeschwindigkeit von 100 Mbit/s auf Twisted-Pair-Kabel.

– 1000-Mbit/s Ethernet – Eine einzelne LAN-Spezifikation, die auch als Gigabit Ethernet bekannt ist, für eine Übertragungsgeschwindigkeit von 1000 Mbit/s (1 Gbps) auf Glasfaser- und Twisted-Pair-Kabel.

Dieses Kapitel bietet einen groben Überblick über jede Technologievariante.

Das Ethernet hat als eine der wichtigsten Medium-Technologien überlebt, weil es äußerst flexibel und relativ leicht zu implementieren und zu verstehen ist. Obwohl andere Technologien als bester Ersatz angepriesen wurden, sind die Netzwerk-Administratoren beim Ethernet und seinen Derivaten geblieben, da sich ihnen hier eine effektive Lösung für eine Vielzahl lokaler Implementationen bot. Die Begrenzungen des Ethernet

wurden immer wieder durch innovative Entwickler (und Standardisierungsgremien) mit ständig erweiterten Ethernet-Pipes überwunden. Auch wenn Kritiker das Ethernet als eine nichtskalierbare Technologie verwerfen, bleibt das zugrundeliegende Übertragungsschema weiterhin eine der Übertragungsmethoden in heutigen lokalen Netzen. Dieses Kapitel beschreibt die verschiedenen Ethernet-Technologien, die bis heute entwickelt wurden.

7.2 Ethernet und IEEE 802.3

Ethernet ist eine Basisband-LAN-Spezifikation, die von der Xerox Corp. erfunden wurde. Ethernet arbeitet mit einer Übertragungsgeschwindigkeit von 10 Mbit/s und verwendet das Fehlerprotokoll CSMA/CD (Carrier Sense Multiple Access/Collision Detect), um auf Coaxial-Kabel zu senden. Ethernet wurde in den 70er Jahren von Xerox entwickelt, wird heute aber oft als Oberbegriff für alle CSMA/CD-LANs verwendet. Ursprünglich wurde das Ethernet für Netzwerke mit sporadischem, gelegentlich hohem Datenverkehr entworfen. Die Spezifikation IEEE 802.3 wurde 1980 entwickelt und basiert auf der originalen Ethernet-Technologie. Die Ethernet Version 2.0 wurde von Digital Equipment Corp., Intel Corp. und Xerox Corp. gemeinsam entwickelt. Sie ist mit IEEE 802.3 kompatibel. Bild 7.1 zeigt ein Ethernet-Netzwerk.

Ethernet und IEEE 802.3 sind normalerweise in einer Schnittstellenkarte oder einer Schaltung der Hauptplatine implementiert. Die Kabelkonvention für Ethernet gibt vor, daß das physische Netzwerk-Medium über einen Transceiver angeschlossen wird. Der Transceiver führt eine Vielzahl der Funktionen der physischen Schicht aus, einschließlich der Kollisionserkennung. Mit einem Transceiver-Kabel werden Stationen an einen Transceiver angeschlossen.

IEEE 802.3 bietet verschiedene Verkabelungen, zu denen die Spezifikation 10Base5 gehört. Diese Spezifikation kommt dem Ethernet am nächsten. Das Anschlußkabel wird als Anschaltschnittstelle (AUI, Attachment Unit Interface) bezeichnet, das Gerät zur Netzwerk-Verbindung als Anschalteinrichtung (MAU, Media Attachment Unit), anstatt eines Transceivers.

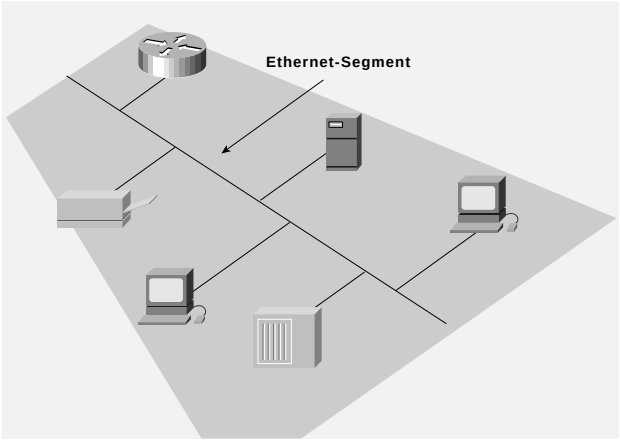


Bild 7.1: Ein Ethernet-Netzwerk betreibt CSMA/CD über Coaxial-Kabel

7.2.1 Ethernet- und IEEE-802.3-Betrieb

In der Broadcast-basierten Umgebung des Ethernet sehen alle Stationen sämtliche Frames, die im Netzwerk übertragen werden. Nach jeder Übertragung muß jede Station jeden Frame überprüfen, ob er für sie bestimmt ist. Falls dies zutrifft, wird der Frame an das Protokoll der nächsthöheren Schicht übergeben.

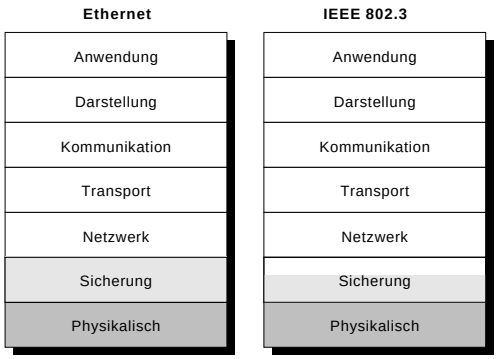
In einem CSMA/CD-LAN kann jede Station jederzeit auf das Netzwerk zugreifen. Bevor Daten gesendet werden, hört eine CSMA/CD-Station das Netz auf Datenverkehr ab. Erst wenn keine Datenübertragung mehr stattfindet, beginnt die Station, ihre Daten zu senden.

Da es sich beim Ethernet um eine kollisionsfreie Umgebung handelt, darf jede Station ihre Daten senden, wenn auf dem Netzwerk »Funkstille« herrscht. Zu einer Kollision kommt es dann, wenn zwei Stationen feststellen, daß der Datenverkehr ruht und nun beide gleichzeitig mit dem Senden beginnen. Dabei werden die Daten beider Stationen zerstört, so daß beide Stationen zu einem späteren Zeitpunkt erneut die Daten senden müssen. Back-off-Algorithmen legen fest, wann mit der erneuten Übertragung der Daten begonnen wird.

7.2.2 Ethernet und IEEE-802.3 – Service-Unterschiede

Obwohl Ethernet und IEEE 802.3 sich in vielerlei Hinsicht sehr ähnlich sind, unterscheiden sich die beiden Spezifikationen bei bestimmten Services. Ethernet bietet Services für die Schichten 1 und 2 des OSI-Referenzmodells, während IEEE 802.3 die physische Schicht (Schicht 1) und den Kanalzugriff der Verbindungsschicht (Schicht 2) spezifiziert. Außerdem definiert IEEE 802.3 kein logisches Verbindungssicherungsprotokoll, es spezifiziert jedoch mehrere verschiedene physische Schichten, wohingegen Ethernet nur eine definiert. Bild 7.2 verdeutlicht die Beziehung von Ethernet und IEEE802.3 zum OSI-Referenzmodell.

Bild 7.2: Ethernet und IEEE im OSI-Referenzmodell



Jedes IEEE-802.3-Protokoll zur physischen Schicht hat einen dreiteiligen Namen, der seine Eigenschaften wiedergibt. Dabei steht je ein Teil für die LAN-Geschwindigkeit, Signalisierungsmethode und den physischen Mediumtyp. Bild 7.3 veranschaulicht diese Namenskonvention.

Bild 7.3: IEEE-802.3-Komponenten sind den Konventionen entsprechend benannt

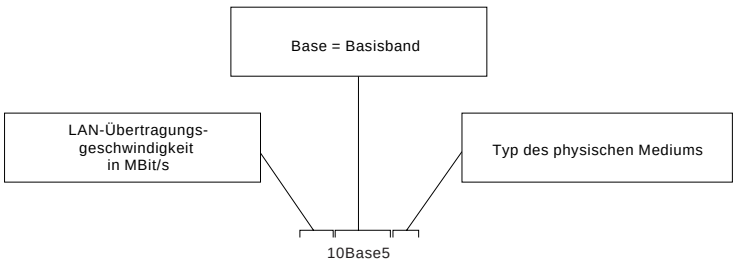


Tabelle 7.1 faßt die Unterschiede zwischen Ethernet und IEEE 802.3 und zwischen den verschiedenen Spezifikationen der physischen Schichten der IEEE 802.3 zusammen.

Eigenschaft	**Ethernet-Wert**	**IEEE 802.3 Werte 10Base5**	**10Base2**	**10BaseT**	**10BaseFL**	**100BaseT**
Datenrate (MBit/s)	10	10	10	10	10	10
Signalisierungsverfahren	Basisband	Basisband	Basisband	Basisband	Basisband	Basisband
Max. Segmentlänge	500	500	185	100	2.000	100
Medium	50ohmiges Coax (thick)	50ohmiges Coax (thick)	50ohmiges Coax (thin)	Ungeschirmtes Twisted Pair-Kabel (UTP)	Optische Faser	Ungeschirmtes Twisted Pair-Kabel (UTP)
Topologie	Bus	Bus	Bus	Stern	Punkt-zu-Punkt	Bus

Tabelle 7.1: Vergleich verschiedener Spezifikationen zur physischen Schicht bei IEEE 802.3

7.2.3 Ethernet- und IEEE-802.3-Frame-Formate

Bild 7.4 zeigt die Frame-Felder in ihrer Zuordnung sowohl in Ethernet- als auch in IEEE-802.3-Frames.

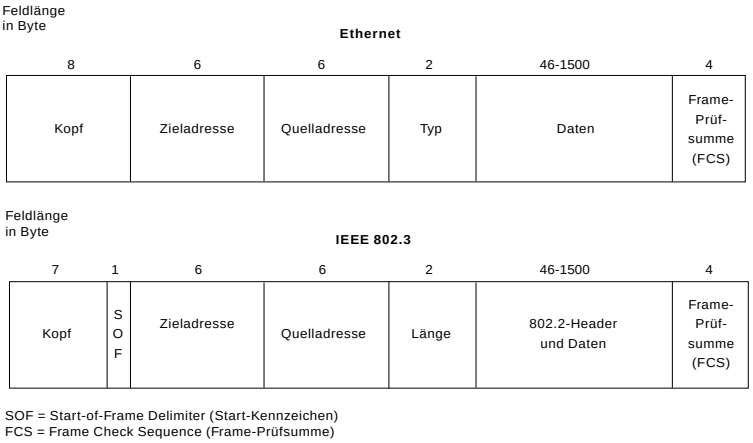


Bild 7.4: Verschiedene Frame-Felder für Ethernet und IEEE 802.3

Die Frame-Felder des Ethernet und IEEE 802.3, welche in Bild 7.4 illustriert sind, werden in der folgenden Übersicht beschrieben:

- *Kopf (Header)* – Die wechselnden Muster von Einsen und Nullen teilen der empfangenden Station mit, daß ein Frame beginnt (Ethernet oder IEEE 802.3). Der Ethernet-Frame enthält ein zusätzliches Byte, das dem Frame-Anfangsfeld (Start of Frame – SOF) des IEEE 802.3 entspricht.
- *Frame-Anfang (Start-of-Frame – SOF)* – Das Trenn-Byte bei IEEE 802.3 endet mit zwei aufeinanderfolgenden Bits mit dem Wert 1, die der Synchronisation aller Stationen im LAN dienen. Der Frame-Anfang ist für das Ethernet explizit spezifiziert.
- *Ziel- und Quelladressen* – Die ersten drei Byte sind vom IEEE herstellerabhängig definiert. Die letzten drei Byte werden vom Ethernet- oder IEEE-802.3-Hersteller spezifiziert. Die Quelladresse ist immer eine Unicast-Adresse (Einzel-Knoten). Die Zieladresse kann eine Unicast- (an einen einzelnen Knoten), Multicast- (an eine Gruppe) oder Broadcast-Adresse (an alle) sein.
- *Typ (Ethernet)* – Der Typ spezifiziert das Protokoll der höheren Schicht, an das die Daten weitergegeben werden, nachdem die Ethernet-Verarbeitung beendet wurde.
- *Länge (IEEE 802.3)* – Die Länge gibt die Anzahl der Daten-Byte an, die diesem Feld folgen.
- *Daten (Ethernet)* – Nach der physischen und der Verbindungsschicht werden die Daten des Frame an ein Protokoll der nächsthöheren Schicht weitergegeben, das im Typ-Feld bezeichnet wird. Obwohl Ethernet Version 2 keine Muster vorgibt (im Gegensatz zu IEEE 802.3), erwartet es doch mindestens 46 Byte mit Daten.
- *Daten (IEEE 802.3)* – Nach der physischen und der Verbindungsschicht werden die Daten des Frame an ein Protokoll der nächsthöheren Schicht weitergegeben, das innerhalb der Frame-Daten definiert ist. Falls weniger als 64 Daten-Byte vorliegen, wird der Frame auf 64 Byte aufgefüllt.
- *Frame-Prüfsumme (Frame Check Sequence [FCS])* – Diese Byte-Folge enthält einen 4 Byte langen CRC-Wert (Cyclic

Redundancy Check), der vom sendenden Gerät erzeugt und vom empfangenden Gerät überprüft wird, um zerstörte Frames zu erkennen.

7.3 100-Mbit/s Ethernet

100-MBit/s Ethernet ist eine sehr schnelle LAN-Technologie, die sowohl dem Desktop-Benutzer im Netz als auch den Servern und Server-Clustern (zuweilen als Server-Farmen bezeichnet) in den Daten-Centern eine größere Bandbreite bietet.

Die Higher Speed Ethernet Study Group des IEEE sollte untersuchen, ob es möglich ist, ein Ethernet mit 100 Mbit/s zu betreiben. Zwar formulierte die Studiengruppe mehrere Ziele für dieses Hochgeschwindigkeits-Ethernet, konnte sich aber bei der Zugriffsmethode nicht einigen. Zur Debatte stand, ob dieses neue schnellere Ethernet CSMA/CD den Zugriff auf das Netzwerk-Medium unterstützen oder eine andere Methode dafür gewählt werden sollte. Über diesem Streit teilte sich die Studiengruppe in zwei Lager: in die Fast-Ethernet-Allianz und das 100VG-AnyLAN-Forum. Von jeder Gruppe wurde eine Spezifikation für den Betrieb eines schnellen Ethernet (bzw. Token Ring für das letztere) erstellt: 100BaseT und 100VG-AnyLAN.

100BaseT ist die IEEE-Spezifikation für die 100-Mbit/s-Ethernet-Implementation mit UTP- (unshielded twisted-pair – ungeschirmte Twisted-Pair-Kabel) und STP-Verkabelung (shielded twisted-pair). Die Schicht Media Access Control (MAC) ist zur IEEE-802.3-MAC-Schicht kompatibel. Grand Junction, heute ein Unternehmen der Cisco Systems Workgroup Business Unit (WBU), entwickelte das Fast Ethernet, das vom IEEE mit der Spezifikation 802.3 zum Standard erklärt wurde.

100VG-AnyLAN ist die IEEE-Spezifikation für 100-Mbit/s-Implementationen von Token Ring und Ethernet mit vieradrigem UTP. In diesem Fall ist die MAC-Schicht *nicht* kompatibel zur IEEE-802.3-MAC-Schicht. 100VG-AnyLAN wurde von Hewlett-Packard (HP) entwickelt, um neuere, zeitkritische Anwendungen wie Multimedia zu unterstützen. Eine Version der HP-Implementationen wurde vom IEEE mit der Spezifikation 802.12 zum Standard erklärt.

7.3.1 100BaseT im Überblick

100BaseT verwendet die bestehende Spezifikation IEEE 802.3 CSMA/CD. Als ein Ergebnis behält 100BaseT das IEEE-802.3 Frame-Format, die Frame-Größe und die Fehlererkennung bei. Zusätzlich unterstützt es alle Anwendungen und Netzwerk-Software, die gegenwärtig in 802.3-Netzwerken läuft. Mit 100BaseT Fast Link Pulses (FLPs) unterstützt 100BaseT sowohl 10 als auch 100 Mbit/s. 100BaseT-Hubs müssen die zwei Übertragungsgeschwindigkeiten wie Token Ring 4/16-Hubs erkennen. Netzwerk-Karten unterstützen 10 Mbit/s, 100 Mbit/s oder beide Geschwindigkeiten. Bild 7.5 zeigt, wie die 802.3-MAC-Subschicht und höhere Schichten unverändert mit 100BaseT betrieben werden können.

Bild 7.5: 802.3 MAC und Protokolle höherer Schichten werden über 100BaseT betrieben

7.3.2 100BaseT-Signalisierung

100BaseT unterstützt zwei Signalisierungsarten:

– 100BaseX

– 4T+

Beide Verfahren sind auf Ebene der Stationen und Hubs interoperabel. Das Media Independend Interface (MII) – eine AUI-ähnliche Schnittstelle – bietet auf Stationsebene lnteroperabilität. Auf Hub-Ebene gilt gleiches.

Das Signalisierungsschema von 100BaseX verfügt über eine Konvergenz-Subschicht. Diese Schicht setzt die kontinuierliche Voll-Duplex-Signalisierung auf der FDDI-Schicht (physical medium dependent – PMD) in die Halb-Duplex-Signalisierung (mit Start-Stop) für die Subschicht des Ethernet Media Access

Control (MAC) um. Da 100BaseTX die vorhandene FDDI-Spezifikation verwendet, kamen die Produkte schnell auf den Markt. 100BaseX ist die Signalisierung, die von den Mediumtypen 100BaseTX und 100BaseFX eingesetzt wird. Bild 7.6 zeigt, wie die Konvergenz-Subschicht des 100BaseX die beiden Signalisierungen umsetzt.

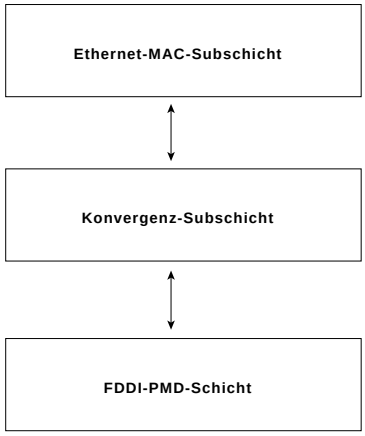


Bild 7.6: Die 100BaseX-Konvergenz-Subschicht ist die Schnittstelle zweier Signalisierungsverfahren

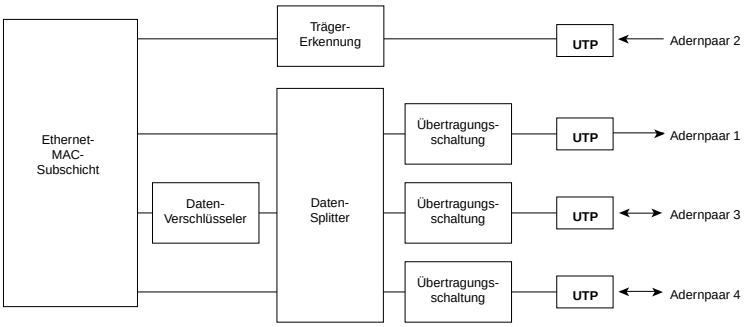
Das Signalisierungsverfahren 4T+ nutzt ein Drahtpaar für die Kollisionserkennung und die anderen drei Drahtpaare für die Datenübertragung. Somit kann 100BaseT auf einer Verkabelung der Kategorie 3 betrieben werden, wenn alle vier Drahtpaare am Desktop angeschlossen sind. 4T+ ist das Signalisierungsverfahren, das für 100BaseT4-Mediumtypen verwendet wird und nur den Halb-Duplex-Betrieb unterstützt. Bild 7.7 zeigt, wie die 4T+-Signalisierung alle vier Drahtpaare des UTP beansprucht.

7.3.3 100BaseT-Hardware

Folgende Komponenten gehören zu einer physischen 100BaseT-Verbindung:

- *Physisches Medium* – Diese Komponente überträgt die Signale zu den Computern und kann einer der drei 100BaseT-Mediumtypen sein:
  - 100BaseTX
  - 100BaseFX
  - 100BaseT4

Bild 7.7: 4T+ benötigt vier UTP-Adernpaare



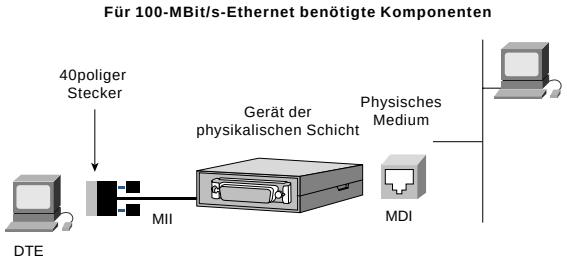
– *Medium-Dependent Interface (MDI)* – Das MDI ist die mechanische und elektrische Schnittstelle zwischen dem Übertragungsmedium und dem Gerät der physischen Schicht (PHY).

– *Gerät der physischen Schicht (PHY)* – Das PHY kann mit 10 Mbit/s oder 100 Mbit/s betrieben werden und besteht aus integrierten Schaltungen (ICs) – oder einer eigenen Karte – mit einer Ethernet-Schnittstelle. Oder es handelt sich um ein externes Gerät mit einem MII-Kabel (Medium Independent Interface), das an die MII-Schnittstelle eines 100BaseT-Geräts (ähnlich einem 10-Mbit/s-Ethernet-Transceiver) angeschlossen wird.

– *Medium Independent Interface (MII)* – Das MII wird zusammen mit externen 100-Mbit/s-Transceivern eingesetzt, um ein 100-Mbit/s-Ethernet-Gerät an einen der drei Mediumtypen anzuschließen. Das MII ist mit einem 40poligen Stecker ausgestattet und bis zu 0,5 Meter lang.

Bild 7.8 zeigt die 100BaseT-Hardware-Komponenten.

Bild 7.8: 100BaseT benötigt mehrere Hardware-Komponenten



7.3.4 100BaseT-Betrieb

100BaseT und 10BaseT setzen beide die gleichen Verfahren nach IEEE 802.3 für den MAC-Zugriff und die Fehlererkennung ein. Beide haben das gleiche Frame-Format und die gleichen Längenanforderungen. Der eigentliche Unterschied zwischen 100BaseT und 10BaseT (außer dem offensichtlichen Geschwindigkeitsunterschied) ist der Netzwerk-Durchmesser. Für 100BaseT liegt das Maximum dafür bei 205 Meter, was ca. um den Faktor 10 kleiner ist als beim 10-Mbit/s Ethernet.

Der Grund für die Verkleinerung des Netzwerk-Durchmessers bei 100BaseT liegt im Verfahren der Fehlererkennung, welches das gleiche ist wie bei 10BaseT. Die maximale Entfernung wurde für 10BaseT so definiert, daß eine Station noch beim Übertragen des kürzesten zulässigen Frame (64 Byte) die Kollision mit einer sendenden Station am entferntesten Punkt der Domäne erkennt.

Um den erhöhten Durchsatz von 100BaseT zu erreichen, mußte die Größe der Kollisionsdomäne verkleinert werden. Dies ist erforderlich, da die Ausbreitungsgeschwindigkeit des Mediums sich nicht geändert hat. Daraus folgt, daß bei zehnmal schnellerer Übertragung die maximale Entfernung nur ein Zehntel betragen kann. So erkennt jede Station bereits beim Senden der ersten 64 Byte, ob es zu einer Kollision mit einer anderen Station gekommen ist.

100BaseT Fast-Link Pulses (FLPs)

Die von 100BaseT verwendeten Signale werden als Fast-Link Pulses (FLPs) bezeichnet und dienen der Verbindungsintegrität zwischen einem Hub und 100BaseT-Gerät. FLPs sind abwärtskompatibel mit den 10BaseT Normal-Link Pulses (NLPs). Die FLPs beinhalten mehr Informationen als NLPs und werden beim automatischen Abstimmungsprozeß zwischen Hub und einem Gerät im 100BaseT-Netzwerk verwendet.

100BaseT Autonegotiation-Option

100BaseT-Netzwerke unterstützen eine optionale Funktion, die automatische Abstimmung, mit der ein Gerät und ein Hub in der Lage sind, Daten über ihre Eigenschaften auszutauschen (mit 100BaseT FLPs), wodurch eine optimale Kommunikationsumgebung geschaffen wird.

Die automatische Abstimmung unterstützt eine Vielzahl von Eigenschaften, einschließlich der Geschwindigkeitsabstimmung für Geräte, die im 10- und 100-Mbit/s-Betrieb zum Einsatz kommen. Außerdem wird der Voll-Duplex-Betrieb für entsprechende Geräte und die automatische Signalisierungskonfiguration für 100BaseT4- und 100BaseTX-Stationen unterstützt.

7.3.5 100BaseT-Mediumtypen

100BaseT unterstützt für die physische OSI-Schicht (Schicht 1) drei Mediumtypen: 100BaseTX, 100BaseFX und 100BaseT4. Diese drei Mediumtypen, die alle mit der MAC-Schicht der IEEE 802.3 kommunizieren können, sind in Bild 7.9 dargestellt. Tabelle 7.2 vergleicht die Schlüsseleigenschaften der drei 100BaseT-Mediumtypen.

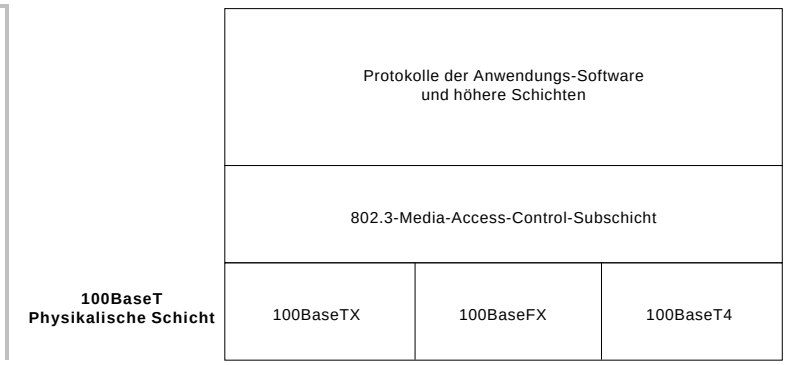


Bild 7.9: Drei 100BaseT-Mediumtypen stehen in der physischen Schicht zur Verfügung

Eigenschaften	100BaseTX	100BaseFX	100BaseT4
Kabel	Kategorie 5 UTP oder Typ 1 und 2 STP	62.5/125 Micron-Multi-Modus-Faser	Kategorie 3, 4 oder 5 UTP
Anzahl der Adern bzw. Fasern	2adrig	2faserig	4adrig
Stecker	ISO 8877 (RJ-45)	Duplex Scmedia-interface connector (MIC) ST	ISO 8877 (RJ-45)
Max. Segmentlänge	100 m	400 m	100 m
Max. Netzwerk-Durchmesser	200 m	400 m	200 m

Tabelle 7.2: Eigenschaften der 100BaseT-Mediumtypen

100BaseTX

100BaseTX basiert auf der Spezifikation Twisted Pair-Physical Medium Dependent (TP-PMD) des American National Standards Institute (ANSI). Diese Spezifikation unterstützt ungeschirmte Twisted-Pair-Kabel (unshielded twisted-pair, UTP) und geschirmte Twisted-Pair-Kabel (shielded twisted-pair, STP). 100BaseTX verwendet das Signalisierungsverfahren von 100BaseX auf 2adrigen Kabeln der Kategorie 5 UTP oder STP.

Die IEEE-802.3u-Spezifikation für 100BaseTX-Netze läßt maximal zwei Repeater-(Hubs-)Netzwerke und einen gesamten Netzwerk-Durchmesser von ca. 200 Metern zu. Ein Link-Segment, das durch die Punkt-zu-Punkt-Verbindung zweier MII-Geräte (Medium Independent Interface) definiert ist, kann einen Durchmesser von bis zu 100 Metern haben. Bild 7.10 illustriert diese Konfigurationsrichtlinien.

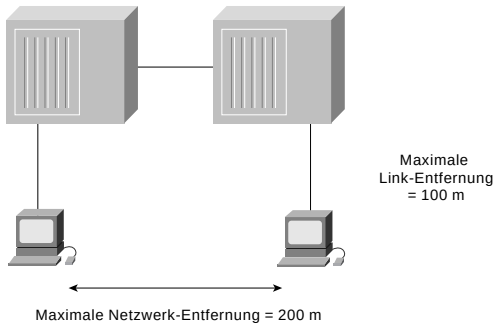
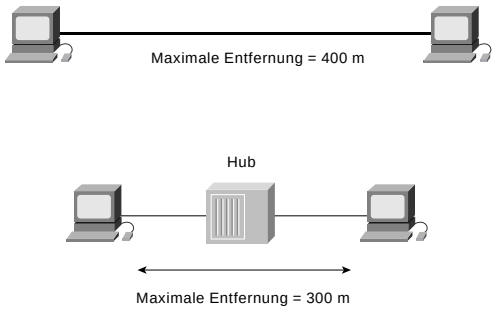


Bild 7.10: 100BaseTX ist auf eine Verbindungslänge von 100 Meter beschränkt

100BaseFX

100BaseFX basiert auf der Spezifikation ANSI TP PMD (Twisted Pair-Physical Medium Dependent) X3T9.5 für FDDI LANs. 100BaseFX verwendet das Signalisierungsverfahren 100BaseX auf einem 2faserigen Multi-Modus-Glasfaser-Kabel (MMF). Die Spezifikation IEEE 802.3u für 100BaseFX-Netze ermöglicht DTE-zu-DTE-Verbindungen (Data Termial Equipment – Datenendeinrichtung, DDE) von bis zu 400 Metern oder ein Repeater-Netzwerk von ca. 300 Metern Länge. Bild 7.11 illustriert diese Konfigurationsrichtlinien.

Bild 7.11: 100BaseFX ist auf eine DTE-zu-DTE-Verbindungslänge von 400 Meter beschränkt



100BaseT4

100BaseT4 erlaubt den Betrieb von 100BaseT auf einer Verkabelung der Kategorie 3, wenn alle vier Kabelpaare an den Desktop angeschlossen sind. 100BaseT4 verwendet die Signalisierung Halb-Duplex 4T+. Die Spezifikation 802.3u für 100BaseT4-Netzwerke läßt maximal zwei Repeater-Netzwerke (Hubs) und einen Gesamtdurchmesser von ca. 200 Metern zu. Ein Link-Segment, das durch eine Punkt-zu-Punkt-Verbindung von zwei MII-Geräten definiert ist, kann bis zu 100 Meter weit reichen. Bild 7.12 illustriert diese Konfigurationsrichtlinien.

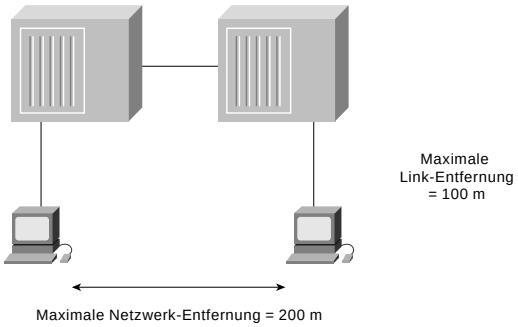


Bild 7.12: 100BaseT4 ist auf eine Verbindungslänge von 100 Meter beschränkt

7.4 100VG-AnyLAN

100VG-AnyLAN wurde von Hewlett-Packard (HP) als Alternative zu CSMA/CD für neuere, zeitkritische Anwendungen, z.B. Multimedia, entwickelt. Die Zugriffsmethode basiert auf Stationsanfrage und wurde als Ausbau von Ethernet und 16-Mbit/s Token Ring entworfen. 100VG-AnyLAN unterstützt folgende Kabeltypen:

- vieradrig, Kategorie 3, ungeschirmtes Twisted-Pair-Kabel (UTP)
- zweiadrig, Kategorie 4 oder 5, UTP
- geschirmtes Twisted-Pair-Kabel (STP)
- Glasfaser

Der Standard IEEE 802.12 100VG-AnyLAN spezifiziert die maximale Link-Länge, Hub-Konfiguration und maximale Netzwerk-Länge. Die Link-Länge zwischen Knoten und Hub beträgt 100 Meter (Kategorie 3, UTP) oder 150 Meter (Kategorie 5, UTP). Bild 7.13 illustriert die maximalen Link-Längen für 100VG-AnyLAN.

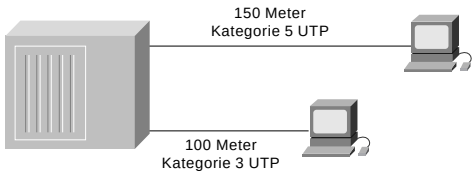


Bild 7.13: 100VG-AnyLAN ist auf verschiedene Längen für Kategorie 3 und 5 UTP beschränkt

100VG-AnyLAN-Hubs sind hierarchisch angeordnet. Jeder Hub hat mindestens einen aufwärtsgerichteten Port (uplink), während alle anderen Ports abwärtsgerichtet (downlink) sein

können. Hubs können dreistufig kaskadiert werden, wenn sie eine Aufwärtsverbindung zu anderen Hubs haben. Die Kabellänge zwischen kaskadierten Hubs kann maximal 100 Meter (Kategorie 3, UTP) bzw. 150 Meter (Kategorie 5, UTP) betragen. Bild 7.14 zeigt die 100VG-AnyLAN-Hub-Konfiguration.

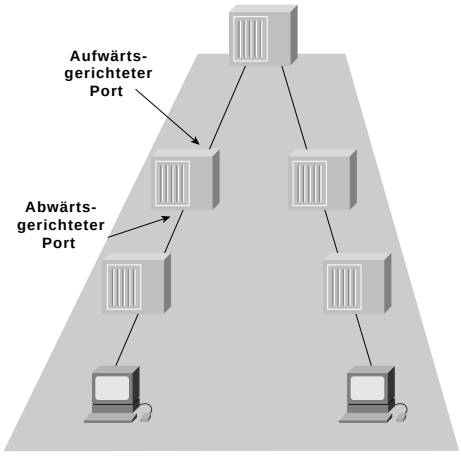


Bild 7.14: 100VG-AnyLAN-Hubs sind hierarchisch angeordnet

Die Entfernung der Enden im Netz ist auf 600 Meter (Kategorie 3, UTP) bzw. 900 Meter (Kategorie 5, UTP) beschränkt. Wenn Hubs im gleichen Verteilerraum stehen, sinkt die maximale Entfernung auf 200 Meter (Kategorie 3, UTP) bzw. auf 300 Meter (Kategorie 5, UTP). Bild 7.15 zeigt die maximalen Entfernungen in 100VG-AnyLAN-Netzen.

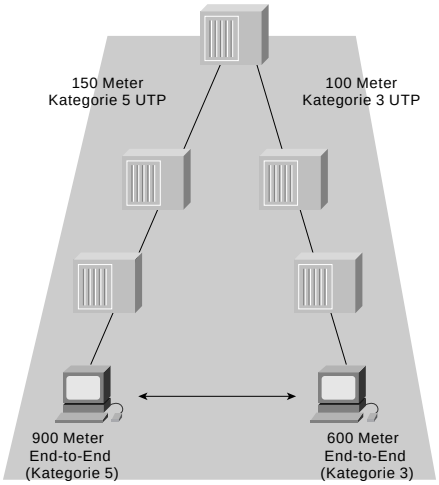


Bild 7.15: Entfernungsbeschränkungen für Ende-zu-Ende-Verbindungen im 100VG-AnyLAN hängen von der Implementation ab

7.4.1 100VG-AnyLAN-Betrieb

100VG-AnyLAN verwendet die Demand-Priority Access Method, die Kollisionen vermeidet und höher belastet werden kann als 100BaseT. Diese Methode ist stärker deterministisch als CSMA/CD, da der Hub den Zugriff auf das Netzwerk steuert.

Der 100VG-AnyLAN-Standard erfordert einen Level-One-Hub oder Repeater, der als Root arbeitet. Dieser Root-Repeater steuert den Betrieb der Priority Domain. Hubs können in einer Sternverkabelung dreistufig kaskadiert werden. Miteinander verbundene Hubs arbeiten wie ein einzelner großer Repeater, wobei der Root Repeater die Ports nacheinander abfragt.

In einem 100VG-AnyLAN-Netz mit Demand-Priority-Betrieb signalisiert ein Knoten, der Daten übertragen will, seine Anforderung dem Hub (oder Switch). Wenn auf dem Netz keine Übertragung stattfindet, quittiert der Hub sofort die Anforderung, und der Knoten überträgt ein Paket an den Hub. Falls mehrere Anforderungen gleichzeitg beim Hub eingehen, verwendet der Hub eine Round-Robin-Technik, um jede Anforderung zu quittieren. Anforderungen von hoher Priorität, z.B. von zeitkritischen Video-Konferenz-Programmen, werden als erste bedient. Um für alle Stationen eine Zugangsmöglichkeit offenzuhalten, läßt ein Hub den priorisierten Zugriff für einen Port nur zweimal hintereinander zu.

7.5 Gigabit Ethernet

Gigabit Ethernet ist eine Erweiterung des IEEE-802.3-Ethernet-Standards. Gigabit Ethernet hat eine Bandbreite von 1000 Mbit/s, wobei die Kompatibilität zu Ethernet- und Fast-Ethernet-Geräten gewahrt bleibt. Gigabit Ethernet bietet Voll-Duplex-Betrieb für die Verbindung Switch-zu-Switch und Switch-zu-Endgerät. Halb-Duplex-Betrieb ist auf gemeinsam nutzbaren Verbindungen möglich, wenn Repeater und CSMA/CD eingesetzt werden. Des weiteren verwendet Gigabit Ethernet das gleiche Frame-Format, die gleiche Frame-Größe und die gleichen Management-Objekte wie vorhandene IEEE-802.4-Netze. Am ehesten ist Gigabit Ethernet für Glasfaser-

Verkabelung gedacht, es kann aber auch mit Kategorie 5, UTP und Coaxial-Kabel betrieben werden.

Die Gigabit-Ethernet-Allianz ist eine offene Vereinigung mehrerer Hersteller, die Firmen bei der Entwicklung von Gigabit Ethernet unterstützt. Von dieser Allianz werden Aktivitäten zur Standardisierung von Gigabit Ethernet unterstützt, die von der Arbeitsgruppe IEEE 802.3 geleitet werden. Außerdem stellt sie technische Ressourcen zur Verfügung, um Konvergenz und Konsens bei der technischen Spezifikation zu erreichen. Ferner werden Ressourcen bereitgestellt, um die Produkt-Interoperabilität einzuführen und zu demonstrieren und die Kommunikation zwischen den potentiellen Anbietern und Käufern von Gigabit-Ethernet-Produkten zu fördern.

Die IEEE-802.3-Arbeitsgruppe hat eine Projektgruppe für das 802.3z Gigabit Ethernet gebildet, die einen Standard entwikkeln soll, der eine Reihe von Anforderungen festschreibt. Dieser Standard muß Voll- und Halb-Duplex-Betrieb bei 1000 Mbit/s erlauben. Entsprechende Implementationen verwenden das Standard IEEE 802.3/Ethernet-Frame-Format und die Medium-Zugriffsmethode CSMA/CD. Gigabit Ethernet-Implementationen werden ebenfalls abwärtskompatibel zu 10BaseT und 100BaseT sein. Des weiteren soll der IEEE-Standard die Unterstützung für Multi-Modus-Glasfaser-Verbindung mit einer maximalen Länge von 500 Metern festlegen; für Single-Modus-Glasfaser-Verbindung mit einer maximalen Länge von 2 km und bei einer Kupferverkabelung mit maximal 25 Metern. Der Gigabit-Ethernet-Standard wird ein Anhang zum bereits vorhandenen Standard 802.3 Ethernet/Fast Ethernet.

7.5.1 Gigabit-Ethernet-Spezifikation

Die gegenwärtigen Bemühungen um einen Standard setzen auf Fibre Channel und andere Netzwerk-Komponenten mit hohem Datendurchsatz. Die Gigabit-Ethernet-Implementationen sind mit schnellen optischen Komponenten für 780 nm (kurzwellige) Fibre Channels ausgestattet, die die optische Datenübertragung übernehmen. 8B/10B-Ver- und Entschlüsselungsverfahren werden für die Serialisierung und Deserialisierung eingesetzt. Für größere Entfernungen werden optische Komponenten für 1300 nm (langwellig) spezifiziert. Um für die Entwicklung bei Halbleitern und der digitalen Signalver-

arbeitung gewappnet zu sein, wird eine logische, medium-unabhängige Schnittstelle für die Schichten MAC und PHY spezifiziert, mit der das Gigabit Ethernet auf ungeschirmter Twisted-Pair-Verkabelung (UTP) betrieben werden kann. Diese logische Schnittstelle wird dazu führen, daß Kodierungen, die für UTP-Verkabelungen geeignet sind, unabhängig von der Fibre-Channel-Kodierung implementiert werden. Bild 7.16 illustriert die funktionalen Elemente des Gigabit Ethernet.

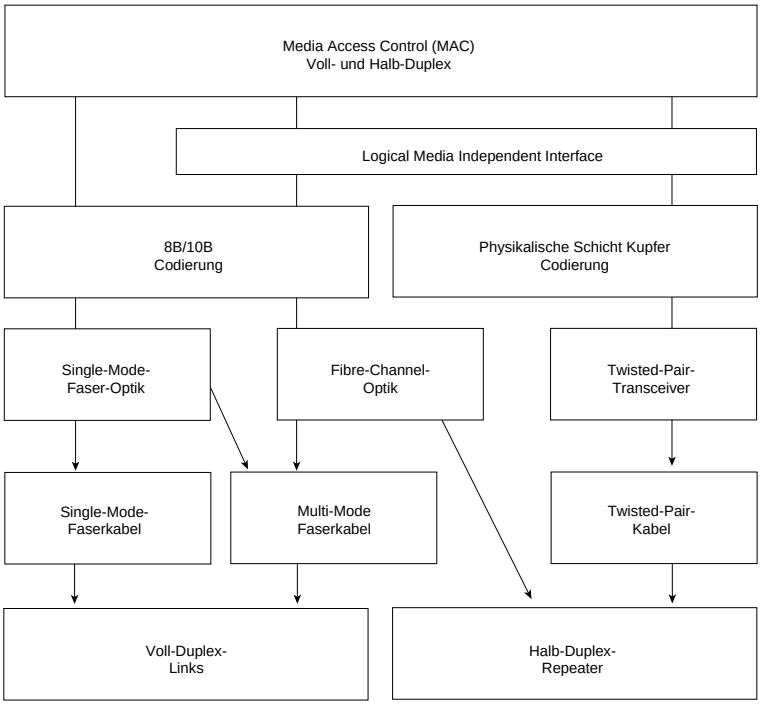


Bild 7.16: Funktionale Elemente des Gigabit Ethernet

7.5.2 Migrieren zum Gigabit Ethernet

Die Migration zum Gigabit Ethernet wird in kleinen Schritten erfolgen, wobei mit den Backbones der vorhandenen Ethernet-LANs begonnen wird. Dann werden die Verbindungen zwischen den Servern umgestellt, und eventuell werden auch die Desktops auf diesen Stand gebracht. Zu den voraussichtlichen Aktionen einer Implementation gehören folgende:

– Ausbauen von Switch-zu-Switch-Verbindungen – 100-Mbit/s-Verbindungen zwischen Fast-Ethernet-Switches oder Repeatern können durch 1000-Mbit/s-Verbindungen

ersetzt werden. Dadurch wird die Kommunikation zwischen Backbone-Switches beschleunigt, und die Switches sind in der Lage, eine größere Anzahl geswitchter und gemeinsamer Fast-Ethernet-Segmente zu unterstützen.

– Ausbauen von Switch-zu-Server-Verbindungen – 1000-Mbit/s-Anbindungen können zwischen Switches und Hochleistungs-Servern implementiert werden. Bei dieser Aktion müssen die Server mit Gigabit-Ethernet-NICs ausgestattet werden.

– Ausbauen eines Fast-Ethernet-Backbone – Ein Fast-Ethernet-Backbone-Switch, an dem 10/100-Mbit/s-Switches angeschlossen sind, kann zu einem Gigabit-Ethernet-Switch ausgebaut werden, der sowohl mehrere 100/1000-Mbit/s-Switches als auch Router und Hubs mit Gigabit-Ethernet-Schnittstelle und Gigabit-Repeater unterstützt.

Damit ist es möglich, die Server mit den Gigabit-Ethernet-NICs direkt an den Backbone anzuschließen, so daß für Benutzer mit sehr bandbreitenintensiven Anwendungen der Durchsatz zum Server vergrößert wird. Ein Gigabit-Ethernet-Netzwerk kann eine größere Anzahl von Segmenten, eine größere Bandbreite je Segment und somit eine größere Anzahl Knoten je Segment unterstützen.

– Ausbauen eines gemeinsam genutzten FDDI-Backbone – Ein FDDI-Backbone kann ausgebaut werden, indem der FDDI-Konzentrator, der Hub oder Ethernet-zu-FDDI-Router durch einen Gigabit-Ethernet-Switch oder -Repeater ersetzt wird. Ansonsten müssen nur noch die neuen Gigabit-Ethernet-Schnittstellen in den Routern, Switches oder Repeatern installiert werden.

– Ausbauen von Hochleistungs-Desktops – Mit Gigabit-Ethernet-NICs können Hochleistungs-Desktop-Computer für das Gigabit Ethernet aufgerüstet werden. Diese Computer werden an Gigabit-Ethernet-Switches oder -Repeater angeschlossen.

KAPITEL 8

Fiber Distributed Data Interface (FDDI)

8.1 Hintergrund

Das Fiber Distributed Data Interface (FDDI) spezifiziert ein 100-Mbps-Doppelring-LAN, das mit Token arbeitet und als Medium eine optische Faser verwendet. FDDI wird am häufigsten in Hochgeschwindigkeits-Backbones eingesetzt, da es eine größere Bandbreite und weitere Entfernungen erlaubt als das Kupferkabel. Es muß allerdings darauf hingewiesen werden, daß es eine relativ neue entsprechende Spezifikation für Kupferkabel gibt, die als Copper Distributed Data Interface (CDDI) bezeichnet wird und auf diesem Medium eine Übertragungsrate von 100 Mbps ermöglicht. CDDI ist die Implementation der FDDI-Protokolle für Twisted-Pair-Kupferkabel. In diesem Kapitel liegt der Schwerpunkt auf der FDDI-Spezifikation, wobei CDDI auch immer wieder zur Sprache kommt.

FDDI verwendet die Doppelring-Architektur mit Datenverkehr auf beiden Ringen in entgegengesetzter Richtung (auch als *gegenläufig* bezeichnet). Der Doppelring besteht aus dem primären und dem sekundären Ring. Im normalen Betrieb dient der primäre Ring der Datenübertragung, während auf dem sekundäre Ring der Datenverkehr ruht. Der eigentliche Zweck des Doppelrings ist eine hohe Zuverlässigkeit und Robustheit des Netzes. Bild 8.1 zeigt die gegenläufigen primären und sekundären FDDI-Ringe.

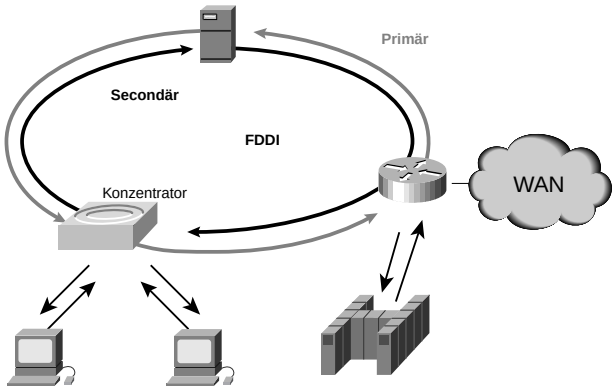


Bild 8.1: FDDI setzt gegenläufige primäre und sekundäre Ringe ein

8.1.1 Standards

FDDI wurde Mitte der 80er Jahre vom X3T9.5-Standardkomitee des American National Standards Institute (ANSI) entwickelt. Zu dieser Zeit stellten schnelle Engineering-Workstations die Bandbreite der vorhandenen Ethernet- oder Token-Ring-LANs hart auf die Probe. Es wurde ein neues LAN-Medium erforderlich, das diese Workstation und ihre neuen verteilten Anwendungen verkraftete. Gleichzeitig wuchs die Bedeutung der Netzwerk-Zuverlässigkeit, da unternehmenswichtige Anwendungen von Großrechnern auf Netzwerke migriert wurden. Um diese Anforderungen zu erfüllen, wurde FDDI entwickelt. Nachdem die FDDI-Spezifikation fertiggestellt war, wurde sie vom ANSI an die International Organization for Standardization (ISO) weitergeleitet. Dort wurde eine internationale Version von FDDI erstellt, die voll zur ANSI-Version kompatibel ist.

8.2 FDDI-Übertragungsmedium

Bei FDDI wird eine optische Faser (Glasfaser) als primäres Übertragungsmedium verwendet. Es ist jedoch auch möglich, FDDI auf Kupferkabeln zu betreiben. Dies wird als *Copper-Distributed Data Interface* (CDDI) bezeichnet. Die optische Faser hat viele Vorteile gegenüber dem Kupfer. Im einzelnen werden die Sicherheit, Zuverlässigkeit und der Durchsatz bei der optischen Faser verbessert, da keine elektrischen Signale

emittiert werden. Ein Medium, das elektrische Signale emittiert, kann abgehört werden und erlaubt somit unbefugten Zugriff auf die übertragenen Daten. Außerdem kann die Übertragung in der Faser nicht durch hochfrequente (RFI) oder elektromagnetische Interferenzen (EMI) gestört werden. Bisher konnte die Faser eine wesentlich höhere Bandbreite (und damit Datendurchsatz) aufweisen als das Kupferkabel, jedoch erlauben die neuesten Entwicklungen auch auf dem Kupferkabel eine Übertragung von 100 Mbps. Werden Multi-Mode-Fasern verwendet, kann mit FDDI eine Entfernung von zwei Kilometern zwischen Stationen liegen, beim Einsatz von Single-Mode-Fasern können wesentlich weitere Distanzen überbrückt werden.

FDDI definiert zwei Arten optischer Fasern: Single-Mode und Multi-Mode. Ein *Mode* ist eine Lichtwelle, die in die Faser unter einem bestimmten Winkel eintritt. Bei Multi-Mode-Fasern dienen LEDs als Lichtquelle, bei Single-Mode-Fasern wird im allgemeinen ein Laser verwendet.

In Multi-Mode-Fasern werden mehrere Modes (Lichtwellen) gleichzeitig übertragen. Da diese Wellen in verschiedenen Winkeln in die Faser eingestrahlt werden, treten sie am Ende der Faser zu unterschiedlichen Zeitpunkten wieder aus. Diese Eigenschaft wird als *modale Dispersion* bezeichnet. Die modale Dispersion begrenzt die Bandbreite und die Entfernung, die mit Multi-Mode-Fasern erreicht werden können. Daher wird diese Technik zumeist für die Gebäudeverkabelung oder über kurze Distanzen verwendet.

In Single-Mode-Fasern kann immer nur eine Welle übertragen werden, so daß hier die modale Dispersion keine Rolle spielt. Dies führt zu höherem Durchsatz und der Überwindung größerer Entfernungen. Darum wird diese Technik für die Verbindung zwischen weit auseinander liegenden Gebäuden verwendet.

Bild 8.2 zeigt eine Single-Mode-Faser mit einem Laser als Lichtquelle und eine Multi-Mode-Faser mit Leuchtdioden (LED) als Lichtquelle.

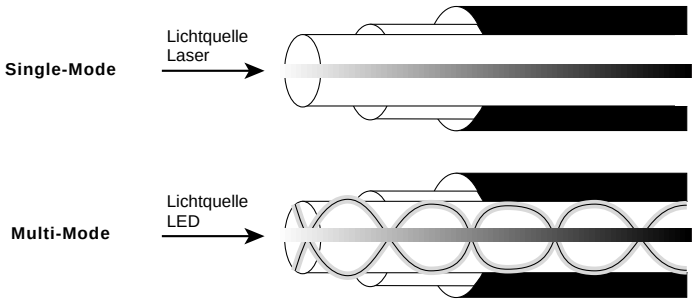


Bild 8.2: Verschiedene Lichtquellen für Single-Mode- und Multi-Mode-Fasern

8.3 FDDI-Spezifikationen

FDDI spezifiziert den physischen Teil und den des Medium-Zugriffs des OSI-Referenzmodells. FDDI ist tatsächlich keine einzelne Spezifikation, sondern setzt sich aus vier einzelnen Spezifikationen mit bestimmten Funktionen zusammen. Zusammen bieten diese Spezifikationen die Hochgeschwindigkeits-Verbindung zwischen den Protokollen der höheren Schichten, z.B. TCP/IP und IPX, und dem Medium, hier die optische Faser.

Bei diesen vier FDDI-Spezifikationen handelt es sich um die Media Access Control (MAC), das Protokoll der physischen Schicht (PHY), das Physical-Medium Dependent (PMD) und das Station Management (SMT). Die MAC-Spezifikation definiert, wie auf das Medium zugegriffen wird, einschließlich Frame-Format, Token-Handling, Adressierung, Algorithmen für die Berechnung der CRC-Werte (cyclic redundancy check) und der Fehlerbeseitigungsmechanismen. Die PHY-Spezifikation definiert z.B. die Kodierungsverfahren, die Taktung und das Framing. Die PMD-Spezifikation definiert die Eigenschaften des Übertragungsmediums, einschließlich Faser-Links, Leistungsstufen, Bit-Fehlerraten, optische Komponenten und Anschlüsse. Die SMT-Spezifikation definiert die FDDI-Stations-Konfiguration, die Ring-Konfiguration und die Ring-Steuerfunktionen, einschließlich Stationseinfügung und -entfernung, Initialisierung, Fehlereingrenzung und -behebung, Planung und statistische Erhebung.

Hinsichtlich des OSI-Modells sind sich FDDI, IEEE 802.3 Ethernet und IEEE 802.5 Token Ring ähnlich. FDDI dient der Verbindung der oberen OSI-Schichten mit ihren gängigen Protokollen und dem Medium, das die Geräte vernetzt. Bild 8.3 zeigt die vier FDDI-Spezifikationen in ihren gegenseitigen Beziehungen und der Beziehung zur IEEE-definierten Subschicht Logical-Link Control (LLC). Diese ist eine Komponente der Schicht 2 (MAC) im OSI-Referenzmodell.

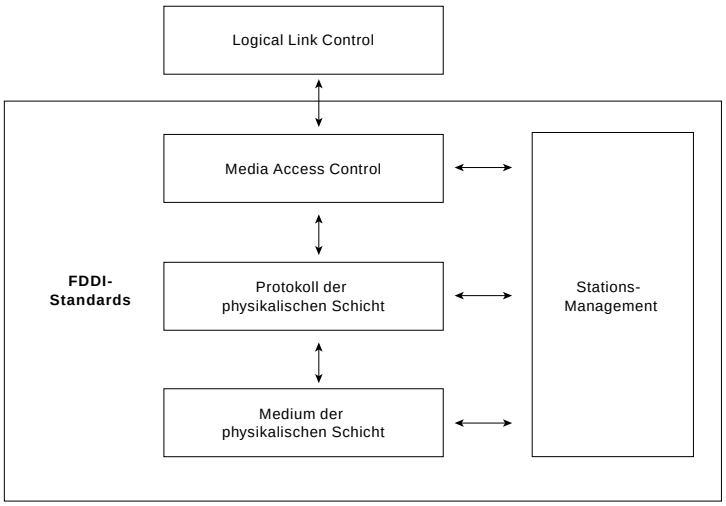


Bild 8.3: FDDI-Spezifikationen entsprechen dem hierarchischen OSI-Modell

8.4 FDDI-Station-Attachment-Typen

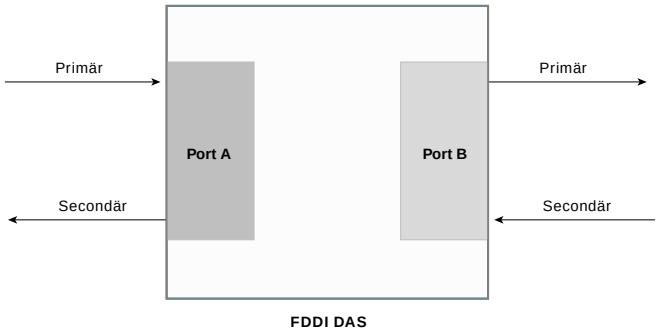
Eine der kennzeichnenden Eigenschaften von FDDI ist die Möglichkeit, FDDI-Geräte auf verschiedene Weise anzuschließen. FDDI definiert drei Gerätetypen: Single-Attachment-Stationen (SAS), Dual-Attachment-Stationen (DAS) und den Konzentrator.

Eine SAS ist über einen Konzentrator nur an einen Ring (den primären) angeschlossen. Einer der Vorteile dieser Anschlußart liegt darin, daß SAS einfach abgehängt oder ausgeschaltet werden können, ohne daß dies Auswirkungen auf den FDDI-Ring hat (Konzentratoren werden weiter unten behandelt).

Jede FDDI-DAS verfügt über zwei Schnittstellen, die als A und B gekennzeichnet sind. Damit wird die DAS an den FDDI-Doppelring angeschlossen. Die beiden Schnittstellen sind für den primären und den sekundären Ring bestimmt. Im folgen-

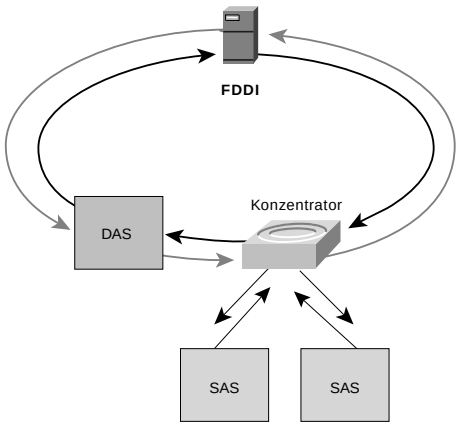
den Abschnitt wird erläutert, warum DAS den FDDI-Ring beeinflussen, wenn sie abgehängt oder ausgeschaltet werden. Bild 8.4 zeigt, wie die Schnittstellen A und B einer FDDI-DAS an den primären und sekundären Ring angeschlossen sind.

Bild 8.4: FDDI-DAS sind an den primären und den sekundären Ring angeschlossen



Ein FDDI-Konzentrator (auch als *Dual-Attachment-Concentrator* [DAC] bezeichnet) ist der grundlegende Baustein eines FDDI-Netzwerks. Er ist direkt mit dem primären und dem sekundären Ring verbunden und stellt sicher, daß ein Fehler oder das Ausschalten einer SAS-Station den Ring nicht unterbricht. Diese Funktion des DAC ist besonders sinnvoll, wenn PCs oder andere Geräte am FDDI-Netz betrieben werden, die öfter ein- und ausgeschaltet werden. Bild 8.5 zeigt die Anschlüsse von SAS-, DAS-Stationen und des Konzentrators.

Bild 8.5: Ein Connector ist an den primären und den sekundären Ring angeschlossen



8.5 FDDI-Fehlertoleranz

FDDI ist mit mehreren Fehlertoleranzfunktionen ausgestattet. Dazu zählen der doppelte FDDI-Ring, die Implementation eines optischen Bypass-Switches und die Unterstützung des Dual-Homing, die FDDI zu einer unverwüstlichen Medientechnologie machen.

8.5.1 Doppelring

Das primäre Feature der FDDI-Fehlertoleranz ist der Doppelring. Falls eine Station im Doppelring ausfällt oder ausgeschaltet oder ein Kabel beschädigt wird, wird aus dem Doppelring automatisch ein einfacher Ring (da der Ringe doppelt rückgekoppelt ist). Dabei wird aus der Doppelring-Topologie eine Einfach-Ring-Topologie. Währenddessen können die Daten ohne Performance-Verlust weiter übertragen werden. Bild 8.6 und Bild 8.7 zeigen die Auswirkung dieser Ring-Umleitung bei FDDI.

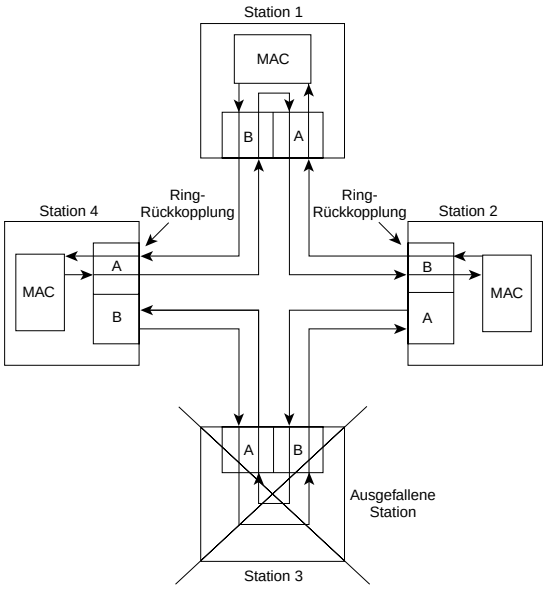


Bild 8.6: Ein Stationsausfall wird überbrückt

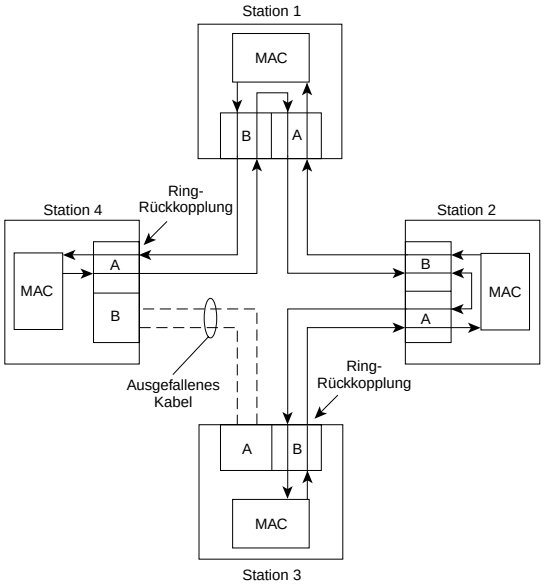


Bild 8.7: Ring-Umleitung wegen Kabelstörung

Wenn eine einzelne Station ausfällt oder ausgeschaltet wird, wie in Bild 8.6 dargestellt, erfolgt die Rückkopplung des Doppelrings zu einem Ring in den beiden benachbarten Stationen. Der Netzwerk-Betrieb wird für die anderen Stationen aufrechterhalten. Gleiches gilt, wenn ein Kabel beschädigt wurde, wie in Bild 8.7 dargestellt, so daß für alle Stationen der Netzwerk-Betrieb weiterläuft.

Dabei ist zu beachten, daß die Fehlertoleranz nur für ein Fehlerereignis gilt. Treten zwei oder mehr Fehler gleichzeitig auf, zerfällt der FDDI-Ring in zwei oder mehr Segmente, die nicht miteinander kommunizieren können.

8.5.2 Optischer Bypass-Switch

Ein optischer Bypass-Switch stellt einen unterbrechungsfreien Doppelringbetrieb sicher, falls ein Gerät im Doppelring ausfällt. Deshalb wird der Bypass-Switch dazu eingesetzt, einer Ring-Segmentierung vorzubeugen und ausgefallene Stationen vom Ring zu isolieren. Um diese Aufgabe zu erfüllen, ist der Bypass-Switch mit optischen Spiegeln ausgestattet, die das Licht bei normalem Betrieb aus dem Ring direkt an die DAS leiten. Im Fehlerfall aber, z.B. durch Ausschalten der DAS, wird das Licht durch den Bypass-Switch im Ring weitergelei-

tet, so daß es zu keiner Unterbrechung des Doppelrings kommt. Der Vorteil dieser Funktion liegt auf der Hand. Bild 8.8 zeigt die Funktion eines optischen Bypass-Switches in einem FDDI-Netzwerk.

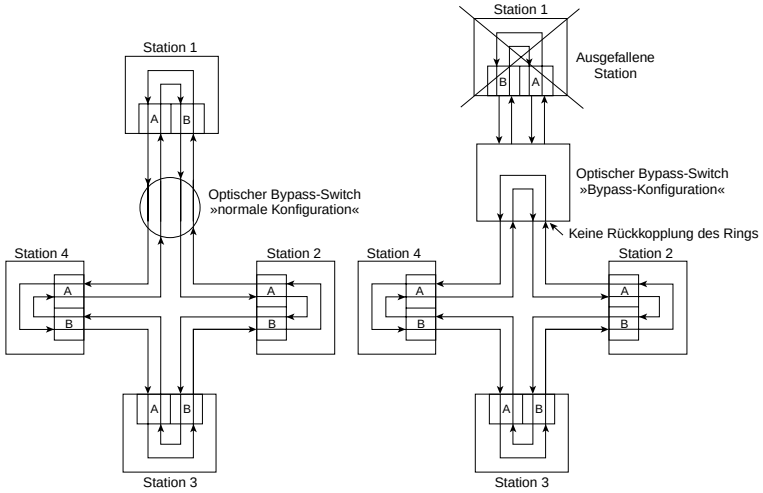


Bild 8.8: Ein optischer Bypass-Switch setzt eingebaute Spiegel ein, um die Netzverfügbarkeit sicherzustellen

8.5.3 Dual-Homing

Bei kritischen Geräten, wie Routern oder Mainframes, kann eine fehlertolerante Technik zum Einsatz kommen, die als *Dual-Homing* bezeichnet wird und mit ihrer Redundanz zur sicheren Verfügbarkeit beiträgt. Beim Dual-Homing ist ein kritisches Gerät an zwei Konzentratoren angeschlossen. Bild 8.9 zeigt eine Dual-Homing-Konfiguration für Geräte wie File-Server und Router.

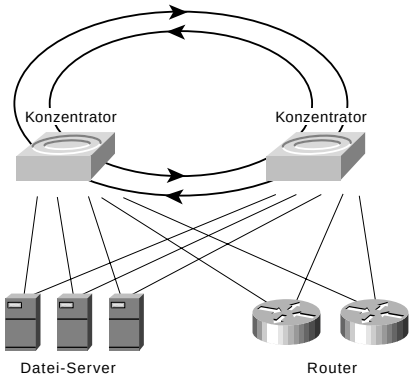


Bild 8.9: Dual-Home-Konfguration stellt Betrieb sicher

Ein Anschlußpaar am Konzentrator wird zum aktiven, das andere zum passiven Anschluß erklärt. Der passive Anschluß läuft im Sicherungsmodus, bis erkannt wird, daß der primäre Anschluß (oder der Konzentrator, an den er angeschlossen ist) ausgefallen ist. In diesem Fall wird der passive Anschluß automatisch aktiviert.

8.6 FDDI-Frame-Format

Das Frame-Format im FDDI ist dem bei Token Ring sehr ähnlich. Dies ist einer der Bereiche, in denen sich FDDI sehr stark an frühere LAN-Technologien wie Token Ring anlehnt. FDDI-Frames können bis zu 4500 Byte lang sein. Bild 8.10 zeigt das Frame-Format eines FDDI-Daten-Frames und -Tokens.

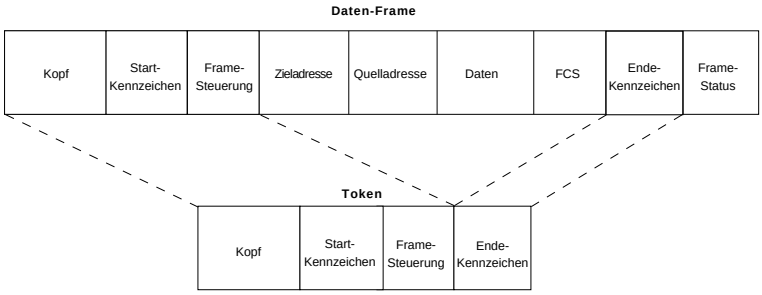


Bild 8.10: Der FDDI-Frame ist dem Token-Ring-Frame sehr ähnlich

8.6.1 FDDI-Frame-Felder

Die folgenden Beschreibungen fassen die in Bild 8.10 gezeigten Felder der FDDI-Daten-Frames und -Tokens zusammen.

- *Kopf (Header)* – Eindeutige Sequenz, die eine Station auf einen eingehenden Frame vorbereitet.
- *Start-Kennzeichen* – Kennzeichnet den Anfang eines Frame anhand eines bestimmten Signalmusters, das sich vom restlichen Frame unterscheidet.
- *Frame-Steuerung* – Gibt die Größe (und weitere Steuer-Informationen) der Adreßfelder an und ob der Frame asynchrone oder synchrone Daten enthält.
- *Zieladresse* – Enthält eine Unicast- (einzelne), Multicast- (Gruppe) oder Broadcast-Adresse (alle Stationen). Wie die

Adressen bei Ethernet und Token Ring sind die Zieladressen bei FDDI 6 Byte lang.

- *Quelladresse* – Gibt die Station an, die den Frame gesendet hat. Wie die Adressen bei Ethernet und Token Ring sind Quelladressen bei FDDI 6 Byte lang.
- *Daten* – Enthält entweder Daten für eine höhere Protokollschicht oder Steuerdaten.
- *Frame-Prüfsumme (Frame Check Sequence [FCS])* – Von der sendenden Station in Abhängigkeit vom Frame-Inhalt berechneter CRC-Wert (wie bei Ethernet und Token Ring). Von der empfangenden Station wird dieser Wert zum Vergleich erneut berechnet, um so festzustellen, ob die Daten fehlerfrei übertragen wurden. Falls ein Fehler auftrat, wird der Frame verworfen.
- *Ende-Kennzeichen* – Enthält eindeutige Symbole, die keine Daten darstellen können, um das Ende eines Frame zu kennzeichnen.
- *Frame-Status* – Dient der sendenden Station dazu, einen Fehler zu erkennen und festzustellen, ob der Frame erkannt und von der empfangenden Station kopiert wurde.

8.7 Copper-Distributed Data Interface (CDDI)

Copper-Distributed Data Interface (CDDI) ist die Implementierung von FDDI-Protokollen für Twisted-Pair-Kupferleitungen. Wie FDDI kann mit CDDI eine Übertragungsrate von 100 Mbps erreicht werden, und es wird auch mit einem Doppelring gearbeitet. Bei CDDI beträgt die maximale Entfernung zwischen Desktop und Konzentrator ca. 100 Meter.

CDDI wurde vom X3T9.5-Komitee des ANSI definiert. Die offizielle Bezeichnung für den CDDI-Standard lautet: *Twisted-Pair Physical Medium Dependent* (TP-PMD). Oft wird aber auch von *Twisted-Pair Distributed Data Interface* (TP-DDI) gesprochen, in Anlehnung an den Begriff *Fiber-Distributed Data Interface* (FDDI). CDDI entspricht den ANSI-Standards für die physische Schicht und die Mediumzugriffssteuerungsschicht.

Im ANSI-Standard sind nur zwei Kabelarten für CDDI vorgesehen: geschirmtes Twisted Pair (STP) und ungeschirmtes Twisted Pair (UTP). Die Impedanz bei STP-Kabeln beträgt 150 Ohm und orientiert sich an den Spezifikationen EIA/TIA 568 (IBM Type 1). UTP ist eine für die Datenübertragung geeignete Verkabelung (Kategorie 5), bei der acht ungeschirmte Drähte zu je zwei verdrillt in einen speziell entwickelten Kunststoff (isolierende Polymere) eingegossen werden, so daß sie der Spezifikation EIA/TIA 568B entspricht.

Bild 8.11 zeigt die Spezifikation CDDI TP-PMD im Verhältnis zu den übrigen FDDI-Spezifikationen.

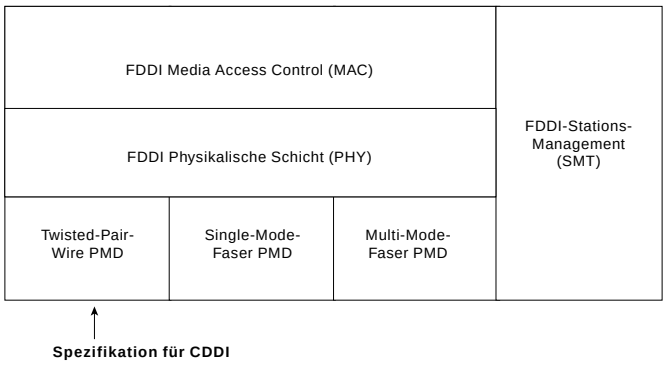


Bild 8.11: Die Spezifikationen CDDI-TP-PMD und FDDI beziehen sich auf veschiedene Standards

KAPITEL 9

Token Ring/IEEE 802.5

9.1 Hintergrund

Das Token-Ring-Netzwerk wurde ursprünglich in den 70er Jahren von IBM entwickelt. Es ist immer noch die wichtigste LAN-Technologie von IBM und steht in der Popularität an zweiter Stelle nach Ethernet/IEEE 802.3. Die Spezifikation IEEE 802.5 ist weitestgehend identisch und vollständig kompatibel zum Token-Ring-Netzwerk von IBM. Die IEE-802.5-Spezifikation lehnt sich sehr stark an jene von Token Ring an und wird weiterhin den Entwicklungen des Token Ring angepaßt. Wenn von *Token Ring* gesprochen wird, bezeichnet man damit sowohl das IBM-Token-Ring-Netzwerk als auch ein IEEE-802.4-Netzwerk. In diesem Kapitel werden beide Spezifikationen behandelt.

Token-Ring- und IEEE-802.5-Netzwerke sind grundsätzlich kompatibel, wenngleich die Spezifikationen ein wenig voneinander abweichen. Von IBM wird bei Token Ring ein Stern spezifiziert, dessen End-Stationen alle an ein Gerät, die sog. *Multistation Access Unit* (MSAU), angeschlossen werden. IEEE 802.5 spezifiziert im Gegensatz dazu keine Topologie, wenn auch virtuell alle IEEE-802.5-Implementationen auf einem Stern basieren. Weitere Unterschiede existieren hinsichtlich des Mediumtyps (IEEE 802.5 spezifiziert keinen Mediumtyp vor, von IBM wird für Token Ring eine Twisted-Pair-Verkabelung angegeben) und der Feldgröße bei den Routing-Informationen. Bild 9.1 faßt die Spezifikationen IBM Token Ring und IEEE 802.5 zusammen.

Bild 9.1: Wenngleich in einigen Punkten verschieden, so sind die Netzwerke IBM Token Ring und IEEE 802.5 im allgemeinen kompatibel

	IBM-Token-Ring-Netzwerk	IEEE 802.5
Datenrate	4.16 MBit/s	4.16 MBit/s
Stationen/ Segment	260 (shielded twisted pair) 72 (unshielded twisted pair)	250
Topologie	Stern	Nicht spezifiziert
Medien	Twisted pair	Nicht spezifiziert
Signalisierung	Baseband	Baseband
Zugriffsmethode	Token passing	Token passing
Kodierung	Differential Manchester	Differential Manchester

9.2 Physische Verbindungen

Die Stationen in einem IBM-Token-Ring-Netzwerk werden direkt an MSAUs angeschlossen, die wiederum zu einem großen Ring miteinander verbunden werden können (siehe Bild 9.2). Die benachbarten MSAUs werden dabei mit Patch-Kabeln verbunden, während die Stationen mit Lobe-Kabeln an die MSAUs angeschlossen werden. In die MSAUs integriert sind Bypass-Relais, mit denen Stationen vom Ring abgekoppelt werden können.

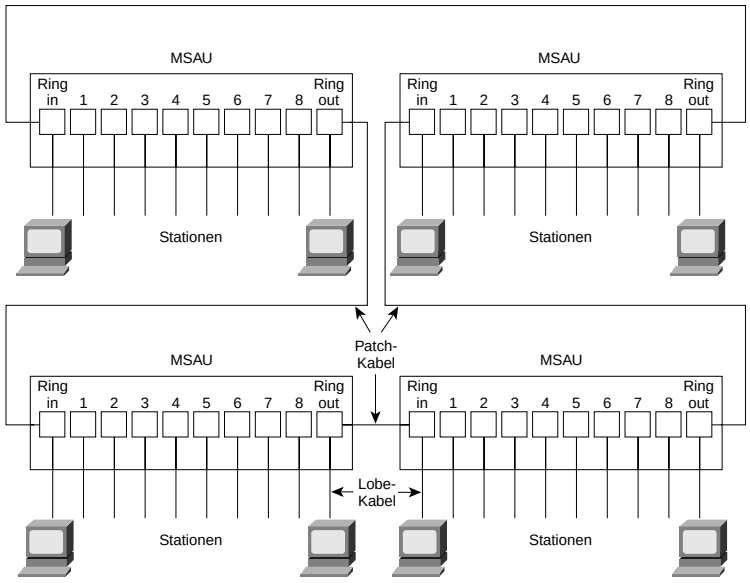


Bild 9.2: In einem IBM-Token-Ring-Netzwerk können MSAUs miteinander zu einem großen Ring verbunden werden

9.3 Betrieb eines Token Ring

Token Ring und IEEE 802.5 sind zwei Beispiele für Token-Passing-Netzwerke (FDDI ist ein weiteres). Token-Passing-Netzwerke senden einen kleinen Frame, das sog. *Token*, durch das Netzwerk. Wer im Besitz des Token ist, der darf Daten übertragen. Wenn ein Knoten das Token erhält, jedoch keine Daten zu übertragen hat, reicht er das Token an die nächste End-Station weiter. Jede Station kann das Token eine bestimmte Zeit lang behalten.

Wenn eine Station, die gerade das Token besitzt, Daten zu übertragen hat, behält sie das Token, verändert darin ein Bit, so daß aus dem Token eine Frame-Anfang-Sequenz wird, hängt die zu übertragenden Daten an und sendet diese Daten an die nächste Station im Ring. Solange dieser Informations-Frame im Ring kreist, wird kein Token weitergereicht (es sei denn, daß der Ring *Early Token Release* unterstützt), so daß andere Stationen, die Daten übertragen wollen, warten müssen. Daraus ergibt sich, daß es in Token-Ring-Netzwerken nicht zu Kollisionen kommen kann. Wenn die frühe Token-Freigabe unterstützt wird, kann ein neues Token gesendet werden, wenn die Frame-Übertragung beendet wurde.

Der Informations-Frame durchläuft so lange den Ring, bis er die Zielstation erreicht. Diese kopiert die Information für die weitere Verarbeitung. Der Informations-Frame wird im Ring weitergereicht, bis er die sendende Station wieder erreicht, von der er gelöscht wird. Die sendende Station kann den zurückkehrenden Frame prüfen, ob er von der Zielstation erkannt und kopiert wurde.

Anders als CSMA/CD-Netzwerke (z.B. Ethernet) sind Token-Passing-Netzwerke deterministisch, d.h., daß es möglich ist, die maximale Zeitspanne zu berechnen, bis zu der jede End-Station einmal die Gelegenheit hatte Daten zu übertragen. Diese Funktion und viele Zuverlässigkeitsfunktionen machen Token-Ring-Netzwerke zum idealen Netz für Anwendungen, die eine vorhersagbare Verzögerung und robusten Netzwerkbetrieb benötigen. Auf diese Funktionen wird weiter unten im Abschnitt »Mechanismen des Ausfall-Managements« näher eingegangen. Ein Beispiel für solche Anwendungen stellt die Fertigungsautomatisierung dar.

9.4 Prioritäten-System

Von Token-Ring-Netzwerken wird ein ausgeklügeltes Prioritäten-System verwendet, das es bestimmten benutzerdefinierten, Stationen mit hoher Priorität erlaubt, das Netzwerk häufiger zu benutzen. Token-Ring-Frames enthalten zwei Felder, die die Priorität steuern: das *Prioritäts*feld und das *Reservierungs*feld.

Nur Stationen, deren Priorität genauso hoch oder höher ist als die im Token angegebene, können das Token halten. Nachdem das Token gehalten und in einen Informations-Frame geändert wurde, können nur Stationen mit höherer Priorität als die gerade übertragende Station das Token für den nächsten Durchlauf durch das Netzwerk für sich reservieren. Beim Generieren des nächsten Token wird in diesem eine höhere Priorität gesetzt als die der reservierenden Station. Stationen, die die Priorität des Token erhöht haben, müssen nach Beendigung ihrer Datenübertragung die Priorität wieder auf die Stufe setzen, die davor galt.

9.5 Mechanismen des Ausfall-Managements

Token-Ring-Netzwerke arbeiten mit mehreren Mechanismen für die Erkennung und Kompensation von Netzwerk-Ausfällen. So wird z.B. eine Station in einem Token-Ring-Netzwerk als *aktive Überwachungsstation (active monitor)* bestimmt. Potentiell kommt dafür jede Station des Netzwerks in Frage. Diese Station ist die zentrale Quelle für Timing-Informationen, die für alle Stationen im Ring gelten. Außerdem führt diese Station verschiedene Wartungsfunktionen im Ring aus, wie z.B. das Entfernen von endlos kreisenden Frames, die entstehen können, wenn ein sendendes Gerät ausfällt. Eine solche Situation kann dazu führen, daß andere Stationen mit der Übertragung ihrer Frames warten und so das Netzwerk blokkiert wird. Die aktive Überwachungsstation erkennt solche Frames, entfernt sie vom Ring und generiert ein neues Token.

Die Stern-Topologie des IBM-Token-Ring-Netzwerks bietet eine umfassende Netzwerk-Zuverlässigkeit. Da sämtliche Informationen von den aktiven MSAUs gelesen werden, können diese so programmiert werden, daß sie Problemlagen erkennen und ggf. einzelne Stationen vom Ring nehmen können.

Ein als *Beaconing* bezeichneter Algorithmus in Token Ring erkennt bestimmte Netzwerk-Fehler und versucht, diese zu beheben. Immer wenn eine Station einen schwerwiegenden Fehler erkennt (z.B. eine unterbrochene Leitung), sendet sie einen Beacon-Frame, der eine ausgefallene Domäne definiert. Diese Domäne enthält Angaben zur Station, die den Fehler meldet, deren nächstgelegenen aktiven Nachbarn (*nearest active upstream neighbor* – NAUN) und alle anderen dazwischen. Das Beaconing startet einen Prozeß, der als *Autorekonfiguration* bezeichnet wird. Dabei führen die Knoten in der fehlerhaften Domäne automatisch Diagnosen durch, um das Netzwerk um den ausgefallenen Bereich herum zu rekonfigurieren. Eine MSAU kann diese Fehlerbehebung durch elektrische Rekonfiguration erreichen.

9.6 Frame-Format

Token Ring und IEEE 802.5 unterstützen zwei grundlegende Frame-Typen: Tokens und Daten-/Befehls-Frames. Tokens sind

drei Byte lang und beginnen mit einem Start-Kennzeichen, es folgen ein Zugriffssteuerungs-Byte und ein Ende-Kennzeichen. Daten-/Befehls-Frames variieren in der Größe, wobei diese vom Umfang des Datenfelds abhängt. Daten-Frames enthalten Informationen für die Protokolle der höheren Schichten, während Befehls-Frames nur Steuerinformationen enthalten. Beide Formate sind in Bild 9.3 dargestellt.

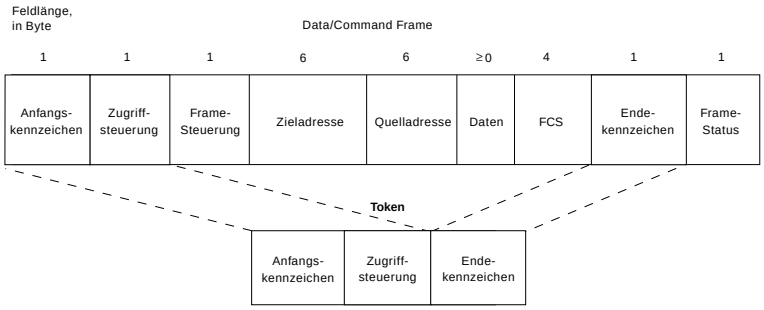


Bild 9.3: IEEE 802.5 und Token Ring spezifizieren Tokens und Daten-/Befehls-Frames

9.6.1 Felder des Token-Frames

Die Felder des Token-Frame, wie in Bild 9.3 dargestellt, werden im folgenden kurz erläutert:

- *Anfangs-Kennzeichen* – Informiert jede Station, daß ein Token eingegangen ist (oder ein Daten-/Befehls-Frame). Dieses Feld enthält Signale, die dieses Byte von allen anderen im Frame unterscheidet, indem es gegen das Kodierungsschema für den restlichen Frame verstößt.
- *Zugriffssteuerungs-Byte* – Enthält das Prioritätsfeld (die drei höherwertigen Bit) und das Reservierungsfeld (die drei niederwertigen Bit), ein Token-Bit (zur Unterscheidung eines Token von einem Daten-/Befehls-Frame) und ein Überwachungsbit (das von der aktiven Überwachungsstation dazu verwendet wird, einen endlos kreisenden Frame zu erkennen).
- *Ende-Kennzeichen* – Kennzeichnet das Ende eines Token oder Daten-/Befehls-Frame. Außerdem enthält dieses Feld Bits, die einen beschädigten Frame oder den Frame als letzten in einer logischen Folge kennzeichnen.

9.6.2 Felder des Daten-/Befehls-Frames

Daten-/Befehls-Frames enthalten auch die drei vorgenannten Felder eines Token-Frame. Die weiteren Felder eines Daten-/Befehls-Frame, wie in Bild 9.3 dargestellt, werden im folgenden kurz erläutert:

- *Anfangs-Kennzeichen* – Informiert jede Station, daß ein Token eingegangen ist (oder ein Daten-/Befehls-Frame). Dieses Feld enthält Signale, die dieses Byte von allen anderen im Frame unterscheidet, indem es gegen das Kodierungsschema für den restlichen Frame verstößt.
- *Zugriffssteuerungs-Byte* – Enthält das Prioritätsfeld (die drei höherwertigen Bit) und das Reservierungsfeld (die drei niederwertigen Bit), ein Token-Bit (zur Unterscheidung eines Token von einem Daten-/Befehls-Frame) und ein Überwachungsbit (wird von der aktiven Überwachungsstation verwendet, um einen endlos kreisenden Frame zu erkennen).
- *Frame-Steuer-Byte* – Gibt an, ob der Frame Daten oder Steuerinformationen enthält. Handelt es sich um einen Steuer-Frame, gibt dieses Byte an, um welche Art der Steuerinformation es sich handelt.
- *Ziel- und Quelladresse* – Zwei jeweils 6 Byte lange Felder geben die Quell- und Zieladresse an.
- *Daten* – Die Länge dieses Felds ist begrenzt durch die Ring-Token-Verweildauer, die definiert, wie lange eine Station das Token halten darf.
- *Frame-Prüfsumme* (FCS) – Von der Quellstation durch Berechnung über den Frame-Inhalt erzeugter Wert. Die Zielstation führt die gleiche Berechnung aus, um festzustellen, ob die Daten bei der Übertragung beschädigt wurden. Ist dies der Fall, wird der Frame entwertet.
- *Ende-Kennzeichen* – Kennzeichnet das Ende eines Token oder Daten-/Befehls-Frame. Außerdem enthält dieses Feld Bits, die einen beschädigten Frame oder den Frame als letzten in einer logischen Folge kennzeichnen.
- *Frame-Status* – Dieses Feld ist ein Byte lang und beendet einen Daten-/Befehls-Frame. Dieses Feld enthält die beiden Kennzeichnungen für die Adressenerkennung und dafür, daß der Frame kopiert wurde.

Kapitel 10: Frame Relay
Kapitel 11: High-Speed Serial Interface
Kapitel 12: Integrated Services Digital Network (ISDN)
Kapitel 13: Point-to-Point Protocol
Kapitel 14: Switched Multimegabit Data Service (SMDS)
Kapitel 15: ADSL – Asymmetric Digital Subscriber Line
Kapitel 16: Synchronous Data-Link Control und Derivate
Kapitel 17: X.25

TEIL 3

WAN-Technologien

Teil 3, »WAN-Technologien«, faßt die Spezifikationen und operationalen Eigenschaften der Schlüsseltechnologien und Protokolle für WAN (wide-area network) zusammen. In den einzelnen Kapiteln werden folgende Themen behandelt:

Frame Relay – In diesem Kapitel werden der Betrieb und die Eigenschaften dieser Hochgeschwindigkeitstechnologie für WAN beschrieben.

High-Speed Serial Interface (HSSI) – Hier wird HSSI definiert und der Einsatz der HSSI-Technologie in T3-WAN-Implementationen zusammengefaßt.

Integrated Services Digital Network (ISDN) – Hier wird ISDN definiert und der Einsatz von ISDN als eine WAN-Technologie zusammengefaßt.

Point-to-Point Protocol (PPP) – Hier wird PPP definiert und der Einsatz von PPP für den Fernzugriff in WAN-Umgebungen beschrieben.

Switched Multimegabit Data Service (SMDS) – In diesem Kapitel werden der Betrieb und die Eigenschaften von SMDS beschrieben. Bei SMDS handelt es sich um eine Einwahl-Technologie für WAN mit hoher Bandbreite.

Asymmetric Digital Subscriber Line (ADSL) – In diesem Kapitel werden der Betrieb und die Eigenschaften von ADSL beschrieben. Bei ADSL handelt es sich um eine Hochgeschwindigkeitsimplementation für WAN.

Synchronous Data-Link Control and Derivatives (SDLC) – In diesem Kapitel geht es um die Rolle von SDLC als ein Protokoll der Verbindungssicherungsschicht in IBM-SNA-Netzen und einen Überblick davon abgeleiteter Protokolle.

X.25 – Dieses Kapitel behandelt den Betrieb und die Eigenschaften von X.25.

KAPITEL 10

Frame Relay

10.1 Hintergrund

Frame Relay ist ein Hochgeschwindigkeitsprotokoll für WAN, das sich auf die physische Schicht und die Sicherungsschicht des OSI-Referenzmodells bezieht. Ursprünglich wurde Frame Relay für den Einsatz unter ISDN (Integrated Services Digital Network) entwickelt. Heute wird es aber auch von vielen anderen Netzwerk-Schnittstellen verwendet. In diesem Kapitel geht es hauptsächlich um die Frame-Relay-Spezifikationen und -Anwendungen im Zusammenhang mit WAN-Services.

Frame Relay ist ein Beispiel für eine paketvermittelte Technologie. Paketvermittelte Netzwerke ermöglichen es den End-Stationen, das Netzwerk-Medium und die verfügbare Bandbreite dynamisch gemeinsam zu nutzen. Dabei werden Pakete variabler Länge versendet, so daß die Übertragung wesentlich effizienter und flexibler erfolgt. Diese Pakete werden dann von den verschiedenen Netzwerk-Segmenten weitervermittelt, bis sie ihr Ziel erreichen. Statistische Multiplex-Techniken steuern den Netzwerk-Zugriff in paketvermittelten Netzen. Der Vorteil dieser Technik besteht darin, daß die Bandbreite flexibler und effizienter genutzt wird. Bei den heute gängigsten LANs, wie Ethernet und Token Ring, handelt es sich um paketvermittelte Netzwerke.

Frame Relay wird oft als eine modernisierte Version von X.25 bezeichnet, ohne einige robuste Eigenschaften von X.25, wie etwa das Windowing und die Wiederholung der letzten Daten.

Dies hat seine Ursache darin, daß Frame Relay über WAN-Facilities betrieben wird, die wesentlich zuverlässigere Verbindungsservices und eine verbesserte Zuverlässigkeit bieten, als dies in den späten 70er und frühen 80er Jahren bei Plattformen der Fall war, von denen X.25 für WANs unterstützt wurde. Wie bereits erwähnt, ist Frame Relay eine Protokollfamilie, die sich strikt auf die Schicht 2 bezieht, während X.25 auch für die Schicht 3 (Vermittlungsschicht) Services anbietet. Deshalb ist mit Frame Relay eine höhere Performance und größere Übertragungseffektivität als mit X.25 möglich. Was wiederum der Grund ist, daß Frame Relay so gut für aktuelle WAN-Anwendungen (z.B. LAN Interconnection) geeignet ist.

10.1.1 Frame-Relay-Standardisierung

Die ersten Vorschläge für die Standardisierung von Frame Relay wurden dem Consultative Committee on International Telephone and Telegraph (CCITT) 1984 unterbreitet. Aufgrund der fehlenden Interoperabilität und Standardisierung kam es zu keinem verstärkten Einsatz von Frame Relay bis in die späten 80er Jahre.

Zu wesentlichen Schritten in der Weiterentwicklung von Frame Relay kam es 1990, als Cisco, Digital Equipment, Northern Telecom und StrataCom ein Konsortium bildeten, das sich Frame Relay auf die Fahnen geschrieben hatte. Von diesem Konsortium wurde eine Spezifikation entworfen, die dem grundlegenden Frame-Relay-Protokoll entsprach und vom CCITT diskutiert wurde. Das CCITT erweiterte das Protokoll mit Funktionen, die zusätzliche Eigenschaften für komplexe Internetworking-Umgebungen boten. Diese Frame-Relay-Erweiterungen werden zusammenfassend als Local Management Interface (LMI) bezeichnet.

Seit von dem Konsortium die Spezifikation entwickelt und veröffentlicht wurde, haben viele Anbieter dieser erweiterten Frame-Relay-Definition ihre Unterstützung zugesagt. ANSI und CCITT haben in der Zwischenzeit ihre eigenen Varianten der ursprünglichen LMI-Spezifikation standardisiert. Diese kommen heute weit häufiger zum Einsatz als die Original-Version.

Auf internationaler Ebene wurde Frame Relay von der International Telecommunications Union – Telecommunications Sector (ITU-T) standardisiert. In den USA ist Frame Relay ein vom American National Standards Institute (ANSI) festgelegter Standard.

10.2 Frame-Relay-Geräte

An ein Frame-Relay-WAN angeschlossene Geräte lassen sich in zwei große Kategorien unterteilen: Data Terminal Equipment (DTE – Datenendeinrichtung [DEE]) und Data Circuit-Terminating Equipment (DCE – Datenübertragungseinrichtung [DÜE]). Bei DTEs handelt es sich im allgemeinen um Endeinrichtungen für ein bestimmtes Netzwerk. Die DTEs sind normalerweise in den Räumlichkeiten des Kunden aufgestellt, der sie ggf. gekauft hat. DTE-Geräte sind z.B. Terminals, Personalcomputer, Router und Bridges.

DCEs sind Internetworking-Geräte im Eigentum des Dienstanbieters. DCE-Geräte werden zur Taktung und zum Switching in einem Netzwerk eingesetzt. Dabei handelt es sich um die Geräte, die die eigentliche Datenübertragung in einem WAN bewerkstelligen. In den meisten Fällen werden dies Paket-Switches sein. Bild 10.1 zeigt die Beziehung zwischen diesen beiden Gerätekategorien.

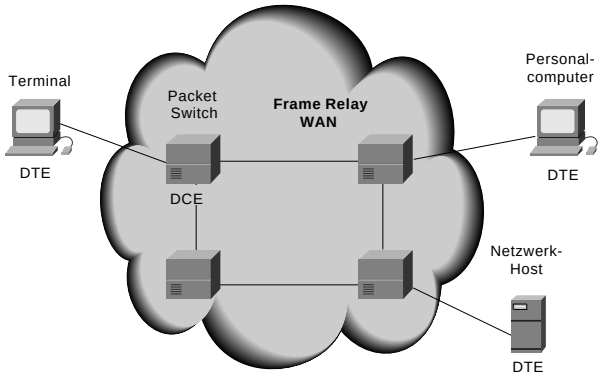


Bild 10.1: DCEs stehen im allgemeinen in WANs, die von Dienstanbietern betrieben werden

Die Verbindung zwischen DTE- und DCE-Gerät wird sowohl von Komponenten der physischen Schicht als auch der Sicherungsschicht getragen. Die physische Komponente spezifiziert

die mechanische, elektrische, funktionale und prozedurale Seite einer Verbindung zwischen Geräten. Eine der gebräuchlichsten Schnittstellen-Spezifikationen der physischen Schicht ist der Recommended Standard (RS)-232. Die Komponente der Sicherungsschicht definiert das Protokoll, das die Verbindung zwischen einem DTE-Gerät (z.B. einem Router) und einem DCE-Gerät (z.B. einem Switch) herstellt. Dieses Kapitel beschreibt eine oft in der WAN-Technologie eingesetzte Protokoll-Spezifikation – das Frame-Relay-Protokoll.

10.3 Frame Relay Virtual Circuits

Frame Relay bietet verbindungsorientierte Sicherungsschicht-Kommunikation. Dies bedeutet, daß zwischen zwei Geräten eine definierte Kommunikation besteht, und daß diese Verbindungen einer Verbindungskennzeichnung zugeordnet sind. Dieser Service wird implementiert durch den Einsatz eines *Frame Relay Virtual Circuit*. Dieser stellt eine logische Verbindung dar, die zwischen zwei Datenendeinrichtungen (DTE/DEE) über ein paketvermitteltes Frame-Relay-Netzwerk erzeugt wurde.

Virtual Circuits (virtuelle Verbindungen) bieten einen bi-direktionalen Kommunikationspfad von einem DTE-Gerät zu einem anderen. Sie sind eindeutig gekennzeichnet anhand eines Data-Link Connection Identifier (DLCI). Mehrere virtuelle Verbindungen können für die Übertragung im Netzwerk über eine physische Verbindung gemultiplext werden. Diese Eigenschaft kann genutzt werden, um die Ausstattung und die Netzwerk-Komplexität zu reduzieren, wenn mehrere DTE-Geräte verbunden werden sollen.

Eine virtuelle Verbindung kann über eine beliebige Anzahl dazwischengeschalteter DCE-Geräte (Swichtes) im Frame Relay PSN geleitet werden.

Virtuelle Frame-Relay-Verbindungen lassen sich in zwei Kategorien unterteilen: Switched Virtual Circuits (SVC – gewählte virtuelle Verbindungen [GVV]) und Permanent Virtual Circuits (PVC – feste virtuelle Verbindungen [FVV]).

10.3.1 Switched Virtual Circuits (SVC/GVV)

Switched Virtual Circuits (SVCs – gewählte virtuelle Verbindungen [GVV]) sind temporäre Verbindungen, die eingesetzt werden, wenn zwischen DTE-Geräten in einem Frame-Relay-Netzwerk nur gelegentliche Datenübertragungen stattfinden. Eine Kommunikationssitzung über einen SVC hat folgende Zustände:

- *Anruf (Call Setup)* – Die virtuelle Verbindung zwischen den beiden Frame-Relay-DTE-Geräten wird aufgebaut.
- *Datenübertragung* – Daten werden zwischen den DTE-Geräten über die virtuelle Verbindung übertragen.
- *Ruhezustand* – Die Verbindung zwischen den DTE-Geräten ist noch aktiv, es werden aber keine Daten übertragen. Wenn ein SVC eine bestimmte Zeit lang in diesem Ruhezustand bleibt, kann die physische Verbindung unterbrochen werden.
- *Verbindungsbeendigung* – Die virtuelle Verbindung zwischen den DTE-Geräten wird beendet.

Nachdem die virtuelle Verbindung beendet wurde, muß für eine weitere Datenübertragung zwischen den DTE-Geräten der SVC erneut aufgebaut werden. Es wird erwartet, daß SVCs mit den gleichen Signalisierungsprotokollen wie unter ISDN aufgebaut, verwaltet und beendet werden. Es gibt jedoch Hersteller von Frame-Relay-DCE-Geräten, die gewählte virtuelle Verbindungen unterstützen. In heutigen Frame-Relay-Netzwerken werden deren Produkte aber nur selten eingesetzt.

10.3.2 Permanent Virtual Circuits (PVC/FVV)

Permanent Virtual Circuits (PVCs – feste virtuelle Verbindungen [FVV]) sind fest eingerichtete Verbindungen, die eingesetzt werden, wenn über ein Frame-Relay-Netzwerk zwischen zwei DTE-Geräten häufige und möglichst fehlerfreie Datenübertragungen erfolgen sollen. Für die Kommunikation über einen PVC sind der Anruf und die Verbindungsbeendigung wie bei einem SVC nicht erforderlich. PVCs befinden sich immer in einem der beiden folgenden operationalen Zustände:

– *Datenübertragung* – Daten werden über die virtuelle Verbindung zwischen den DTE-Geräten übertragen.

– *Ruhezustand* – Die Verbindung zwischen den DTE-Geräten ist aktiv, es werden jedoch keine Daten übertragen. Anders als bei SVCs werden PVCs unter keinen Umständen beendet, auch nicht im Ruhezustand.

DTE-Geräte können jederzeit mit der Datenübertragung beginnen, da die Verbindung ständig aufgebaut bleibt.

10.3.3 Data-Link Connection Identifier (DLCI)

Virtuelle Frame-Relay-Verbindungen sind gekennzeichnet durch Data-Link Connection Identifiers (DLCIs). Normalerweise nimmt der Dienstanbieter (z.B. die Telefongesellschaft) des Frame-Relay-Netzwerks die Zuordnung der DLCI-Werte vor. Frame-Relay-DLCIs sind lokal eindeutig, nicht jedoch in einem Frame-Relay-WAN. So können z.B. zwei über eine virtuelle Verbindung kommunizierende DTE-Geräte verschiedene DLCI-Werte verwenden, um die gleiche Verbindung zu verwenden. Bild 10.2 zeigt, wie einer einzelnen virtuellen Verbindung am jeweiligen Ende verschiedene DLCI-Werte zugeordnet werden.

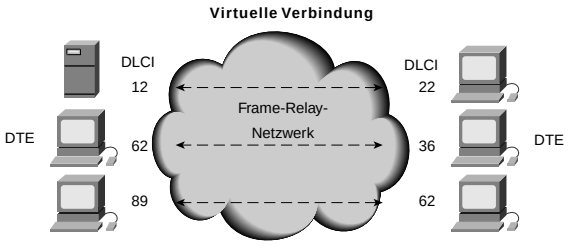


Bild 10.2: Einer virtuellen Frame-Relay-Verbindung können an jedem Ende verschiedene DLCI-Werte zugeordnet werden

10.4 Congestion-Control-Mechanismen

Frame Relay reduziert den Netzwerk-Overhead, indem einfache Stau-Erkennungsmechanismen anstelle expliziter Flußsteuerung für jede virtuelle Verbindung implementiert werden. Da Frame Relay auf zuverlässigen Netzwerk-Medien implementiert wird, muß auf die Datenintegrität nicht verzichtet

werden, weil die Flußsteuerung den Protokollen höherer Schichten überlassen werden kann. Frame Relay implementiert zwei Stau-Erkennungsmechanismen:

– Forward-explicit Congestion Notification (FECN)

– Backward-Explicit Congestion Notification (BECN)

FECN und BECN werden durch ein einzelnes Bit gesteuert, das sich im Frame-Header befindet. Der Frame-Header bei Frame Relay enthält ein DE-Bit (*DE – Discard Eligibility*), mit dem weniger wichtiger Datenverkehr gekennzeichnet wird, der in Stau-Situationen nicht weiterübertragen wird.

Das FECN-Bit ist Teil des Adreßfelds im Frame-Header. Der FECN-Mechanismus wird gestartet, wenn ein DTE-Gerät Frame-Relay-Frames in das Netzwerk sendet. Ist ein Netzwerk überlastet, setzen DCE-Geräte (Switches) den Wert des FECN-Bit eines Frame auf 1. Wenn der Frame sein Ziel erreicht, zeigt das Adreßfeld (mit dem gesetzten FECN-Bit) an, daß der Frame auf der Strecke zwischen Quelle und Ziel in einen Stau geraten war. Das DTE-Gerät kann diese Information an das Protokoll einer höheren Schicht zur Bearbeitung weitergeben. Abhängig von der Implementation wird dann die Flußsteuerung aktiviert, oder die Meldung wird ignoriert.

Das BECN-Bit ist ein Teil des Adreßfelds im Frame-Header. DCE-Geräte setzen das BECN-Bit eines Frame auf 1, wenn dieser in die entgegengesetzte Richtung übertragen wird, aus der Frames kommen, deren FECN-Bit gesetzt ist. Damit wird das empfangende DTE-Gerät darüber informiert, daß ein bestimmter Pfad im Netzwerk überlastet ist. Das DTE-Gerät kann diese Information an das Protokoll einer höheren Schicht zur Bearbeitung weitergeben. Abhängig von der Implementation wird dann die Flußkontrolle aktiviert, oder die Meldung wird ignoriert.

10.4.1 Frame Relay Discard Eligibility (DE)

Das Discard Eligibility-Bit (DE) wird dazu verwendet, einen weniger wichtigen Frame zu kennzeichnen. Das DE-Bit ist Teil des Adreßfelds im Frame-Header.

DTE-Geräte können den Wert des DE-Bits eines Frame auf 1 setzen, um den Frame als weniger wichtig zu kennzeichnen. Bei einer Überlastung des Netzwerkes leiten DCE-Geräte Frames mit gesetztem DE-Bit nicht weiter, bevor andere Frames nicht mehr bearbeitet werden. So wird die Wahrscheinlichkeit reduziert, daß kritische Daten in Phasen der Überlastung von DCE-Geräten nicht mehr bearbeitet werden.

10.4.2 Frame-Relay-Fehlererkennung

Frame Relay verwendet einen üblichen Fehlererkennungsmechanismus, der als *Cyclic Redundancy Check* (CRC) bekannt ist. Der CRC vergleicht zwei berechnete Werte, um festzustellen, ob bei der Übertragung zwischen Quelle und Ziel ein Fehler aufgetreten ist. Frame Relay reduziert den Netzwerk-Overhead durch die Implementation einer Fehlererkennung statt einer Fehlerkorrektur.

Da Frame Relay auf zuverlässigen Netzwerk-Medien implementiert wird, muß auf die Daten-Integrität nicht verzichtet werden, weil die Fehlerkorrektur den Protokollen höherer Schichten überlassen werden kann.

10.5 Frame Relay Local Management Interface (LMI)

Das Local Management Interface (LMI) ist ein Satz von Erweiterungen der grundlegenden Frame-Relay-Spezifikation. Das LMI wurde 1990 von Cisco Systems, StrataCom, Northern Telecom und Digital Equipment Corporation entwickelt. Es bietet eine Vielzahl an Eigenschaften (als *Erweiterungen* bezeichnet) für die Verwaltung komplexer Internetworks. Zu den wichtigen Frame-Relay-LMI-Erweiterungen gehören die globale Adressierung, Statusmeldungen virtueller Verbindungen und Multicasting.

Aufgrund der globalen Adressierungserweiterung des LMI sind die von Frame Relay für *Data-Link Connection Identifier (DLCI)* verwendeten Werte nicht nur lokal, sondern global eindeutig. DLCI-Werte werden so zu DTE-Adressen, die in

einem Frame-Relay-WAN eindeutig sind. Mit der globalen Adressierungserweiterung wächst die Funktionalität und wird die Verwaltung eines Frame-Relay-Internetwork verbessert. So können z.B. einzelne Netzwerk-Schnittstellen und die daran angeschlossenen Endknoten anhand einer Standard-Adreßauflösung und Suchtechnik identifiziert werden. Außerdem stellt sich für die Router das gesamte Frame-Relay-Netzwerk als ein typisches LAN dar.

Frame-Relay-DTE- und -DCE-Geräte kommunizieren und synchronisieren sich über Statusmeldungen virtueller LMI-Verbindungen. Diese Meldungen dienen dazu, regelmäßig über den Status eines PVC zu berichten, um Daten nicht in »Schwarze Löcher« (d.h. über längst beendete PVCs) zu senden.

Mit der LMI-Multicast-Erweiterung können Multicast-Gruppen eingerichtet werden. *Multicasting* hilft Ihnen, Bandbreite zu sparen, indem Routing-Aktualisierungen und Adreßauflösungsmeldungen nur an bestimmte Router-Gruppen gesendet werden. Die Erweiterung überträgt in Aktualisierungsmeldungen auch Statusberichte von Multicast-Gruppen.

10.6 Frame-Relay-Netzwerk-Implementation

Für eine übliche, private Implementation eines Frame-Relay-Netzwerks müssen Sie lediglich einen T1-Multiplexer mit einer Frame-Relay- und einer anderen Schnittstelle ausstatten. Der Frame-Relay-Datenverkehr wird über die Frame-Relay-Schnittstelle gesendet und auf das Netzwerk gebracht. Der andere Datenverkehr geht an die entsprechende Anwendung oder einen geeigneten Service, z.B. eine Nebenstellenanlage oder eine Video-Telekonferenz-Anwendung.

Zu einem typischen Frame-Relay-Netzwerk gehören mehrere DTE-Geräte, z.B. Router, die an die Remote-Ports eines Multiplexer über herkömmliche Punkt-zu-Punkt-Services wie T1, Fractional T1 oder 56K-Verbindungen angeschlossen sind. Ein Beispiel für ein einfaches Frame-Relay-Netzwerk zeigt Bild 10.3.

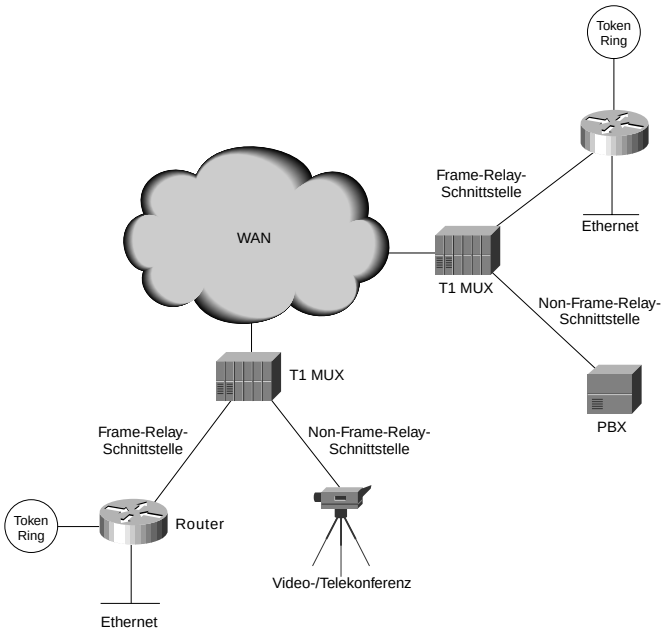


Bild 10.3: Ein einfaches Frame-Relay-Netzwerk verbindet mehrere Geräte über ein WAN mit verschiedenen Services

Die meisten der heute betriebenen Frame-Relay-Netzwerke werden von Dienstanbietern unterhalten, die damit ihren Kunden Übertragungsdienste anbieten, einen sogenannten öffentlichen Frame-Relay-Service. Frame Relay findet sich sowohl in den Netzwerken öffentlicher Anbieter als auch in privaten Unternehmensnetzen. Im folgenden Abschnitt werden die zwei Möglichkeiten für den Einsatz von Frame Relay näher beleuchtet.

10.6.1 Öffentliche Netzwerke

In öffentlichen Frame-Relay-Netzwerken befindet sich die Frame-Relay-Switching-Ausrüstung in der zentralen Niederlassung eines Telekommunikationsanbieters. Die Kosten für den Kunden richten sich nach der Nutzung des Netzwerks, wobei der Anbieter die Administration und Wartung der Ausstattung und Services übernimmt.

Meistens bleiben die DCE-Geräte auch im Eigentum des Telekommunikationsanbieters, der dem Kunden damit einen weiteren Service anbietet.

Die meisten der heute betriebenen Frame-Relay-Netzwerke werden von öffentlichen Anbietern unterhalten.

10.6.2 Private Unternehmensnetze

Immer häufiger unterhalten weltweite Unternehmen eigene Frame-Relay-Netze. Bei solchen privaten Frame-Relay-Netzwerken ist das Unternehmen (eine private Firma) selbst für die Administration und Wartung verantwortlich. Die gesamte Ausstattung, einschließlich der Switching-Geräte, gehört den Kunden.

10.7 Frame-Formate des Frame-Relay

Um den gesamten Funktionsumfang von Frame Relay zu verstehen, ist es hilfreich, sich mit der Struktur des Frame-Relay-Frame vertraut zu machen. Bild 10.7 zeigt das Basisformat eines Frame, Bild 10.9 dessen LMI-Version.

Flags markieren den Frame-Anfang und das Frame-Ende. Der Frame-Relay-Frame besteht aus drei Hauptkomponenten: Header- und Adreßbereich, Benutzerdaten-Bereich und Frame-Prüfsumme (FCS – Frame Check Sequence). Der Adreßbereich ist 2 Byte lang, wobei 10 Bit für die Verbindungskennzeichnung (circuit identifier) und 6 Bit für das Stau-Management reserviert sind. Die Kennzeichnung wird gemeinhin als DLCI (Data-Link Connection Identifier) bezeichnet. Im folgenden Abschnitt wird näher auf diese Komponenten eingegangen.

10.7.1 Standard-Frame des Frame Relay

Die Standard-Frames bei Frame Relay bestehen aus den in Bild 10.4 gezeigten Feldern.

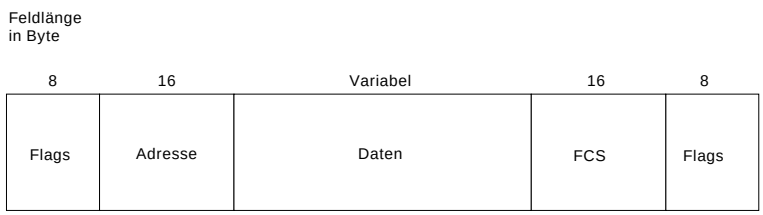


Bild 10.4: Ein Frame des Relay-Frame besteht aus fünf Feldern

Die folgenden Beschreibungen geben einen kurzen Überblick zu diesen grundlegenden Frame-Feldern.

- *Flags* – Begrenzen Anfang und Ende eines Frame. Der Wert dieses Felds ist immer gleich. Er kann hexadezimal als 7E oder binär als 01111110 dargestellt werden.
- *Adresse* – Dieses Feld enthält folgende Informationen:
  - DLCI: Der 10 Bit lange DLCI ist die Essenz des Frame-Relay-Headers. Dieser Wert gibt die virtuelle Verbindung zwischen einem DTE-Gerät und einem Switch wieder. Jede virtuelle Verbindung, die über einen physischen Kanal gemultiplext wird, wird mit einem eindeutigen DLCI dargestellt. Die DLCI-Werte haben nur lokale Bedeutung, d.h., daß sie nur für den physischen Kanal eindeutig sind. Deshalb können Geräte am anderen Ende der Verbindung einen anderen DLCI-Wert für die gleiche virtuelle Verbindung benutzen.
  - Erweiterte Adresse (EA): Die EA dient dazu anzugeben, ob das Byte, in dem der EA-Wert 1 ist, das letzte Adreßfeld ist. Wenn der Wert 1 ist, ist das aktuelle Byte das letzte DLCI-Oktett. Zur Zeit verwenden Frame-Relay-Implementationen nur einen zwei Oktett langen DLCI, aber auf die beschriebene Weise ist sichergestellt, daß in Zukunft auch längere DLCIs genutzt werden können. Jeweils das achte Bit in einem Byte des Adreßfelds wird für die EA verwendet.
  - C/R: Das C/R-Bit folgt dem höherwertigen DLCI-Byte im Adreßfeld. Zur Zeit ist dieses Bit noch nicht definiert.
  - *Stau-Steuerung:* Dafür stehen drei Bits zur Verfügung, über die der Stau-Erkennungsmechanismus gesteuert wird. Das FECN-, BECN- und DE-Bit zählen dazu. Diese sind die letzten drei Bit im Adreßfeld.

    FECN (Forward-Explicit Congestion Notification) ist ein Feld mit einem einzigen Bit, das von einem Switch auf 1 gesetzt werden kann, um dem End-DTE-Gerät (z.B. einem Router) anzuzeigen, daß es zu einer Überlastung auf der Übertragungsstrecke von der Quelle zum Ziel kam. Der primäre Vorteil der FECN- und BECN-

Felder ist die Fähigkeit von Protokollen höherer Schichten, auf diese Indikatoren intelligent zu reagieren. Zu den einzigen Protokollen höherer Schichten, in denen diese Funktionen implementiert sind, gehören DECnet und OSI.

BECN (Backward-Explicit Congestion Notification) ist ein Feld mit einem einzigen Bit, das von einem Switch auf 1 gesetzt werden kann, um anzuzeigen, daß es im Netzwerk zu einer Überlastung kam, und zwar in entgegengesetzter Richtung zur Übertragungsrichtung des Frame (von der Quelle zum Ziel).

Das DE-Bit (Discard Eligibility) wird von einem DTE-Gerät gesetzt, z.B. einem Router, um den Frame als weniger wichtig zu kennzeichnen. So gekennzeichnete Frames werden in einem überlasteten Netzwerk ausgeschieden. Damit ist eine grundlegende Priorisierung in einem Frame-Relay-Netzwerk möglich.

– *Daten* – Enthält die gekapselten Daten höherer Schichten. Jeder Frame in diesem Feld variabler Länge enthält ein Benutzerdaten- bzw. Nutzlast-Feld. Die Länge kann bis zu 16000 Oktetts betragen. Dieses Feld dient dazu, das Protokoll-Paket höherer Schichten (PDU) durch ein Frame-Relay-Netzwerk zu transportieren.

– *Frame-Prüfsumme (FCS)* – Stellt die Integrität der übertragenen Daten sicher. Dieser Wert wurde vom Quellgerät berechnet und wird vom Empfänger überprüft, um die Integrität der Übertragung sicherzustellen.

10.7.2 LMI-Frame-Format

Frame-Relay-Frames, die der LMI-Spezifikation entsprechen, bestehen aus den in Bild 10.5 gezeigten Feldern.

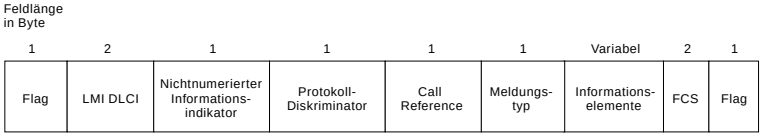


Bild 10.5: Ein Frame, der dem LMI-Format entspricht, besteht aus neun Feldern

Die folgenden Beschreibungen geben einen kurzen Überblick zu diesen Feldern.

- *Flag* – Begrenzt Anfang und Ende eines Frame.
- *LMI DLCI* – Kennzeichnet den Frame als LMI-Frame. Der LMI-spezifische DLCI-Wert, der vom LMI-Konsortium festgelegt wurde, lautet DLCI = 1023.
- *Nicht numerierter Informationsindikator* – Setzt das Poll-/Final-Bit auf Null.
- *Protokoll-Diskriminator* – Enthält immer einen Wert, der anzeigt, daß es sich um einen LMI-Frame handelt.
- *Call Reference* – Dieses Feld enthält nur Nullen. Es ist für zukünftige Verwendung reserviert.
- *Meldungstyp* – Markiert den Frame mit einem der folgenden Meldungstypen:
  - Statusabfrage-Meldung: Erlaubt es einem Anwendergerät, den Status des Netzwerks abzufragen.
  - Status-Meldung: Antwort auf die Statusabfrage-Meldung. Status-Meldungen schließen Keep-Alives und PVC-Status-Meldungen ein.
- *Informations-Elemente* – Enthalten eine unterschiedliche Anzahl einzelner Informations-Elemente (IEs). Diese setzen sich aus folgenden Feldern zusammen:
  - IE-Kennzeichnung: Kennzeichnet das IE eindeutig.
  - IE-Länge: Gibt die Länge des IE an.
  - Daten: Besteht aus einem oder mehr Byte mit gekapselten Daten höherer Schichten.
- *Frame-Prüfsumme (FCS)* – Stellt die Integrität übertragener Daten sicher.

KAPITEL 11

High-Speed Serial Interface

11.1 Hintergrund

Das *High-Speed Serial Interface* (HSSI) ist eine DTE/DCE-Schnittstelle, die von Cisco Systems und T3plus Networking entwickelt wurde, um den Anforderungen nach schneller Kommunikation über WAN-Verbindungen nachzukommmen. Die HSSI-Spezifikation steht jedermann/-frau zur Verfügung.

HSSI wird vom American National Standards Institute (ANSI) und dem Electronic Industries Association (EIA)/TIA TR30.2 Komitee formal standardisiert. Kürzlich wurde HSSI vom International Telecommunication Union Telecommunication Standardization Sector (ITU-T) (früher das Consultative Committee for International Telegraph and Telephone [CCITT]) und der International Organization for Standardization (ISO) angenommen und wird wahrscheinlich von diesen Einrichtungen standardisiert.

11.2 Grundlagen zum HSSI

HSSI definiert sowohl die elektrischen als auch die physischen DTE/DCE-Schnittstellen. Deshalb entspricht es der physischen Schicht des OSI-Referenzmodells. Die technischen Daten sind in der folgenden Tabelle 11.1 zusammengestellt.

Tabelle 11.1: Technische Daten zum HSSI

Eigenschaft	Wert
Übertragungsrate, max.	52 MBit/s
Kabellänge, max.	15,24 m
Stecker-Pole	50
Schnittstelle	DTE-DCE
Elektrik	Differential ECL
Leistungsaufnahme	610 mW
Topologie	Punkt-zu-Punkt
Kabeltyp	STP

Die maximale Übertragungsrate für HSSI beträgt 52 Mbit/s. Damit ist HSSI in der Lage, sowohl die T3-Geschwindigkeit (45 Mbit/s) vieler heutiger schneller WAN-Technologien zu bedienen als auch die Geschwindigkeit nach *Office Channel -1* (OC-1) mit 52 Mbit/s in der *Synchronous Digital Hierarchy* (SDH). Außerdem kann HSSI auf einfache Weise schnelle Verbindungen zwischen LANs aufbauen, z.B. bei Token Ring oder Ethernet.

Der Einsatz differentieller *Emitter-Coupled Logic* (ECL) verhilft HSSI dazu, hohe Datenraten und niedriges Rauschen zu erreichen. ECL wird in Cray-Schnittstellen seit Jahren verwendet und ist ebenfalls vom ANSI als Kommunikationsstandard *High-Performance Parallel Interface* (HIPPI) für die Kommunikation von Supercomputern mit LAN spezifiziert. ECL ist eine Standard-Technik, die hervorragendes Retiming auf dem Receiver ermöglicht, so daß zuverlässige Latenzzeiten erreicht werden.

HSSI verwendet einen sehr kleinen FCC-genehmigten 50poligen Stecker, der kleiner als sein V.35-Gegenstück ist. Um nicht ständig mit Adaptern hantieren zu müssen, sind die HSSI-Kabel als Stecker ausgelegt. HSSI verwendet die gleiche Anzahl Pins und Leitungen wie das *Small Computer Systems Interface 2* (SCSI-2)-Kabel, jedoch ist beim HSSI die elektrische Spezifikation enger gefaßt.

11.3 HSSI-Betrieb

Die Flexibilität des HSSI-Takt- und Dtaenübertragungs-Protokolls ermöglicht die Zuordnung von Benutzer- (oder Lieferanten-)Bandbreiten. Das DCE steuert den Takt, indem es die

Frequenz erhöht oder verringert. Auf diese Weise kann das DCE einzelnen Anwendungen eine Bandbreite zuordnen. Ein Nebenstellenanlage z.B. benötigt eine gewisse Bandbreite, ebenso ein Router und ein Channel Extender. Die Zuordnung von Bandbreiten ist der Schlüssel, der T3 und andere Breitband-Services erschwinglich und beliebt gemacht hat.

HSSI geht von einer Peer-to-Peer-Intelligenz beim DCE und DTE aus. Das Steuerungs-Protokoll wurde vereinfacht, so daß es nur zwei Steuersignale erfordert (»DTE verfügbar« und »DCE verfügbar«). Beide Signale müssen eingehen, bevor die Datenverbindung verfügbar ist. Bei den DCE und DTE wird davon ausgegangen, daß sie in der Lage sind, die Netzwerke hinter ihren Schnittstellen zu verwalten. Durch die Reduzierung der Steuersignale wird die Zuverlässigkeit der Verbindung verbessert, da so die Anzahl der möglichen fehlerhaften Verbindungen verringert wird.

11.3.1 Rückkopplungstests

HSSI bietet vier Rückkopplungstest an, die in Bild 11.1 dargestellt sind. Im ersten Test wird das lokale Kabel geprüft, indem das Signal, nachdem es den DTE-Port erreicht hat, zurückgeleitet wird. Beim zweiten Test läuft das Signal bis zum Leitungs-Port des lokalen DCE. Beim dritten Test läuft das Signal bis zum Leitungs-Port des fernen DCE. Der vierte Test schließlich ist ein vom DCE initiierter Test des DCE-Port am DTE.

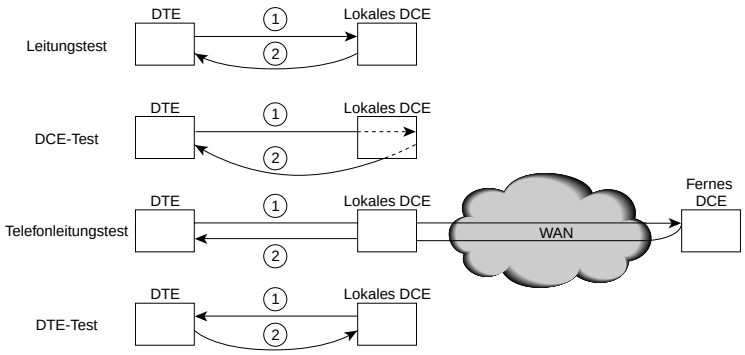


Bild 11.1: HSSI unterstützt vier Rückkopplungstests

KAPITEL 12

Integrated Services Digital Network (ISDN)

12.1 Hintergrund

Integrated Services Digital Network (ISDN – diensteintegrierendes digitales Telekommunikationsnetz) umfaßt digitales Telefonieren und digitale Datenübertragung, das von Telefongesellschaften angeboten wird. ISDN ist die Digitalisierung des Telekommunikationsnetz, das die Übertragung von Sprache, Daten, Texten, Grafiken, Musik, Video usw. über vorhandene Telefonleitungen ermöglicht. Die Notwendigkeit von ISDN zeigen die Bemühungen um Standardisierung der Teilnehmer-Services, Benutzer-/Netzwerk-Schnittstellen und der Netzwerk- und Internetwork-Fähigkeiten. Zu den ISDN-Anwendungen gehören schnelle bildgebende Anwendungen (z.B. Gruppe IV-Fax), zusätzliche Telefonleitungen bei den Teilnehmern (für die Telecommuting-Industrie), schnelle Dateiübertragung und Videokonferenzen. Die Sprachübermittlung zählt auch zu den Anwendungen des ISDN. In diesem Kapitel werden die grundlegenden Technologien und Dienste des ISDN erläutert.

12.2 ISDN-Komponenten

Die ISDN-Komponenten umfassen Endgeräte, Terminal-Adapter (TAs), Netzwerk-Abschlußgeräte, Leitungsabschlußgeräte und Vermittlungsabschlußgeräte. Bei den ISDN-Endgeräten lassen sich zwei Typen unterscheiden. Spezielle ISDN-Endgeräte werden als *Endgeräte vom Typ 1* (terminal equipment type 1 – TE1) bezeichnet. Nicht-ISDN-Engeräte, wie DTEs,

die es bereits vor der Entwicklung von ISDN gab, werden als *Endgeräte vom Typ 2* (terminal equipment type 2 – TE2) bezeichnet. TE1-Geräte sind mit dem ISDN-Netz über einen vierdrahtigen, digitalen Twisted-Pair-Anschluß verbunden. TE2-Geräte werden an das ISDN über einen Terminal-Adapter (TA) angeschlossen. Dabei kann es sich um ein einzelnes Gerät oder eine eingebaute Karte im TE2-Gerät handeln. Wenn im TE2 kein TA eingebaut ist, erfolgt die Verbindung über eine Standardschnittstelle der physischen Schicht, wie EIA/TIA-232-C (früher RS-232-C), V.24 und V.35.

Der nächste Anschlußpunkt im ISDN-Netzwerk nach den TE1- und TE2-Geräten sind die Geräte des Netzwerk-Abschluß Typ 1 (network termination type 1 – NT1) oder Netzwerk-Abschluß Typ 2 (network termination type 2 – NT2). Mit diesen Geräten erfolgt der Anschluß der vierdrahtigen Teilnehmerverkabelung an die zweidrahtige lokale Verkabelung. In Nordamerika wird der NT1 als *kundeneigenes Gerät* (customer premises equipment – CPE) bezeichnet. In den meisten Ländern ist der NT1 jedoch Teil des vom Betreiber bereitgestellten Netzwerks. Der NT2 ist ein komplizierteres Gerät, das in digitalen Nebenstellenanlagen eingesetzt wird und Funktionen der Schicht-2- und -3-Protokolle und Konzentrator-Funktionen übernimmt. NT1 und NT2 können auch in einem Gerät, dem NT1/2, vereint werden.

ISDN spezifiziert mehrere Referenzpunkte, die logische Schnittstellen zwischen funktionalen Gruppierungen definieren, z.B. zwischen TAs und NT1. Es gibt folgende Referenzpunkte:

- *R* – Der Referenzpunkt zwischen Nicht-ISDN-Geräten und einem TA.
- *S* – Der Referenzpunkt zwischen Benutzer-Endgeräten und einem NT2.
- *T* – Der Referenzpunkt zwischen NT1- und NT2-Geräten.
- *U* – Der Referenzpunkt zwischen NT1-Geräten und Leitungsabschlußgeräten im Betreiber-Netzwerk. Der U-Referenzpunkt ist nur in Noramerika von Bedeutung, da dort der NT1 nicht vom Netzwerk-Betreiber zur Verfügung gestellt wird.

Bild 12.1 zeigt eine Beispiel-Konfiguration für ISDN mit drei Geräten, die an einen ISDN-Switch in der Zentrale angeschlossen sind. Zwei dieser Geräte sind ISDN-kompatibel, so daß sie über einen S-Referenzpunkt an ein NT2-Gerät angeschlossen werden können. Das dritte Gerät (ein normales, nicht für ISDN ausgerüstetes Telefon) wird über den R-Referenzpunkt an einen TA angeschlossen. Alle diese Geräte könnten an ein NT1/2-Gerät angeschlossen werden, das die beiden NT1- und NT2-Geräte ersetzen würde. Weitere Geräte könnten an den rechts gezeigten ISDN-Switch angeschlossen werden.

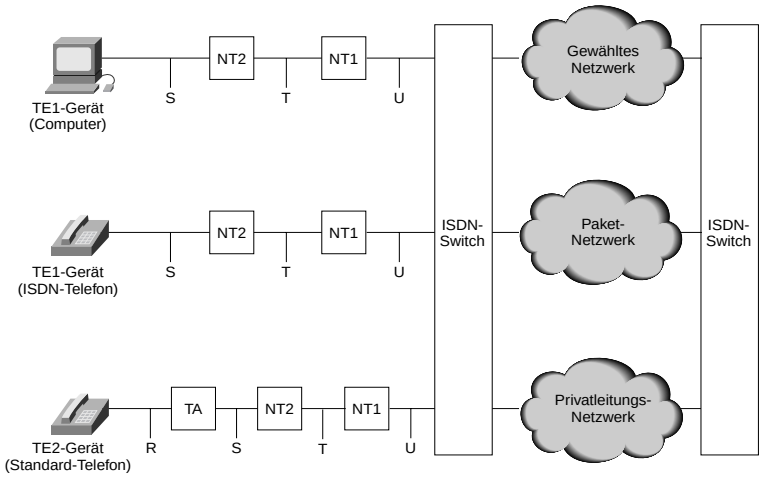


Bild 12.1: Die Beispiel-Konfiguration für ISDN zeigt die Beziehungen zwischen Geräten und Referenzpunkten

12.3 Dienste

Der ISDN-Basisanschluß (*Basic Rate Interface* – BRI) bietet zwei *B-Kanäle* und einen *D-Kanal* (*2B+D*). Der BRI *B-Kanal-Service* arbeitet mit einer Übertragungsrate von 64 Kbit/s und überträgt die Benutzerdaten; der D-Kanal-Service arbeitet mit einer Übertragungsrate von 16 Kbit/s und überträgt Steuer- und Signalisierungsdaten, kann aber auch unter gewissen Umständen für Benutzerdaten verwendet werden. Das Signalisierungs-Protokoll für den D-Kanal umfaßt die Schichten 1 bis 3 des OSI-Referenzmodells. BRI bietet auch Framing-Steuerung und anderen Overhead, wobei die Gesamtübertragungsrate auf 192 Kbit/s steigt. Die Spezifikation für die physische Schicht des BRI ist der *International Telecommunication*

Union Telecommunication Standardization Sector (ITU-T) (früher *Consultative Committee for International Telegraph and Telephone* [CCITT]) I.430.

ISDN *Primary Rate Interface* (PRI) bietet 23 B-Kanäle und einen D-Kanal in Nordamerika und Japan mit einer Gesamtübertragungsrate von 1,544 Mbit/s (der PRI D-Kanal arbeitet mit 64 Kbit/s). ISDN PRI wird in Europa, Australien und anderen Ländern/Kontinenten mit 30 B-Kanälen und einem D-Kanal angeboten. Hier erreicht die Übertragungsrate 2,048 Mbit/s. Die Spezifikation für die physische Schicht des ISDN PRI ist die ITU-T I.431.

12.4 Schicht 1

Die Frame-Formate in der physischen Schicht des ISDN (Schicht 1) unterscheiden sich nach ausgehendem Frame (vom Endgerät zum Netzwerk) und eingehendem Frame (vom Netzwerk zum Endgerät). Beide Schnittstellen der physischen Schicht sind in Bild 12.2 dargestellt.

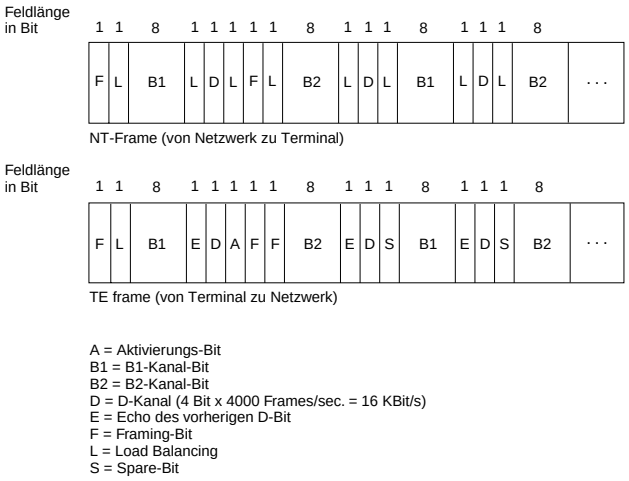


Bild 12.2: Die Frame-Formate der physischen ISDN-Schicht unterscheiden sich abhängig von ihrer Übertragungsrichtung

Die Frames sind 48 Bit lang, wovon 36 Bit Daten sind. Die Bits für den Frame der physischen ISDN-Schicht werden wie folgt verwendet:

– *F* – Dient der Synchronisation

– *L* – Reguliert den durchschnittlichen Bit-Wert

- *E* – Stellt die Konfliktlösung sicher, wenn mehrere Endgeräte von einem passiven Bus gleichzeitig eine Kanal anfordern.
- *A* – Aktiviert Geräte
- *S* – Nicht belegt
- B1, B2 und D – Übertragen Benutzerdaten

Mehrere ISDN-Benutzergeräte können physisch an eine Leitung angeschlossen werden. Dabei kann es zu Kollisionen kommen, weshalb ISDN Funktionen bietet, die Verbindungskonflikte feststellen. Wenn ein NT ein D-Bit von einem TE empfängt, liefert es dieses Bit an der nächsten E-Bit-Position zurück. Das TE-Gerät erwartet, daß das nächste E-Bit mit dem von ihm zuletzt übertragenen D-Bit identisch ist.

Endgeräte können erst dann auf dem D-Kanal Daten übertragen, wenn sie zuvor eine bestimmte Anzahl Einsen empfangen haben (was so viel bedeutet wie »Kein Signal«), entsprechend einer voreingestellten Priorität. Wenn das TE-Gerät auf dem Echo-(E-)Kanal ein anderes Bit als sein D-Bit empfängt, muß es die Übertragung sofort unterbrechen. Mit dieser einfachen Technik ist sichergestellt, daß immer nur ein Endgerät seine D-Nachricht überträgt. Nachdem die D-Nachricht erfolgreich übertragen wurde, senkt das Endgerät seine Priorität, indem es mehrere aufeinanderfolgende Einsen erwartet, bevor es mit der Übertragung beginnt. Endgeräte können ihre Priorität erst dann erhöhen, nachdem alle anderen Geräte, die die gleiche Leitung benutzen, eine D-Nachricht senden konnten. Telefonverbindungen haben eine höhere Priorität als andere Dienste, und Signalisierungsinformationen haben eine höhere Priorität als andere Informationen.

12.5 Schicht 2

Die Schicht 2 des ISDN-Signalisierungs-Protokolls ist die *Link Access Procedure, D-Kanal (LAPD)*. LAPD ähnelt der *High-Level Data-Link Control* (HDLC) und *Link Access Procedure, Balanced* (LAPB). (Weitere Informationen zu diesen Protokollen finden Sie in den Kapiteln 16, »Synchronous Data-Link Control und Derivate« und 17, »X.25«.) Wie das ausgeschriebene Akronym LAPD sagt, wird diese Schicht vom D-

Kanal benutzt, um sicherzustellen, daß Steuer- und Signalisierungs-Informationen ungehindert übertragen und unbeschädigt empfangen werden können. Das Frame-Format des LAPD (siehe Bild 12.3) lehnt sich stark an das von HDLC an und verwendet, wie auch HDLC, *Steuer-, Informations-* und *nichtnumerierte* Frames. Die formale Spezifikation des LAPD-Protokolls ist festgelegt in ITU-T Q.920 und ITU-T Q.921.

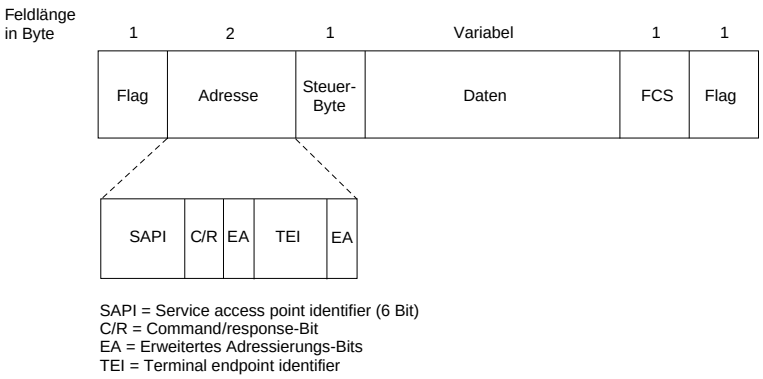


Bild 12.3: Das Frame-Format von LAPD ist angelehnt an HDLC und LAPB

Die *Flag-* und *Steuer-*Felder in LAPD sind identisch mit denen in HDLC. Das *Adreßfeld* in LADP kann 1 oder 2 Byte lang sein. Wenn das Bit für die erweiterte Adresse im ersten Byte gesetzt ist, ist die Adresse nur 1 Byte lang; ist das Bit nicht gesetzt, ist die Adresse 2 Byte lang. Das erste Byte im Adreßfeld enthält den *Service Access Point Identifier* (SAPI – Dienstzugangspunkt), der den Zugangspunkt bezeichnet, an dem LAPD-Dienste für die Schicht 3 angeboten werden. Das C/R-Bit gibt an, ob der Frame einen Befehl oder eine Antwort enthält. Das Feld *Terminal End-Point Identifier* (TEI) bezeichnet ein einzelnes oder mehrere Endgeräte. Ein TEI, der nur Einsen enthält, ist ein Broadcast.

12.6 Schicht 3

Für die ISDN-Signalisierung werden zwei Schicht-3-Spezifikationen verwendet: ITU-T (früher CCITT) I.450 (auch bekannt unter ITU-T Q.930) und ITU-T I.451 (auch bekannt unter ITU-T Q.931). Alle diese Protokolle zusammen unterstützen Benutzer-zu-Benutzer-, verbindungsvermittelte und paketvermittelte Verbindungen. Eine Vielzahl von Anruf-Aufbau-,

Anruf-Beendigungs-, Informations- und verschiedene Nachrichten sind spezifiziert, einschließlich SETUP, CONNECT, RELEASE, USER INFORMATION, CANCEL, STATUS und DISCONNECT. Diese Nachrichten entsprechen funktional denen des X.25-Protokolls (weitere Informationen dazu finden Sie in Kapitel 17). Bild 12.4, das der Spezifikation ITU-T I.451 entnommen wurde, zeigt die typischen Zustände eines verbindungsvermittelten ISDN-Anrufs.

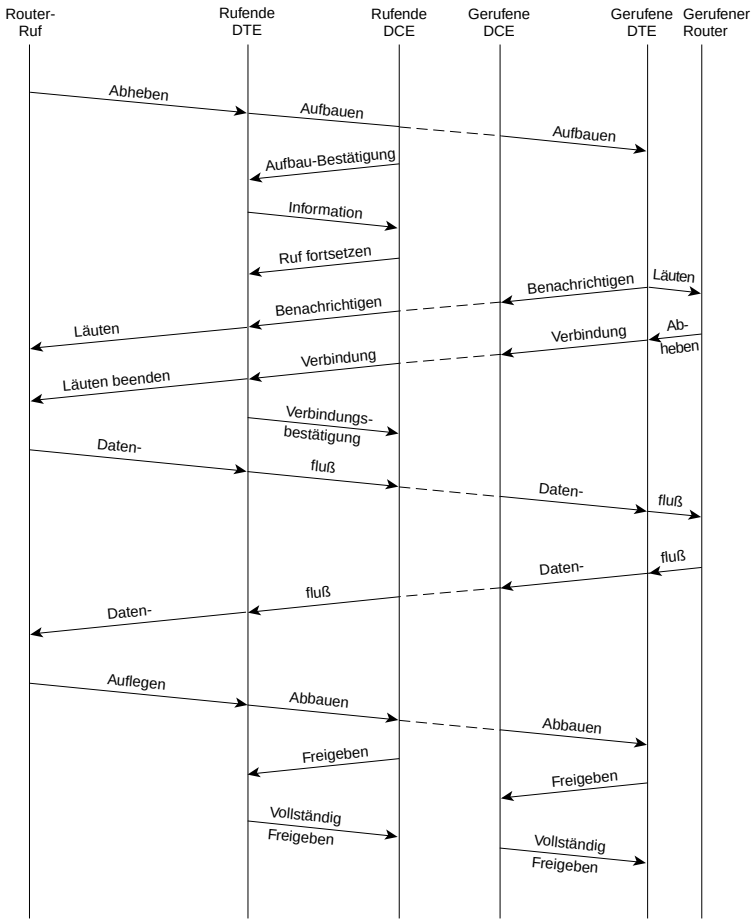


Bild 12.4: Ein verbindungsvermittelter ISDN-Anruf durchläuft verschiedene Zustände bis zu seinem Ziel

KAPITEL 13

Point-to-Point Protocol

13.1 Background

Das Point-to-Point Protocol (PPP – Punkt-zu-Punkt-Protokoll) kam ursprünglich als Kapselungs-Protokoll für die Übertragung von IP-Datenverkehr über Punkt-zu-Punkt-Verbindungen auf. Mit PPP wurde ein Standard gesetzt für die Zuordnung und Verwaltung von IP-Adressen, asynchrone (Start/Stop) und bit-orientierte synchrone Kapselung, das Netzwerk-Protokoll-Multiplexing, die Verbindungskonfiguration, das Testen der Verbindungsqualität, die Fehlererkennung und die Aushandlung von Optionen für z.B. Adressen in der Vermittlungsschicht und Datenkompression. PPP unterstützt diese Funktionen, indem es das erweiterbare *Link Control Protocol* (LCP) und eine Familie von *Network Control Protocols* (NCPs) zur Verfügung stellt, mit denen optionale Konfigurationsparameter und -einrichtungen ausgehandelt werden können. Außer IP unterstützt PPP weitere Protokolle, einschließlich Novells Internetwork Packet Exchange (IPX) und DECnet. Dieses Kapitel gibt einen Überblick über die grundlegenden Protokollelemente und die Funktionen von PPP.

13.2 PPP-Komponenten

PPP bietet eine Methode zur Übertragung von Datagrammen über serielle Punkt-zu-Punkt-Verbindungen. PPP umfaßt drei Hauptkomponenten:

- Eine Methode zur Kapselung von Datagrammen über serielle Verbindungen – PPP verwendet das Protokoll *High-Level Data-Link Control* (HDLC) als Basis für die Kapselung von Datagrammen über Punkt-zu-Punkt-Verbindungen (weitere Informationen zu HDLC finden Sie in Kapitel 16, »Synchronous Data-Link Control und Derivate«).
- Ein erweiterbares LCP, um Datenverbindungen aufzubauen, zu konfigurieren und zu testen.
- Eine Familie von NCPs, um verschiedene Protokolle der Vermittlungsschicht zu etablieren und zu konfigurieren – PPP wurde dazu entwickelt, die gleichzeitige Nutzung mehrerer Protokolle der Vermittlungsschicht zu ermöglichen.

13.3 Das Verfahren

Um die Kommunikation über eine Punkt-zu-Punkt-Verbindung zu etablieren, sendet das Ursprungs-PPP zuerst LCP-Frames, um die Datenverbindung zu konfigurieren und zu testen (optional). Nachdem die Verbindung aufgebaut und optionale Einrichtungen entsprechend den Anforderungen der LCP ausgehandelt wurden, sendet das Ursprungs-PPP NCP-Frames, mit denen ein oder mehrere Protokolle der Vermittlungsschicht ausgewählt und konfiguriert werden. Wenn jedes der ausgewählten Protokolle der Vermittlungsschicht konfiguriert ist, können Pakete von jedem Vermittlungsschicht-Protokoll über die Verbindung gesendet werden. Die Verbindung bleibt so lange erhalten, bis sie von LCP- oder NCP-Frames explizit geschlossen wird oder wenn ein externes Ereignis dies veranlaßt (z.B. wenn ein Timer abgelaufen ist oder ein Benutzer eingreift).

13.4 Anforderungen der physischen Schicht

PPP kann über jede DTE/DCE-Schnittstelle betrieben werden. Dazu gehören z.B. EIA/TIA-232-C (früher RS-232-C), EIA/TIA-422 (früher RS-422), EIA/TIA-423 (früher RS-423) und International Telecommunication Union Telecommunication Standardization Sector (ITU-T) (früher CCITT) V.35. Das einzige absolute Muß für PPP ist eine Duplex-Verbindung: eine dedizierte oder geswitchte, die entweder im asynchronen

oder synchronen bit-seriellen Modus betrieben werden kann und die für PPP-Frames der Verbindungsschicht transparent ist. PPP gibt keine Beschränkungen hinsichtlich der Übertragungsrate vor, außer denen, die sich von den einzelnen benutzten DTE/DCE-Schnittstellen herleiten.

13.5 PPP-Verbindungsschicht

PPP verwendet die Prinzipien, Terminologie und Frame-Struktur wie sie vorgegeben sind in den HDLC-Prozeduren der International Organization for Standardization (ISO 3309-1979), modifiziert durch ISO 3309:1984/PDAD1 »Addendum 1: Start/stop transmission«. ISO 3309-1979 spezifiziert die HDLC-Frame-Struktur für den Einsatz in synchronen Umgebungen. ISO 3309:1984/PDAD1 spezifiziert beantragte Modifikationen zu ISO 3309-1979, um den Einsatz auch in asynchronen Umgebungen zuzulassen. Die PPP-Steuerprozeduren verwenden die Kodierung von Definitionen und Steuerfeldern entsprechend ISO 4335-1979 und ISO 4335-1979/Addendum 1-1979. Das PPP-Frame-Format zeigt Bild 13.1.

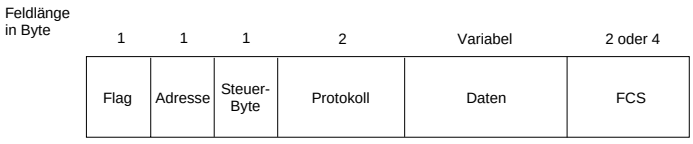


Bild 13.1: Der PPP-Frame besteht aus sechs Feldern

Im folgenden werden die PPP-Frame-Felder, wie in Bild 13.1 dargestellt, kurz beschrieben:

- *Flag* – Ein einzelnes Byte, das Anfang und Ende des Frame kennzeichnet. Dieses Feld enthält die Bit-Folge 01111110.
- *Adresse* – Ein einzelnes Byte mit der Bit-Folge 11111111, die die Standard-Broadcast-Adresse darstellt. PPP weist keine individuellen Stations-Adressen zu.
- *Steuer-Bit* – Ein einzelnes Byte mit der Bit-Folge 00000011, das zur Übertragung von Benutzerdaten in einem Frame außerhalb der Reihenfolge auffordert. Es wird ein verbindungsloser Link-Service angeboten, der dem der Logical Link Control (LLC) Type 1 entspricht (weitere Informationen zu LLC-Typen und Frame-Typen finden Sie in Kapitel 16, »Synchronous Data-Link Control und Derivate«).

- *Protokoll* – Zwei Byte, die das im Daten-Feld des Frame gekapselte Protokoll bezeichnen. Die aktuellen Werte für das Protokollfeld sind in den neuesten *Assigned Numbers Request for Comments* (RFC) definiert.
- *Daten* – Kein oder mehrere Byte, die das Datagramm für das im Protokollfeld spezifizierte Protokoll enthalten. Das Ende des Datenfelds wird ermittelt über das End-Flag plus 2 Byte für das FCS-Feld. Die standardmäßige maximale Länge des Datenfelds beträgt 1500 Byte. Aufgrund früherer Vereinbarungen kann bei zustimmenden PPP-Implementationen die maximale Länge dieses Felds von diesem Wert abweichen.
- *Frame-Prüfsequenz* (FCS) – Normalerweise 16 Bit (2 Byte). Aufgrund früherer Vereinbarung kann bei zustimmenden PPP-Implementationen ein 32 Bit (4 Byte) langes FCS-Feld für genauere Fehlererkennung verwendet werden.

Das LCP kann Modifikationen an der Standardstruktur des PPP-Frame aushandeln. Modifizierte Frames sind jedoch von Standard-Frames eindeutig zu unterscheiden.

13.5.1 PPP-Verbindungssteuerungs-Protokoll

Das Verbindungssteuerungs-Protokoll (Link-Control Protocol – LCP) des PPP stellt eine Methode für die Etablierung, Konfigurierung, Verwaltung und Beendigung einer Punkt-zu-Punkt-Verbindung zur Verfügung. LCP durchläuft vier einzelne Phasen:

In der ersten Phase erfolgt der Verbindungsaufbau, und die Konfiguration wird ausgehandelt. Bevor irgendwelche Datagramme der Vermittlungsschicht (z.B. IP) ausgetauscht werden können, muß LCP die Verbindung eröffnet und Konfigurationsparameter ausgehandelt haben. Diese Phase ist beendet, wenn sowohl ein Frame gesendet als auch einer empfangen wurde, der die Konfigurationsbestätigung enthält.

Dann wird die Verbindungsqualität ermittelt. Diese Phase ist optional. In dieser Phase wird die Verbindung getestet, um festzustellen, ob die Qualität dafür ausreicht, daß die Protokolle der Vermittlungsschicht gestartet werden können. LCP kann die Übertragung von Protokoll-Informationen der Ver-

mittlungsschicht so lange hinauszögern, bis diese Phase beendet ist.

Zu diesem Zeitpunkt erfolgt die Konfigurationsaushandlung des Vermittlungschicht-Protokolls. Nachdem LCP die Phase der Qualitätsprüfung beendet hat, können die Vermittlungsschicht-Protokolle vom entsprechenden NCP einzeln konfiguriert werden und jederzeit gestartet oder beendet werden. Wenn LCP die Verbindung beendet, informiert es die Protokolle der Vermittlungsschicht, damit diese entsprechende Schritte einleiten.

Zuletzt erfolgt die Verbindungsbeendigung. LCP kann die Verbindung jederzeit beenden. Eine Verbindung wird im allgemeinen auf Anforderung eines Benutzers beendet, kann aber auch aufgrund eines physischen Ereignisses beendet werden, z.B. wenn das Trägersignal verloren geht oder ein Timer abläuft.

Es gibt drei Klassen von LCP-Frames. Verbindungsaufbau-Frames dienen dazu, eine Verbindung aufzubauen und zu konfigurieren. Verbindungsbeendigungs-Frames dienen dazu, eine Verbindung zu beenden, während Verbindungverwaltungs-Frames eine Verbindung verwalten und debuggen.

Diese Frames werden für die Durchführung der einzelnen LCP-Phasen benötigt.

KAPITEL 14

Switched Multimegabit Data Service (SMDS)

14.1 Hintergrund

Switched Multimegabit Data Service (SMDS) ist eine schnelle, paketvermittelte, datagrammbasierende WAN-Technologie, die für die Kommunikation über öffentliche Datennetze eingesetzt wird. SMDS kann über faser- oder kupferbasierte Medien betrieben werden und unterstützt eine Übertragungsgeschwindigkeit von 1,544 Mbit/s bei Übertragungseinrichtungen der Digitalen Signalstufe 1 (DS-1) oder 44,736 Mbit/s bei Übertragungseinrichtungen der Digitalen Signalstufe 3 (DS-3). Außerdem sind SMDS-Dateneinheiten groß genug, um ganze Frames der Spezifikationen IEEE 802.3, IEEE 802.5 und Fiber-Distributed Data Interface (FDDI) zu kapseln. Dieses Kapitel gibt einen Überblick zu den operationalen Elementen der SMDS-Umgebung und behandelt das zugrundeliegende Protokoll. Außerdem werden entsprechende Technologien diskutiert, z.B. der Distributed Queue Dual Bus (DQDB). Das Kapitel schließt mit der Betrachtung der SMDS-Zugriffsklassen und -Zellformate.

14.2 SMDS-Netzwerk-Komponenten

SMDS-Netzwerke befähigen viele zugrundeliegende Einrichtungen, Hochgeschwindigkeits-Daten-Service anzubieten. Dazu gehören kundeneigene Geräte (*customer premises equipment – CPE), Betreiber-Geräte* und *Teilnehmer-Netzschnittstellen (subscriber network interface – SNI*). Zum CPE zählen Endgeräte,

die im Eigentum des Kunden sind und von diesem gewartet/verwaltet werden. Dies können Terminals, Personalcomputer und Zwischenknoten (z.B. Router, Modems und Multiplexer) sein. Zwischenknoten werden manchmal auch vom SMDS-Betreiber gestellt. Zu den Geräten des Betreibers gehören Hochgeschwindigkeits-WAN-Switches, die bestimmten Netzwerkspezifikationen entsprechen müssen, z.B. denen von Bell Communications Research (Bellcore). Diese Spezifikationen definieren Netzwerk-Operationen, die Schnittstelle zwischen dem örtlichen Netzwerk des Betreibers und dem Fernverkehrs-Netzwerk eines anderen Betreibers, und die Schnittstelle zwischen zwei Switches im Netzwerk eines einzelnen Betreibers.

Das SNI ist die Schnittstelle zwischen CPE und Betreiber-Geräten. An dieser Schnittstelle endet das Kunden-Netzwerk, und das Betreiber-Netzwerk beginnt. Das SNI hat die Funktion, die Technolgie und Arbeitsweise des SMDS-Betreiber-Netzwerks dem Kunden gegenüber transparent zu halten. Bild 14.1 zeigt den Zusammenhang zwischen diesen drei Komponenten eines SMDS-Netzwerks.

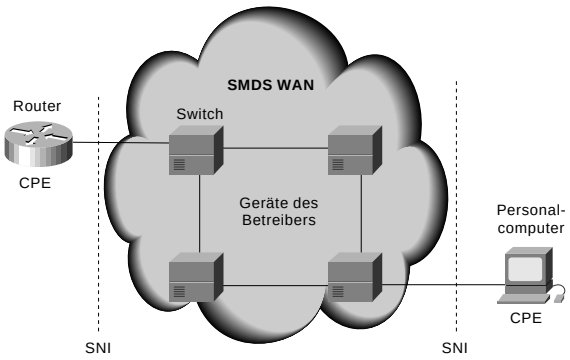


Bild 14.1: Das SNI bietet eine Schnittstelle zwischen CPE und Betreiber-Geräten in einem SMDS-Netz

14.3 SMDS Interface Protokoll (SIP)

Das SMDS-Interface-Protokoll (SIP) dient zur Kommunikation zwischen CPE und SMDS-Betreiber-Geräten. SIP bietet verbindungslosen Service über das *Subscriber-Network Interface* (SNI), so daß die CPE-Geräte auf das SMDS-Netzwerk zugreifen können. SIP basiert auf dem Standard IEEE 802.6 *Distributed Queue Dual Bus (DQDB)* für Zell Relay über

Metropolitan-Area Networks (MANs). Der DQDB wurde als Basis für SIP gewählt, da es sich um einen offenen Standard handelt, der alle SMDS-Service-Funktionen unterstützt. Außerdem wurde der DQDB so entworfen, daß er mit den aktuellen Übertragungsstandards der Betreiber kompatibel ist. Des weiteren orientiert sich DQDB an zukünftigen Standards für Breitband-ISDN (BISDN), die es ermöglichen, mit Breitband-Video und -Sprachdiensten zusammenzuarbeiten. Bild 14.2 zeigt, an welchen Stellen im SMDS-Netzwerk SIP zum Einsatz kommt.

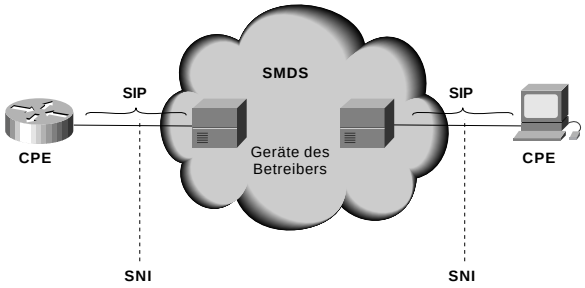


Bild 14.2: SIP bietet einen verbindungslosen Service zwischen CPE und Betreiber-Geräten

14.3.1 SIP-Stufen

SIP ist aus drei Stufen aufgebaut. *SIP-Stufe 3* läuft auf der MAC-Subschicht (Media-Access Control) der Sicherungsschicht. *SIP-Stufe 2* läuft auf der MAC-Subschicht der Sicherungsschicht. *SIP-Stufe 1* läuft auf der physikalischen Schicht des OSI-Referenzmodells. Bild 14.3 zeigt die Entsprechungen von SIP im OSI-Referenzmodell, einschließlich der IEEE-Verbindungs-Subschichten.

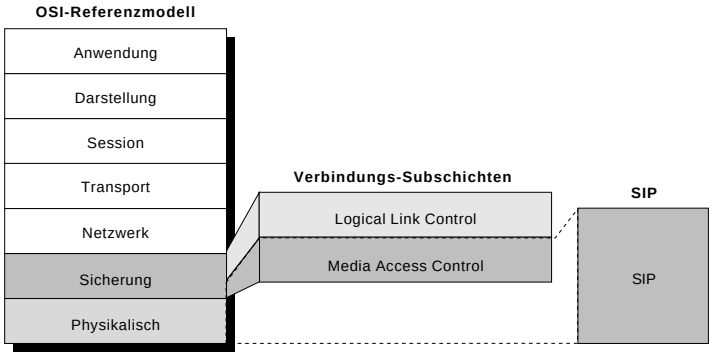


Bild 14.3: SIP bietet Services, die den Schichten (physikalische und Sicherungsschicht) des OSI-Referenzmodells zugeordnet werden können

SIP-Stufe 3 startet, wenn Benutzerdaten in Form von SMDS Service Data Units (SDUs) eingehen. SMDS SDUs werden dann vom Header und Trailer der SIP-Stufe 3 gekapselt. Der sich daraus ergebende Frame wird als Stufe-3-Protocol-Data-Unit (PDU) bezeichnet. PDUs der SIP-Stufe 3 werden dann an SIP-Stufe 2 übergeben.

SIP-Stufe 2 läuft auf der MAC-Subschicht (Media Access Control) der Sicherungsschicht. Diese Stufe startet, wenn SIP-Stufe-3-PDUs eingehen. Die PDUs werden in gleichgroße Stufe-2-PDUs (53-Oktetts) segmentiert, den sogenannten Zellen. Die Zellen werden an SIP-Stufe 1 übergeben, damit sie auf dem physischen Medium übertragen werden.

SIP-Stufe 1 läuft auf der physikalischen (oder physischen) Schicht und bietet das physische Verbindungsprotokoll, das mit Übertragungsraten von DS-1 oder DS-3 zwischen CPE-Geräten und dem Netzwerk arbeitet. SIP-Stufe 1 besteht aus dem Übertragungssystem und den Subschichten des Physical-Layer Convergency Protocol (PLCP). Die Transmission-System-Subschicht definiert die Eigenschaften und Methode für den Anschluß an eine DS-1- oder DS-3-Übertragungsverbindung. Das PLCP spezifiziert, wie Zellen der SIP-Stufe 2 in bezug zu DS-1- oder DS-3-Frames angeordnet werden sollen. PLCP definiert noch weitere Management-Informationen.

14.4 Distributed Queue Dual Bus (DQDB)

Der Distributed Queue Dual Bus (DQDB) ist ein Kommunikationsprotokoll der Sicherungsschicht, das für den Einsatz in Metropolitan-Area Networks (MANs) entworfen wurde. DQDB spezifiziert eine Netzwerk-Topologie, die sich aus zwei unidirektionalen logischen Bussen zusammensetzt, die mehrere Systeme miteinander verbinden. Definiert ist dieser Bus im Standard IEEE 802.6 DQDB.

Der Zugriffs-DQDB beschreibt genau die Funktionsweise des DQDB-Protokolls (in SMDS, SIP) über eine Benutzer/Netzwerk-Schnittstelle (in SMDS über SNI). Diese Funktionsweise unterscheidet sich von der des DQDB-Protokolls in jeder anderen Umgebung (z.B. zwischen Betreiber-Geräten innerhalb des SMDS PDN).

Der Zugriffs-DQDB setzt sich zusammmen aus den folgenden grundlegenden SMDS-Netzwerk-Komponenten:

– *Betreiber-Geräte* – Ein Switch im SMDS-Netzwerk wird als eine Station am Bus betrieben.

– *CPE* – Ein oder mehrere CPE-Geräte werden als Stationen am Bus betrieben.

– *SNI* – Das SNI fungiert als Schnittstelle zwischen CPE und Betreiber-Geräten.

Bild 14.4 zeigt einen einfachen Zugriffs-SQDB mit zwei CPE-Geräten und einem Switch (Betreiber-Gerät), die an einen doppelten Bus angeschlossen sind.

Ein SMDS-Zugriffs-DQDB wird normalerweise in einer Einzel-CPE- oder Multi-CPE-Konfiguration eingerichtet.

Ein Zugriffs-DQDB in einer Einzel-CPE-Konfiguration besteht aus einem Switch im Betreiber-SMDS-Netz und einer CPE-Station auf Teilnehmerseite. Einzel-CPE-DQDB-Konfigurationen ergeben ein zweiknotiges DQDB-Subnetzwerk. Die Kommunikation erfolgt nur zwischen dem Switch und dem einen CPE-Gerät über das SNI. Auf diesem Bus kommt es zu keinen Konflikten, da kein anderes CPE-Gerät darauf zugreifen kann.

Eine Multi-CPE-Konfiguration besteht aus einem Switch im Betreiber-SMDS-Netz und mehreren miteinander verbundenen CPE-Geräten auf Teilnehmerseite (die alle zum gleichen Teilnehmer gehören). In Multi-CPE-Konfigurationen ist die lokale Kommunikation zwischen CPE-Geräten möglich. Manchmal wird die lokale Kommunkation am Switch sichtbar, der das SNI bedient.

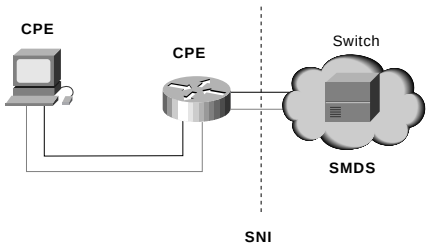


Bild 14.4: Ein einfacher Zugriffs-DQDB kann aus einem Endknoten, Router und einem Switch bestehen

Konflikte auf dem Bus aufgrund mehrerer Geräte erfordern den Einsatz von DQDM-Algorithmen für verteilte Warteschlangen, weshalb die Implementation einer Multi-CPE-Konfiguration komplizierter ist als die einer Einzel-CPE-Konfiguration.

14.5 SMDS-Zugriffsklassen

SMDS-Zugriffsklassen ermöglichen es SMDS-Netzwerken, sich einem weiten Bereich von Datenverkehrsanforderungen und Geräteeigenschaften anzupassen. Zugriffsklassen erzwingen von den CPE-Geräten eine dauerhafte oder durchschnittliche Übertragungsrate, indem sie eine maximal dauerhafte Datenübertragungsrate und einen maximalen Grad an Verkehrs-*Burstiness* etablieren (Burstiness meint in diesem Zusammenhang die Neigung eines Netzwerks, eine plötzliche Erhöhung der angeforderten Bandbreite zu erfahren). SMDS-Zugriffsklassen sind manchmal so implementiert, daß sie ein Haben-Management-Schema verwenden. In diesem Fall erzeugt und verfolgt ein Haben-Management-Algorithmus die Haben-Salden für jede Kunden-Schnittstelle. Wenn Pakete in das Netzwerk gesendet werden, verringert sich der Saldo. Neue Habenpunkte werden regelmäßig bis zu einem festgelegten Maximum zugewiesen. Ein Haben-Management wird nur von SMDS-Schnittstellen mit DS-3-Übertragungsraten, nicht mit DS-1-Raten eingesetzt.

Für Zugriffe mit DS-3-Übertragungsraten (entsprechend den dauerhaften Datenraten) gibt es fünf Zugriffsklassen. Unterstützt werden Datenraten von 4, 10, 16, 25 und 34 Mbit/s.

14.6 Überblick zur SMDS-Adressierung

Die Protocol Data Units (PDUs) von SMDS enthalten sowohl die Quell- als auch die Zieladresse. SMDS-Adressen sind zehnstellige Werte, die an konventionelle Telefonnummern erinnern.

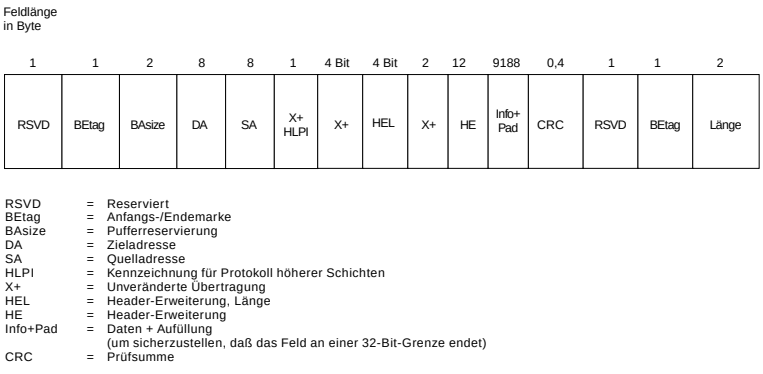
Die SMDS-Adressierung bietet Gruppen-Adressierung und Sicherheitsfunktionen.

Mit der SMDS-Gruppen-Adressierung können über eine einzelne Adresse mehrere CPE-Stationen angesprochen werden, die die Gruppenadresse im Ziel-Adreßfeld der PDU angeben. Das Netzwerk kopiert die PDU mehrfach und sendet die Kopien an alle Mitglieder der Gruppe. Die Gruppen-Adressierung reduziert die erfoderlichen Netzwerk-Ressourcen bei der Verteilung von Routing-Informationen, beim Auflösen von Adressen und dem dynamischen Erkennen von Netzwerk-Ressourcen. Die SMDS-Gruppen-Adressierung entspricht dem Multicasting im LAN.

SMDS implementiert zwei Sicherheitsfunktionen: Quelladreß-Überprüfung und Adreßüberwachung. Die *Quelladreß-Überprüfung* stellt sicher, daß die PDU-Quelladresse rechtmäßig der SNI zugeordnet ist, von der sie stammt. Quelladreß-Überprüfung beugt dem Adreß-Spoofing vor, bei dem ein unerwünschter Besucher die Quelladresse eines rechtmäßigen Geräts vorspiegelt. *Adreß-Überwachung* erlaubt es einem Teilnehmer, ein privates virtuelles Netzwerk aufzubauen, das unerwünschten Datenverkehr ausschließt. Wenn eine Adresse nicht zugelassen ist, wird die Dateneinheit nicht ausgeliefert.

14.7 SMDS-Referenz: SIP-Stufe-3-PDU-Format

Bild 14.5 zeigt das Format einer Protocol Data Unit (PDU) des SMDS-Interface-Protokolls (SIP) der Stufe 3.



RSVD = Reserviert
BEtag = Anfangs-/Endemarke
BAsize = Pufferreservierung
DA = Zieladresse
SA = Quelladresse
HLPI = Kennzeichnung für Protokoll höherer Schichten
X+ = Unveränderte Übertragung
HEL = Header-Erweiterung, Länge
HE = Header-Erweiterung
Info+Pad = Daten + Aufüllung
(um sicherzustellen, daß das Feld an einer 32-Bit-Grenze endet)
CRC = Prüfsumme

Bild 14.5: Eine Data Unit der SIP-Stufe 3 besteht aus 15 Feldern

Die folgende Beschreibung faßt die Funktion der PDU-Felder der SIP-Stufe 3, wie in Bild 14.5 gezeigt, zusammen:

- *X+* – Stellt sicher, daß das SIP-PDU-Format am DQDB-Format ausgerichtet wird. SMDS verarbeitet oder ändert die Werte dieser Felder nicht, die von Systemen verwendet werden können, die an das SMDS-Netz angeschlossen sind.
- *RSVD* – Besteht aus Nullen.
- *BEtag* – Bildet eine Zuordnung von erstem und letztem Segment der segmentierten SIP-Stufe-3-PDU. Beide Felder enthalten die gleichen Werte und werden dazu verwendet, festzustellen, ob das letzte Segment einer PDU und das erste Segment der folgenden PDU verloren gegangen sind. Dies hat den Empfang einer ungültigen PDU der Stufe 3 zur Folge.
- *BAsize* – Enthält die Pufferzuordnungsgröße.
- *Zieladresse (Destination Address – DA)* – Besteht aus zwei Teilen:
  - Adreßtyp: Belegt die vier höherwertigen Bit des Felds. Der Adreßtyp wird mit 1100 oder 1110 angegeben. Erstere bezeichnet eine 60 Bit lange Einzeladresse, letztere eine 60 Bit lange Gruppenadresse.
  - Adresse: Die Einzel- oder Gruppen-SMDS-Adresse des Ziels. Die SMDS-Adreßformate sind konsistent mit dem North American Numbering Plan (NANP).

    Die vier höherwertigen Bit des Zieladressen-Subfelds enthalten den Wert 0001 (der internationale Landesschlüssel für Nordamerika). Die folgenden 40 Bit enthalten den binärkodierten Wert der zehnstelligen SMDS-Adresse. Die letzten 16 (niederwertigen) Bit werden mit Einsen aufgefüllt.
- Quelladresse *(*Source Address – *SA)* – Besteht aus zwei Teilen:
  - Adreßtyp: Belegt die vier höherwertigen Bit des Felds. Das Feld des Quelladreßtyps kann nur eine Einzeladresse enthalten.

  - Adresse: Belegt die individuelle SMDS-Adresse der Quelle. Dieses Feld hat das gleiche Format wie das Adressen-Subfeld des Zieladressen-Felds.

- *Higher Layer Protocol Identifier (HLPI)* – Bezeichnet den Typ des im Datenfeld gekapselten Protokolls. Der Wert ist für SMDS uninteressant, kann aber von Systemen, die an das Netz angeschlossen sind, verwendet werden.
- *Header Extension Length (HEL)* – Gibt die Anzahl der 32-Bit-Wörter im Header-Extension-Feld (HE) an. Zur Zeit ist die Feldgröße für SMDS auf 12 Byte beschränkt (weshalb der HEL-Wert immer 0011 lautet).
- *Header Extension (HE)* – Enthält die SMDS-Versionsnummer. Dieses Feld übermittelt auch den Betreiber-Auswahlwert, der dazu dient, einen bestimmten zwischengeschalteten Betreiber zu wählen, der die SMDS-Daten von einem lokalen Betreibernetz zum anderen überträgt.
- *Daten- und Füll-Feld (Info + Pad)* – Enthält eine gekapselte SMDS Service Data Unit (SDU) und Füllbits, so daß das Feld immer an der 32-Bit-Grenze endet.
- *Cyclic Redundancy Check (CRC)* – Enthält einen Wert für die Fehlerprüfung.
- *Länge* – Gibt die Länge einer PDU an.

14.8 SMDS-Referenz: SIP-Stufe-2-Zellformat

Bild 14.6 zeigt das Zellformat für SMDS Interface Protocol (SIP) Stufe 2.

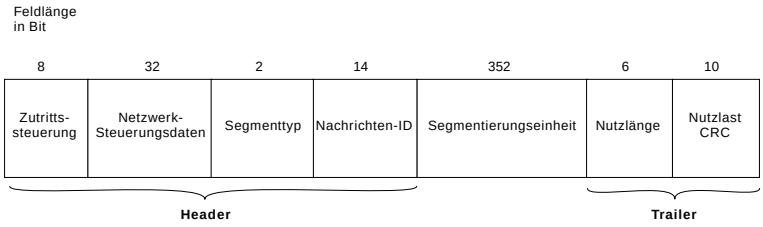


Bild 14.6: Eine Zelle der SMDS-SIP-Stufe 2 besteht aus sieben Feldern

Die folgenden Beschreibungen fassen die Funktionen der PDU-Felder der SIP-Stufe 2 zusammen, wie in Bild 14.6 gezeigt:

- *Zugriffssteuerung* – Enthält verschiedene Werte, die abhängig sind von der Übertragungsrichtung. Wenn die Zelle vom Switch zum CPE-Gerät gesendet wurde, ist nur die Angabe wichtig, ob die PDU der Stufe 3 Daten enthält. Wenn die Zelle vom CPE-Gerät an den Switch gesendet wurde und es sich um eine Multi-CPE-Konfiguration handelt, dann kann dieses Feld Anforderungs-Bits enthalten, die Senderechtanforderungen für Zellen auf dem Bus enthalten, die vom Switch zum CPE-Gerät übertragen werden.
- *Netzwerk-Steuerungsdaten* – Enthält einen Wert, der angibt, ob die PDU Daten enthält.
- *Segment-Typ* – Gibt an, ob die Zelle die erste, letzte oder eine dazwischenliegende Zelle in einer segmentierten Stufe-3-PDU ist. Es gibt vier Segment-Typwerte:
  - 00: Fortsetzung der Nachricht
  - 01: Ende der Nachricht
  - 10: Anfang der Nachricht
  - 11: Einsegmentige Nachricht
- *Nachrichten-ID* – Ordnet Stufe-2-Zellen einer Stufe-3-PDU zu. Die Nachrichten-ID ist für alle Segmente einer Stufe-3-PDU gleich. In einer Multi-CPE-Konfiguration müssen die Stufe-3-PDUs, die von verschiedenen CPE-Geräten stammen, eine unterschiedliche Nachrichten-ID aufweisen. Damit ist es dem SMDS-Netzwerk möglich, verschachtelte Zellen von verschiedenen Stufe-3-PDUs jeder Stufe-2-Zelle der korrekten Stufe-3-PDU zuzuordnen.
- *Segmentierungs-Einheit* – Enthält die Daten einer Zelle. Wenn die Stufe-2-Zelle leer ist, wird dieses Feld mit Nullen aufgefüllt.
- *Nutzlänge* – Gibt an, wieviel Byte einer Stufe-3-PDU im Segmentierungs-Einheiten-Feld tatsächlich enthalten sind. Wenn die Stufe-2-Zelle leer ist, wird es mit Nullen aufgefüllt.

- *Nutzlast Cyclic Redundancy Check (CRC)* – Enthält den CRC-Wert, der dazu verwendet wird, Fehler in den folgenden Feldern zu ermitteln:
  - Segment-Typ
  - Nachrichten-ID
  - Segmentierungs-Einheit
  - Nutzlänge
  - Nutzlast-CRC

Der Nutzlast-CRC-Wert betrifft nicht die Felder der Zugriffssteuerung oder Netzwerk-Steuerungsdaten.

KAPITEL 15

ADSL – Asymmetric Digital Subscriber Line

15.1 Hintergrund

Asymmetric Digital Subscriber Line (ADSL) ist eine Modem-Technologie, die auf vorhandenen Twisted-Pair-Telefonleitungen hoch-bandbreite Daten, z.B. Multimedia- oder Videodaten, zu den Service-Teilnehmern überträgt. Diese Technologie gehört zu einer größeren Familie, die insgesamt als *xDSL* bezeichnet wird. ADSL wird von Implementierern und Dienstanbietern mit besonderem Interesse betrachtet (einschließlich der anderen xDSL-Technologien), da diese Technologie verspricht, hoch-bandbreite Datenraten in einem stark dispersen Netz zu übertragen, so daß an der vorhandenen Telekommunikationsstruktur nur wenige Änderungen erforderlich sind. Ziel von ADSL ist es, eine Übertragungsrate von mehr als 6 Mbit/s je Teilnehmerleitung zu unterstützen. In diesem Kapitel wird ein Überblick zu den Eigenschaften und Funktionen von ADSL gegeben.

15.1.1 ADSL-Standardisierung

Die Arbeitsgruppe T1E1.4 des American National Standards Institute (ANSI) hat vor kurzem einen ADSL-Standard (ANSI Standard T1.413) herausgegeben, mit dem Übertragungsraten bis zu 6,1 Mbit/s möglich sind. Das European Technical Standards Institute (ETSI) fügte dem noch einen Anhang hinzu, um europäische Anforderungen zu berücksichtigen. T1.413 enthält zur Kundenseite eine Einzelterminal-Schnittstelle. Aus-

gabe II befindet sich unter der Bezeichnung T1E1.4 in Entwicklung und soll den Standard u.a. um eine multiplexende Schnittstelle auf der Kundenseite und Protokolle für die Konfiguration und das Netzwerk-Management erweitern.

Das ATM-Forum und DAVIC erkennen ADSL als ein Übertragungsprotokoll der physikalischen Schicht für UTP-Medien an.

Das ADSL-Forum wurde im Dezember 1994 gebildet, um das ADSL-Konzept zu fördern und die Entwicklung von ADSL-Systemarchitekturen, -Protokollen und -Schnittstellen für wichtige ADSL-Anwendungen zu unterstützen.

15.2 Überblick zur ADSL-Technologie

Eine ADSL-Verbindung verbindet zwei Modems über eine normale Telefonleitung (Twisted-Pair-Kabel) und erzeugt drei Datenkanäle: einen Hochgeschwindkeits-Downstream-Kanal, einen mittelschnellen Duplex-Kanal und einen POTS-Kanal (Plain Old Telephone Service – einfacher, alter Telefondienst). Der POTS-Kanal wird vom digitalen Modem über Filter abgetrennt, damit dieser Dienst immer verfügbar ist, auch wenn ADSL ausfällt. Der Hochgeschwindigkeits-Kanal arbeitet mit einer Übertragungsrate von 1,5 bis 6,1 Mbit/s, während der Duplex-Kanal 16 bis 640 Kbit/s überträgt. Jeder Kanal kann eigens gemultiplext werden, um mehrere, langsamere Kanäle zu erzeugen.

Die von ADSL-Modems unterstützten Übertragungsgeschwindigkeiten sind konform zu den nordamerikanischen und europäischen digitalen Hierarchien (siehe Tabelle 15.1). Solche Modems sind erhältlich mit verschiedenen Übertragungsgeschwindigkeiten und -eigenschaften. Bei minimaler Konfiguration werden ein Downstream-Kanal mit 1,5 oder 2,0 Mbit/s und ein Duplex-Kanal mit 16 Kbit/s unterstützt. Andere Konfigurationen unterstützen 6,1 Mbit/s und 64 Kbit/s. Produkte mit Downstream-Raten von 9 Mbit/s und auf dem Duplex-Kanal mit bis zu 640 Kbit/s sind seit 1996 verfügbar. Wenn die ATM-Technologie und deren Marktanforderungen ausgereift sind, werden ADSL-Modems die ATM-Übertragung mit variablen Raten und Komprimierung für den ATM-Overhead übernehmen können.

Die Downstream-Datenraten hängen von einer Vielzahl von Faktoren ab, wie der Länge der Kupferleitung, deren Durchmesser, vorhandenen Stichleitungen und Interferenzen. Die Leitungsdämpfung nimmt mit der Leitungslänge und höherer Frequenz zu. Je größer der Leitungsdurchmesser, um so niedriger die Dämpfung. Tabelle 15.1 faßt die angegebene Leistung von ADSL für verschiedene physische Medien zusammen (entsprechend den Veröffentlichungen des ADSL-Forums).

Datenrate (Mbit/s)	Durchmesser (AWG)	Leitungs-durchmesser (mm)	Leitungslänge (km)
1,5 oder 2	24	0,5	5,5
1,5 oder 2	26	0,4	4,6
6,1	24	0,5	3,7
6,1	26	0,4	2,7

Tabelle 15.1: Vorgegebene Leistung von ADSL für physische Medien

Viele der Anwendungen, die für ADSL vorgesehen sind, bearbeiten komprimierte digitale Videos. Da es sich bei digitalem Video um Echtzeit-Signale handelt, können diese die normalerweise von Datenkommunikationssystemen verwendeten Fehlerkontrollverfahren auf Verbindungs- oder Netzwerk-Ebene nicht einsetzen. ADSL-Modems arbeiten mit eingebauter Forward-Fehlerkorrektur, um Fehler, die durch Signalrauschen entstehen, zu reduzieren. Symbolweise Fehlerkorrektur reduziert ebenfalls die Fehlerzahl, die von ständigem Leitungsrauschen herrührt.

Zur Zeit bieten ADSL-Modelle die digitalen Schnittstellen T1/E1 und V.35 für Continuous Bit Rate (CBR)-Signale.

15.3 ADSL-Funktionen

Um mehrere Kanäle zu erzeugen, müssen ADSL-Modems die verfügbare Bandbreite einer Telefonleitung aufteilen. Dazu gibt es zwei Möglichkeiten: Frequency Division Multiplexing (FDM) oder Echo Cancellation. Bei FDM wird jeweils ein Band für ausgehende (upstream) und eingehende (downstream) Daten zugewiesen. Der eingehende Pfad wird dann per Zeitscheiben-Multiplexing in einen oder mehrere Hochgeschwindigkeits-Kanäle und einen oder mehrere langsame Kanäle zerlegt. Der ausgehende Pfad wird in entpsrechende

langsame Kanäle gemultiplext. Die Echo Cancellation ordnet das Eingangsband so an, daß es das Ausgangsband überlappt. Dabei werden die beiden Bänder per lokaler Echo Cancellation getrennt – eine Technik, die bei V.32- und V.34-Modems eingeführt ist. Mit der Echo Cancellation wird die Bandbreite effektiver genutzt, jedoch mit einer höheren Komplexität und Kosten. Mit beiden Techniken teilt ADSL einen 4 kHz-Bereich für POTS am DC-Ende des Bands ab.

Ein ADSL-Modem organisiert den aggregierten Datenstrom, der beim Multiplexen der Downstream-, Duplex- und Verwaltungskanäle entsteht, in Blöcke, denen es einen Fehlerkorrektur-Code anhängt. Der Empfänger berichtigt anschließend Fehler, die bei der Übertragung zustande kamen, bis zu einem Grenzwert, der vom Code und der Blocklänge bestimmt wird. Ein Modem kann – aus Benutzersicht – Superblöcke erzeugen, indem es Daten in Subblöcke schachtelt, wodurch der Empfänger in die Lage versetzt wird, jede Kombination von Fehlern innerhalb einer bestimmten Bit-Spanne zu berichtigen.

Bild 15.1 zeigt den Geschwindigkeitsbereich von ADSL für den Downstream, wie er vom ADSL-Forum erklärt wurde. Die Geschwindigkeit für Upstreams reicht von 16 Kbit/s bis zu 640 Kbit/s. Einzelne Produkte sind heute mit einer Vielzahl eingebauter Geschwindigkeitskombinationen versehen, mit einer minimalen Geschwindigkeit von 1,544/2,048 Mbit/s (ausgehend) und 640 Kbit/s (eingehend). Alle diese Einrichtungen werden in einem Frequenzband oberhalb POTS betrieben, wobei der POTS serviceunabhängig und -ungestört bleibt, auch wenn ein ADSL-Modem auf Benutzerseite ausfällt. Bild 15.1 zeigt eine grundlegende ADSL-Verbindung.

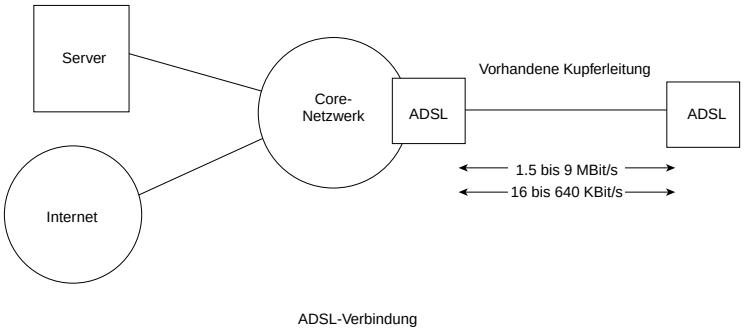


Bild 15.1: Eine ADSL-Verbindung erreicht einen Server oder das Internet über ein Core-Netzwerk

15.4 ADSL-Referenzmodell

Bild 15.2 und die folgenden Definitionen geben eine Überblick über die Hauptelemente des ADSL-Referenzmodells.

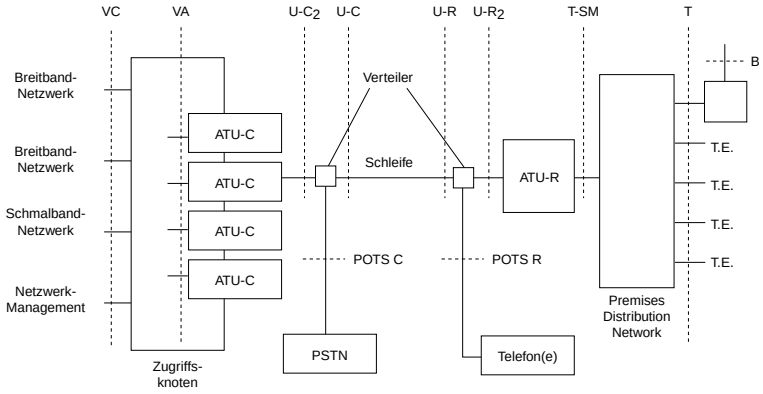


Bild 15.2: Das ADSL-Referenzmodell enthält eine Vielzahl miteinander verbundener Komponenten

Die folgenden Beschreibungen fassen die Elemente des ADSL-Referenzmodells, wie in Bild 15.2 gezeigt, zusammen:

– *ATU-C* – ADSL-Übertragungseinheit am Netzwerk-Ende. Die ATU-C kann in einen Zugriffsknoten integriert sein.

– *ATU-R* – ADSL-Übertragungseinheit auf Seiten des Kunden. Die ATU-R kann in ein SM integriert sein.

– *Zugriffsknoten* – Konzentrationspunkt für breit- und schmalbandige Daten. Ein Zugriffsknoten kann in der Zentrale oder einer entfernten Niederlassung stehen. Ein ferner Zugriffsknoten kann auch von einem zentralen Zugriffsknoten mitverwaltet werden.

– *B* – Hilfsdaten-Eingang (z.B. Satelliten-Einspeisung) für ein Service-Modul (z.B. Top-Box).

– *Broadcast* – Breitband-Dateneingang im einfachen Modus (normalerweise Broadcast-Video).

– *Breitband-Netzwerk* – Switching-System für Datenraten über 1,5/2,0 Mbit/s.

– *Schleife* – Twisted-Pair-Telefonleitung. Schleifen können in der Länge, im Durchmesser, Alter und den Übertragungseigenschaften verschieden sein, abhängig vom Leitungsnetz.

- *Schmalband-Netzwerk* – Switching-System für Datenraten unter oder bis zu 1,5/2,0 Mbit/s.
- *POTS* – Plain Old Telephone Service (einfacher, alter Telefondienst).
- *POTS-C* – Schnittstelle zwischen PSTN- und POTS-Verteiler am Netzwerk-Ende.
- *POTS-R* – Schnittstelle zwischen Telefonen und POTS-Verteiler auf Kundenseite.
- *Premises Distribution Network (PDN)* – System für den Anschluß einer ATU-R an Service-Module. PDN kann ein Netz mit Punkt-zu-Punkt-Verbindungen oder Mehrpunkt-Verbindungen sein, mit einer passiven Verkabelung oder einem aktiven Netzwerk. Mehrpunkt-Verbindungen können über einen Bus oder eine Stern-Verkabelung erfolgen.
- *PSTN* – Public Switched Telephone Network (Öffentliches Telefonnetz).
- *Service-Modul (SM)* – Geräte, die der Terminaladaptierung dienen. Zum Beispiel Set-Top-Boxen, PC-Schnittstellen oder LAN-Router.
- *Verteiler* – Filter, die hoch- (ADSL) und niederfrequente (POTS) Signale am Netzwerk-Ende und auf Kundenseite trennen können. Ein Verteiler kann in die ATU integriert sein, physisch von ihr getrennt sein oder in Hoch- und Tiefpass-Filter aufgeteilt sein, wobei der Tiefpass-Filter physisch von der ATU getrennt ist. Die Bereitstellung von POTS-Verteilern und POTS-bezogener Funktionen ist optional.
- *T-SM* – Schnittstelle zwischen ATU-R und PDN, die wie T ausgelegt sein kann, wenn es sich bei dem Netzwerk um eine passive Punkt-zu-Punkt-Verkabelung handelt. Eine ATU-R kann mit mehr als einem Typ von T-SM-Schnittstellen ausgestattet sein, z.B. für T1/E1-Verbindungen und Ethernet-Verbindungen. Die T-SM-Schnittstelle kann in einem Service-Modul integriert sein.
- *T* – Schnittstelle zwischen PDN und Service-Modulen, die mit T-SM identisch sein kann, wenn es sich bei dem Netzwerk um eine passive Punkt-zu-Punkt-Verkabelung

handelt. Beachten Sie, daß T-Schnittstellen auf der physischen Ebene entfallen, wenn die ATU-R in einem Service-Modul integriert ist.

- *U-C* – Schnittstelle zwischen Schleife und POTS-Verteiler auf Netzwerk-Seite. Definiert die beiden Enden der Schleifenschnittstelle separat, aufgrund der Asymmetrie der Leitungssignale.
- *U-C2* – Die Schnittstelle zwischen POTS-Verteiler und ATU-C. Beachten Sie, daß z. Zt. ANSI T1.413 eine solche Schnittstelle nicht definieren, so daß die Separierung eines POTS-Verteilers von der ATU-C einige technische Schwierigkeiten hinsichtlich der Standardisierung bereitet.
- *U-R* – Die Schnittstelle zwischen der Leitungschleife und dem POTS-Verteiler auf Kundenseite.
- *U-R2* – Die Schnittstelle zwischen POTS-Verteiler und der ATU-R. Zu beachten ist dabei, daß gegenwärtig eine solche Schnittstelle vom Standard ANSI T1.413 nicht definiert wird. Da der POTS-Verteiler vom ATU-R getrennt wird, ergeben sich einige technische Schwierigkeiten in Hinblick auf die Standardisierung der Schnittstelle.
- *VA* – Die logische Schnittstelle zwischen der ATU-C und dem Zugriffsknoten. Da diese Schnittstelle oft mit Schaltungen auf einer normalen Karte realisiert ist, hat das ADSL-Forum von der Spezifizierung einer physischen VA-Schnittstelle abgesehen. Die V-Schnittstelle kann das Synchronous Transport Module (STM), ATM oder beide Übertragungsmodi enthalten. Im einfachsten Fall der Punkt-zu-Punkt-Verbindung zwischen einem Switch-Port und einer ATU-C (wenn es keine Konzentration oder kein Multiplexing gibt) sind VA- und VC-Schnittstelle identisch (oder die VA-Schnittstelle entfällt).
- *VC* – Die Schnittstelle zwischen Zugriffsknoten und dem Netzwerk, das mit mehreren physischen Verbindungen angeschlossen ist (wie in der Darstellung). Es ist auch möglich, daß ein Netzwerk mit nur einer physischen Leitung angeschlossen ist, über die alle Signale laufen. Eine digitale Einrichtung des Betreibers, z.B. ein Synchronous Optical Network (SONET) oder eine Synchronous Digital Hierarchy (SDH)-Erweiterung, kann zwischen VC-Schnittstelle

und dem Zugriffsknoten geschaltet sein, wenn der Zugriffsknoten und die ATU-Cs auf der fernen Kundenseite stehen. Die Schnittstelle zum PSTN kann eine universale Tip-Ring-Schnittstelle oder ein gemultiplextes Telefon-Interface sein, z.B. wie es in Bellcore TR-08 bzw. TR-303 spezifiziert ist. Beim Breitband-Segment der VC-Schnittstelle kann es sich um die Verbindungstypen STM-Switching, ATM-Switching oder Privatleitung handeln.

KAPITEL 16

Synchronous Data-Link Control und Derivate Statusbar

16.1 Hintergrund

Von IBM wurde Mitte der 70er Jahre das Protokoll Synchronous Data-Link Control (SDLC) für den Einsatz in Systems Network Architecture (SNA)-Umgebungen entwikkelt. SDLC war das erste Protokoll der Verbindungssicherungsschicht, das auf synchronem, bit-orientiertem Verfahren aufsetzt. Dieses Kapitel gibt einen kurzen Überblick über die grundlegenden funktionalen Eigenschaften von SDLC und erwähnt auch verschiedene davon abgeleitete Protokolle.

Nach der Entwicklung von SDLC legte IBM das Protokoll mehreren Standardisierungs-Komitees vor. Die International Organization for Standardization (ISO) modifizierte SDLC zum Protokoll High-Level Data-Link Control (HDLC). Die International Telecommunication Union – Telecommunication Standardization Sector (ITU-T) (früher CCITT) änderte HDLC, so daß daraus die Link-Access Procedure (LAP) wurde und anschließend Link-Access Procedure, Balanced (LAPB). Das Institute of Electrical and Electronic Engineers (IEEE) änderte HDLC in den Standard IEEE 802.2. Jedes dieser Protokolle wurde in seinem bestimmten Bereich wichtig, jedoch blieb SDLC das primäre Protokoll der SNA-Verbindungsschicht für WAN-Verbindungen.

16.2 SDLC-Typen und Topologien

SDLC unterstützt eine Vielzahl von Verbindungstypen und Topologien. Es kann eingesetzt werden für Punkt-zu-Punkt- und Mehrpunkt-Verbindungen, auf begrenzten und unbegrenzten Medien, mit Halb- und Vollduplex-Übertragungseinrichtungen und in verbindungsvermittelten und paketvermittelten Netzwerken.

SDLC erkennt zwei Typen von Netzwerk-Knoten: *primäre* und *sekundäre*. Primäre Knoten steuern den Betrieb anderer Stationen, den sogenannten Sekundären. Die primären Stationen pollen die sekundären in einer vordefinierten Reihenfolge. Auf diese Anforderung hin können die sekundären Stationen ihre Daten senden. Von den primären Stationen werden Verbindungen auf- und abgebaut und verwaltet. Die *sekundären* Knoten werden von den primären gesteuert, d.h., die sekundären Knoten können nur dann Daten an die primären Knoten senden, wenn diese das zulassen.

Primäre und sekundäre SDLC-Knoten können auf vier Weisen miteinander verbunden werden:

- *Punkt-zu-Punkt* – Es werden nur zwei Knoten verbunden, ein primärer und ein sekundärer.
- *Mehrpunkt* – Es werden an einen primären Knoten mehrere sekundäre angeschlossen.
- *Schleife* – Schleifen-Topologie, in der an den primären Knoten die erste und letzte sekundäre Station angeschlossen ist. Die dazwischenliegenden sekundären Stationen reichen den Datenverkehr zueinander weiter, wie auch die Antworten auf eine Anforderung des primären Knotens.
- *Hub go-ahead* – Eingangs- und Ausgangskanal. Die primären Knoten kommunizieren über den Ausgangskanal mit den sekundären Stationen. Die sekundären Stationen kommunizieren dementsprechend über den Eingangskanal mit dem primären Knoten. Der Eingangskanal wird über jede sekundäre Station zum primären Knoten zurückgeschleift.

16.3 SDLC-Frame-Format

Der SDLC-Frame ist in Bild 16.1 dargestellt.

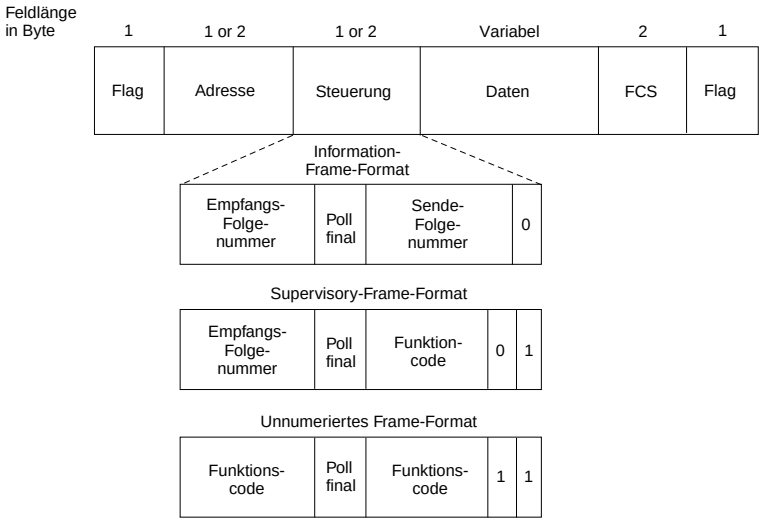


Bild 16.1: Der SDL-Frame setzt sich aus sechs Feldern zusammen

Im folgenden werden die dargestellten Felder aus Bild 16.1 kurz beschrieben.

- *Flag* – Kennzeichnet Anfang und Ende der Fehlerprüfung.
- *Adresse* – Enthält die SDLC-Adresse der sekundären Station, woran zu erkennen ist, ob der Frame von der primären oder sekundären Station gesendet wurde. Diese Adresse kann eine einzelne, eine Gruppen- oder eine Broadcast-Adresse sein. Die primäre Station ist immer entweder Quelle oder Ziel, weshalb deren Adresse nicht angegeben zu werden braucht.
- *Steuerung* – Verwendet drei verschiedene Formate abhängig vom SDLC-Frame-Typ:
  - Information- (I-)Frame: Überträgt Informationen der höheren Protokollschichten und bestimmte Steuerinformationen.

    Dieser Frame sendet und empfängt Folgenummern; das Poll-Final-Bit (P/F) dient der Fluß- und Fehlersteuerung. Die Sende-Folgenummer bezieht sich auf die Nummer des als nächsten zu sendenden Frame. Die Empfangs-

Folgenummer stellt die Nummer für den Frame zur Verfügung, der als nächster empfangen werden soll.

Sowohl Sender als auch Empfänger verwalten Sende- und Empfangs-Folgenummern.

Eine primäre Station verwendet das P/F-Bit dazu, einer sekundären Station mitzuteilen, ob eine sofortige Antwort erforderlich ist. Mit dem P/F-Bit teilt die sekundäre Station der primären mit, ob der aktuelle Frame der letzte seiner Antwort ist.

- Supervisory-(S-)Frame: Stellt Steuerinformationen zur Verfügung. Ein S-Frame kann die Übertragung anfordern und unterbrechen, gibt Auskunft über den Status und bestätigt den Eingang von I-Frames. S-Frames haben kein Informations-Feld.
- Unnumerierter-(U-)Frame: Dient zur Steuerung und ist nicht an die Reihenfolge gebunden. Ein U-Frame kann dazu verwendet werden, eine sekundäre Station zu initialisieren. Abhängig von der Funktion des U-Frame ist das Steuerungsfeld 1 oder 2 Byte lang. Manche U-Frames haben auch ein Informations-Feld.

- *Daten* – Enthält die Daten der Path-Information Unit (PIU) oder zur Exchange Identification (XID).
- *Frame-Prüfsequenz (FCS)* – Ist dem Ende-Flag vorangestellt und ein Wert des Cyclic Redundancy Check (CRC). Die CRC-Berechnung erfolgt ein zweites Mal im Empfänger. Wenn das Ergebnis vom ursprünglichen Wert abweicht, wird von einem Fehler ausgegangen.

In Bild 16.2 wird eine typische SDLC-basierte Netzwerk-Konfiguration dargestellt. Ein IBM-Establishment-Controller (früher bezeichnet als Cluster Controller) verbindet in der entfernten Niederlassung »dumme« Terminals mit einem Token-Ring-Netzwerk. In der lokalen Niederlassung ist ein IBM-Host (über Kanalanschluß-Techniken) an einen Vorrechner (FEP) angeschlossen, der auch in Verbindung stehen kann mit lokalen Token-Ring-LANs und einem SNA-Backbone. Die beiden Niederlassungen sind über eine geleaste Leitung mit 56-Kbit/s und SDLC miteinander verbunden.

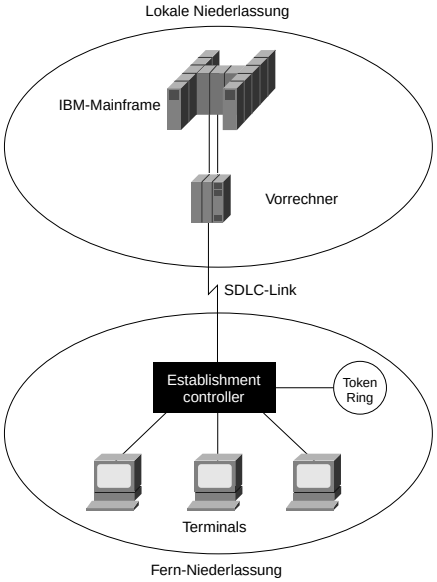


Bild 16.2: Mit SDLC sind eine lokale und ferne Niederlassung über eine serielle Leitung verbunden

16.4 Abgeleitete Protokolle

Obwohl HDLC einige Funktionen nicht aufweist, die in SDLC verwendet werden, wird HDLC immer als ein zu SDLC kompatibles, umfangreicheres Protokoll betrachtet. LAP ist ein Subset von HDLC und wurde entwickelt, um weiterhin die Kompatibilität mit HDLC sicherzustellen, das in den frühen 80er Jahren modifiziert wurde. IEEE 802.2 ist eine für LAN-Umgebungen angepaßte Fassung von HDLC. *Qualified Logical Link Control* (QLLC) ist ein Protokoll der Verbindungsschicht, das von IBM definiert wurde, und das dazu dient, SNA-Daten über X.25-Netzwerke zu senden.

16.4.1 High-Level Data-Link Control (HDLC)

HDLC hat das gleiche Frame-Format wie SDLC, wobei die HDLC-Felder die gleichen Funktionen bereitstellen wie die in SDLC. HDLC unterstützt ebenso synchronen Voll-Duplex-Betrieb.

HDLC unterscheidet sich jedoch geringfügig von SDLC. Als erstes bietet HDLC eine Option für eine 32-Bit-Prüfsumme.

Anders als bei SDLC werden von HDLC die Schleifen- und Hub-go-ahead-Konfiguration nicht unterstützt.

Die Hauptdifferenz zwischen HDLC und SDLC ist, daß SDLC nur einen Übertragungsmodus unterstützt, während HDLC drei Modi unterstützt:

- *Normal response mode* (NRM) – Dieser Übertragungsmodus wird auch von SDLC verwendet. In diesem Modus können sekundäre Stationen mit der primären erst auf deren Anforderung kommunizieren.
- *Asynchronous response mode* (ARM) – In diesem Modus können sekundäre Stationen selbst die Kommunikation mit der primären Station beginnen, ohne von dieser erst aufgefordert werden zu müssen.
- *Asynchronous balanced mode* (ABM) – ABM führt den *kombinierten* Knoten ein, der sowohl primärer als auch sekundärer Knoten sein kann, abhängig von der Situation. Die gesamte ABM-Kommunikation erfolgt zwischen mehreren kombinierten Knoten. In einer IBM-Umgebung kann jede kombinierte Station die Datenübertragung starten, ohne dazu die Erlaubnis anderer Stationen einholen zu müssen.

16.4.2 Link-Access Procedure, Balanced (LAPB)

LAPB ist bekannt aufgrund seiner Präsenz im X.25-Protokoll-Stack. LAPB arbeitet mit dem gleichen Frame-Format, den gleichen Frame-Typen und Feldfunktionen wie SDLC und HDLC. Aufgrund seiner Unterschiede zu beiden Protokollen ist LAPB jedoch beschränkt auf den ABM-Tranfer-Modus und nur für kombinierte Stationen sinnvoll. Ebenso können LAPB-Verbindungen vom Data Terminal Equipment (DTE) oder Data Circuit-Terminating Equipment (DCE) aufgebaut werden. Die Station, die die Anforderung absetzt, wird als primäre Station festgelegt, während die antwortende Station als sekundäre Station läuft.

Schließlich ist die Behandlung des P/F-Bit etwas anders als bei den anderen Protokollen. Weitere Informationen zu LAPB finden Sie im Kapitel 17, »X.25«.

16.4.3 IEEE 802.2

IEEE 802.2 wird oft als *Logical Link Control* (LLC) bezeichnet. In LAN-Umgebungen ist es sehr weit verbreitet, wo es mit Protokollen wie *IEEE 802.3*, *IEEE 802.4* und *IEEE 802.5* zusammenarbeitet. IEEE 802.2 bietet drei Service-Typen.

Typ 1 ist ein verbindungsloser Service, der ohne Bestätigung arbeitet. Das heißt, daß LLC Typ 1 die Datenübertragung nicht quittiert. Da viele Protokolle der höheren Schichten (z.B. *Transmission Control Protocol/Internet Protocol* – TCP/IP) eine zuverlässige Datenübertragung bieten, die sich nicht auf Protokolle der unteren Schichten stützt, ist Typ 1 ein oft verwendeter Service.

Typ 2 bietet einen verbindungsorientierten Service. Der Service LLC Typ 2 (oft als *LLC2* bezeichnet) baut zwischen Sender und Empfänger eine logische Verbindung auf und ist deshalb verbindungsorientiert. LLC2 quittiert die Übertragung beim Empfang und findet sich in Kommunikationssystemen von IBM.

Type 3 arbeitet als verbindungsloser Service mit Quittung. Auch wenn LLC Typ 3 die Datenübertragung quittiert, baut er jedoch keine logischen Verbindungen auf. Als Kompromiß zu den anderen beiden LLC-Services kann LLC Typ 3 in der Automatisierungstechnik eingesetzt werden, wo die Fehlererkennung wichtig, der Speicherplatz (für virtuelle Verbindungen) aber sehr begrenzt ist.

End-Stationen können mehrere LLC-Service-Typen gleichzeitig unterstützen. Ein Gerät der Klassse I unterstützt aber nur den Typ-1-Service. Klasse-II-Geräte unterstützen die Typen 1 und 2, Klasse-III-Geräte Typ 1 und 3, während Klasse-IV-Geräte alle drei Service-Typen unterstützen.

Prozesse der höheren Schichten greifen auf IEEE-802.2-Services über *Service Access Points* (SAPs) zu. Der IEEE-802.2-Header beginnt mit einem *Destination Service Access Point* (DSAP)-Feld, das den eingehenden Prozeß der höheren Schichten erkennt. In anderen Worten: Nachdem die IEEE-802.2-Implementation des empfangenden Knotens die Verarbeitung beendet hat, werden die verbleibenden Daten an den im DSAP-Feld bezeichneten Prozeß weitergegeben. Der DSAP-

Adresse folgt die Adresse des *Source Service Access Point* (SSAP), die den sendenden Prozeß der höheren Schichten bezeichnet.

16.4.4 Qualified Logical-Link Control (QLLC)

QLLC bietet die DLC-Fähigkeiten, die erforderlich sind, um SNA-Daten über ein X.25-Netzwerk zu übertragen. QLLC und X.25 ersetzen SDLC im SNA-Protokoll-Stack. QLLC verwendet die Packet-Level-Schicht (Schicht 3) des X.25-Protokoll-Stack. Um QLLC mitzuteilen, daß ein X.25-Paket der Schicht 3 verarbeitet werden muß, wird ein spezielles Bit, das *Qualifier Bit*, im *General Format Identifier* (GFI) gesetzt, der zum X.25-Paket-Header der Schicht 3 gehört. Die SNA-Daten werden in X.25-Paketen der Schicht 3 als Benutzerdaten übertragen. Weitere Informationen zum X.25-Protokoll-Stack finden Sie in Kapitel 17, »X.25«.

KAPITEL 17

X.25

17.1 Hintergrund

X.25 ist ein Protokoll-Standard für WAN-Kommunikation des International Telecommunication Union Telecommunication Standardization Sector (ITU-T). In diesem Standard ist definiert, wie Verbindungen zwischen Benutzergeräten und Netzwerk-Geräten aufgebaut und verwaltet werden. X.25 ist so ausgelegt, daß es effektiv arbeitet – unabhängig vom System, das an das Netz angeschlossen ist. Am häufigsten ist X.25 in den paketvermittelten Netzen (PSNs) der normalen Betreiber im Einsatz, z.B. bei den Telefongesellschaften. Den Teilnehmern werden die Kosten für die Verwendung des Netzwerks berechnet. In den 70er Jahren veranlaßten die gängigen Betreiber die Entwicklung des X.25-Standards. Damals war die Anforderung, ein WAN-Protokoll zu entwickeln, das Verbindungen über öffentliche Datennetze (PDNs) ermöglichte. Heute wird X.25 als ein internationaler Standard von der ITU-T betreut. In diesem Kapitel werden die grundlegenden Funktionen und Komponenten von X.25 erläutert.

17.2 X.25-Geräte und -Protokollfunktion

Die X.25-Netzwerk-Geräte lassen sich in drei große Kategorien einteilen: Datenendeinrichtungen (DEE/DTE), Datenübertragungseinrichtungen (DÜE/DCE) und paketvermittlende Datenaustauscher (PSE). DTE-Geräte sind Endgeräte, die über das X.25-Netzwerk miteinander kommunizieren. Dabei handelt es sich meistens um Terminals, Personalcomputer oder Netzwerk-Hosts, die bei den Teilnehmern stehen. DCE-Geräte sind spezielle Kommunikationseinrichtungen, z.B. Modems oder Paket-Switches. Sie bilden die Schnittstelle zwischen den DTE-Geräten und dem PSE und sind meistens in den Räumlichkeiten des Netzbetreibers untergebracht. PSEs sind Switches, die den eigentlichen Teil des Betreiber-Netzwerks ausmachen. Sie übertragen die Daten von einem DTE-Gerät zu einem anderen über das öffentliche X.25-Netz. Bild 17.1 zeigt den Zusammenhang dieser drei Gerätetypen im X.25-Netz.

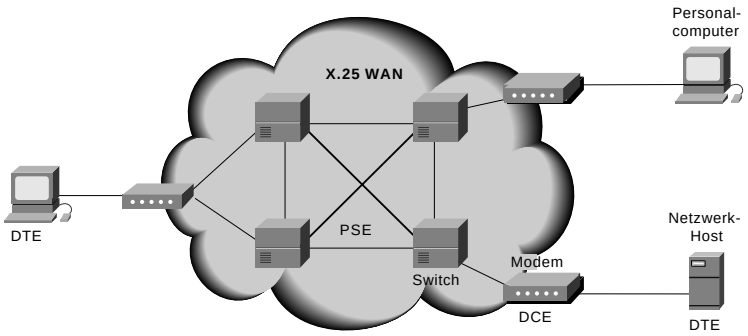


Bild 17.1: DTEs, DCEs und PSEs bilden das X.25-Netzwerk

17.2.1 Packet Assembler/Disassembler (PAD)

Der Packet Assembler/Disassembler (PAD) ist ein in X.25-Netzen häufig anzutreffendes Gerät. PADs werden immer dann benötigt, wenn ein DTE-Gerät, z.B. ein zeichenorientiertes Terminal, zu »dumm« ist, um die volle X.25-Funktionalität zu unterstützen. Der PAD ist zwischen DTE- und DCE-Gerät geschaltet und hat drei wichtige Aufgaben: Pufferung, Paketzusammenstellung (assembly), Paketzerlegung (disassembly). Der PAD puffert die Daten, die vom DTE-Gerät gesendet werden. Dann stellt er die ausgehenden Daten zu

Paketen zusammen und sendet sie an das DCE-Gerät (wobei ein X.25-Header dem Paket hinzugefügt wird). Eingehende Datenpakete werden vom PAD zerlegt, bevor die Daten an das DTE-Gerät weitergeleitet werden (zuvor wird noch der X.25-Header entfernt). Bild 17.2 zeigt die Grundfunktion des PAD bei eingehenden Paketen aus dem X.25-WAN.

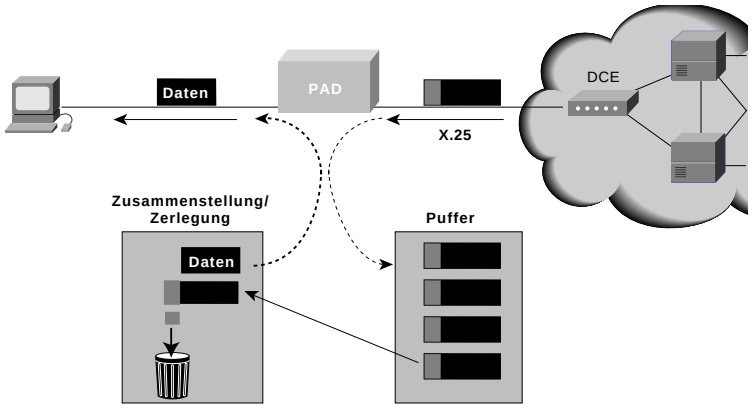


Bild 17.2: Der PAD puffert, stellt zusammen und zerlegt Datenpakete

17.2.2 X.25-Sitzung einrichten

Eine X.25-Sitzung wird aufgebaut, wenn ein DTE-Gerät ein anderes kontaktiert, um eine Kommunikationssitzung anzufordern. Das DTE-Gerät, bei dem diese Anforderung eingeht, kann die Verbindung annehmen oder ablehnen. Wenn es die Verbindung annimmt, tauschen die beiden Geräte ihre Daten im Voll-Duplex-Betrieb aus. Die Verbindung kann von beiden Geräten beendet werden. Wurde eine Sitzung einmal beendet, muß für jede weitere Kommunikation eine neue Sitzung aufgebaut werden.

17.2.3 Virtuelle Verbindungen bei X.25

Eine virtuelle Verbindung (virtual circuit) ist eine logische Verbindung, die dazu dient, eine zuverlässige Kommunikation zwischen zwei Netzwerk-Geräten sicherzustellen. Eine virtuelle Verbindung setzt einen logischen, bidirektionalen Pfad von einem DTE-Gerät über ein X.25-Netz zu einem anderen Gerät

voraus. Physisch kann die Verbindung über eine beliebige Anzahl dazwischenliegender Knoten (z.B. DCE-Geräte und PSEs) geleitet werden. Mehrere virtuelle Verbindungen (d.h. logische Verbindungen) können über eine physische Verbindung (eine Leitung) gemultiplext werden. Am anderen Ende der Leitung werden die virtuellen Verbindungen wieder demultiplext, und die Daten werden an das Ziel gesendet. Bild 17.3 zeigt vier einzelne virtuelle Verbindungen, die auf eine einzelne physische Verbindung gemultiplext werden.

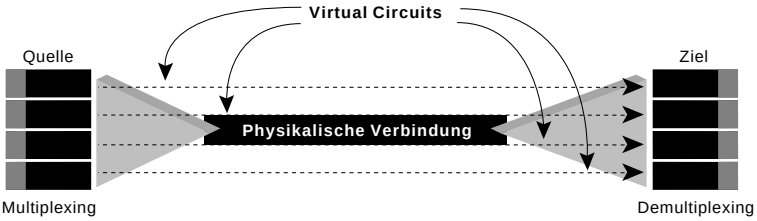


Bild 17.3: Virtuelle Verbindungen können auf eine einzelne physische Verbindung gemultiplext werden

Es gibt zwei Arten virtueller Verbindungen: gewählte und feste Verbindungen. Gewählte virtuelle Verbindungen (GVV – Switched Virtual Connection [SVC]) sind temporäre Verbindungen, die für gelegentliche Datenübertragung gedacht sind. Dazu müssen zwei DTE-Geräte für jede Kommunikation eine Sitzung aufbauen, verwalten und beenden. Feste virtuelle Verbindungen (FVV – Permanent Virtual Connection [PVC]) sind dauerhaft aufgebaute Verbindungen, die für häufige und zuverlässige Datenübertragungen verwendet werden können. Bei PVCs müssen keine Verbindungen aufgebaut und abgebaut werden. So kann von DTE-Geräten jederzeit eine Datenübertragung gestartet werden, da die Sitzung immer aktiv ist.

Der grundlegende Betrieb einer virtuellen X.25-Verbindung beginnt, indem das sendende DTE-Gerät die virtuelle Verbindung (die im Paket-Header eingetragen wird) angibt, über die Daten übertragen werden sollen. Dann sendet das DTE-Gerät seine Daten an das lokale DCE-Gerät. Jetzt überprüft das lokale DCE-Gerät den Paket-Header, um festzustellen, welche virtuelle Verbindung genutzt werden soll, und sendet das Paket zum nächstgelegenen PSE auf dem Pfad dieser virtuellen Verbindung. PSEs (Switches) leiten den Datenverkehr zum

nächsten zwischengeschalteten Knoten, wobei es sich um einen weiteren Switch oder das ferne DCE-Gerät handeln kann.

Gehen die Daten beim fernen DCE-Gerät ein, wird von diesem der Paket-Header überprüft und die Zieladresse bestimmt. Dann werden die Pakete zum entsprechenden DTE-Gerät gesendet. Wenn die Kommunikation über eine gewählte virtuelle Verbindung (SVC) erfolgt und keines der beiden Endgeräte mehr Daten überträgt, wird die virtuelle Verbindung abgebaut.

17.3 X.25-Protokolle

Die von X.25 betriebenen Protokolle betreffen die unteren drei Schichten des OSI-Referenzmodells. Die folgenden Protokolle kommen bei X.25-Implementationen meistens zum Einsatz: Packet-Layer Protocol (PLP), Link-Access Procedure, Balanced (LAPB) und neben anderen seriellen Schnittstellen der physikalischen Schicht (z.B. EIA/TIA-232, EIA/TIA-449, EIA-530 und G.703) das Protokoll X.21bis. Bild 17.4 zeigt die Zuordnung der wichtigen X.25-Protokolle zu den Schichten des OSI-Referenzmodells.

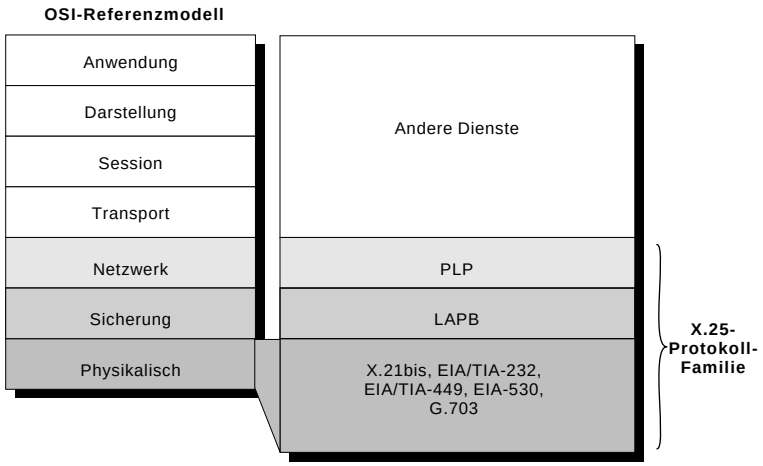


Bild 17.4: Zuordnung der wichtigen X.25-Protokolle zu den unteren drei Schichten des OSI-Referenzmodells

17.3.1 Packet-Layer Protocol (PLP)

Das *Packet Layer Protocol* (PLP) ist das X.25-Protokoll der Vermittlungsschicht. PLP verwaltet den Paketaustausch zwischen DTE-Geräten über virtuelle Verbindungen. PLP-Verbindungen können auch über Logical-Link Control 2 (LLC2)-Implementationen in LANs geführt werden und über ISDN-Schnittstellen, die mit der Link-Access Procedure auf dem D-Kanal (LAPD) arbeiten.

PLP kennt fünf verschiedene Modi: *Ruf-Aufbau*, *Datenübertragung*, *Ruhezustand*, *Ruf-Clearing* und *Neustart*.

Der Modus Ruf-Aufbau dient dazu, einen SVC zwischen DTE-Geräten aufzubauen. Das PLP verwendet dafür das X.121-Adressierungs-Schema. Der Modus Ruf-Aufbau wird von jeder virtuellen Verbindung durchlaufen, so daß sich eine virtuelle Verbindung in diesem Modus befindet, während eine andere sich im Datenübertragungs-Modus befindet. Der Modus Ruf-Aufbau ist nur bei SVCs erforderlich (nicht bei PVCs).

Der Datenübertragungs-Modus dient der Datenübertragung zwischen zwei DTE-Geräten über eine virtuelle Verbindung. In diesem Modus erfolgt durch das PLP die Segmentierung und die Zusammenstellung, das Bit-Padding und die Fehler- und Flußsteuerung. Dieser Modus wird von jeder virtuellen Verbindung durchlaufen (PVC und SVC).

Der Modus Ruhezustand tritt ein, wenn eine virtuelle Verbindung aufgebaut ist, aber keine Datenübertragung stattfindet. Dieser Modus wird von jeder einzelnen virtuellen Verbindung durchlaufen, allerdings nur von SVCs.

Der Modus Ruf-Clearing dient dazu, die Kommunikationssitzung zwischen DTE-Geräten zu beenden und einen SVC abzubauen. Dieser Modus wird von jeder virtuellen Verbindung durchlaufen, und zwar nur von SVCs.

Der Modus Neustart dient dazu, die Übertragung zwischen einem DTE-Gerät und einem lokal angeschlossenen DCE-Gerät zu synchronisieren. Dieser Modus wird nicht von jeder virtuellen Verbindung einzeln durchlaufen, sondern betrifft alle von DTE-Geräten aufgebaute virtuelle Verbindungen.

PLP kennt vier Paketfeld-Typen:

- *General Format Identifier (GFI)* – Kennzeichnet Paket-Parameter, z.B. ob ein Paket Benutzerdaten oder Steuerdaten überträgt, welche Art des Windowing verwendet wird und ob eine Quittierung erforderlich ist.
- *Logical Channel Identifier (LCI)* – Kennzeichnet die virtuelle Verbindung über die lokale DTE/DCE-Schnittstelle.
- *Packet Type Identifier (PTI)* – Kennzeichnet ein Paket mit einem der 17 verschiedenen PLP-Pakettypen.
- *Benutzerdaten* – Enthält gekapselte Daten höherer Schichten. Dieses Feld ist nur in Datenpaketen enthalten. Ansonsten werden Felder mit Steuerdaten hinzugefügt.

17.3.2 Link-Access Procedure, Balanced (LAPB)

Bei Link-Access Procedure, Balanced (LAPB) handelt es sich um ein Protokoll der Verbindungssicherungsschicht, das die Kommunikation und das Paket-Framing zwischen DTE- und DCE-Geräten verwaltet. LAPB ist ein bit-orientiertes Protokoll, das sicherstellt, daß Frames richtig geordnet werden und fehlerfrei sind.

Es gibt drei Arten von LAPB-Frames: *Information*, *Supervisory* und *Unnumbered*. Der **Information-Frame (I-Frame)** enthält Daten der höheren Schichten und einige Steuerdaten. Zu den Funktionen des I-Frame gehören die Folge-, Flußsteuerung und die Fehlererkennung und -behebung. I-Frames enthalten Sende- und Empfangsfolgenummern. Der **Supervisory-Frame (S-Frame)** überträgt Steuerdaten. Zu den Funktionen des S-Frame gehören das Anfordern und Unterbrechen von Übertragungen, die Bekanntgabe des Status und die Bestätigung ein-

gegangener I-Frames. S-Frames enthalten nur Empfangsfolgenummern. Der **Unnumbered Frame (U-Frame)** überträgt Steuerdaten. Zu den Funktionen des U-Frame gehören das Aufbauen und Beenden von Verbindungen und die Fehlermeldung. U-Frames enthalten keine Folgenummern.

17.3.3 X.21bis-Protokoll

X.21bis ist ein Protokoll der physikalischen Schicht, das in X.25 verwendet wird. Es definiert die elektrischen und mechanischen Verfahren für das physische Medium. X.21bis bearbeitet die Aktivierung und Deaktivierung des physischen Mediums, über das DTE- und DCE-Geräte verbunden sind. Es unterstützt Punkt-zu-Punkt-Verbindungen, Übertragungsraten bis zu 19,2 Kbit/s und synchrone Voll-Duplex-Übertragung über Vierdraht-Medien. Bild 17.5 zeigt das Format eines LPL-Pakets im Zusammenhang mit einem LAPB-Frame und einem X.21bis-Frame.

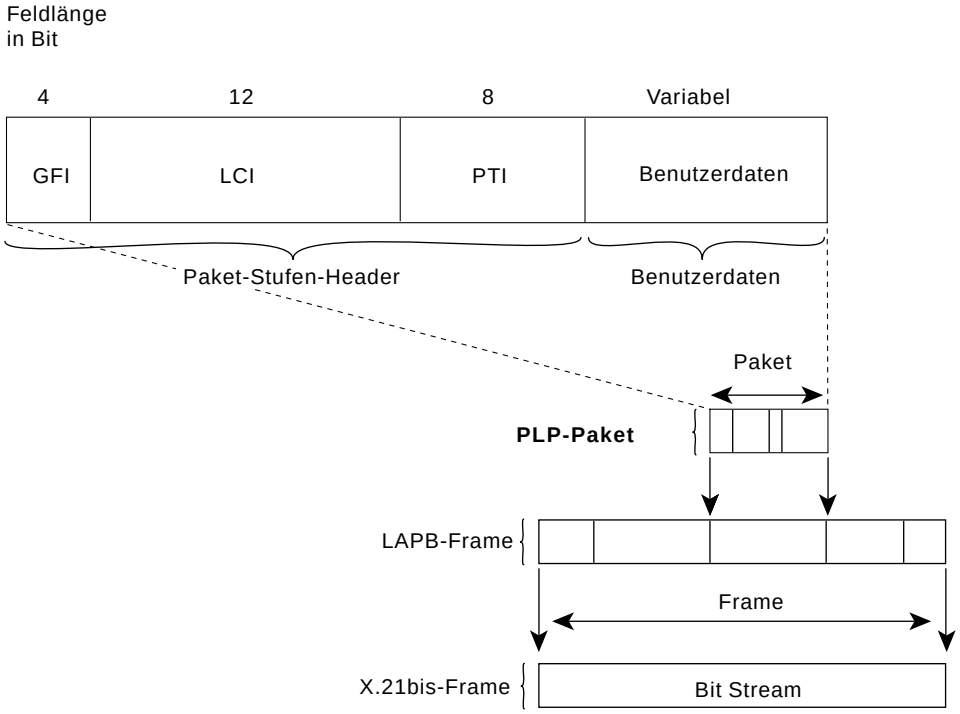


Bild 17.5: Das PLP-Paket ist in den LAPB- und X.21bis-Frame gekapselt

17.4 LAPB-Frame-Format

LAPB-Frames bestehen aus dem Header, gekapselten Daten und dem Trailer. Bild 17.6 zeigt das Format eines LAPB-Frames im Zusammenhang mit einem PLP-Paket und einem X.21bis-Frame.

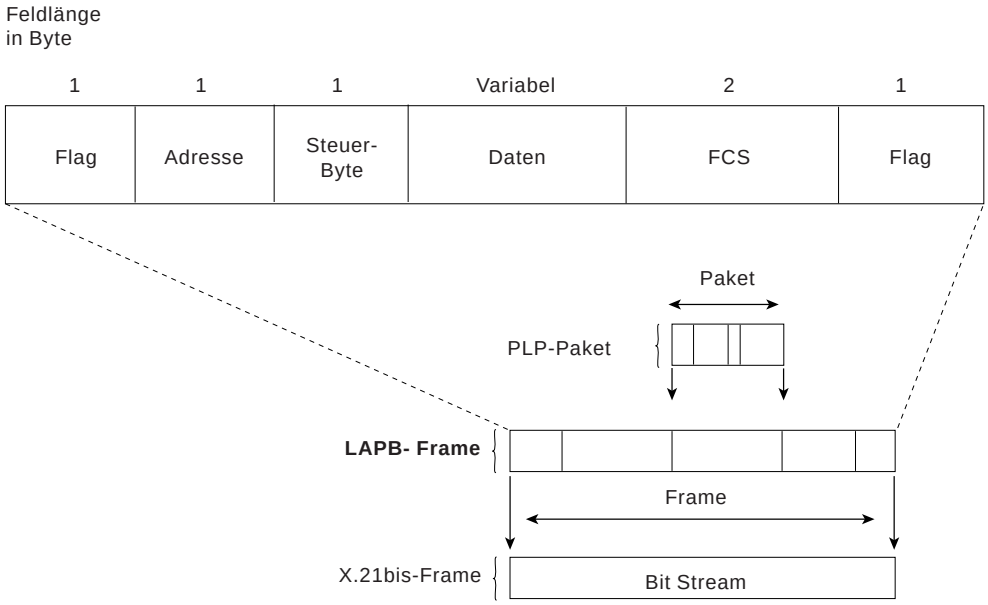


Bild 17.6: Ein LAPB-Frame beinhaltet einen Header, einen Trailer und gekapselte Daten

Im folgenden werden die in Bild 17.6 gezeigten Felder kurz beschrieben:

– *Flag-Frame-Feld* – Kennzeichnet Anfang und Ende eines LAPB-Frame. Dabei besteht das Flag aus gleichen Bits, um sicherzustellen, daß das Bit-Muster des Flag nicht auch im Innern des Frame vorkommt.

– *Adreßfeld* – Gibt an, ob der Frame einen Befehl oder eine Antwort enthält.

– *Steuerfeld* – Kennzeichnet Befehls- und Antwort-Frames und gibt an, ob es sich bei dem Frame um einen I-Frame, S-Frame oder U-Frame handelt. Außerdem enthält dieses

Feld die Folgenummer des Frame und seine Funktion (z.B. empfangsbereit oder unterbrochen). Steuer-Frames sind unterschiedlich lang, was vom Frame-Typ abhängt.

– *Datenfeld* – Enthält Daten höherer Schichten in Form gekapselter PLP-Pakete.

– *FCS* – Bearbeitet die Fehlerprüfung und stellt die Integrität der Datenübertragung sicher.

17.5 X.121-Adreß-Format

X.121-Adressen werden vom X.25-PLP im Modus Ruf-Aufbau verwendet, um SVCs aufzubauen. Bild 17.7 zeigt das Format der X.121-Adressen.

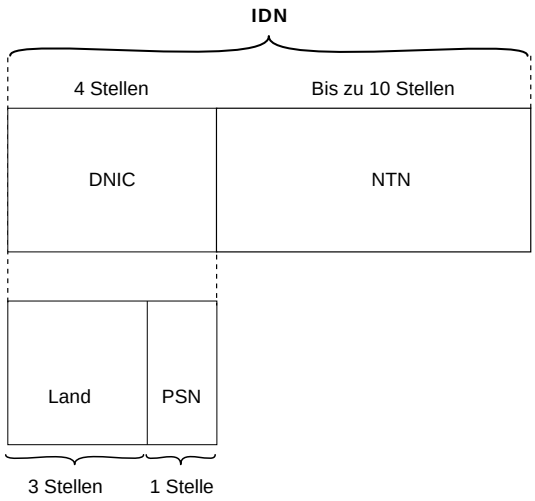


Bild 17.7: Eine X.121-Adresse besteht aus drei IDN-Felder

Das X.121-Adreßfeld besteht aus der International Data Number (IDN), die sich aus zwei Feldern zusammensetzt: dem *Data Network Identification Code (DNIC)* und der *National Terminal Number (NTN).*

Der DNIC ist ein optionales Feld, das das PSN genau bezeichnet, in dem sich das Ziel-DTE-Gerät befindet. Bei Anrufen innerhalb eines PSN wird dieses Feld manchmal weggelassen. Der DNIC besteht aus zwei Unterfeldern: Land und PSN. Mit Land wird das Land angegeben, in dem sich das Ziel-PSN be-

findet. Im PSN-Feld wird genau das PSN angegeben, in dem sich das Ziel-DTE-Gerät befindet.

Mit der NTN wird das DTE-Gerät im PSN genau identifiziert, für das ein Paket bestimmt ist. Die Länge dieses Feldes ist variabel.

Kapitel 18: Asynchronous Transfer Mode (ATM)
Kapitel 19: Data-Link Switching
Kapitel 20: LAN-Switching
Kapitel 21: Tag Switching
Kapitel 22: Mixed-Media-Bridging
Kapitel 23: Source-Route Bridging (SRB)
Kapitel 24: Transparent-Bridging

TEIL 4

Bridging und Switching

Teil 4, »Bridging und Switching«, behandelt die Schlüssel-Technologien, die das Bridging und Switching betreffen. In den einzelnen Kapiteln werden folgende Themen behandelt:

Asynchronous Transfer Mode (ATM) Switching – Dieses Kapitel steigt in die ATM-Technologie, einschließlich der Komponenten, Verbindungstypen, Adressierung, Multicasting-Anforderungen und LAN-Emulation (LANE), ein.

Data-Link Switching (DLSw) – In diesem Kapitel wird der Betrieb von DLSw für die Implementation von SNA- und NetBIOS-Verkehr über IP-WANs definiert und beschrieben.

LAN Switching – Dieses Kapitel behandelt Ursprünge, Vorteile, Einsatz, Arten und Anwendungen des LAN-Switching.

Tag Switching – Dieses Kapitel betrifft das Tag-Switching, einschließlich TDP, Tag-Allocation-Methoden und die verschiedenen Routing-Module.

Mixed-Media Bridging – In diesem Kapitel werden das übersetzende Bridging und Source Route Transparent Bridging definiert und beschrieben. Ferner werden die Schlüsselanforderungen für die Implementation aufgezeigt.

Source-Route Bridging (SRB) – Dieses Kapitel beleuchtet SRB-Konzepte, -Standards, -Route-Discovery-Prozesse und das Frame-Format RIF.

Transparent Bridging – Diese Kapitel faßt den Betrieb transparenten Bridging, Bridging Loops und den Spanning-Tree-Algorithmus zusammen.

KAPITEL 18

Asynchronous Transfer Mode (ATM)

18.1 Grundlagen

Der Asynchronous Transfer Mode (ATM) ist ein Standard der International Telecommunication Standardization Organization (ITU-T) für die Zellenvermittlung, wobei Informationen für verschiedene Dienste, wie Sprache, Video und Daten in kleinen Paketen fester Größe übertragen werden. Dieses Kapitel gibt einen Überblick zu ATM-Protokollen, -Diensten und -Einsatz. Bild 18.1 zeigt ein abgeschlossenes Teilnehmernetz und ein öffentliches ATM-Netz zur Übertragung von Sprache, Video und Datenverkehr.

18.1.1 Standards

ATM beruht auf dem ITU-T »Broadband Integrated Services Digital Network«-Standard (BISDN). Es war ursprünglich als Hochgeschwindigkeits-Übertragungstechnologie für Sprache, Video und Daten über öffentliche Netze gedacht. Das ATM-Forum erweiterte den Vorschlag auf den Einsatz in öffentlichen und privaten Netzwerken. Das ATM-Forum hat seine Arbeiten in den folgenden Spezifikationen veröffentlicht:

- User-to-Network Interface (UNI) 2.0
- UNI 3.0
- UNI 3.1
- Public-Network Node Interface (P-NNI)
- LAN Emulation (LANE)

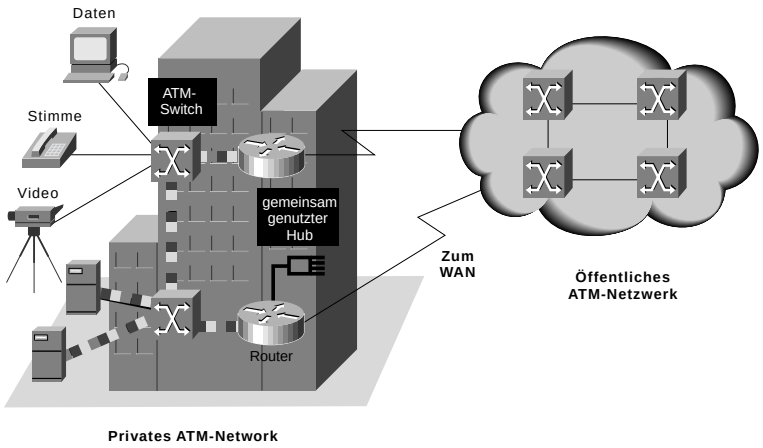


Bild 18.1: Sowohl ein privates als auch ein öffentliches ATM-Netzwerk können Sprache, Video und Datenverkehr übertragen

18.2 ATM-Geräte und Netzwerkumgebungen

ATM ist eine Zellenvermittlungs- und Multiplex-Technologie, welche die Vorteile der Durchschaltevermittlung (garantierte Qualität und konstante Durchlaufzeit) mit denen der Paketvermittlung (Flexibilität und Effizienz des intermittierenden Verkehrs) verbindet. Es ermöglicht skalierbare Bandbreiten zwischen einigen Megabit pro Sekunde (Mbps) bis zu vielen Gigabit pro Sekunde (Gbps). Als asynchrone Technologie ist ATM effizienter als synchrone Technologien, wie z.B. das Time-Division Multiplexing (TDM). Bei TDM wird den Benutzern ein Zeitkanal zugewiesen, in dem keine andere Station senden kann. Wenn eine Station viele Daten zu versenden hat, kann sie nur dann senden, wenn ihr Zeitkanal an der Reihe ist, auch dann, wenn alle anderen Zeitkanäle frei sind. Wenn dagegen eine Station nichts zu verschicken hat, wenn ihr Zeitkanal an der Reihe ist, dann wird er leer verschickt und verschwendet. Weil ATM ein asynchrones Verfahren ist, werden die Zeitkanäle nach Bedarf bereitgestellt, wobei der Header jeder ATM-Zelle die Informationen zur Quelle der Übertragung enthält.

18.2.1 ATM-Zellen-Basisformat

ATM überträgt Informationen in Einheiten fester Größe, die als Zellen bezeichnet werden. Jede Zelle besteht aus 53 Oktetts oder Bytes. Die ersten fünf Bytes enthalten Zellenkopfin-

formationen und die übrigen 48 die eigentlichen »Nutzdaten« (Anwenderinformation). Kleine Zellen mit fester Länge sind für die Übertragung von Sprach- und Videoverkehr besser geeignet, weil diese Verkehrsarten u.a. Verzögerungen auf Grund der Wartezeit auf große Datenpakete nicht tolerieren. Bild 18.2 zeigt das Basisformat einer ATM-Zelle.

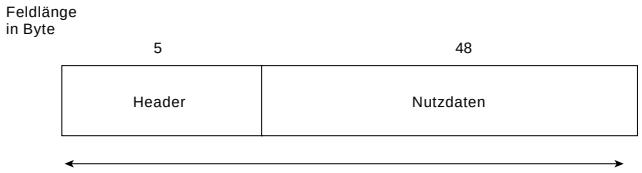


Bild 18.2: Eine ATM-Zelle besteht aus einem Header- und einem Nutzdatenteil

18.2.2 ATM-Geräte

Ein ATM-Netzwerk besteht aus ATM-Switch- und ATM-Endpunkten. Ein ATM-Switch ist für die Übertragung aller Zellen im Netzwerk verantwortlich. Die Aufgabe eines ATM-Switch ist klar definiert: Er nimmt die Zellen von einem ATM-Endpunkt oder anderen ATM-Switch an. Anschließend liest und aktualisiert er die Informationen des Zellenkopfes und leitet die Zelle an eine Ausgangsschnittstelle in Versandrichtung weiter. Ein ATM-Endpunkt (oder Endsystem) enthält einen ATM-Netzwerk-Adapter. Beispiele für ATM-Endpunkte sind Workstations, Router, Data-Service-Units (DSU), LAN-Switches und Video-Koder/Dekoder (CODEXs). Bild 18.3 zeigt ein ATM-Netzwerk, das aus ATM-Switches und ATM-Endpunkten besteht.

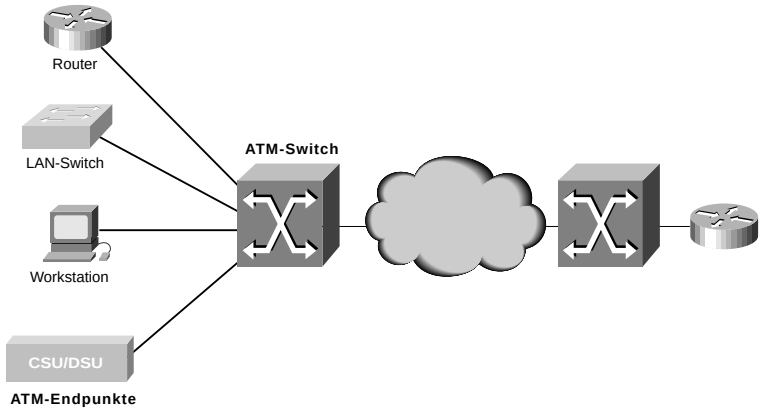


Bild 18.3: Ein ATM-Netzwerk besteht aus ATM-Switches und -Endpunkten

18.2.3 ATM-Netzwerkschnittstellen

Ein ATM-Netzwerk besteht aus ATM-Switches, die durch Punkt-zu-Punkt-ATM-Verbindungen oder -Schnittstellen verbunden sind. ATM-Switches unterstützen zwei primäre Schnittstellentypen: User-Network-Interfaces (UNI) und Network-Node-Interfaces (NNI). Die UNI verbinden ATM-Endsysteme (wie Hosts und Router) mit einem ATM-Switch. Die NNI verbinden zwei ATM-Switches.

Abhängig davon, ob der Switch im Besitz des Kunden ist und ihm zur Verfügung steht, oder im öffentlichen Besitz ist und durch eine Telefongesellschaft vertrieben wird, können UNI und NNI in weitere öffentliche oder private UNIs und NNIs aufgeteilt sein. Ein privates UNI verbindet einen ATM-Endpunkt und einen privaten ATM-Switch. Sein öffentliches Äquivalent verbindet einen ATM-Endpunkt oder privaten Switch mit einem öffentlichen Switch. Ein NNI verbindet zwei ATM-Switches innerhalb der gleichen privaten Organisation, ein öffentlicher verbindet dagegen zwei ATM-Switches innerhalb der gleichen öffentlichen Organisation.

Eine weitere Spezifikation, die Broadband Interexchange Carrier Interconnect (B-ICI), verbindet zwei öffentliche Switches unterschiedlicher Dienstanbieter. Bild 18.4 verdeutlicht die ATM-Schnittstellen-Spezifikationen für private und öffentliche Netzwerke.

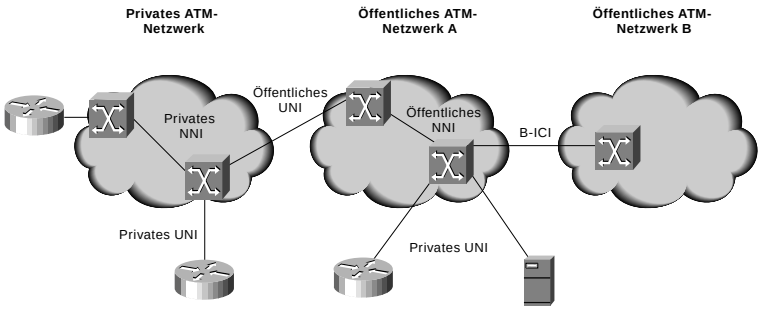


Bild 18.4: ATM-Schnittstellen-Spezifikationen unterscheiden sich für private und öffentliche Netzwerke

18.3 ATM-Zellenkopf-Format

Ein ATM-Zellenkopf kann eines der Formate UNI und NNI besitzen. Der UNI-Header wird für die Kommunikation zwischen ATM-Endpunkten und ATM-Switches in privaten Netzwerken verwendet. Der NNI-Header dient zur Übertragung zwischen ATM-Switches. Bild 18.5 zeigt das grundlegende ATM-Zellenformat, das ATM-UNI-Zellenkopfformat und das ATM-NNI-Zellenkopfformat.

Anders als bei UNI besitzt der NNI-Header kein General Flow Control-Feld (GFC). Außerdem beinhaltet der NNI-Header ein Feld Virtual Path Identifier (VPI), das die ersten 12 Bit einnimmt und größere Strecken zwischen öffentlichen ATM-Switches erlaubt.

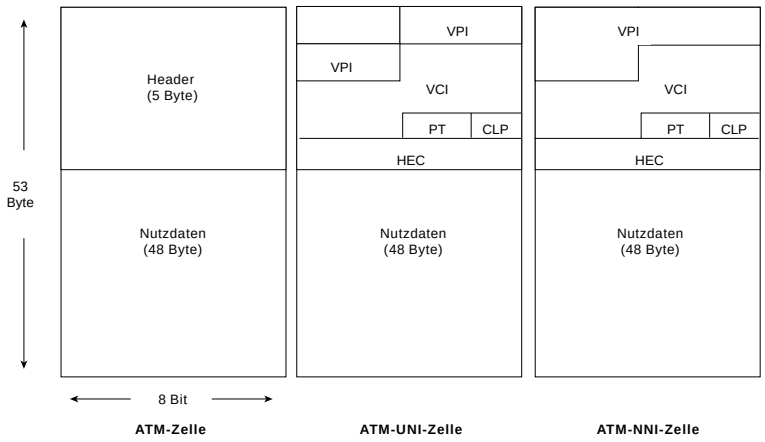


Bild 18.5: ATM-Zellen-, ATM-UNI-Zellen- und ATM-NNI-Zellen-Header mit jeweils 48 Byte Nutzdaten

18.3.1 Felder im ATM-Zellenkopf

Außer den Feldern GFC und VPI werden verschiedene weitere ATM-Zellenkopffelder eingesetzt. Die folgenden Beschreibungen behandeln die Kopffelder in Bild 18.5:

– *Generic Flow Control (GFC)* – stellt lokale Funktionen bereit, wie die Identifikation mehrerer Stationen, die sich eine einzelne ATM-Schnittstelle teilen. Dieses Feld wird gewöhnlich nicht verwendet und ist mit einem Voreinstellungswert belegt.

- *Virtual Path Identifier (VPI)* – identifiziert zusammen mit dem VCI das nächste Ziel einer Zelle, während diese auf dem Weg zum Ziel eine Reihe von ATM-Switches durchläuft.
- *Virtual Channel Identifier (VCI)* – identifiziert zusammen mit dem VPI das nächste Ziel einer Zelle, während diese auf dem Weg zum Ziel eine Reihe von ATM-Switches durchläuft.
- *Payload Type (PT)* – das erste Bit gibt an, ob die Zelle Benutzerdaten oder Steuerdaten enthält. Wenn die Zelle Benutzerdaten enthält, gibt Bit 2 die Blockierung an, Bit 3 beinhaltet, ob die Zelle die letzte in einer Folge von Zellen ist, die einen einzelnen AAL5-Frame bilden.
- *Congestion Loss Priority (CLP)* – gibt an, ob die Zelle verworfen werden soll, wenn sie im Verlauf der Übertragung durch das Netzwerk auf eine starke Blockierung trifft. Wenn das CLP-Bit gleich 1 ist, ist die Zelle eher zu verwerfen als Zellen, deren CLP-Bit gleich 0 ist.
- *Header Error Control* – berechnet eine Prüfsumme des Header selbst.

18.4 ATM-Dienste

Es gibt drei verschiedene ATM-Dienstarten: Permanent-Virtual-Connections (PVC), Switched-Virtual-Connections (SVC) und Connectionless-Services (ähnlich wie SMDS).

Eine PVC ermöglicht eine Direktverbindung zwischen Sites. Damit ist eine PVC mit einer Standleitung vergleichbar. Einer der Vorzüge der PVC ist die garantierte Verfügbarkeit einer Verbindung und daß sie keine Einrichtungsprozeduren zwischen Switches erfordert. Nachteil der PVC sind die statische Verbindung und das manuelle Setup.

Eine SVC wird dynamisch auf- und abgebaut und bleibt nur für die Dauer der Datenverbindung bestehen. In diesem Sinne ist sie einer Telefonverbindung sehr ähnlich. Die dynamische Anrufsteuerung erfordert ein Signalprotokoll zwischen ATM-Endpunkt und dem ATM-Switch. Zu den Vorteilen der SVC gehört eine hohe Verbindungsflexibilität und die automatische

Konfigurierbarkeit über Netzwerk-Geräte. Nachteil sind u.a. Zeitaufwand und Überhang für die Einrichtung der Verbindung.

18.4.1 ATM Virtual Connections

ATM-Verbindungen sind grundsätzlich verbindungsorientiert. Das bedeutet, es muß erst ein virtueller Kanal (VC) durch das Netzwerk eingerichtet werden, bevor Daten übertragen werden können. (Ein virtueller Kanal ist ungefähr mit einer virtuellen Schaltung vergleichbar.)

Es gibt zwei ATM-Verbindungstypen: virtuelle Pfade, die durch virtuelle Pfadidentifikatoren gekennzeichnet sind, und virtuelle Kanäle, die durch eine Kombination aus VPI und Virtual Channel Identifier (VCI) gekennzeichnet sind.

Ein virtueller Pfad ist ein Bündel virtueller Kanäle, die alle im ATM-Netzwerk transparent auf Basis des gemeinsamen VPI vermittelt werden. Alle VCIs und VPIs haben aber in einer speziellen Verbindung nur lokale Bedeutung und werden durch jeden Switch nach Bedarf umgeordnet.

Der Übertragungspfad ist ein Bündel von VPs. Bild 18.6 zeigt, wie VCs zu VPs verbunden werden, welche ihrerseits einen Übertragungspfad bilden.

Bild 18.6: VCs bilden VPs

18.5 ATM-Switch-Betrieb

Die Hauptaufgabe von ATM ist einfach: Die Zelle wird innerhalb einer Verbindung mit einem bekannten VCI- oder VPI-Wert empfangen. Der Switch sucht den Verbindungswert in einer lokalen Übersetzungstabelle, um den Ausgangsanschluß und den neuen VPI/VCI-Wert der Verbindung dieses Link festzustellen. Der Switch sendet die Zelle anschließend mit den entsprechenden Verbindungsidentifikatoren über diese Verbindung weiter. Weil alle VCIs und VPIs nur lokale Bedeutung

über eine spezielle Verbindung haben, werden diese Werte durch jeden Switch nach Bedarf verändert.

18.6 ATM-Referenzmodell

Die ATM-Architektur verwendet ein lokales Modell zur Beschreibung der unterstützten Funktionalität. ATM entspricht seinen Funktionen nach der physikalischen und einem Teil der Sicherungsschicht des OSI-Referenzmodells.

- Das ATM-Referenzmodell ist aus den folgenden Ebenen zusammengesetzt, die alle Schichten überspannen.
- Kontrolle – Diese Ebene ist für die Erstellung und Verwaltung von Signalisierungsanforderungen zuständig.
- Benutzer – Diese Ebene ist für Verwaltung des Datenverkehrs zuständig.
- Verwaltung – Diese besteht aus zwei Komponenten:
  - Die Schichten-Verwaltung bearbeitet Ebenenfunktionen wie die Ausfallerkennung und Protokollprobleme.
  - Die Ebenen-Verwaltung bearbeitet und koordiniert Funktionen des Gesamtsystems.

Das ATM-Referenzmodell besteht aus den folgenden ATM-Schichten:

- Physikalische Schicht – Analog zur physikalischen Schicht des OSI-Referenzmodells verwaltet die physikalische Schicht die medienabhängige Übertragung.
- ATM-Schicht – Ist in Verbindung mit der ATM-Anpassungsschicht der Sicherungsschicht des OSI-Referenzmodells vergleichbar. Die ATM-Schicht ist für die Einrichtung von Verbindungen und die Weiterleitung von Zellen durch das ATM-Netzwerk zuständig. Sie verwendet dafür die Informationen des Zellenkopfes.
- ATM-Anpassungsschicht (AAL) – Ist in Verbindung mit der ATM-Schicht der Sicherungsschicht des OSI-Referenzmodells vergleichbar. Der AAL ist zuständig für die Isolation der Protokolle höherer Schichten vom ATM-Prozeß.

Als Abschluß nehmen die höheren Schichten über dem AAL Benutzerdaten entgegen, verpacken sie in Pakete und übergeben diese an den AAL. Bild 18.7 zeigt das ATM-Referenzmodell.

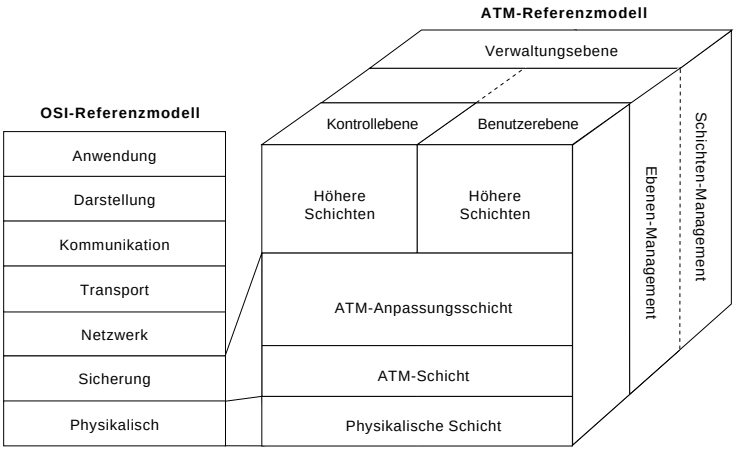


Bild 18.7: Das ATM-Referenzmodell entspricht den untersten beiden Schichten des OSI-Referenzmodells

18.6.1 ATM, Physikalische Schicht

Die physikalische Schicht von ATM erfüllt vier Funktionen: Bits werden in Zellen konvertiert, Empfang und Senden der Bits durch das physikalische Medium werden gesteuert, ATM-Zellenränder werden verfolgt, und die Zellen werden in den entsprechenden Frame-Typ des physikalischen Mediums verpackt.

Die physikalische ATM-Schicht besteht aus zwei Teilen: dem physikalischen Physical Medium-Dependent Sublayer (PMD) und dem Transmission Convergence (TC) Sublayer.

Der PMD stellt zwei Schlüsselfunktionen bereit. Er synchronisiert den Empfang durch Versand und Empfang eines Bitstroms mit zugehöriger Timing-Information. Zweitens gibt er das verwendete physikalische Medium an (einschließlich Anschlüssen und Kabel). Beispiele für physikalische Medienstandards für ATM sind Synchronous Optical Network/Synchronous Digital Hierarchy (SONET/SDH), DS-3/E3, 155 MBps oder Multimodefiber (MMF) unter Verwendung des 8B/10B-Kodierungsschemas über geschirmte Twisted-Pair-Kabel (STP).

Der TC-Sublayer erfüllt vier Funktionen: Zellendilineation, Header-Fehlerkontrolle (Folgengenerierung und -prüfung), Zellenfrequenzentkopplung und Senderahmenanpassung. Die Zellendilineation verwaltet ATM-Zellengrenzen, so daß Geräte die Zellen in einem Bitstrom erkennen können. Die Header-Fehlerkontrolle (Folgengenerierung und -prüfung) erstellt und prüft Header-Steuercodes zur Sicherung einer stabilen Datenübertragung. Die Zellenfrequenzentkopplung verwaltet die Synchronisation und ergänzt oder unterdrückt leere ATM-Zellen zur Anpassung der Frequenz der gültigen ATM-Zellen an die Kapazität des Übertragungssystems. Die Senderahmenanpassung verpackt ATM-Zellen in Frames, welche die jeweilige physikalische Umsetzung der physikalischen Schicht übertragen kann.

18.6.2 ATM-Anpassungsschichten: AAL1

AAL1 ist ein verbindungsorientiertes Gerät, das in der Lage ist, Anwendungen mit Simulation einer Schaltung zu verwalten, wie z.B. Sprach- und Video-Konferenzen. Die Schaltungsemulationsdienste bieten auch einen Anschluß von Ausrüstungen, die momentan Standleitungen zur Verbindung mit einem ATM-Backbone-Netzwerk einsetzen. AAL1 benötigt eine Timing-Synchronisation zwischen Quelle und Ziel. Deshalb ist AAL1 von einem Medium wie SONET abhängig, das eine Taktung unterstützt. Der AAL1-Prozeß bereitet Zellen in drei Schritten für die Übertragung vor. Zuerst werden synchrone Samples in das Nutzlastfeld eingefügt (1 Datenbyte mit einer Samplingrate von 125 µs). Zweitens werden die Felder Sequence Number (SN) und Sequence Number Protection (SNP) ergänzt, so daß der empfangende AAL1 mit Hilfe der Zusatzinformationen die Richtigkeit der Reihenfolge überprüfen kann. Der Rest des Nutzdaten-Feldes wird mit einzelnen Bytes auf 48 Byte aufgefüllt. Bild 18.8 zeigt, wie AAL1 die Zelle für die Übertragung vorbereitet.

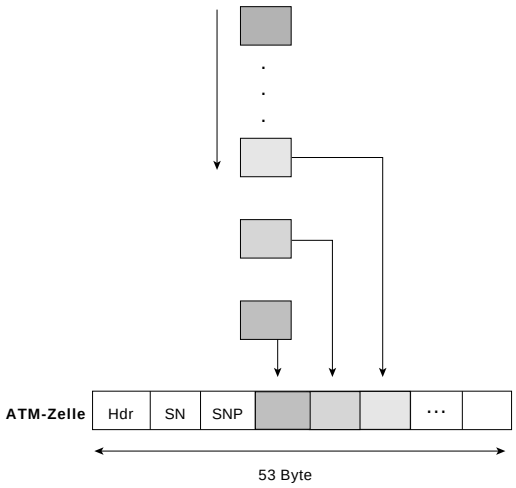


Bild 18.8: AAL1 bereitet die Zelle für die Übertragung vor

18.6.3 ATM-Anpassungsschichten: AAL3/4

AL3/4 unterstützt sowohl verbindungsorientierte als auch verbindungslose Daten. Es wurde für Netzwerk-Dienstanbieter entwickelt und steht in engem Zusammenhang mit dem Switched Multimegabit Data Service (SMDS). AAL3/4 wird für den Versand von SMDS-Paketen über ein ATM-Netzwerk eingesetzt.

AAL3/4 bereitet Zellen in drei Schritten für die Übertragung vor. Zuerst erzeugt der Convergence Sublayer eine Protocol Data Unit (PDU), indem er dem Frame einen Start/End-Tag-Header voranstellt und ein Längenfeld anhängt. Danach fragmentiert der Segmentation and Reassembly Sublayer (SAR) die PDU und stellt ihr einen Header voran. Dann ergänzt der SAR-Sublayer einen CRC-10-Anhang zur Fehlerkontrolle an jedes PDU-Fragment. Abschließend erhält die komplette SAR-PDU das Nutzlast-Feld einer ATM-Zelle, welcher der ATM-Schicht den Standard-ATM-Header voranstellt.

Eine AAL-3/4-SAR-PDU-Header enthält Felder für Typ, Sequenznummer und Multiplexing Identifier. Typ zeigt, ob eine Zelle Anfang, Fortsetzung oder Ende einer Meldung enthält. Die Sequenznummer gibt die Reihenfolge an, in der die Zellen wieder zusammengesetzt werden sollen. Der Multiplexing Identifier gibt an, welche Zellen aus unterschiedlichen Quellen am gleichen VCC verschachtelt sind, so daß am Ziel die richtigen Zellen wieder verbunden werden.

18.6.4 ATM-Anpassungsschicht: AAL5

AAL5 ist der primäre AAL für Daten und unterstützt sowohl verbindungsorientierte als auch verbindungslose Daten. Er wird für die Übertragung der meisten Nicht-SMDS-Daten über ATM und LAN-Emulation (LANE) verwendet, wie z.B. das klassische IP. AAL5 ist als eine einfache und effiziente Anpassungsschicht bekannt (SEAL), weil der SAR-Sublayer die CS-PDU einfach annimmt und ohne Ergänzung zusätzlicher Felder in 48-Byte-SAR-PDUs umwandelt.

AAL5 bereitet eine Zelle in fünf Schritten für die Übertragung vor. Als erstes ergänzt der CS-Sublayer den Frame mit einem Polster variabler Länge und einem 8 Byte langen Anhang. Das Polster stellt sicher, daß die entstandene PDU in die 48 Byte einer ATM-Zelle paßt. Der Anhang enthält die Länge des Frame und einen 32-Bit-CRC, der aus der gesamten PDU berechnet wurde. Dies ermöglicht es dem AAL5-Empfänger, Bitfehler, verlorene Zellen und Zellen, die in der falschen Reihenfolge ankommen, zu erkennen. Danach unterteilt der SAR-Sublayer die CS-PDU in 48 Byte lange Blöcke. Es wird kein Header oder Anhang ergänzt (wie etwa in AAL3/4), so daß keine Meldungen verschachtelt werden können. Abschließend verpackt die ATM-Schicht jeden Block in ein Nutzdaten-Feld einer ATM-Zelle. Für alle Zellen, bis auf die letzte, wird ein Bit im Feld Nutzdaten-Typ auf Null gesetzt, um anzuzeigen, daß die Zelle nicht die letzte innerhalb einer Serie ist, die einen einzelnen Frame darstellt. In der letzten Zelle ist dieses Bit auf 1 gesetzt.

18.7 ATM-Adressierung

Der ITU-T-Standard basiert auf der Verwendung von E.164-Adressen (ähnlich wie Telefonnummern) für öffentliche ATM-(BISDN-)Netzwerke. Das ATM-Forum erweiterte die ATM-Adressierung um den Einsatz in privaten Netzwerken. Es entschied sich dabei für das Subnet oder Overlay-Modell der Adressierung, in dem die ATM-Schicht für die Zuordnung von Netzwerk-Adressen zu ATM-Adressen verantwortlich ist. Das Sub-Netzwerk ist eine Alternative zum Einsatz von Protokolladressen auf Netzwerk-Schicht-Ebene (wie bei IPX und IP) und zu bereits vorhandenen Routing-Protokollen (wie IGRP

und RIP). Das ATM-Forum definiert ein Adreßformat auf Basis der Struktur von »Network Service Access Point«-Adressen (NSAP).

18.7.1 Sub-Netzwerk-Modell der Adressierung

Das Sub-Netzwerk-Modell der Adressierung entkoppelt die ATM-Schicht von anderen höheren Netzwerkschichten wie beispielsweise IP oder IPX. Als solches benötigt es ein völlig neues Adressierungsschema und Routing-Protokoll. Allen ATM-Systemen muß zusätzlich zu den Adressen der oberen Schichten eine ATM-Adresse zugewiesen sein. Dies erfordert ein *ATM address resolution protocol* (ATM-ARP), um die Adressen der oberen Schichten mit ihren ATM-Adressen in Verbindung bringen zu können.

18.7.2 NSAP-Format-ATM-Adresse

Die 20-Byte-NASP-Format-ATM-Adressen sind für den Gebrauch von privaten Netzwerken konzipiert, während in öffentlichen Netzwerken typischerweise E.164-Adressen verwendet werden, deren Formatierung von ITU-T definiert ist. Das ATM-Forum hat keine NSAP-Entschlüsselung für E.164-Adressen spezifiziert, was für die Entschlüsselung der E.164-Adressen in privaten Netzwerken benötigt wird.

Derartige private Netze können ihre eigene Adressierung im NSAP-Format auf Basis der E.164-Adresse des öffentlichen User-Network Interface (UNI) bilden, mit dem sie verbunden sind, und sie können ihr Adreßprefix aus der E.164-Nummer bilden, wobei lokale Knoten durch die niederwertigen Bits identifiziert werden.

Alle ATM-Adressen im NSAP-Format enthalten drei Bestandteile, den Authoritiy and Format Identifier (AFI), den Initial Domain Identifier (IDI) und den Domain Specific Part (DSP). Der AFI gibt Typ und Format des IDI an, welcher wiederum die Adreßzuordnung und Verwaltungsautorität angibt. Der DPS enthält aktuelle Routing-Informationen.

Die drei ATM-Adressierungsformate unterscheiden sich in Abhängigkeit von AFI und IDI. Im nach NSAP kodierten E.164-Format ist der IDI eine Nummer. Im DCC-Format ist der IDI ein Data Country Code (DCC), der bestimmte Länder

kennzeichnet, wie in der ISO 3166 festgelegt. Diese Adressen werden durch den ISO-National Members Body jedes Landes verwaltet. Im ICD-Format ist der IDI ein International Code Designator (ICD), der von der ISO-6523-Registrierungsbehörde (dem British Standards Institute) zugeordnet wird. ICD-Codes identifizieren bestimmte nationale Organisationen.

Das ATM-Forum empfiehlt, daß Organisationen und Dienstanbieter mit privaten Netzwerken entweder DCC oder ICD für die Erstellung eines eigenen Numerierungsplans verwenden.

Bild 18.9 zeigt die drei Formate für ATM-Adressen in privaten Netzwerken.

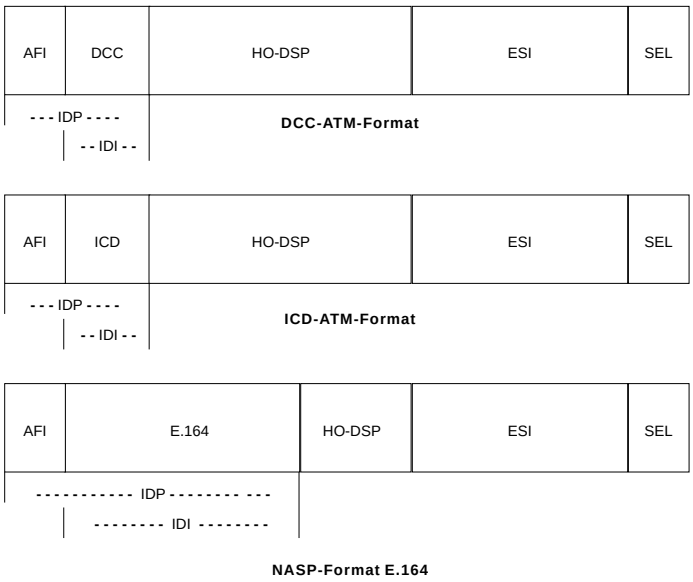


Bild 18.9: In privaten Netzwerken werden drei Formate von ATM-Adressen verwendet

18.7.3 ATM-Adreßfelder

Die folgenden Beschreibungen geben einen Überblick zu den in Bild 18.9 gezeigten Feldern:

Authority and Format Identifier (AFI) – Identifiziert Typ und Format der Adresse (E.164, ICD oder DCC).

Data Country Code (DCC) – Identifiziert bestimmte Länder.

High Order Domain Specific Part (HO-DSP) – Kombiniert die Routing Domain (RD) und Area Identifier (AREA) der NSAP-Adresse. Das ATM-Forum kombiniert diese Felder zur Unterstützung einer flexiblen Adressierungshierarchie auf mehreren Ebenen für Prefix-basierte Routing-Protokolle.

End System Identifier (ESI) – Legt die 46 Bit lange MAC-Adresse entsprechend dem Institute of Electrical and Electronic Engineers (IEEE) fest.

Selector (SEI) – Wird für die lokale Verteilung in den Endstationen verwendet und hat keine Bedeutung für das Netzwerk.

International Code Designator (ICD) – Gibt bestimmte internationale Organisationen an.

E.164 – BISDN-E.164-Adresse.

18.8 ATM-Verbindungen

ATM unterstützt zwei Verbindungstypen: Point-to-Point und Point-to-Multipoint.

Point-to-Point verbindet zwei ATM-Endsysteme und kann unidirektional (in einer Richtung) oder bidirektional (in beiden Richtungen) sein. Point-to-Multipoint verbindet ein einzelnes Quell-Endsystem (als Root-Knoten bezeichnet) mit mehreren Ziel-Endsystemen (als Leaves [Blätter] bezeichnet). Solche Verbindungen sind immer unidirektional. Root-Knoten können zu Leaves übertragen, aber Leaves können nicht mit dem Root oder miteinander über die gleiche Verbindung kommunizieren. Die Vervielfältigung der Zellen findet im ATM-Netzwerk durch die ATM-Switches statt, wenn die Verbindung auf zwei oder mehr Zweige aufgeteilt wird.

Es wäre wünschenswert, in ATM-Netzen bidirektionale Multipoint-to-Multipoint-Verbindungen aufbauen zu können. Solche Verbindungen wären mit den Broadcasting- oder Multicasting-Fähigkeiten von verteilten LANs, wie Ethernet und Token-Ring, vergleichbar. In verteilten LANs ist Broadcasting einfach zu realisieren, weil alle Knoten eines LAN-Segments alle in diesem Segment versendeten Pakete bearbeiten müssen. Mit AAL5 (der verbreitetsten ATM-Anpassungsschicht) kann eine solche Lösung zur Multipoint-to-Multipoint-Übertragung

leider nicht realisiert werden. Anders als AAL3/4 mit seinem Message Identifier Field, stellt AAL5 in seinem Zellenformat keine Möglichkeit bereit, Zellen aus verschiedenen AAL5-Paketen in einer einzigen Verbindung zu verschachteln. Dies bedeutet, daß alle AAL5-Pakete, die über eine bestimmte Verbindung an ein bestimmtes Ziel gesendet werden, nacheinander empfangen werden müssen. Sonst ist der Wiederherstellungsprozeß am Ziel nicht in der Lage, die Pakete wieder zu verbinden. Deshalb können AAL5-Point-to-Multicast-Verbindungen nur unidirektional sein. Wenn ein Leaf-Knoten ein AAL5-Packet in die Verbindung sendet, würde dieses beispielsweise von dem Root-Knoten und ebenso von den anderen Leaf-Knoten empfangen. Dabei würden sich Pakete, die vom Root-Knoten stammen, mit Paketen von anderen Leaf-Knoten überlagern, wodurch die Wiederherstellung aller verschachtelten Pakete unmöglich würde.

18.9 ATM und Multicasting

ATM benötigt eine Multicast-Fähigkeit. AAL5 (der verbreitetste Daten-AAL) unterstützt momentan keine verschachtelten Pakete und somit auch kein Multicasting.

Wenn ein Leaf-Knoten ein Paket an eine AAL5-Verbindung schickt, kann es vorkommen, daß das Paket mit anderen Paketen vermischt wird und falsch wieder zusammengesetzt wird. Es wurden drei Methoden zur Lösung dieses Problems vorgeschlagen: VP-Multicasting, Multicast-Server und Overlaid-Point-to-Multipoint-Verbindung.

Bei der ersten Lösung verbindet ein Multipoint-to-Multipoint-VP alle Knoten der Multicast-Gruppe. Jedem Knoten wird ein in diesem VP einmaliger Virtual Channel Identifier (VCI) zugeordnet. Von nun an können verschachtelte Pakete eindeutig nach dem VCI-Wert der Quelle identifiziert werden. Allerdings würde dieser Mechanismus ein Protokoll zur eindeutigen Zuordnung von VCI-Codewerten erfordern, und ein solches Protokoll existiert momentan nicht. Es ist auch unklar, ob vorhandene Segmentierungs- und Wiederherstellungsgeräte (SAR) einen solchen Betriebsmodus unterstützen könnten.

Ein Multicast-Server ist ein anderer Lösungsansatz für das Problem des Multicasting über ein ATM-Netzwerk. In diesem

Fall stellen alle Knoten, die in einer Multicast-Gruppe senden wollen, eine Verbindung zu einem externen Gerät, dem sogenannten Multicast-Server, her (der besser als Serilisierer oder Umordner beschrieben werden kann). Der Multicast-Server ist seinerseits an alle Knoten, die Multicast-Pakete empfangen wollen, über eine Point-to-Multicast-Verbindung angeschlossen. Der Multicast-Server empfängt Pakete über eine Point-to-Point-Verbindung und sendet diese anschließend über eine Point-to-Multipoint-Verbindung weiter, nachdem er sichergestellt hat, daß die Pakete serialisiert wurden (d.h. jedes Paket wird komplett übertragen, bevor das nächste gesendet wird). So wird eine Verschachtelung der Zellen ausgeschlossen.

Eine Overlaid-Point-to-Multipoint-Verbindung stellt die dritte Lösung für das Problem des Multicasting über ein ATM-Netzwerk dar. In diesem Fall stellen alle Knoten einer Multicast-Gruppe eine Point-to-Multipoint-Verbindung zu jedem anderen Knoten der Gruppe her und werden damit Leaves (Blätter) in den entsprechenden Verbindungen der anderen Knoten. Damit können alle Knoten miteinander kommunizieren. Bei dieser Lösung muß jeder Knoten eine Verbindung mit jedem sendenden Mitglied der Gruppe aufbauen, während der Multicast-Server-Mechanismus nur zwei Verbindungen erfordert. Es wäre außerdem ein Registrierungsprozeß erforderlich, der die Knoten informiert, die einer Gruppe von anderen Knoten beitreten, so daß die neuen Knoten die Point-to-Multipoint-Verbindung herstellen können. Die anderen Knoten müssen auch von dem neuen Knoten erfahren, so daß sie ihn zu ihren Point-to-Multipoint-Verbindungen ergänzen können. Der Multicast-Server-Mechansimus ist im Sinne der Verbindungsresourcen besser skalierbar, benötigt aber einen zentralen Umordner, der sowohl ein möglicher Engpaß als auch ein Punkt ist, dessen Ausfall einem Totalausfall gleichkommt.

18.10 ATM Quality of Service (QoS)

ATM unterstützt Garantien für die Qualität des Dienstes. Dazu gehört der Traffic-Contract, Traffic-Shaping und Traffic-Policing.

Ein Traffic-Contract legt einen Umschlag fest, der den geplanten Datenfluß beschreibt. Dazu gehört u.a. die maximale

Bandbreite, die mittlere ständige Bandbreite und die Burst-Größe. Wenn ein ATM-Endsystem eine Verbindung mit einem ATM-Netz aufbaut, geht es einen auf den QoS-Parametern basierenden Vertrag mit dem Netzwerk ein.

Als Traffic-Shaping bezeichnet man den Einsatz von Warteschlangen zur Einschränkung von Daten-Bursts, Begrenzung der Spitzenübertragungsrate sowie Glättung von Jittern, so daß der Verkehr die zugestandenen Rahmenbedingungen nicht überschreitet. ATM-Geräte sind für die Einhaltung des Vertrags im Sinne des Traffic-Shaping verantwortlich. ATM-Switches können Traffic-Policing einsetzen, um die Einhaltung des Vertrags zu erzwingen. Der Switch kann den Datenfluß messen und mit dem bei Verbindungsaufbau verhandelten Wert des Umschlags vergleichen. Stellt der Switch fest, daß der Datenfluß die abgesprochenen Parameter überschreitet, kann er die Cell-Loss-Priority-Bits (CLP) der betroffenen Zellen setzen. Dadurch wird es legalisiert, die Zelle zu verwerfen, was jeder folgende Switch in Blockierungssituationen ausführen kann.

18.11 ATM-Signalisierung und Verbindungsaufbau

Wenn ein ATM-Gerät eine Verbindung zu einem anderen ATM-Gerät aufbauen will, sendet es eine Signalisierungsanforderung an seinen direkt angeschlossenen ATM-Switch. Diese Anforderung enthält die ATM-Adresse des gewünschten ATM-Endpunkts sowie einige für die Verbindung notwendige QoS-Parameter.

ATM-Signalisierungsprotokolle sind vom Typ der ATM-Verbindung abhängig, es können User-Network-Interface-(UNI-) oder Network-Node-Interface-(NNI-)Signale sein. UNI wird zwischen ATM-Endsystem und ATM-Switch über ATM-UNI verwendet, während NNI bei NNI-Verbindungen eingesetzt wird.

Die UNI-3.1-Spezifikation des ATM-Forums ist der aktuelle Standard für ATM-UNI-Signalisierung. Die UNI-3.1-Spezifikation basiert auf dem Q.2931-Public-Network-Signaling-Protocol des ITU-T. UNI-Signalisierungsanforderungen werden mit der bekannten Standardverbindung übertragen: VPI=0, VPI=5.

Momentan existieren nur Standards für die ATM-UNI-Signalisierung, aber die Standardisierung wird mit NNI fortgesetzt.

18.11.1 ATM-Verbindungsaufbau

Die ATM-Signalisierung verwendet die One-Pass-Methode, die in allen modernen Telekommunikationsnetzen (wie dem Telefonnetz) eingesetzt wird. Ein ATM-Verbindungsaufbau läuft wie folgt ab. Als erstes sendet das Quell-Endsystem eine Anforderung in Form eines Verbindungssignals. Die Anforderung zum Verbindungsaufbau wird über das Netzwerk verbreitet. Infolgedessen wird über das Netzwerk eine Verbindung aufgebaut. Letztendlich erreicht die Anforderung das Ziel, welches die Verbindungsanforderung annehmen oder ablehnen kann.

18.11.2 Routing und Verhandlung der Verbindungsanforderung

Das Routing der Verbindungsanforderung wird durch ein ATM-Routing-Protokoll geregelt (dieses vermittelt Verbindungen aufgrund der Ziel- und Quelladressen). Der Datenverkehr und die QoS-Parameter dafür werden durch das Quell-Endsystem bereitgestellt. Die Verhandlung einer Verbindungsanforderung ist nur eingeschränkt möglich, weil das Call-Routing von den Anfangsparametern abhängig ist. Eine Änderung der Parameter kann Rückwirkungen auf das Routing der Verbindung haben. Bild 18.10 erklärt die One-Pass-Methode für den ATM-Verbindungsaufbau.

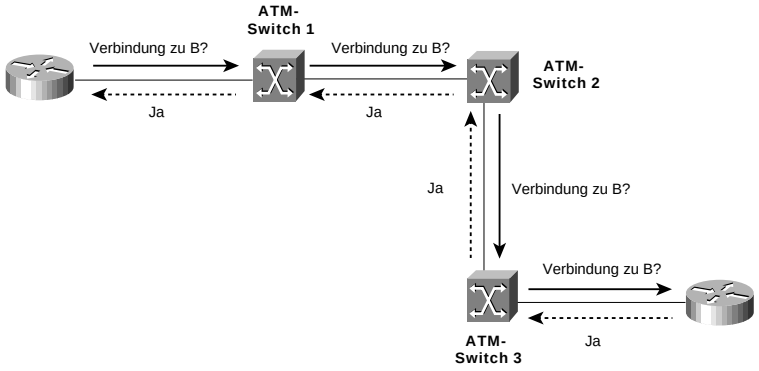


Bild 18.10: ATM-Geräte bauen mit Hilfe einer One-Pass-Methode Verbindungen auf

18.12 ATM-Meldungen für die Verbindungsverwaltung

Für den Aufbau und Abbruch einer ATM-Verbindung wird eine Gruppe von Meldungen verwendet, zu der auch Setup, Call-Proceeding, Connect und Release gehört. Das Quell-Endsystem sendet eine Setup-Meldung (mit der Adresse des Zielendsystems und den QoS-Parametern), wenn es eine Verbindung aufbauen will. Der Anschluß-Switch aus dem Netz sendet als Antwort auf diese Setup-Meldung eine Call-Proceeding-Meldung zurück. Als nächstes sendet das Ziel-Endsystem eine Connect-Meldung, wenn die Verbindung angenommen wurde oder eine Release-Meldung, wenn die Verbindung abgelehnt wurde. Es gibt damit die Verbindung wieder frei.

Die Meldungen für die Verbindungsverwaltung werden wie folgt für den Aufbau einer ATM-Verbindung verwendet: Als erstes sendet das Quell-Endsystem eine Setup-Meldung, welche an den ersten ATM-Switch (den Anschluß-Switch) des Netzwerks weitergeleitet wird. Dieser Switch sendet daraufhin eine Call-Proceeding-Meldung und ruft ein ATM-Routing-Protokoll auf. Die Signalisierungsanforderung wird über das Netz verbreitet. Der Ausgangs-Switch aus dem Netz (auch als Egress-Switch bezeichnet), der mit dem Ziel-Endsystem verbunden ist, empfängt die Setup-Meldung. Der Egress-Switch leitet die Setup-Meldung über sein UNI an das Ziel-Endsystem weiter, worauf dieses eine Connect-Meldung sendet, wenn es die Verbindung annimmt. Die Connect-Meldung durchquert das Netz auf demselben Weg zurück zum Quell-Endsystem, welches eine Connect-Acknowledge-Meldung zurück an das Ziel sendet und damit die Verbindung bestätigt. Danach kann die Datenübertragung beginnen.

18.13 LAN-Emulation (LANE)

Die LAN-Emulation ist ein Standard des ATM-Forums, der über ATM verbundenen Stationen die gleichen Möglichkeiten bieten soll, wie die gewöhnlichen LANs (z.B. Ethernet und Token Ring). Wie der Name bereits andeutet, besteht die Aufgabe des LANE-Protokolls in der Emulation eines LAN auf Basis eines ATM-Netzwerks. Das LANE-Protokoll stellt Mechanismen zur Emulation eines »IEEE 802.3«-Ethernet

oder eines 802.5-Token-Ring-LAN bereit. Das aktuelle LANE-Protokoll sieht keine separate Kapselung für FDDI vor (d.h., FDDI-Pakete müssen mit Hilfe von Translational-Bridges in Ethernet oder Token-Ring-LAN umgewandelt werden). Fast Ethernet und IEEE 802.12 (100VG-AnyLAN) können beide ohne Änderung abgebildet werden, weil sie das gleiche Paketformat verwenden. Bild 18.11 stellt echtes LAN und ELAN gegenüber.

Das LANE-Protokoll legt eine Diensteschnittstelle für die höhere Schicht fest (d.h. die Netzwerk-Schicht), welche identisch mit der vorhandener LANs ist. Daten werden für den Versand über das ATM-Netzwerk in das entsprechende LAN-MAC-Paketformat verpackt. Vereinfacht gesagt, das LANE-Protokoll läßt ein ATM-Netzwerk so aussehen und auftreten, wie ein Ethernet oder Token-Ring-Netzwerk, obwohl es wesentlich schneller ist als aktuelle Ethernet oder Token-Ring-Netzwerke.

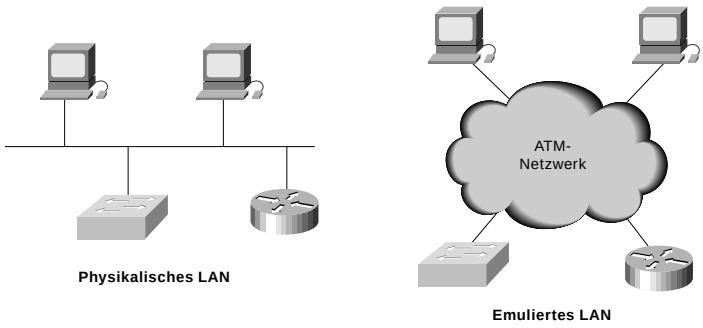


Bild 18.11: ATM-Netzwerke können ein echtes LAN emulieren

Es ist wichtig zu verstehen, daß LANE nicht versucht, das jeweilige MAC-Protokoll des vorhandenen LAN zu emulieren (d.h. CSMA/CD für Ethernet oder Token-Passing für IEEE 802.5). LANE verlangt keine Anpassung von Protokollen höherer Schichten für deren Betrieb über ein ATM-Netzwerk. Weil der LANE-Dienst der Netzwerk-Schicht die gleichen Schnittstellen wie die vorhandenen MAC-Protokolle bereitstellt (wie NDIS- oder ODI-artige Treiberschnittstellen), sind an diesen Treibern keine Änderungen erforderlich.

18.13.1 LANE-Protokoll-Architektur

Die Hauptaufgabe des LANE-Protokolls ist die Umwandlung der MAC-Adressen in ATM-Adressen. Das Ziel dabei ist, die Adressen so zuzuordnen, daß LANE-Endsysteme Direktverbindungen für die Datenübertragung untereinander einrichten können, um dann die Daten weiterzuleiten. Das LANE-Protokoll wird durch zwei an ATM angeschlossene Ausrüstungen eingesetzt: ATM-Netzwerkkarten (engl. network interface cards, kurz NIC) und netzübergreifende und LAN-Switching-Technik.

ATM-NICs setzen das LANE-Protokoll und die Schnittstelle zum ATM-Netz um, stellen dabei den angeschlossenen Endsystemen die jeweilige LAN-Diensteschnittstelle zu Protokolltreibern für die höhere Schicht bereit. Die Netzwerk-Schicht-Protokolle der Endsysteme kommunizieren miteinander, als wären sie über ein gewohntes LAN unter Verwendung der gewohnten Prozeduren verbunden. Allerdings sind sie in der Lage, die wesentlich größere Bandbreite des ATM-Netzwerks auszunutzen.

Die zweite Klasse von Netzwerk-Ausrüstungen, die LANE einsetzen, sind mit ATM verbundene LAN-Switches und Router. Diese Geräte werden im Verbund mit direkt angeschlossenen und mit ATM-NICs ausgerüsteten ATM-Hosts verwendet, um unabhängig vom wirklichen Ort einen virtuellen LAN-Dienst anzubieten, bei dem den Anschlüssen der LAN-Switches einzelne virtuelle LANs zugewiesen werden. Bild 18.12 zeigt die LANE-Protokoll-Architektur, wie sie in ATM-Netzwerk-Ausrüstungen eingesetzt wird.

Das LANE-Protokoll nimmt keinen Einfluß auf ATM-Switches. LANE ist wie die meisten anderen ATM-Netzwerkprotokolle nach dem Overlay-Modell aufgebaut, d.h., die LANE-Protokolle funktionieren transparent durch und über ATM-Switches, wobei sie nur die standardisierten ATM-Signalisierungsprozeduren verwenden.

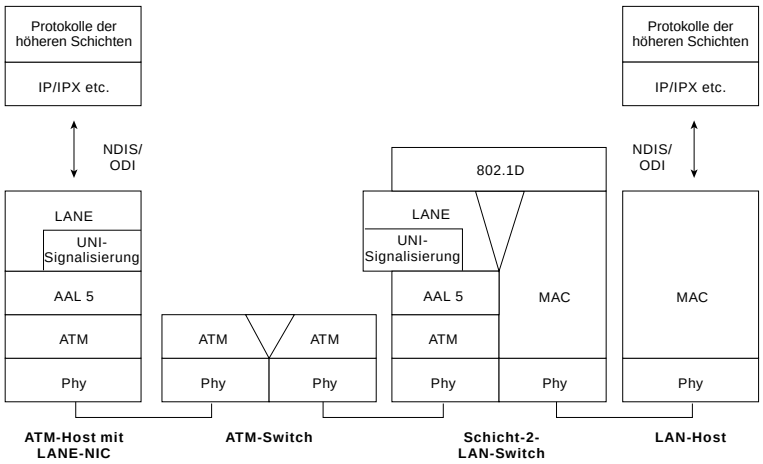


Bild 18.12: Die LANE-Protokoll-Architektur, kann in ATM-Netzwerk-Ausrüstungen eingesetzt werden

18.13.2 Bestandteile des LANE

Das LANE-Protokoll definiert die Funktion eines einzelnen emulierten LAN (ELAN). (Ein ELAN ist ein Äquivalent zum virtual LAN [VLAN].) Obwohl in einem ATM-Netzwerk mehrere LANs vorhanden sein können, emuliert ein ELAN entweder ein Ethernet oder ein Token-Ring-Netz. Es besteht aus den folgenden Bestandteilen:

– *LAN Emulation Client (LEC)* – Der LEC ist ein Element eines Systems, das für die Datenweiterleitung, Adreßauflösung und Registrierung von MAC-Adressen des LAN-Emulation-Server (LES) zuständig ist. Der LEC stellt gewöhnlichen LANs außerdem eine LAN-Standardschnittstelle zu höheren Protokollebenen bereit. Ein ATM-Endsystem verwendet für die Verbindung mit mehreren ELANs je einen LEC pro LAN.

– *LAN-Emulation Server (LES)* – der LES stellt den LECs einen zentralen Steuerungspunkt zur Weiterleitung von Registrierungen und Steuerinformationen bereit. (Pro ELAN ist nur ein LES vorhanden.)

– *Broadcast and Unknown Server (BUS)* – Der BUS ist ein Multicast-Server, der für die Verteilung von Verkehr mit unbekannter Zieladresse sowie Multicast- und Broadcast-Verkehr an die Clients eines bestimmten LAN zuständig ist.

– *LAN Emulation and Configuration Server (LECS)* – Der LECS verwaltet eine Datenbank von LECs und der LANs,

zu denen diese gehören. Der Server nimmt Anfragen von LECs an und antwortet mit dem entsprechenden LAN-Identifikator, d.h. der ATM-Adresse des LES, der das entsprechende ELAN verwaltet. Ein LECS pro administrativer Domain verwaltet alle ELANs innerhalb dieser Domain.

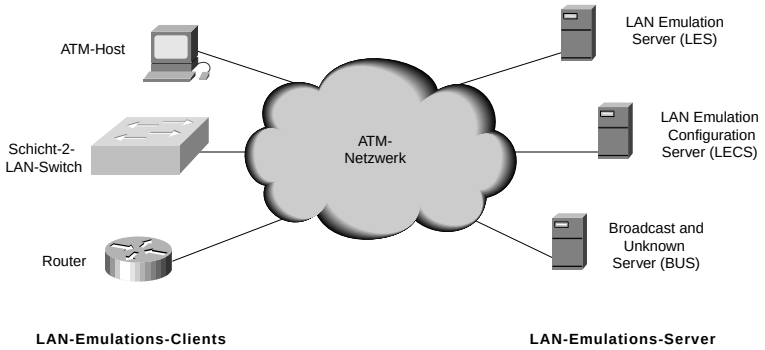


Bild 18.13: Ein ELAN besteht aus Clients, Servern und verschiedenen Zwischenknoten

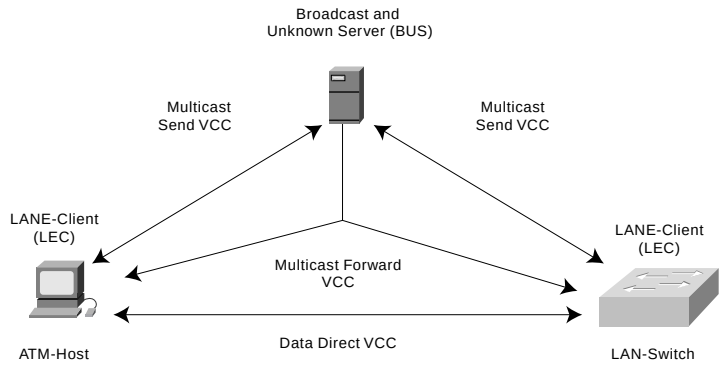
18.13.3 Verbindungsarten der LAN-Emulation

Die LANE-Elemente der Phase 1 kommunizieren mit Hilfe einer Gruppe von ATM-Virtual Circuit Connections (VCCs) miteinander. LECs verwalten die getrennten Verbindungen für den Daten- und Steuerungsverkehr. Die LANE-Datenverbindungen sind Data-Direct-VCC, Multicast-Send-VCC und Multicast-Forward-VCC.

Data-Direct-VCC ist ein bidirektionaler Point-to-Point-VCC, der zwischen zwei LECs eingerichtet wird, die Daten austauschen wollen. Zwei LECs verwenden üblicherweise den gleichen Data-Direct-VCC zur Übertragung aller Pakete zwischen ihnen, statt einen neuen VCC für jedes MAC-Adreßpaar zu öffnen. Diese Technik spart Ressourcen beim Verbindungsaufbau und der Einrichtung.

Multicast-Send-VCC ist ein bidirektionaler Point-to-Point-VCC, der vom BUS für den LEC eingerichtet wird.

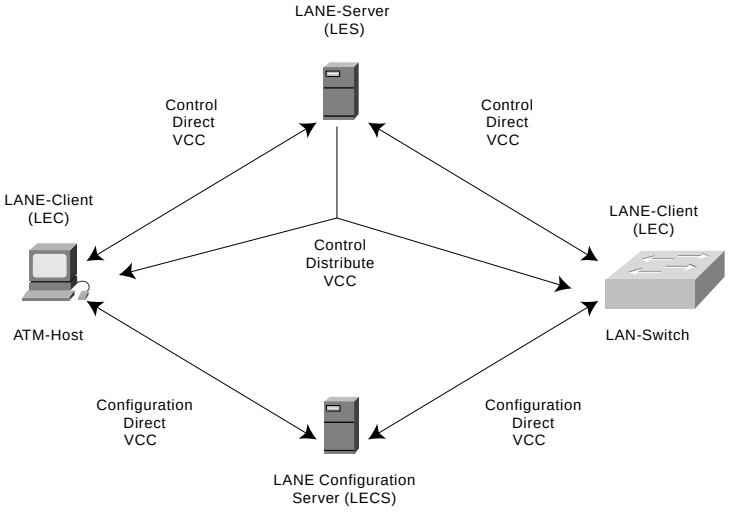
Multicast-Forward-VCC ist ein unidirektionaler VCC, der vom BUS für den LEC eingerichtet wird. Es handelt sich dabei üblicherweise um eine Point-to-Multipoint-Verbindung, mit einem LEC als Leaf. Bild 18.14 zeigt die LANE-Datenverbindungen.



LAN-Emulations-Datenverbindungen

Bild 18.14: LANE-Datenverbindungsserver verwenden eine Gruppe virtueller Schaltungsverbindungen zur Verbindung von LAN-Switch und ATM-Hosts

Zu den Steuerungsverbindungen gehören Configuration-Direct-VCC, Control-Direct-VCC und Control-Distribute-VCC. Configuration-Direct-VCC ist ein bidirektionaler Point-to-Point-VCC, der vom LEC zum LECS eingerichtet wird. Control-Direct-VCC ist ein bidirektionaler VCC, der vom LEC zum LECS eingerichtet wird. Control-Distribute-VCC ist ein unidirektionaler VCC, der vom LES zurück zum LEC führt (üblicherweise eine Point-to-Multipoint-Verbindung). Bild 18.15 zeigt die LANE-Steuerungsverbindungen.



LAN-Emulations-Steuerverbindungen

Bild 18.15: LANE-Steuerungsverbindungen verbinden LES, LECS, LAN-Switch und ATM-Host

18.13.4 LANE im Betrieb

Den Betrieb eines LANE-Systems und dessen einzelner Bestandteile versteht man am besten, wenn man die folgenden Abschnitte des LEC-Betriebs einzeln betrachtet: Initialisierung und Konfiguration, Beitritt und Registrierung beim LES, Suche und Beitritt zum BUS und Datenverkehr.

Initialisierung und Konfiguration

Nach der Initialisierung sucht ein LEC den LECS, um die benötigten Konfigurationsinformation zu erhalten. Es beginnt mit diesem Vorgang, sobald der LEC seine eigene ATM-Adresse erhält, was üblicherweise während der Adreßregistrierung geschieht.

Dann muß der LEC den LECS finden. Dazu muß der LEC den LECS mit einer der folgenden Methoden suchen: mit Hilfe einer vorgegebenen ILMI-Prozedur zur Bestimmung der LECS-Adresse, mit Hilfe einer bekannten LECS-Adresse oder über eine bekannte ständige Verbindung zum LECS (VPI=0, VCI=17).

Wenn der LECS gefunden ist, richtet der LEC einen Configuration-Direct-VCC zum LECS ein und sendet ein LE_CONFIGURE_REQUEST. Wenn ein passender Eintrag gefunden wurde, antwortet der LECS dem LEC mit einem LE_CONFIGURE_RESPONSE und den Konfigurationsinformationen, die dieser für die Verbindung zu seinem Ziel-ELAN benötigt. Dazu gehört: ATM-Adresse des LES, der Typ des zu emulierenden LAN, die maximale Paketgröße im ELAN und der Name des ELAN (ein Kurztext für Anzeigezwecke).

Beitritt und Registrierung beim LES

Wenn ein LEC dem LES beitritt und seine eigenen ATM- und MAC-Adressen registriert, geschieht dies wie folgt:

1. Nachdem der LEC die LES-Adresse erhalten hat, gibt er wahlweise die Verbindung zum LECS frei, richtet einen Control-Direct-VCC zum LES ein und sendet einen LE_JOIN_REQUEST an diesen VCC. Dies ermöglicht es dem LEC, seine eigenen ATM- und MAC-Adressen bei dem LES zu registrieren und (wahlweise) weitere MAC-Adressen, für die er als Proxy läuft. Diese Informationen werden

so verwaltet, daß kein neuer LEC die gleichen MAC- oder ATM-Adressen registriert.

2. Nach dem Empfang des LE_JOIN_REQUEST prüfen LEC und LES die offene Verbindung, vergleichen die Anforderung und bestätigen die Mitgliedschaft des Client.

3. Nach einem erfolgreichen Vergleich fügt der LES den LEC als Leaf seines Point-to-Multipoint-Control-Distribute-VCC ein und sendet dem LEC eine LE_JOIN_RESPONSE mit einer einmaligen LAN-Emulations-Client-ID (LECID). Die LECID wird vom LEC für die Filterung seiner eigenen Broadcasts aus dem BUS benötigt.

Suche und Beitritt zum BUS und Datenverkehr

Nachdem der LEC dem LECS beigetreten ist, besteht seine erste Aufgabe in der Suche der BUS-ATM-Adresse, um der Broadcast-Gruppe beizutreten und Mitglied des ELAN zu werden. Als erstes erzeugt der LEC ein LE_ARP_REQUEST-Paket mit der MAC-Adresse 0xFFFFFFFF. Dann sendet der LEC dieses spezielle LE_ARP-Paket auf dem Control-Direct-VCC an den LES. Der LES erkennt, daß der LEC den BUS sucht, und antwortet mit der ATM-Adresse auf dem Control-Distribute-VCC.

Sobald der LEC die ATM-Adresse des BUS erhalten hat, tritt er diesem BUS bei, indem er zuerst ein Signalpaket mit der Adresse des BUS erstellt und ein Multicast-Send-VCC mit dem BUS aufbaut. Nach Empfang des Signals fügt der BUS den LEC als weiteres Leaf zu seiner Point-to-Multipoint-Multicast-Forward-VCC hinzu. Der LEC ist nun ein Mitglied des ELAN und bereit für die Datenübertragung.

Datenübertragung

Das Endstadium, der Datentransfer, beinhaltet die Auflösung der ATM-Adresse des Ziel-LEC und die eigentliche Datenübertragung, zu der auch die Flush-Prozedur gehören kann.

Wenn ein LEC ein Datenpaket an eine unbekannte Ziel-MAC-Adresse sendet, muß er die ATM-Adresse des Ziel-LEC herausfinden, über die diese spezielle Adresse erreicht werden kann. Dazu sendet der LEC den Daten-Frame zuerst an den BUS (per Multicast-Send-VCC), damit dieser ihn per Multi-

cast-Forward-VCC an alle LECs des ELAN sendet. Dies geschieht, weil eine Auflösung der ATM-Adresse Zeit kosten würde und viele Netzwerk-Protokolle keine Verzögerung tolerieren.

Danach sendet der LEC einen Steuerungs-Frame mit einem LAN-Emulation Address Resolution Protocol Request (LE_ARP_Request) per Control-Direct-VCC an den LEC.

Wenn der LEC die Antwort hat, antwortet er mit der ATM-Adresse des LEC, der die gesuchte MAC-Adresse besitzt. Wenn der LEC die Antwort nicht kennt, gibt er die Anfrage an einige oder auch alle LECs weiter (nach Regeln, die der Verteilung der jeweiligen Daten-Frames über den BUS ähnlich sind, aber per Control-Direct und Control-Distribute-VCCs, statt der Multicast-Send- und -Forward-VCCs des BUS). Wenn Bridges/Switches mit LEC-Software im ELAN eingebunden sind, dann übersetzen diese den ARP und leiten ihn an ihre LAN-Schnittstellen weiter.

Im Falle eines aktuellen Datentransfers richtet der LEC auf den Empfang eines LE_ARP eine Data-Direct-VCC zum Zielknoten ein und verwendet diese anstelle des Buspfads für die Datenübertragung. Bevor dies geschehen kann, ist möglicherweise die LANE-Flush-Prozedur auszuführen.

Die LANE-Flush-Prozedur stellt sicher, daß alle Pakete, die vorher an den BUS gesendet wurden, am Ziel abgeliefert wurden, bevor der Data-Direct-VCC verwendet wird. Im Verlauf der Flush-Prozedur wird dem letzten Paket eine Steuerungszelle nachgeschickt. Der LEC wartet danach, bis das Ziel den Empfang des Flush-Pakets bestätigt, bevor es den zweiten Pfad zum Senden der Pakete einsetzt.

KAPITEL 19

Data-Link-Switching

19.1 Grundlagen

Data-Link-Switching (DLSw) ist ein Hilfsmittel zur Übertragung von IBM *Systems Network* Architecture (SNA) und *Network Basic Input/Output System* (NetBIOS)-Datenverkehr über ein IP-Netzwerk. Es dient als Alternative zum *Source Route Bridging* (SRB), einem Protokoll für die Übertragung von SNA- und NetBIOS-Verkehr in Token-Ring-Umgebungen, das vor der Einführung von DLSw sehr verbreitet war. Grundsätzlich dient DLSw zur Behebung einiger Fehler von SRB bei bestimmten Übertragungsanforderungen – speziell bei Anwendung in WANs. Dieses Kapitel enthält eine Gegenüberstellung von DLSw und SRB sowie eine Zusammenfassung zu den darunterliegenden Protokollen und stellt eine Übersicht gebräuchlicher Protokolloperationen bereit.

DLSw wurde als firmeneigene Lösung von IBM im Jahre 1992 entwickelt. Es wurde dem IETF im Jahr 1993 erstmals als RFC 1434 vorgelegt. Die detaillierte Dokumentation von DLSw ist heute im IETF RFC 1795 festgelegt, der im April 1995 vorgestellt wurde. DLSw ist eine gemeinsame Entwicklung des *Advanced Peer-to-Peer Networking* (APPN), *Implementors Workshop* (AIW) und der *Data-Link Switching Related Interest Group* (DLSw RIG).

RFC 1795 beschreibt die drei Primärfunktionen von DLSw:

- Das Switch-to-Switch Protocol (SSP) ist das Protokoll, das zwischen zwei DLSw-Knoten oder Routern ausgehandelt wird.
- Der Abschluß von SNA »Data-Link Control«-Verbindungen (DLC) führt zu einer Verringerung der Wahrscheinlichkeit von Timeouts innerhalb der Verbindungsschicht (engl. link-layer) von WANs.
- Die lokale Zuweisung von DLC-Verbindungen an eine DLSw-Schaltung.
- Alle diese Funktionen werden im folgenden Kapitel im Detail behandelt.

Bild 19.1 zeigt eine verallgemeinerte DLSw-Umgebung.

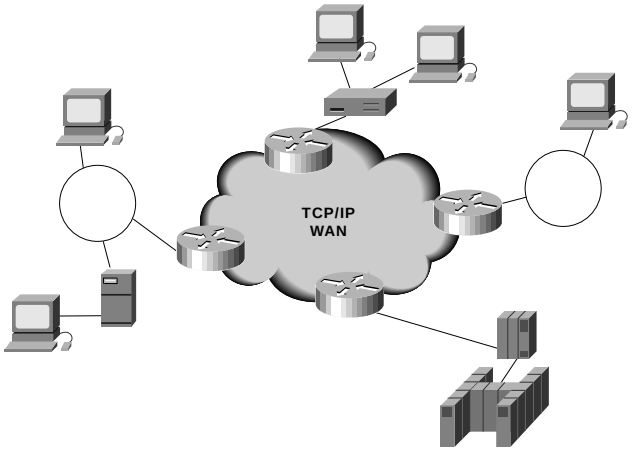


Bild 19.1: Ein DLSw-Kreis erleichtert SNA-Connectivity über ein IP-WAN

19.2 DLSw im Vergleich mit Source-Route-Bridging

Der grundsätzliche Unterschied zwischen Source-Route-Bridging (SRB) und DLSw liegt in der Unterstützung des lokalen Abschlusses. Der SNA- und NetBIOS-Verkehr ist auf Empfangsbestätigungen und Meldungen zur Erhaltung der Verbindung in der Verbindungsschicht angewiesen, um die Integrität

der Verbindung und Datenübertragung zu gewährleisten. Für die Verbindungsdaten schließt der lokale DLSw-Knoten oder Router die Steuerung der Datenverbindung ab. Infolgedessen wird das WAN nicht durch Empfangsbestätigungen und Meldungen zur Erhaltung der Verbindung belastet. Im Gegensatz dazu funktioniert DLC für SRB auf einer Ende-zu-Ende-Basis, was die Wahrscheinlichkeit von DLC-Timeouts bei WAN-Verbindungen erhöht.

Obwohl Source-Route-Bridging in vielen Umgebungen eine realisierbare Lösung darstellt, ist der Nutzen von SRB für die Übertragung bei SNA und NetBIOS bei WAN-Anwendungen beschränkt. Die folgenden Nachteile stechen dabei besonders hervor:

– Einschränkung des SRB Hop-Counts auf 7 Hops (Sprünge)
– Punkt-zu-Mehrpunkt-Betriebs-Behandlung (von SRB Explorer-Frames oder NetBIOS-Namensanfragen)
– Weiterleitung von überflüssigem Verkehr (Empfangsbestätigungen und Meldungen zur Erhaltung der Verbindung)
– Fehlende Flußsteuerung und Prioritätensetzung

Bild 19.2 zeigt eine SRB-Verbindung über ein WAN (Ende-zu-Ende).

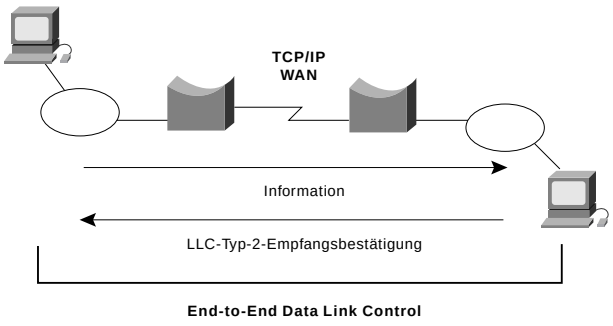


Bild 19.2: SRB bietet eine Ende-zu-Ende-Verbindung über ein IP-WAN

Der lokale Abschluß von DLC-Verbindungen durch DLSw ist in vielerlei Hinsicht vorteilhafter als SRB-basierte Umgebungen. Der lokale Abschluß von DLSw beseitigt die Notwendigkeit von Empfangsbestätigungen und Meldungen zur Erhal-

tung der Verbindung innerhalb der Verbindungsschicht des WAN. Weiterhin verringert der lokale Abschluß die Wahrscheinlichkeit von Timeouts. Außerdem stellt DLSw sicher, daß die Verbreitung von Explorer-Frames (Such-Frames) durch DLSw gesteuert wird, sobald das Zielsystem gefunden wurde. Bild 19.3 zeigt den Informationsfluß und die Verwendung lokaler Empfangsbestätigungen in einer DLSw-Umgebung.

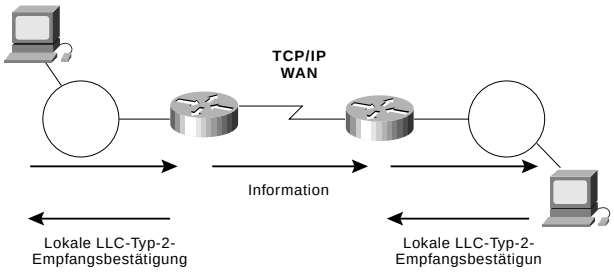


Bild 19.3: DLSw verwendet lokale Empfangsbestätigungen zur Steuerung des Datenflusses

19.3 DLSw-SNA-Unterstützung

Als weiteren Vorteil bringt DLSw eine breitere Geräte- und Medienunterstützung, als bisher mit SRB verfügbar war. DLSw schließt eine Reihe von typischen SNA-Umgebungen ein und stellt eine LAN-Unterstützung nach IEEE 802.2 bereit, zu der auch die Unterstützung von SNA-»Physical Unit« (PU) 2, PU 2.1 und PU-4-Systemen und NetBIOS-basierten Systemen gehört.

DLSw unterstützt Synchronous Data-Link Control (SDLC) und deckt damit PU-2- (primary und secondary) und PU-2.1-Systeme ab. Bei SDLC-verbundenen Systemen stellt jede SDLC-PU für das DLSw-Switch-to-Switch-Protocol (SSP) ein einmaliges Media Access Control (MAC)/service access point (SAP)-Adreßpaar dar. In durch Token Ring verbundenen Systemen erscheinen DLSw-Knoten als Source-Route-Bridge. Entfernte Token-Ring-Systeme, auf die über einen DLSw-Knoten zugegriffen wird, werden als benachbarter Ring angesehen. Dieser Nachbarring wird als virtueller Ring bezeichnet und

innerhalb jedes DLSw-Knotens erzeugt. Bild 19.4 zeigt verschiedene IBM-Knoten, die über ein TCP/IP-WAN mit DLSw-Geräten verbunden sind, wobei es sich in diesem Fall um Router handelt.

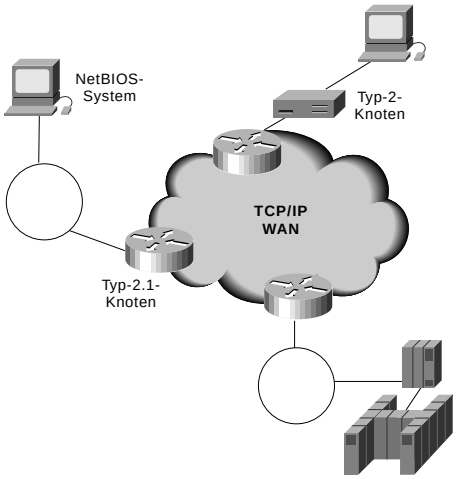


Bild 19.4: Verbindung von SNA-Knoten mit einem TCP/IP-WAN über DLSw

19.4 DLSw-Switch-to-Switch Protocol (SSP)

Das Switch-to-Switch Protocol (SSP) wird als Protokoll zwischen DLSw-Knoten (Routern) für den Aufbau von Verbindungen, das Lokalisieren von Ressourcen, die Weiterleitung von Daten sowie für Flußsteuerung und Fehlerkorrektur verwendet. Dies sind die wesentlichen Besonderheiten von DLSw. Im allgemeinen unterstützt SSP kein komplettes Routing zwischen Knoten, weil dies grundsätzlich durch allgemeine Routing-Protokolle, wie RIP, OSPF oder IGRP/EIGRP abgedeckt wird. Statt dessen vermittelt SSP Pakete auf der SNA-Sicherungsschicht. Es verpackt außerdem Pakete per TCP/IP für die Übertragung in IP-basierten Netzwerken und verwendet TCP als Hilfsmittel für die zuverlässige Übertragung zwischen DLSw-Knoten. Bild 19.5 zeigt die Einordnung von SSP innerhalb der SNA-Architektur sowie im Verhältnis zum OSI-Referenzmodell.

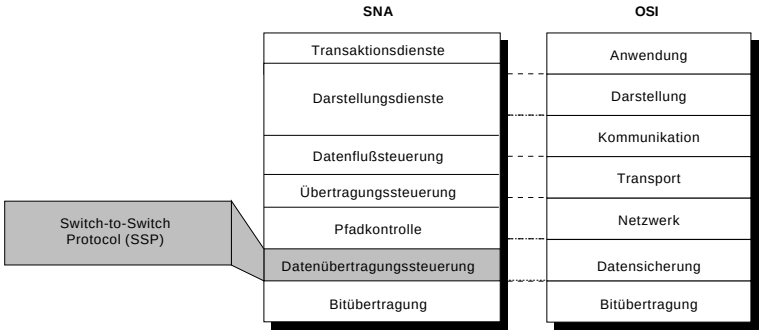


Bild 19.5: SSP wird den Datensicherungskomponenten von SNA und OSI-Referenzmodell zugeordnet

19.5 DLSw-Betrieb

DLSw beinhaltet verschiedene Betriebszustände. Zwei DLSw-Partner bauen zwei TCP-Verbindungen miteinander auf. Diese TCP-Verbindung ist die Grundlage der DLSw-Übertragung. Da TCP eine zuverlässige und garantierte Ankunft des IP-Verkehrs garantiert, stellt es auch die Übertragung und Integrität des durch das IP-Protokoll gekapselten Datenverkehrs sicher, welcher in diesem Fall SNA- und NetBIOS-Verkehr ist. Nach dem Aufbau einer Verbindung tauschen die DLSw-Partner eine Liste der unterstützten Fähigkeiten aus. Dies ist besonders wichtig, wenn die DLSw-Partner von unterschiedlichen Herstellern stammen. Als nächstes schalten die DLSw-Partner die SNA- oder NetBIOS-Endsysteme durch, so daß der Informationsfluß über die Durchschaltung hergestellt ist.

19.5.1 DLSw-Prozesse

Der gesamte DLSw-Betrieb kann in drei Grundkomponenten aufgeteilt werden: *Austausch der Fähigkeiten, Herstellung der Durchschaltung und Flußsteuerung.* Der Austausch der Fähigkeiten der DLSws schließt die Verhandlung von Informationen über Fähigkeiten der DLSw-Sitzungen ein. Dieser Informationsaustausch wird bei der Initialisierung der Sitzung und im Verlauf von Sitzungsoperationen ausgehandelt. Die Durchschaltung der DLSws findet zwischen den Abschlußsystemen statt. Dazu gehört die Zuordnung des Endsystems des Ziels und die Einrichtung der Datenübertragungssteuerung zwischen den Abschlußsystemen und deren lokalen Routern. Die DLSw-Flußsteuerung ermöglicht den Aufbau einer unabhän-

gigen, einseitigen Flußsteuerung zwischen den Partnern. Alle diese Prozesse werden in den folgenden Abschnitten diskutiert.

Austausch der Fähigkeiten

Der Austausch der Fähigkeiten zwischen den DLSws basiert auf einer Switch-to-Switch-Steuerungsmeldung, welche die Fähigkeiten des sendenden Datenübertragungs-Switch beschreibt. Steuerungsmeldungen mit den Fähigkeiten werden nach Aufbau der Switch-to-Switch-Verbindung oder zur Laufzeit ausgetauscht, wenn bestimmte Betriebsparameter verändert wurden, die an den Partner übergeben werden müssen. Im Verlauf des Austauschs der Fähigkeiten wird eine Anzahl von Fähigkeiten ausgewiesen und verhandelt. Zu den zwischen den DLSw-Partnern ausgetauschten Fähigkeiten gehört:

- Die DLSw-Versionsnummer
- anfängliche Pacing-Fenstergröße (Empfangsfenstergröße)
- NetBIOS-Unterstützung
- Liste unterstützter Link-Service Access Points (SAPs)
- Anzahl der unterstützten TCP-Sitzungen
- MAC-Adreßliste
- NetBIOS-Namenslisten
- Suchrahmenunterstützung

DLSw-Verbindungsaufbau

Der Vorgang des Verbindungsaufbaus zwischen zwei Endsystemen schließt bei DLSw das Auffinden des Zielsystems und die Einrichtung der Datenübertragungssteuerung zwischen dem jeweiligen Endsystem und seinem lokalen Router ein. Der Verbindungsaufbau unterscheidet sich je nach Art des Datenverkehrs.

Eine der DLSw-Hauptfunktionen besteht in der Bereitstellung von Übertragungsmechanismen für SNA-Verkehr. Der SNA-Verbindungsaufbau durchläuft mehrere unabhängige Stadien. Als erstes suchen die Geräte eines LAN andere SNA-Geräte, indem sie einen Such-Frame mit der MAC-Adresse des SNA-Zielgeräts aussenden. Sobald ein DLSw-Internet-Knoten einen

Such-Frame empfängt, sendet dieser Knoten einen »canureach«-Frame an alle seine DLSw-Partner. Die Funktion dieses Frame besteht in der Abfrage aller DLSw-Partner, um festzustellen, ob das gesuchte Gerät gefunden werden kann. Wenn einer der DLSw-Partner die angegebene MAC-Adresse erreicht, antwortet der Partner mit einem »icanreach«-Frame, der anzeigt, daß ein bestimmter DLSw-Partner einen Übertragungsweg zu dem gesuchten Gerät herstellen kann. Nachdem die »canureach«- und »icanreach«-Frames ausgetauscht wurden, stellen die beiden DLSw-Partner eine Verbindung her, die aus einer Verbindung zur Datenübertragungssteuerung zwischen den Routern und dem lokal angeschlossenen SNA-Endsystem (für insgesamt zwei Verbindungen) und einer TCP-Verbindung zwischen den DLSw-Partnern besteht. Die entstandene Schaltung wird durch die Quell- und Zielschaltungs-IDs eindeutig identifiziert. Jede SNA-DLSw-Schaltungs-ID beinhaltet eine MAC-Adresse, einen Zugriffspunkt auf den Verbindungsservice (engl. link-service access point, kurz LSAP) und die Anschlußport-ID der Datenübertragungssteuerung (engl. data-link-control port ID). Die Priorität der Schaltung wird zum Zeitpunkt der Einrichtung der Schaltung verhandelt.

Der NetBIOS-Verbindungsaufbau läuft bis auf einige wenige Unterschiede genauso ab wie der SNA-Verbindungsaufbau. Als erstes senden die DLSw-Knoten bei einem NetBIOS-Verbindungsaufbau eine Namensanfrage mit einem NetBIOS-Namen (statt eines »canureach«-Frame, der eine MAC-Adresse bestimmt). Weiterhin senden die DLSw-Knoten beim Aufbau einer NetBIOS-Schaltung einen »name recognized«-Frame (statt eines »icanreach«-Frame).

DLSw-Flußsteuerung

DLSw-Flußsteuerung schließt *adaptives Pacing* zwischen DLSw-Routern ein. Im Verlauf der Verhandlung der Flußsteuerung werden zwei unabhängige, einseitige Flußkontrollmechanismen zwischen den DLSw-Partnern aufgebaut. Adaptives Pacing nutzt einen Fenstersteuerungsmechanismus, der sich je nach Pufferverfügbarkeit dynamisch anpaßt. Fenster können vergrößert, verkleinert, halbiert oder auf Null zurückgesetzt werden. Dies ermöglicht es den DLSw-Knoten, das Datenverkehrstempo durch das Netzwerk zu steuern und somit die Integrität und Übergabe aller Daten zu sichern.

DLSw-Flußkontroll-Indikatoren

Die Anzahl der genehmigten Einheiten (Anzahl der Einheit, die der Absender senden darf) wird mit einer Flußkontrollanzeige vom Empfänger erhöht (einer der verschiedenen möglichen Indikatoren). Die DLSw-Flußsteuerung stellt die folgenden Indikatorfunktionen bereit:

- *Repeat* – (Wiederholen) erhöht die Anzahl der genehmigten Einheiten um die aktuelle Fenstergröße.
- *Increment* – (Erhöhen) Vergrößert die Fenstergröße um 1 und die Anzahl der genehmigten Einheiten um die neue Fenstergröße.
- *Decrement* – (Verringern) Verkleinert die Fenstergröße um 1 und die Anzahl der genehmigten Einheiten um die neue Fenstergröße.
- *Reset* – (Zurücksetzen) Setzt das Fenster auf 0 und die genehmigten Einheiten auf 0; dadurch wird die Übertragung in einer Richtung angehalten, bis ein Flußkontroll-Indikator »Erhöhen« gesendet wird.
- *Half* – (Halbieren) Halbiert die Fenstergröße und erhöht die Anzahl der genehmigten Einheiten um die neue Fenstergröße.

Flußkontroll-Indikatoren und Flußkontroll-Empfangsbestätigungen können huckepack mit Informations-Frames oder als unabhängige Flußsteuerung-Meldungen übertragen werden. Reset-Indikatoren werden immer als unabhängige Meldungen verschickt.

Beispiele für Adaptives Pacing

Beispiele für »Adaptive Pacing«-Kriterien sind u.a. Pufferverfügbarkeit, Übertragungseffizienz, Länge der Sende-Warteschlange und die Verkehrspriorität. Es folgen Beispiele dafür, wie diese im einzelnen auf das Pacing einwirken:

- *Buffer availability* – (Pufferverfügbarkeit) Wenn die verfügbaren Pufferspeicher eines Vermittlungsknotens einer Datenverbindung kritisch niedrig sind, kann der Knoten die Fenstergröße verringern, um den Datenfluß zu reduzieren. Mit der Vergrößerung der Pufferverfügbarkeit kann

der Knoten die Fenstergröße erhöhen, um den Datenfluß zwischen den DLSw-Partnern zu beschleunigen.

- *Transport utilization* – (Übertragungseffizienz) Wenn die Verbindung zwischen zwei DLSw-Partnern eine hohe Übertragungseffizienz erreicht, kann die Fenstergröße zur Verringerung der Ausnutzung verkleinert werden, um den Verlust von Paketen zwischen den Knoten zu verhindern.
- *Outbound queue length* – (Länge der Sende-Warteschlange) Der von einem DLSw-Knoten weitergeleitete Verkehr wird üblicherweise in eine Ausgangswarteschleife geleitet, die durch einen Speicher gebildet wird, der für die Weiterleitung des Verkehrs von einem Gerät zum anderen zuständig ist. Wenn diese Warteschlange einen bestimmten Schwellwert erreicht oder sogar voll ist, kann die Anzahl der genehmigten Einheiten verringert werden, bis die Ausnutzung der Warteschlange auf ein ausreichendes Niveau reduziert wurde.
- *Trafficpriority* – (Verkehrspriorität) Eine der speziellen Fähigkeiten von SSP ist die Möglichkeit der Priorisierung des Verkehrs. Diese Prioritäten werden durch das »circuit-priority«-Feld im DLSw-Meldungs-Frame festgelegt. Durch Festlegung einer veränderlichen Anzahl von genehmigten Einheiten für bestimmte DLSw-Schaltungen können die Knoten die verschiedenen Prioritätsniveaus der einzelnen Schaltungen verwalten.

19.6 DLSw-Meldungsformate

Zwischen DLSw-Knoten werden zwei Formate für Meldungsköpfe ausgetauscht:

- Control
- Information

Der Control-Meldungskopf wird für alle Meldungen, außer *information frames* (Iframes) und *independent flow-control messages* (IFCMs), verwendet, die im Informations-Format verschickt werden.

Bild 19.6 zeigt das Format der DLSw-Control- und Information-Felder. Diese Felder werden in den folgenden Beschreibungen detailliert besprochen.

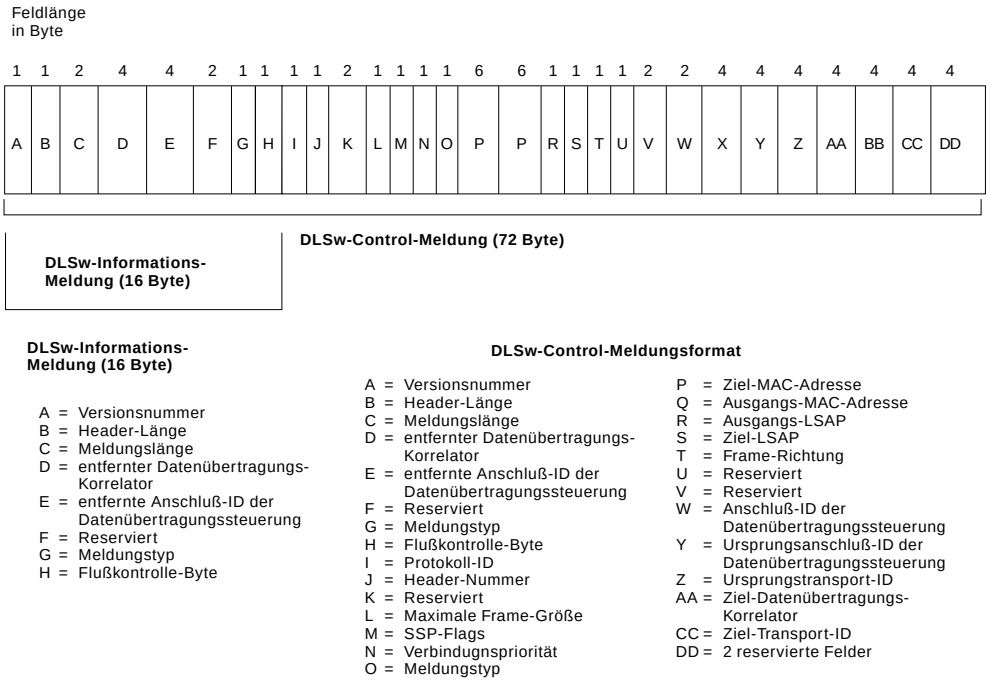


Bild 19.6: DLSw-Control- und -Informations-Frames haben die ersten 16 Bytes gemeinsam

Der folgende Abschnitt ist eine Zusammenfassung der Felder aus Bild 19.6 (die Felder der ersten 16 Bytes aller DLSw-Meldungsköpfe sind gleich):

– *Version Number* – (Versionsnummer) Ein Wert von 0x31 (ASCII 1) entspricht dem Dezimalwert 49 und gibt an, daß das Gerät DLSw-Version 1 verwendet. Dies stellt die Kompatibilität zwischen DLSw-Knoten mit verschiedenen Versionen des DLSw-Standards für die Zukunft sicher. Momentan nutzen alle Geräte DLSw-Version 1, weshalb dieses Feld immer den Dezimalwert 49 hat.

– *Header-Länge* – Ein Wert von 0x48 gibt bei Control-Meldungen einen Dezimalwert von 72 Byte an. Dieser Wert wird für Informations- und Independent-Flow-Control-

Meldungen auf 0x10 gesetzt, was einem Dezimalwert von 16 Bytes entspricht.

- *Meldungslänge* – Legt die Anzahl der Bytes fest, die im Datenfeld nach dem Header kommen.
- *Entfernter Datenübertragungs-Korrelator* – Bildet zusammen mit der entfernten Anschluß-ID der Datenübertragungssteuerung eine 64 Bit lange Schaltungs-ID, welche die DLC-Schaltung innerhalb eines DLSw-Knotens bestimmt. Die Schaltungs-ID wird lokal vergeben und ist innerhalb eines einzelnen DLSw-Knotens einmalig. Eine »end-to-end«-Schaltung wird durch ein Paar aus Schaltungs-IDs bestimmt, welche eine einzelne »end-to-end«-Schaltung im Verbund mit den Datenübertragungs-IDs eindeutig bestimmen. Jeder DLSw-Knoten besitzt eine Tabelle für diese Schaltungs-ID-Paare: eine für das lokale Ende der Schaltung und die andere für das entfernte Ende der Schaltung. Der entfernte Data-Link Correlator wird so wie der Data-Link Correlator des Ziels gesetzt, wenn das Frame-Direction-Feld gleich 0x01 ist. Er entspricht dem Data-Link Correlator der Quelle, wenn das Frame-Direction-Feld gleich 0x02 ist.
- *Entfernte Anschluß-ID der Datenübertragungssteuerung* – Bildet zusammen mit dem entfernten Correlator der Datenübertragungssteuerung eine 64 Bit lange Schaltungs-ID, welche die DLC-Schaltung innerhalb eines DLSw-Knotens bestimmt. Die Schaltungs-ID wird lokal vergeben und ist innerhalb eines einzelnen DLSw-Knotens einmalig. Eine »end-to-end«-Schaltung wird durch ein Paar aus Schaltungs-IDs bestimmt, welche eine einzelne »end-to-end«-Schaltung im Verbund mit den Datenübertragungs-IDs eindeutig bestimmen. Jedes DLSw-Gerät besitzt eine Tabelle für diese Schaltungs-ID-Paare: eine für das lokale Ende der Schaltung und die andere für das entfernte Ende der Schaltung. Die entfernte DLC-Port-ID wird so wie die DLC-Port-ID des Ziels gesetzt, wenn das Frame-Direction-Feld gleich 0x01 ist. Sie entspricht der DLC-Port-ID der Quelle, wenn das Frame Direction-Feld gleich 0x02 ist.
- *Meldungstyp* – weist auf einen speziellen DLSw-Meldungstyp hin. Der Wert wird in zwei verschiedenen Feldern (dezimaler Offset 14 und 23) des Control-Message-Kopfes

angegeben. Beim Empfang einer SSP-Meldung wird nur das erste Feld eingelesen. Das zweite Feld wird von neuen Realisierungen empfangsseitig ignoriert, wurde aber der Rückwärtskompatibilität zu Realisierungen nach RFC 1434 wegen beibehalten und kann in zukünftigen Versionen bei Bedarf verwendet werden.

- *Flußkontrolle-Byte* – Enthält den Flußkontrollindikator, die Empfangsbestätigung der Flußsteuerung und Operatorbits der Flußsteuerung.
- *Protokoll-ID* – Gibt (falls auf sie 0x42 gesetzt ist) einen Dezimalwert von 66 an.
- *Header-Nummer* – Gibt (falls auf sie 0x01 gesetzt ist) einen Dezimalwert von 1 an.
- *Maximale Frame-Größe* – Überträgt die »Maximale Frame-Größe«-Bits über die DLSw-Verbindung. Dieses Feld soll sicherstellen, daß die beiden Endstationen eine Frame-Größe für eine Verbindung verhandeln, die keine Resegmentierung der Frames durch die DLSw-Partner erforderlich macht.
- *Switch-to-Switch Protocol (SSP)-Flags* – Enthält zusätzliche Informationen zur SSP-Meldung. Die Zuordnung der Flags (Bit 7 ist das hochwertigste und Bit 0 das niederwertigste Bit des Bytes) wird in der Tabelle 19.1 gezeigt.

Bit-Position	Name	Bedeutung
7	SSPex	1 = Suchmeldung (»canureach« oder »icanreach«)
6 bis 0	Reserviert	Keine. Reservierte Felder werden bei Übertragung auf 0 gesetzt und bei Empfang ignoriert.

Tabelle 19.1: SSP-Flag-Definitionen

- *Verbindungspriorität* – Steht für nicht unterstützte, niedrige, mittlere, hohe und höchste Verbindungspriorität in den drei niederwertigen Bits dieses Bytes. Zum Einrichtungszeitpunkt der Schaltung stellt jeder Schaltungsendpunkt seinem Partner Prioritätsinformationen bereit. Der Auslöser der Verbindung legt fest, welche Priorität über die Existenzdauer der Verbindung hinweg verwendet wird.

Wenn die Knoten diese Priorität nicht unterstützen, wird die unterstützungslose Priorität verwendet.

- *Ziel-MAC-Adresse* – Wird mit der Zielverbindungs-SAP, der Ausgangs-MAC-Adresse und der Ausgangs-SAP zur Festlegung einer logischen End-To-End-Zuordnung, der sogenannten Data-Link-ID, kombiniert.
- *Ausgangs-MAC-Adresse* – Dient als MAC-Adresse der auslösenden Endstation.
- *Ausgangs-LSAP (Link Service Access Point)* – Dient als SAP des Quellengeräts. Die SAP wird zur logischen Identifikation des übertragenen Verkehrs verwendet.
- *Ziel-LSAP (Link Service Access Point)* – Dient als SAP des Zielgeräts.
- *Frame-Richtung* – Enthält den Wert 0x01 für Frames, die vom Ursprungs-DLSw- zum Ziel-DLSw-Knoten, bzw. 0x02 für Frames, die vom Ziel-DLSw- zum Ursprungs-DLSw-Knoten gesendet werden.
- *Data-Link-Control (DLC) Header-Länge* – Gibt mit der Einstellung 0 für SNA und 0x23 für NetBIOS-Datagramme eine Länge von 35 Byte an. Der NetBIOS-Header beinhaltet die folgenden Informationen:
  - Access Control (AC)-Feld
  - Frame Control (FC)-Feld
  - Destination-MAC-Adresse (DA)
  - Source-MAC-Adresse (SA)
  - Routing Information (RI)-Feld (auf 18 Byte aufgefüllt)
  - Destination Service Access Point (DSAP)
  - Source SAP (SSAP)
  - LLC-Steuerungsfeld (UI)
- *Anschluß-ID der Datenübertragungssteuerung* – Bildet zusammen mit dem Korrelator der Ursprungsdatenverbindung eine 64-Bit-Schaltungs-ID, welche die DLC-Schaltung innerhalb eines einzelnen DLSw-Knoten definiert. Die

Schaltungs-ID wird lokal vergeben und ist innerhalb eines einzelnen DLSw-Knotens einmalig. Die End-To-End-Schaltung wird durch ein Paar von Schaltungs-IDs identifiziert, welche zusammen mit den Datenübertragungs-IDs eine einzelne End-To-End-Schaltung eindeutig identifizieren. Jeder DLSw-Knoten besitzt eine Tabelle für diese Schaltungs-ID-Paare: eine für das lokale Ende der Schaltung und die andere für das entfernte Ende der Schaltung.

– *Ursprungs-Datenübertragungs-Korrelator* – Bildet zusammen mit der Ursprungsanschluß-ID der Datenübertragungssteuerung eine 64 Bit lange Schaltungs-ID, welche die DLC-Schaltung innerhalb eines DLSw-Knotens bestimmt. Die Schaltungs-ID wird lokal vergeben und ist innerhalb eines einzelnen DLSw-Knotens einmalig. Die End-To-End-Schaltung wird durch ein Paar von Schaltungs-IDs identifiziert, welche zusammen mit den Datenübertragungs-IDs eine einzelne End-To-End-Schaltung eindeutig identifizieren. Jeder DLSw-Knoten besitzt eine Tabelle für diese Schaltungs-ID-Paare: eine für das lokale Ende der Schaltung und die andere für das entfernte Ende der Schaltung.

– *Ursprungs-Transport-ID* – Identifiziert den jeweiligen TCP/IP-Anschluß eines DLSw-Knotens. Die Werte haben nur eine lokale Bedeutung. Jeder DLSw-Knoten muß diese Werte zusammen mit den zugehörigen Werten, der Datenübertragungskontroll-(DLC-)Anschluß-ID und dem Datenübertragungs-Korrelator bei der Rückgabe einer Meldung an den DLSw-Partner wiedergeben.

– *Ziel-Anschluß-ID der Datenübertragungssteuerung* – Bildet zusammen mit dem Datenübertragungs-Korrelator des Ziels eine 64 Bit lange Schaltungs-ID, welche die DLC-Schaltung innerhalb eines DLSw-Knotens bestimmt. Die Schaltungs-ID wird lokal vergeben und ist innerhalb eines einzelnen DLSw-Knotens einmalig. Die End-To-End-Schaltung wird durch ein Paar von Schaltungs-IDs identifiziert, welche zusammen mit den Datenübertragungs-IDs eine einzelne End-To-End-Schaltung eindeutig identifizieren. Jeder DLSw-Knoten besitzt eine Tabelle für diese Schaltungs-ID-Paare: eine für das lokale Ende der Schaltung und die andere für das entfernte Ende der Schaltung.

- *Ziel-Datenübertragungs-Korrelator* – Bildet zusammen mit der Ziel-Anschluß-ID der Datenübertragungssteuerung eine 64-Bit lange Schaltungs-ID, welche die DLC-Schaltung innerhalb eines DLSw-Knotens bestimmt. Die Schaltungs-ID wird lokal vergeben und ist innerhalb eines einzelnen DLSw-Knotens einmalig. Die End-To-End-Schaltung wird durch ein Paar von Schaltungs-IDs identifiziert, welche zusammen mit den Datenübertragungs-IDs eine einzelne End-To-End-Schaltung eindeutig identifizieren. Jeder DLSw-Knoten besitzt eine Tabelle für diese Schaltungs-ID-Paare: eine für das lokale Ende der Schaltung und die andere für das entfernte Ende der Schaltung.
- *Transport-ID* – Identifiziert den jeweiligen TCP/IP-Anschluß an einem DLSw-Knoten. Die Werte haben nur eine lokale Bedeutung. Jeder DLSw-Knoten muß diese Werte zusammen mit den zugehörigen Werten, der Datenübertragungskontroll-(DLC-)Anschluß-ID und dem Datenübertragungs-Korrelator bei der Rückgabe einer Meldung an den DLSw-Partner wiedergeben.

KAPITEL 20

LAN-Switching

20.1 Grundlagen

Ein LAN-Switch ist ein Gerät, das eine wesentlich höhere Anschlußdichte ermöglicht als herkömmliche Bridges. Deshalb kann man mit LAN-Switches Netzwerk-Entwürfe mit weniger Nutzern pro Segment realisieren und infolgedessen die mittlere verfügbare Bandbreite pro Benutzer vergrößern. Dieses Kapitel beinhaltet eine Zusammenfassung des allgemeinen LAN-Switch-Betriebs und ordnet das LAN-Switching den Schichten des OSI-Referenzmodells zu.

Der Trend zu weniger Nutzern pro Segment wird als *Microsegmentation* bezeichnet. Die Microsegmentation ermöglicht das Erstellen von persönlichen oder zugeordneten Segmenten, d.h. einem Benutzer pro Segment. Jeder Benutzer hat unmittelbaren Zugriff auf die volle Bandbreite und muß diese nicht mit anderen Benutzern teilen. Infolgedessen können keine Kollisionen (ein übliches Phänomen in Netzwerken mit gemeinsam benutzten Medien und Hubs) auftreten. Ein LAN-Switch leitet Frames weiter, die entweder auf der Layer-2-Adresse (Layer-2-LAN-Switch) oder in manchen Fällen auf der Layer-3-Adresse des Frames basieren (Multi-Layer-LAN-Switch). LAN-Switches werden auch als Frame-Switches bezeichnet, weil sie Layer-2-Frames vermitteln, während ein ATM-Switch Zellen (engl. cells) weiterleitet. Trotz der hohen Verbreitung von Ethernet-LAN-Switches gewinnen Token-Ring- und FDDI-LAN-Switches mit steigender Netzwerk-Auslastung an Bedeutung.

Bild 20.1 zeigt einen LAN-Switch, der den Geräten die jeweiligen dedizierten Bandbreiten zuweist, und verdeutlicht das Layer-2-LAN-Switching im Verhältnis zur OSI-Sicherungsschicht:

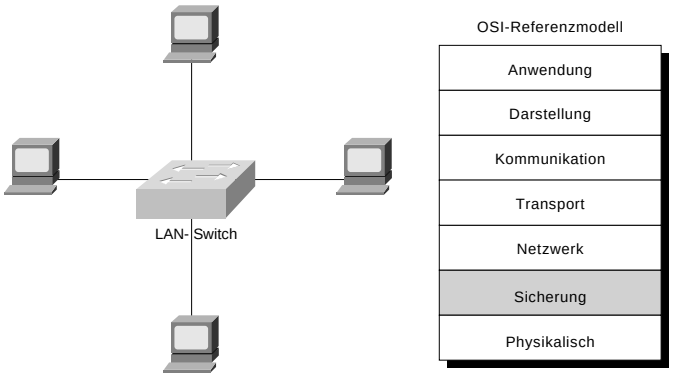


Bild 20.1: Ein LAN-Switch ist ein Gerät der Sicherungsschicht

20.1.1 Zur Geschichte

Die ersten LAN-Switches wurden 1990 entwickelt. Diese waren Layer-2-Geräte und dienten der Lösung von Bandbreitenproblemen. Moderne LAN-Switches haben sich zu Multi-Layer-Geräten entwickelt, die in der Lage sind, Protokollaufgaben in Anwendungen mit sehr hoher Bandbreite zu lösen, die vorher nur von Routern bewältigt werden konnten. Heute lösen LAN-Switches zunehmend die Hubs bei der Vermittlung ab, weil die Benutzeranwendungen immer größere Bandbreiten erfordern.

20.2 Einsatz von LAN-Switches

LAN-Switches ähneln der Funktionsweise von transparenten Bridges bei Funktionen, wie dem Erlernen der Topologie, der Vermittlung und Filterung. Diese Switches beherrschen weitere neue und einmalige Leistungsmerkmale, wie dedizierte Kommunikation zwischen Geräten, mehrere Verbindungen gleichzeitig, Vollduplex-Übertragung und Anpassung an das Medium.

Die dedizierte, kollisionsfreie Übertragung zwischen Netzwerk-Geräten erhöht den Durchsatz bei der Dateiübertragung. Mehrere gleichzeitige Verbindungen können beim Weiterleiten

oder Switching mehrerer Pakete zur selben Zeit aufgebaut werden, wodurch die Netzwerkleistung um die Anzahl der Verbindungen erhöht wird. Eine Vollduplex-Übertragung verdoppelt den Durchsatz, während die Anpassung an das Medium dem LAN-Switch Übertragungen zwischen 10 und 100 Mbps ermöglicht, wodurch die Bandbreite nach Bedarf zugeordnet werden kann.

Der Einsatz von LAN-Switches erfordert keine weiteren Änderungen an vorhandenen Hubs, Netzwerkkarten oder Verkabelungen.

20.2.1 Vermittlung beim LAN-Switching

LAN-Switches können durch die unterstützten Vermittlungsmethoden charakterisiert werden. Bei der Switching-Methode mit Speichern und Weiterleiten findet eine Fehlerprüfung statt, wonach fehlerhafte Frames abgelehnt werden. Bei der durchgehenden Switching-Methode wird die Verzögerung durch den Verzicht auf die Fehlerprüfung verringert.

Bei der Switching-Methode mit Speichern und Weiterleiten kopiert der LAN-Switch den gesamten Frame in seine internen Pufferspeicher und berechnet eine zyklische Blocksicherung (engl. cyclic redundancy check, CRC). Der Frame wird abgelehnt, wenn er einen CRC-Fehler enthält bzw. ein *Runt* (weniger als 64 Byte inklusive CRC lang) oder ein *Giant* (mehr als 1518 Byte inklusive CRC lang) ist. Wenn der Frame keinerlei Fehler aufweist, sucht der LAN-Switch die Zieladresse in seiner Weiterleitungs- oder Switching-Tabelle und legt die Ausgangsschnittstelle fest. Danach wird der Frame in Richtung des Ziels verschickt.

Bei der durchgehenden Switching-Methode kopiert der LAN-Switch nur die Zieladresse (die ersten 6 Byte nach dem Vorspann) in seine internen Puffer. Anschließend sucht der LAN-Switch die Zieladresse in seiner Weiterleitungs- oder Switching-Tabelle, legt die Ausgangsschnittstelle fest und verschickt den Frame in Richtung der Zieladresse. Ein durchgehender Switch ermöglicht geringere Verzögerungen, weil er den Frame weiterleitet, sobald er die Zieladresse gelesen und die Ausgangsschnittstelle festgelegt hat.

Einige Switches können so konfiguriert werden, daß sie solange durchgehendes Switching einsetzen, bis ein benutzerdefinierter Fehlerschwellwert erreicht wird, und dann automatisch mit der Switching-Methode mit Speichern und Weiterleiten fortsetzen. Wenn die Fehlerrate den Schwellwert wieder unterschreitet, wechselt der Anschluß automatisch zurück in den durchgehenden Modus.

20.2.2 Bandbreite des LAN-Switching

LAN-Switches können auch anhand der Bandbreite charakterisiert werden, die jedem Anschluß zur Verfügung steht. Symmetrisches Switching stellt jedem Anschluß die gleiche Bandbreite zur Verfügung, während asymmetrisches Switching den Anschlüssen unterschiedliche Bandbreiten zuweist.

Ein asymmetrischer LAN-Switch ermöglicht vermittelte Verbindungen zwischen Anschlüssen mit unterschiedlicher Bandbreite, wie Kombinationen aus 10BaseT und 100BaseT. Diese Switching-Art wird auch als *10/100-Switching* bezeichnet. Asymmetrisches Switching ist für Client-Server-Datenverkehr optimiert, bei dem mehrere Clients gleichzeitig mit einem Server kommunizieren, wobei am Server-Port eine größere Bandbreite benötigt wird, um dort einen Engpaß zu vermeiden.

Ein symmetrischer Switch ermöglicht vermittelte Verbindungen zwischen Anschlüssen mit der gleichen Bandbreite, wie alle 10BaseT und 100BaseT. Symmetrisches Switching ist für eine zweckmäßig verteilte Datenverkehrsbelastung optimiert, z.B. für eine Peer-to-Peer-Desktop-Umgebung.

Der Netzwerk-Verwalter muß die benötigte Bandbreite für Verbindungen zwischen den Geräten abschätzen, um den Bedarf für den Datenfluß netzwerkgestützter Anwendungen erfüllen zu können, und daraufhin entscheiden, ob ein asymmetrischer oder ein symmetrischer Switch benötigt wird.

20.3 LAN-Switch und das OSI-Modell

LAN-Switches können entsprechend der OSI-Schicht charakterisiert werden, auf der sie Frames filtern und weiterleiten bzw. vermitteln (engl. switch). Diese Kategorien sind: Layer 2, Layer 2 mit Layer-3-Merkmalen oder Multi-Layer.

Ein Layer-2-LAN-Switch ist in seinen Betriebseigenschaften einer Multiport-Bridge sehr ähnlich, verfügt aber über eine wesentlich höhere Kapazität und besitzt viele neue Merkmale, wie z.B. den Vollduplex-Betrieb. Ein Layer-2-LAN-Switch vermittelt und filtert auf Grundlage der MAC-Adresse der OSI-Sicherungsschicht (Layer 2). Wie Bridges, ist er für Netzwerkprotokolle und Benutzeranwendungen vollständig transparent.

Ein Layer-2-LAN-Switch mit Layer-3-Merkmalen kann Switching-Entscheidungen auf Grund weiterer Informationen als nur der Layer-2-MAC-Adresse treffen. Ein solcher Switch kann einige Layer-3-Verkehrssteuerungsmöglichkeiten einschließen, wie Broadcast und Multicast-Verkehrsmanagement, Sicherheitsprüfungen über Zugriffslisten und IP-Fragmentierung.

Ein Multi-Layer-Switch trifft Switching- und Filterungsentscheidungen auf Grundlage der OSI-Sicherungsschicht-Adressen (Layer 2) und OSI-Netzwerkschicht-Adressen (Layer 3). Dieser Switch-Typ entscheidet dynamisch, ob der eintreffende Verkehr vermittelt (Layer 2) oder »geroutet« (Layer 3) wird. Ein Multi-Layer-LAN-Switch vermittelt (engl. switch) innerhalb von Arbeitsgruppen und routet zwischen unterschiedlichen Arbeitsgruppen.

KAPITEL 21

Tag-Switching

21.1 Grundlagen

Radikale Änderungen der Qualität (und Quantität) des über das Internet abgewickelten Datenverkehrs und das explosive Ansteigen der Anzahl der Internet-Benutzer belasten die Infrastruktur des Internet auf bisher nicht vorhersehbare Art und Weise. Diese Entwicklung sorgte für völlig neue Lösungen zur Verwaltung des Datenverkehrs. *Tag-Switching* zielt darauf ab, viele der Herausforderungen des in der Entwicklung begriffenen Internet und der Datenübertragung mit hoher Geschwindigkeit im allgemeinen zu lösen. Dieses Kapitel gibt einen Überblick zu den Grundlagen des Tag-Switching-Betriebs, der Architektur und den Einsatzumgebungen. Tag-Switching liegt momentan in Form einer Serie von Internet-Entwürfen vor. Zu diesen gehört u.a.:

- »Tag Distribution Protocol«, P. Doolan, B. Davie, D. Katz
- »Tag Switching Architecture Overview«, Y. Rekhter, B. Davie, D. Katz
- »Use of Flow Label for Tag Switching«, F. Baker, Y. Rekhter
- »Use of Tag Switching With ATM«, B. Davie, P. Doolan, J. Lawrence, K. McCloghrie
- »Tag Switching: Tag Stack Encoding«, E. Rosen, D. Tappan, D. Farinacci

21.2 Die Tag-Switching-Architektur

Tag-Switching verwendet zwei Grundkomponenten: Weiterleitung (engl. forwarding) und Steuerung (engl. control). Die Forwarding-Komponente verwendet die Tag-Information (Tags), die mit Paketen übertragen wird, und die Tag-Forwarding-Information, die von einem Tag-Switch verwaltet wird, für die Weiterleitung von Paketen. Die Control-Komponente ist für die Handhabung der korrekten Tag-Forwarding-Information innerhalb einer Gruppe von verbundenen Tag-Switches zuständig. Weitere Einzelheiten zum Forwarding und Steuerungsmechanismen beim Tag-Switching finden Sie im weiteren Verlauf dieses Kapitels.

21.2.1 Die Forwarding-Komponente

Das beim Tag-Switching verwendete Forwarding-Paradigma basiert auf der Idee des *Label-Swapping*. Beim Empfang eines Pakets mit einem Tag durch einen Tag-Switch verwendet der Switch das Tag als Index in seiner Tag Information Base (TIB). Jeder Eintrag in der TIB besteht aus einem Eingangs-Tag und einem oder mehr Untereinträgen (in der Folge Ausgangs-Tag, Ausgangs-Schnittstelle, Ausgangs-Verbindungsschichtinformation). Wenn der Switch einen Eintrag findet, dessen Eingangstag dem mit dem Paket übertragenen entspricht, ersetzt der Switch für alle Bestandteile des Eintrags das Tag im Paket durch das Ausgangs-Tag, die Informationen der Verbindungsschicht (wie die MAC-Adresse) durch die Informationen der Verbindungsschicht am Ausgang und leitet das Paket über die Ausgangsschnittstelle weiter.

Anhand der oben gegebenen Beschreibung der Forwarding-Komponente, können wir verschiedene Betrachtungen anstellen. Die Forwarding-Entscheidung basiert auf einem Algorithmus mit genauer Übereinstimmung mit einem relativ kurzen Index-Tag fester Länge. Dies ermöglicht eine einfachere Forwarding-Prozedur als mit der längsten Übereinstimmung, die in der Netzwerkschicht eingesetzt wird.

Dadurch wird die Forwarding-Geschwindigkeit vergrößert (eine höhere Menge von Paketen pro Sekunde). Die Forwarding-Prozedur ist einfach genug, um als Hardware realisiert zu werden. Weiterhin von Bedeutung ist die Tatsache, daß die

Forwarding-Entscheidung unabhängig von der Forwarding-Granularität des Tag ist. Der gleiche Forwarding-Algorithmus verarbeitet sowohl Unicast als auch Multicast: ein Unicast-Eintrag hätte einen einzelnen Untereintrag (Ausgangs-Tag, Ausgangsschnittstelle, Ausgangs-Verbindungsschichtinformation), während ein Multicast-Eintrag ein oder mehr Untereinträge haben kann. Dies verdeutlicht, wie das gleiche Forwarding-Paradigma für das Tag-Switching mit Unterstützung verschiedener Routing-Funktionen verwendet werden kann.

Die einfache Forwarding-Prozedur ist somit von der Steuerungskomponente des Tag-Switching völlig entkoppelt. Neue Routing-(Steuerung-)Funktionen können einfach eingesetzt werden, ohne das Forwarding-Paradigma zu stören. Dies bedeutet auch, daß bei Ergänzung neuer Funktionen keine Neuoptimierung der Forwarding-Leistung (durch Änderung von Hardware oder Software) notwendig ist.

Tag-Encapsulation

Tag-Informationen können in Paketen auf verschiedene Art und Weise übertragen werden:

- Als kleiner »Shim«-Tag-Kopf zwischen Layer 2 und den Netzwerkschicht-Kopfzeilen
- Als Teil des Layer-2-Kopfes, wenn der Layer-2-Kopf eine verwendbare Semantik besitzt (wie bei ATM)
- Als Teil des Netzwerkschicht-Kopfes (wie beim Einsatz des Flow-Label-Felds in IPv6 mit entsprechend modifizierter Semantik)

Infolgedessen kann Tag-Switching für beliebige Medientypen verwendet werden, für Point-To-Point-Verbindungen, Multi-Access-Verbindungen und ATM. Die Tag-Forwarding-Komponente ist vom Netzwerkschicht-Protokoll unabhängig. Die Verwendung von spezifischen Steuerungskomponenten für spezielle Netzwerkschicht-Protokolle ermöglicht den Einsatz des Tag-Switching mit verschiedenen Netzwerkschicht-Protokollen.

21.2.2 Steuerungskomponenten

Eine Besonderheit des Tag-Switching liegt in der Idee der Verbindung zwischen einem Tag und dem Netzwerkschicht-Routing (Routen). Um eine gute Skalierbarkeit zu gewährleisten und gleichzeitig verschiedene Routing-Funktionen umzusetzen, unterstützt Tag-Switching eine Vielzahl von Forwarding-Granularitäten. Als Extremfall kann ein Tag einer Gruppe von Routen zugewiesen werden (genauer die Reachability-Information der Netzwerkschicht der Routen dieser Gruppe). Der andere Extremfall wäre, daß ein Tag einem speziellen Anwendungsfluß (wie einem RSVP-Fluß) zugewiesen wird oder an einen Multicast-Tree.

Die Control-Komponente ist für die Erstellung von Tag-Bindings und die anschließende Verteilung der Tag-Bindings an die Tag-Switches zuständig.

Die Steuerungskomponente ist als Modulsammlung organisiert, von denen jede für die Unterstützung einer speziellen Routing-Funktion entworfen ist. Für die Unterstützung neuer Routing-Funktionen können neue Module ergänzt werden. Im Folgenden werden einige dieser Module beschrieben.

21.3 Destination-Based Routing (zielbasiertes Routing)

Beim Destination-Based-Routing entscheidet ein Router über die Weiterleitung aufgrund der Zieladresse, die im Paket enthalten ist, und der Informationen, die in der Forwarding Information Base (FIB) des Router enthalten sind. Ein Router konstruiert seine FIB mit Hilfe der Informationen, die er von Routing-Protokollen wie OSPF und BGP empfängt.

Zur Unterstützung des Destination-Based-Routing durch das Tag-Switching nimmt ein Tag-Switch an den Routing-Protokollen teil und baut seine FIB mit Hilfe der Informationen auf, die er über diese Protokolle empfängt. Damit funktioniert er ähnlich wie ein Router.

Es gibt drei zulässige Methoden für die Tag-Zuweisung und Verwaltung der Tag Information Base (TIB):

– Downstream-Tag-Zuweisung

– Downstream-Tag-Zuweisung auf Anforderung

– Upstream-Tag-Zuweisung

In alle Fällen sucht ein Tag-Switch Tags und weist diese den Adreßpräfixen in seiner FIB zu. Bei der Downstream-Zuweisung wird das Tag (welches in einem Paket übertragen wird) generiert und vom Switch einem Präfix am Downstream-Ende der Verbindung zugewiesen (unter Berücksichtigung der Richtung des Datenflusses). Bei der Upstream-Zuweisung werden die Tags ausgelesen und dem Upstream-Ende der Verbindung zugewiesen. Zuweisung auf Anforderung bedeutet, daß die Tags nur auf Anforderung durch den Upstream-Switch ausgelesen und verteilt werden.

Downstream- und Upstream-Tag-Zuweisung auf Anforderung sind besonders in ATM-Netzwerken sehr nützlich. Bei der Downstream-Zuweisung ist ein Switch für die Erstellung der Tag-Zuordnungen zuständig, die den eingehenden Datenpaketen zugeordnet werden, und außerdem für den Empfang von Tag-Zuordnungen für abgehende Pakete von seinen Nachbarn. Bei der Upstream-Zuweisung ist ein Switch für die Erstellung der Tag-Zuordnungen der abgehenden Datenpakete zuständig und außerdem für den Empfang von Zuordnungen für eingehende Tags von Nachbarn. Betriebsbedingte Unterschiede werden in der folgenden Zusammenfassung herausgestellt.

21.3.1 Downstream-Tag-Zuweisung

Bei der Downstream-Tag-Zuweisung weist der Switch jeder Route einen Tag in seinem FIB zu, erstellt einen Eintrag in seiner TIB mit dem eingehenden Tag auf das zugeordnete Tag gesetzt und gibt die Zuordnung zwischen dem (eingehenden) Tag und der Route den anderen benachbarten Tag-Switches bekannt. Die Bekanntgabe kann durch Versand der Zuordnung mit den bestehenden Routing-Protokollen oder unter Verwendung eines getrennten Tag Distribution Protocol (TDP) erfolgen. Wenn ein Tag-Switch Tag-Zuordnungsinformationen für

eine Route empfängt, und diese Information stammt vom nächsten Hop dieser Route, dann legt der Switch das Tag (das als Teil der Zuordnungsinformation übertragen wird) als Ausgangs-Tag des TIB-Eintrags ab, dem diese Route zugeordnet ist. So wird die Zuordnung zwischen dem Ausgangs-Tag und der Route hergestellt.

21.3.2 Downstream-Tag-Zuweisung auf Anforderung

Für jede Route in seinem FIB identifiziert der Switch den nächsten Hop für diese Route. Anschließend löst er eine Anforderung (per TDP) an den nächsten Hop für eine Tag-Zuordnung für diese Route aus. Wenn der nächste Hop die Anforderung erhält, liest er ein Tag aus, erstellt einen Eintrag in seiner TIB mit dem eingehenden Tag auf das zugeordnete Tag gesetzt und gibt die Zuordnung zwischen dem (eingehenden) Tag und der Route an den Switch zurück, welcher die Anforderung auslöste. Wenn der Switch eine Zuordnungsinformation (engl. binding) empfängt, erstellt der Switch einen Eintrag in seiner TIB und besetzt das Ausgangs-Tag des Eintrags auf den vom nächsten Hop empfangenen Wert.

21.3.3 Upstream-Tag-Zuweisung

Bei Upstream-Tag-Zuweisung erstellt ein Tag-Switch ein Tag für jede Route in seinem FIB, deren nächster Hop über eine dieser Schnittstellen erreichbar ist (wenn er eine oder mehr Point-To-Point-Schnittstellen besitzt). Es erstellt einen Eintrag in seiner TIB, dessen Ausgangs-Tag mit dem ausgelesenen Tag belegt wird, und teilt die Zuordnung zwischen (Ausgangs-)Tag und der Route dem nächsten Hop mit (per TDP). Wenn der als nächster Hop fungierende Tag-Switch Tag-Zuordnungsinformationen empfängt, dann legt der Switch das Tag (das als Teil der Zuordnungsinformation übertragen wird) als Eingangs-Tag des TIB-Eintrags ab, dem diese Route zugeordnet ist.

Nachdem ein TIB-Eintrag sowohl mit Eingangs- als auch Ausgangs-Tags belegt wurde, kann der Tag-Switch mit Hilfe des Weiterleitungsalgorithmus für das Tag-Switching Pakete für die zugeordneten Routen weiterleiten.

Wenn ein Tag-Switch eine Zuordnungsinformation (engl. binding) zwischen einem Ausgangs-Tag und einer Route erstellt, belegt er nicht nur seine TIB, sondern aktualisiert auch seine FIB mit der Zuordnungsinformation. Dies ermöglicht dem Switch eine Ergänzung von Tags bei bislang tag-losen Paketen. Beachten Sie, daß die Gesamtmenge der Tags die ein Tag-Switch verwaltet, nicht größer als die Anzahl der Routen in der FIB des Switch sein kann. Vielmehr ist in den meisten Fällen ein einzelnes Tag einer Gruppe von Routen zugeordnet, statt nur einer einzelnen Route. So sind weniger Zustandsinformationen notwendig als für den Fall, daß die Tags einzelnen Routen zugeordnet sind.

Im allgemeinen versucht ein Tag-Switch, seine TIB mit eintreffenden und ausgehenden Tags für alle erreichbaren Routen zu füllen, wonach alle Pakete per einfachem Label-Swapping weitergeleitet werden können. Tag-Zuweisung wird demzufolge durch die Topologie (Routing) und nicht den Datenverkehr geleitet. Das Vorhandensein eines FIB-Eintrags bewirkt die Tag-Zuweisung, nicht das Eintreffen von Datenpaketen.

Die Verwendung von Tags, welche Routen zugeordnet sind statt Verbindungen, bedeutet auch, daß es nicht notwendig ist, Fluß-Klassifizierungsprozeduren für all die Verbindungen auszuführen, um festzustellen, ob einer Verbindung ein Tag zuzuordnen ist. Dies vereinfacht das allgemeine Routing-Schema und erstellt eine robustere und stabilere Umgebung.

Beim Einsatz des Tag-Switching zur Unterstützung des Destination-Based-Routing, bleibt das Forwarding auf der normalen Netzwerkschicht weiterhin notwendig. Als erstes ist für das Ergänzen eines Tag zu einem vorher unmarkierten Paket normales Netzwerkschicht-Forwarding erforderlich. Diese Funktion kann durch den ersten Hop-Router oder den ersten Router im Pfad ausgeführt werden, der am Tag-Switching teilnehmen kann. Immer dann, wenn ein Tag-Switch eine Gruppe von Routen zu einem einzigen Tag verbindet und die Routen keinen gemeinsamen nächsten Hop besitzen, muß der Switch ein Netzwerkschicht-Forwarding für die Pakete ausführen, die das Tag übertragen. Allerdings ist die Anzahl der Stellen, an denen Routen zusammengefaßt werden, geringer als die Gesamtanzahl der Stellen, an denen Forwarding-Entscheidungen getroffen werden müssen. Außerdem werden Untergruppen

von Routen, die von einem Tag-Switch bearbeitet werden, oft zusammengefaßt. Infolgedessen können Pakete gewöhnlich mit dem Tag-Switching-Algorithmus weitergeleitet werden.

21.4 Hierarchical-Routing

Die IP-Routing-Architektur modelliert ein Netzwerk als Sammlung von Routing-Domains. Innerhalb einer Domain wird das Routing als internes Routing realisiert (z.B. OSPF), während das Routing über Domains hinweg als externes Routing umgesetzt wird (z.B. BGP). Allerdings müssen alle Router innerhalb einer Domain, die Transit-Datenverkehr übertragen (wie die Domains, die durch Internet Service Provider gebildet werden) auch die Informationen verwalten, die durch das externe Routing geliefert werden, und nicht nur das interne Routing.

Tag-Switching ermöglicht das Entkoppeln von internem und externem Routing, so daß nur an den Grenzen der Domain-Tag-Switches für die Verwaltung der Routing-Information des externen Routing benötigt werden. Alle anderen Switches innerhalb der Domain verwalten die Routing-Information, die durch das interne Routing der Domain bereitgestellt wird. Deren Umfang ist gewöhnlich geringer als der des externen Routing. Dies verringert wiederum die Routing-Belastung der nicht an den Außengrenzen liegenden Switches und verkürzt die Konvergenzzeit des Routing.

Zur Unterstützung dieser Funktionalität erlaubt Tag-Switching die Übertragung mehrerer als Stapelspeicher organisierter Tags in einem Paket. Ein Tag-Switch kann das Tag an die Spitze des Stapelspeichers verlegen, den Stapelspeicher abheben oder ein und mehr Tags auf den Stapelspeicher legen. Wenn ein Paket zwischen zwei (Außen-)Tag-Switches unterschiedlicher Domains übertragen wird, enthält der Tag-Stapelspeicher in dem Paket nur ein Tag.

Bei der Weiterleitung eines Pakets innerhalb einer Domain enthält der Tag-Stapelspeicher in dem Paket allerdings zwei und nicht ein Tag (das zweite Tag wird durch den grenzüberschreitenden Tag-Switch der Domain eingefügt). Das Tag an der Spitze des Stapelspeichers ermöglicht die Paketweiterleitung an einen passenden Domaingrenzen-Tag-Switch, wäh-

rend das nächste Tag auf dem Stapelspeicher für die korrekte Weiterleitung des Pakets durch diesen Switch sorgt. Der Stapelspeicher wird entweder durch den Grenz-Switch oder den vorletzten Switch (relativ zum Grenz-Switch) zurückgesetzt.

Die Steuerungskomponente in diesem Beispiel funktioniert ähnlich der für Destination-Based-Routing verwendeten. Der einzige wesentliche Unterschied ist, daß in diesem Beispiel die Tag-Binding-Information sowohl unter benachbarten Tag-Switches als auch unter Grenz-Tag-Switches innerhalb einer einzelnen Domain ausgetauscht wird.

21.5 Flexibles Routing durch explizite Routen

Eine der grundlegenden Eigenschaften des Destination-Based-Routing ist, daß die einzige von einem Paket für die Weiterleitung verwendete Information die Zieladresse ist. Obwohl dies ein sehr gut skalierbares Routing ermöglicht, schränkt es gleichzeitig die Möglichkeiten zur Beeinflussung der durch die Pakete zurückgelegten Pfade ein. Dies begrenzt die Verfügbarkeit eines gleichmäßig auf mehrere Verbindungen verteilten Datenverkehrs, wobei die Belastung von stark ausgelasteten Verbindungen durch Verlagerung auf weniger ausgelastete Verbindungen verringert würde. Für Internet Service Provider (ISPs), die unterschiedliche Serviceklassen anbieten, schränkt Destination-Based-Routing deren Fähigkeiten zur Auftrennung in unterschiedliche Klassen, unter Berücksichtigung der von diesen Klassen verwendeten Verbindungen, ebenfalls ein. Einige ISPs verwenden heute bereits Frame Relay oder ATM, um die durch das Destination-Based-Routing verursachten Beschränkungen zu vermeiden. Tag-Switching kann aufgrund der flexiblen Granularität der Tags diese Beschränkungen überwinden, ohne Frame Relay oder ATM einzusetzen. Um eine Weiterleitung auf Pfaden zu ermöglichen, die sich von den per Destination-Based-Routing vorgegebenen unterscheiden, erlaubt die Steuerungskomponente des Tag-Switching die Einrichtung von Tag-Bindings in Tag-Switches, die nicht zu den Pfaden des Destination-Based-Routing gehören.

21.6 Multicast-Routing

In einer Multicast-Routing-Umgebung sind Multicast-Routing-Prozeduren (wie das Protocol-Independent Multicast [PIM]) für die Erstellung von Spanning Trees mit den Empfängern als Leaves zuständig. Multicast-Forwarding ist für die Weiterleitung von Multicast-Paketen über diese Bäume verantwortlich.

Tag-Switches unterstützen Multicast, indem sie die Multicast-Fähigkeiten der Daten-Verbindungsschicht nutzen, wie die von Ethernet. Alle Tag-Switches in einem vorgegebenen Multicast-Tree eines gemeinsamen Unternetzes müssen sich auf ein gemeinsames Tag für das Forwarding eines Multicast-Pakets an alle in Verteilungsrichtung liegenden Switches dieses Unternetzes einigen. So wird das Paket über die Daten-Verbindungsschicht auf das Unternetz verteilt. Tag-Switches, die zu einem gemeinsamen Multicast-Tree eines gemeinsamen Daten-Verbindungsunternetzes gehören, einigen sich auf einen Tag-Switch, der für die Bereitstellung eines Tag für diesen Baum zuständig ist. Der Tag-Raum wird in nicht überlappende Regionen für alle Tag-Switches eingeteilt, die mit einem gemeinsamen Subnetz verbunden sind. Jeder Tag-Switch erhält eine Region des Tag-Raums und gibt diese Region seinen Nachbarn bekannt. Konflikte werden auf Basis der IP-Adresse der einbezogenen Switches gelöst. Multicast-Tags sind, statt einem gesamten Tag-Switch, den Schnittstellen eines Tag-Switch zugeordnet. Deshalb verwaltet der Switch TIBs in Verbindung mit einzelnen Schnittstellen, statt eines einzelnen TIB für den gesamten Switch. Die Tags ermöglichen es dem empfangenden Switch, sowohl bestimmte Multicast-Gruppen als auch den Quell-Tag-Switch (den vorherigen Hop), der das Paket gesendet hat, zu erkennen.

Es gibt zwei Möglichkeiten zur Erstellung von Zuordnungen zwischen Tags und Multicast-Trees (Routen). In einer Gruppe von Tag-Switches, die ein gemeinsames Datensubnet teilen, weist der Tag-Switch, der relativ zu einem speziellen Multicast-Tree als Wurzel steht, einer Multicast-Route ein Tag zu und teilt diese Zuordnung anschließend allen Switches in der Verteilungsrichtung des Subnetzes mit. Diese Methode funktioniert ähnlich, wie die Destination-Based-Tag-Zuweisung entgegen der Verteilungsrichtung. Ein Tag-Switch, der relativ

zu einem speziellen Multicast-Tree als Root steht, weist einer Multicast-Route einen Tag zu und teilt diese Zuordnung anschließend allen Switches in und entgegen der Verteilungsrichtung des Subnetzes mit. Der erste Tag-Switch, der der Gruppe beitritt, ist gewöhnlich derjenige, der die Zuweisung vornimmt.

21.7 Tag-Switching mit ATM

Weil das Tag-Switching-Forwarding-Paradigma genau wie ATM-Forwarding auf Label-Swapping beruht, kann die Tag-Switching-Technologie auf ATM-Switches angewendet werden, indem man die Steuerungskomponenten des Tag-Switching einsetzt. Die für das Tag-Switching benötigte Tag-Information kann das ATM-VCI-Feld aufnehmen. Wenn zwei Tag-Ebenen benötigt werden, kann ebenfalls das ATM-VPI-Feld verwendet werden, allerdings schränkt die Größe des VPI-Felds die Größe eines Netzwerks in der Praxis ein. Das VCI-Feld ist für die meisten Anwendungen mit einer Tag-Ebene ausreichend.

Um die notwendigen Steuerungsinformationen zu erhalten, verhält sich der Switch wie ein Peer in Routing-Protokollen der Netzwerk-Schicht, wie OSPF und BGP. Wenn der Switch außerdem eine Sammlung der Routing-Information anlegt, führt er für einen Teil des Datenverkehrs auch Forwarding auf Netzwerk-Ebene aus, um ein zielbasiertes Unicast-Routing zu realisieren. Durch die Unterstützung der Destination-Based-Routing-Funktion mit Tag-Switching auf einem ATM-Switch muß dieser u.U. nicht nur ein, sondern mehrere Tags zu einer Route zuordnen (oder eine Gruppe von Routen mit dem gleichen nächsten Hop). Dies ist notwendig, um die Verschachtelung von Paketen zu vermeiden, die von unterschiedlichen Tag-Switches eintreffen, aber gleichzeitig an den nächsten Hop verschickt werden. Für die Tag-Zuweisung und TIB-Wartungsprozeduren mit ATM-Switches kann sowohl Tag-Zuweisung auf Anforderung (in Übertragungsrichtung) als auch ein Tag-Zuweisungsschema entgegen der Übertragungsrichtung eingesetzt werden.

Deshalb kann ein ATM-Switch Tag-Switching unterstützen; er muß dazu aber mindestens Routing-Protokolle auf Netzwerk-

Ebene und die Tag-Switching-Steuerungskomponente im Switch unterstützen. Es kann auch notwendig werden, Forwarding auf Ebene der Netzwerk-Schicht zu unterstützen.

Der Einsatz des Tag-Switching in einem ATM-Switch würde die Integration von ATM-Switches und Routern stark vereinfachen. Ein ATM-Switch, der Tag-Switching beherrscht, erscheint benachbarten Routern als Router. Dies ermöglicht eine skalierbare Alternative zum »Overlay«-Modell und beseitigt die Notwendigkeit von ATM-Adressierungs-, Routing- und Signalsierungsschemata. Weil das Destination-Based-Forwarding durch die Topologie und nicht durch den Datenverkehr bestimmt wird, beinhaltet die Anwendung dieses Ansatzes auf ATM-Switches keine hohen Ruf-Einrichtungsraten und ist auch nicht von der Entfernung der Übertragung abhängig.

Die Einrichtung des Tag-Switching auf einem ATM-Switch stört die Fähigkeiten der traditionellen ATM-Control-Plane (z.B. PNNI) auf dem gleichen Switch nicht. Die beiden Komponenten, Tag-Switching und die ATM-Control-Plane, funktionieren unabhängig voneinander mit aufgeteilten VPI/VCI-Räumen und anderen Ressourcen, so daß die Komponenten nicht aufeinander einwirken.

21.8 Quality of Service

Ein wichtige Fähigkeit des Tag-Switching ist die Unterstützung von Quality-of-Service (QoS). Zwei Mechanismen werden benötigt, um einen QoS-Bereich für Pakete bereitzustellen, die einen Router oder Tag-Switch durchqueren:

- Zuordnung der Pakete zu verschiedenen Klassen
- Behandlung der Pakete über entsprechende QoS-Merkmale (wie Bandbreite und Verlust)

Tag-Switching stellt eine einfache Möglichkeit zur Markierung von Paketen und damit zur Zuordnung zu einer bestimmten Klasse nach deren erstmaliger Klassifizierung bereit.

Die Anfangsklassifizierung kann mit der in den Headern der Netzwerk-Schicht oder höherer Schichten übertragenen Information durchgeführt werden. Infolgedessen wird diesem Paket ein der resultierenden Klasse entsprechendes Tag zugeordnet.

Markierte Pakete können auf ihrem weiteren Weg durch Tag-Switching-Router ohne eine Neuklassifizierung sehr effektiv bearbeitet werden. Das jeweilige Paket-Scheduling und Queuing ist weitestgehend vergleichbar: Der Punkt dabei ist, daß Tag-Switching das Scheduling mit einer einfachen Logik ermöglicht und so die Information gefunden wird, die festlegt, wie der zeitliche Versand des Pakets auszuführen ist.

Der korrekte Einsatz des Tag-Switching für QoS-Zwecke erfordert genaue Kenntnisse über den Einsatz von QoS. Wenn für die Anforderung einer bestimmten QoS für eine Klasse von Paketen RSVP verwendet wird, dann ist es notwendig, ein Tag für jede RSVP-Sitzung zuzuweisen, für die auf einem Tag-Switch ein Status installiert ist. Dies kann per TDP oder mit einer Erweiterung des RSVP ausgeführt werden.

21.9 IP-Switching

IP-Switching ist eine ähnliche Technologie, die ATM-Layer-2-Switching mit Layer-3-Routing verbindet. Somit ist es eine andere Art des Multi-Layer-Switching. IP-Switching weist üblicherweise ein Label pro Quelle/Ziel-Paketfluß zu. Ein IP-Switch bearbeitet die Anfangspakete bei einer Übertragung, indem er sie an das Standard-Router-Modul weiterleitet, das Bestandteil des IP-Switch ist.

Wenn ein IP-Switch eine ausreichende Anzahl von Paketen von einer Verbindung gesehen hat, um davon ausgehen zu können, daß dieses langfristig ist, richtet er Labels für diese Verbindung mit seinen benachbarten IP-Switches oder Edge-Routern ein, so daß weitere Pakete der Verbindung mit hoher Geschwindigkeit nach dem Label vermittelt werden können (so wie eine ATM-Vermittlungseinheit) und die langsameren Router-Module umgehen. IP-Switching-Gateways sind für die Umwandlung von Paketen von Formaten ohne Labels in Formate mit Labels und von Paket-Medien in ATM zuständig.

KAPITEL 22

Mixed-Media-Bridging

22.1 Grundlagen

Transparent Bridges werden hauptsächlich in Ethernet-Netzwerken eingesetzt. Source-Route-Bridges (SRBs) finden Sie dagegen fast ausschließlich in Token-Ring-Netzwerken. Sowohl transparente Bridges als auch SRBs sind weit verbreitet. Deshalb ist es sinnvoll zu fragen, ob eine Methode existiert, die beide vereint. Es wurden mehrere solche Lösungen entwickelt.

Translational-Bridging stellt eine relativ preiswerte Lösung für einige der Probleme beim Bridging zwischen Transparent-Bridging- und SRB-Domains dar. Translational-Bridging erschien erstmalig Mitte bis Ende der 80er Jahre, wurde aber bisher von keiner Standardisierungsorganisation favorisiert. Deshalb sind viele Aspekte des Translational-Bridging abhängig vom jeweiligen Einsatz.

Im Jahre 1990 löste IBM einige Schwächen des Translational-Bridging durch Einführung des Source-Route-Transparent (SRT). Bridging-SRT-Bridges können sowohl Datenverkehr von Transparent als auch Source-Route-Endknoten weiterleiten und einen gemeinsamen Baum mit transparenten Bridges bilden, und ermöglichen damit Endstationen jedes Typs, mit Endstationen des gleichen Typs, innerhalb eines Netzwerks mit beliebiger Topologie zu kommunizieren. SRT wurde in der IEEE 802.1d Anhang C spezifiziert.

Das Ziel der Verbindung von Transparent-Bridging und SRB-Domains ist, eine Übertragung zwischen transparenten Bridges

und SRB-Endstationen zu ermöglichen. Dieses Kapitel beschreibt die technischen Probleme, für die Algorithmen gefunden werden müssen, und stellt zwei mögliche Lösungen vor: Translational-Bridging und SRT-Bridging.

22.2 Übertragungsanforderungen

Viele Anforderungen stehen in Zusammenhang mit der Notwendigkeit, Endstationen einer Ethernet/Transparent-Bridging-Domain die Kommunikation mit Endstationen einer SRB/Token-Ring-Domain zu ermöglichen:

- Inkompatible Bitreihenfolge – Obwohl sowohl Ethernet als auch Token-Ring 48-bit Media Access Control (MAC)-Adressen unterstützen, unterscheidet sich die interne Darstellung dieser Adressen in der Hardware. Im seriellen Bitstrom einer Adresse versteht ein Token-Ring-Gerät das erste gefundene Bit als hochwertigstes Bit eines Byte. Ethernet sieht auf der anderen Seite das erste gefundene Bit als niederwertigstes Bit an.
- Eingebettete MAC-Adressen – In manchen Fällen werden MAC-Adressen im Datenteil eines Frame übertragen. Das *Address Resolution Protocol* (ARP) – ein verbreitetes Protokoll in *Transmission Control Protocol/Internet Protocol* (TCP/IP)-Netzwerken, legt z.B. Hardware-Adressen im Datenteil des Sicherungsschicht-Frame ab. Die Umwandlung von Adressen, die im Datenteil eines Frame auftauchen können (oder auch nicht), ist kompliziert, weil sie von Fall zu Fall unterschiedlich behandelt werden.
- Inkompatible Maximum-Transfer-Unit-(MTU-)Größen – Token-Ring und Ethernet unterstützen eine unterschiedliche maximale Frame-Größe. Die Ethernet-MTU ist etwa 1500 Byte, während Token-Ring-Frames wesentlich größer sein können. Weil Bridges nicht in der Lage sind, Frames zu fragmentieren und wieder zusammenzusetzen, müssen Pakete, welche die MTU eines Netzwerks überschreiten, verworfen werden.
- Behandlung von Frame-Statusbit-Aktionen – Token-Ring-Frames enthalten drei Frame-Statusbits: A, C und E. Sinn dieser Bits ist die Mitteilung an die Frame-Quelle, ob das

Ziel den Frame erkannte (A-Bit gesetzt), den Frame kopierte (C-Bit gesetzt) oder Fehler im Frame fand (E-Bit gesetzt). Weil Ethernet diese Bits nicht unterstützt, bleibt die Frage nach deren Behandlung dem Hersteller von Ethernet-Token-Ring-Bridges überlassen.

– Behandlung ausschließlicher Token-Ring-Funktionen – Für bestimmte Token-Ring-Bits gibt es kein Äquivalent im Ethernet. Ethernet beinhaltet z.B. keinen Prioritätsmechanismus, während Token-Ring über einen solchen verfügt. Weitere Token-Ring-Bits, die bei der Umwandlung eines Token-Ring-Frame in einen Ethernet-Frame entfernt werden, sind das Token-Bit, das Monitor-Bit und das Reservierungsbit.

– Behandlung von Explorer-Frames – Transparente Bridges wissen grundsätzlich nicht, wie sie SRB-Explorer-Frames behandeln sollen. Transparente Bridges erlernen die Netzwerk-Topologie über eine Analyse der Quellenadressen der eintreffenden Frames. Sie besitzen keine Informationen über den SRB-Route-Erkennungsprozeß.

– Behandlung von Routing Information Field (RIF)-Informationen in Token-Ring-Frames – Der SRB-Algorithmus legt Routing-Informationen im RIF-Feld ab. Der Transparent-Bridging-Algorithmus besitzt kein RIF-Äquivalent, und der Gedanke, Routing-Informationen in einem Frame abzulegen, ist bei Transparent-Bridging völlig unbekannt.

– Inkompatible Spanning-Tree-Algorithmen – Transparent-Bridging und SRB verwenden den Spanning-Tree-Algorithmus, um Schleifen zu vermeiden, aber die jeweiligen Algorithmen der beiden Bridging-Methoden sind nicht kompatibel.

– Behandlung von Frames ohne Route-Information – SRB erwartet von allen LAN-überschreitenden Frames, daß sie Route-Informationen enthalten. Ein Frame ohne RIF-Feld (einschließlich Transparent-Bridging-Konfigurations- und Topologieänderungsmeldungen, ebenso wie MAC-Frames, die von einer Transparent-Bridging-Domain stammen), der in einer SRB-Bridge ankommt, wird ignoriert.

22.3 Translational-Bridging

Weil es keine wirkliche Standardisierung dazu gibt, wie die Übertragung zwischen zwei unterschiedlichen Medienarten stattfinden soll, kann keine einzige Translational-Bridging-Umsetzung als korrekt bezeichnet werden. Im Folgenden werden einige verbreitete Methoden der Umsetzung des Translational-Bridging beschrieben.

Translational-Bridges ordnen Adreßbits von Quelle und Ziel neu an, wenn sie Ethernet und Token-Ring-Frames übersetzen. Das Problem der eingebetteten MAC-Adressen kann gelöst werden, indem man die Bridge so programmiert, daß sie nach verschiedenen MAC-Adreßtypen sucht, aber diese Lösung muß an jeden neuen eingebetteten MAC-Adreßtyp angepaßt werden. Einige Translational-Bridging-Lösungen prüfen einfach auf die verbreitetsten eingebetteten Adressen. Wenn Translational-Bridging-Software auf einem Multiprotocol-Router läuft, kann der Router dieses Protokoll erfolgreich vermitteln und das Problem völlig vermeiden.

Das RIF-Feld besitzt ein Unterfeld, welches die maximale Frame-Größe angibt, die eine spezielle SRB-Umsetzung akzeptiert. Translational Bridges setzen das MTU-Größenfeld gewöhnlich auf 1500 Byte, wenn sie Frames von Transparent-Bridging-Domains an SRB-Domains versenden, und beschränken so die Größe von Token-Ring-Frames beim Übergang in die Transparent-Bridging-Domain. Manche Hosts können dieses Feld nicht richtig verarbeiten, was dazu führt, daß Translational-Bridges Frames verwerfen, die die Ethernet-MTU-Größe überschreiten.

Bits, die Token-Ring-Funktionen ohne Ethernet-Äquivalent darstellen, werden üblicherweise von den Translational-Bridges entfernt. Token-Ring-Priorität-, Reservation- und Monitorbits (im Zugriffssteuerungs-Byte) werden z.B. verworfen. Statusbits aus Token-Ring-Frames (im Byte nach dem Ending-Delimiter, der dem Datenfeldende folgt) werden von verschiedenen Bridge-Herstellern unterschiedlich behandelt. Manche Bridge-Hersteller ignorieren diese Bits. Andere Bridges setzen das C-Bit (und weisen damit darauf hin, daß der Frame kopiert wurde), aber nicht das A-Bit (welches anzeigt, daß die Zielstation die Adresse anerkennt). In diesem

Fall stellt ein Token-Ring-Quellknoten fest, ob der gesendete Frame verlorenging. Gegner dieses Ansatzes vertreten die Meinung, daß Zuverlässigkeitsmechanismen, wie das Verfolgen verlorengegangener Frames, in Schicht 4 des OSI-Modells realisiert werden sollten. Gegner des Ansatzes, das »C-Bit« zu setzen, argumentieren, daß dieses Bit zur Verfolgung verlorener Frames gesetzt werden muß, das A-Bit aber nicht gesetzt werden kann, weil die Bridge nicht das Endziel ist.

Translational-Bridges können einen Software-Gateway zwischen zwei Domains erstellen. Für SRB-Endstationen besitzt die Translational-Bridge eine Ring-Nummer und eine zugeordnete Bridge-Nummer und erscheint als Standard-SRB. Die Ring-Nummer stellt in diesem Fall die gesamte Transparent-Bridging-Domain dar. Für die Transparent-Bridging-Domain ist die Translational Bridge einfach eine andere Transparent Bridge.

Beim Bridging von einer SRB-Domain zu einer Transparent-Bridging-Domain wird die SRB-Information entfernt. RIFs werden gewöhnlich zur Verwendung mit dem folgenden Antwortverkehr zwischengespeichert. Beim Bridging vom Transparent-Bridging zur SRB-Domain kann die Translational Bridge den Frame auf ein Unicast-Ziel prüfen. Wenn der Frame ein Multicast- oder Broadcast-Ziel besitzt, wird er als Spanning-Tree-Explorer in die SRB-Domain verschickt. Wenn der Frame eine Unicast-Adresse besitzt, sucht die Translational-Bridge das Ziel im RIF-Cache. Wenn der Pfad gefunden wurde, wird er verwendet und die RIF-Information dem Frame hinzugefügt; sonst wird der Frame als Spanning-Tree-Explorer verschickt. Weil die beiden Spanning-Tree-Umsetzungen inkompatibel sind, sind mehrere Pfade zwischen SRB- und Transparent-Bridging-Domains üblicherweise nicht zulässig. Bild 22.2 bis Bild 22.3 zeigen Frame-Umwandlungen, die beim Translational-Bridging stattfinden können.

Bild 22.1 verdeutlicht die Frame-Umwandlung zwischen IEEE 802.3 und Token-Ring. Die Destination and Source Adresses (DASA), der Service-Access Point (SAP), die Logical-Link Control (LLC)-Information und die Daten werden in die entsprechenden Felder des Ziel-Frame übertragen. Die Ziel- und Quelladreßbits werden umgestellt. Beim Bridging von IEEE 802.3 zu Token-Ring wird das Längenfeld des IEEE-802.3-

Frame entfernt. Beim Bridging von Token-Ring zu IEEE 802.3 werden Zugriffsteuerbyte und RIF entfernt. Der RIF kann in der Translational-Bridge zur Verwendung mit dem Antwortverkehr zwischengespeichert werden.

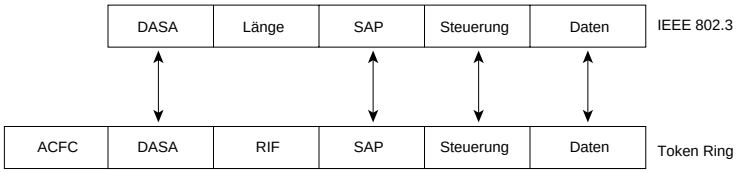


Bild 22.1: Vier Felder bleiben bei der Frame-Umwandlung zwischen IEEE 802.3 und Token Ring gleich

Bild 22.2 verdeutlicht die Frame-Umwandlung zwischen Ethernet-Typ II und Token-Ring Subnetwork Access Protocol (SNAP). (SNAP fügt Verteiler- und Typencodes zum Datenfeld des Token-Ring-Frame hinzu.) Die Ziel- und Quellenadressen, Typeninformation und Daten werden in die entsprechenden Felder des Ziel-Frame übertragen, und die DASA-Bits werden neu angeordnet. Beim Bridging von Token-Ring-SNAP zu Ethernet-Typ II werden RIF-Information, SAP, LLC-Information und Vendorcode entfernt. Der RIF kann in der Translational-Bridge zur Verwendung mit dem Antwortverkehr zwischengespeichert werden. Beim Bridging von Ethernet-Typ II zu Token-Ring-SNAP werden keine Informationen entfernt.

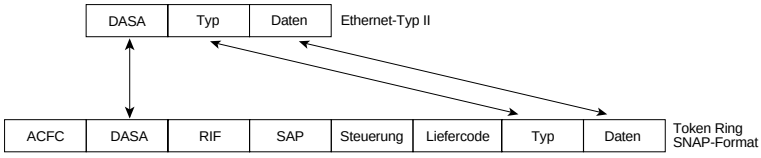


Bild 22.2: Bei der Frame-Umwandlung zwischen Ethernet-Typ II und Token-Ring-SNAP bleiben drei Felder gleich

Bild 22.3 verdeutlicht die Frame-Umwandlung zwischen Ethernet-Typ-II-»0x80D5«-Format und Token Ring. (Ethernet-Typ II »0x80D5« überträgt IBM-SNA-Daten in Ethernet-Frames.) Die DASA-, SAP-, LLC-Informationen und Daten werden in die entsprechenden Felder des Ziel-Frame übertragen, und die DASA-Bits werden neu angeordnet. Beim Bridging von Ethernet-Typ II »0x80D5« zu Token Ring werden die Typen- und 80D5-Headerfelder entfernt. Beim Bridging von Token Ring zu Ethernet-Typ II »0x80D5« wird der

RIF entfernt. Der RIF kann in der Translational-Bridge zur Verwendung mit dem Antwortverkehr zwischengespeichert werden.

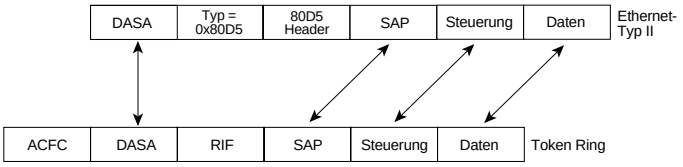


Bild 22.3: Bei der Frame-Umwandlung zwischen Ethernet-Typ II »0x80D5« und Token Ring bleiben vier Felder gleich

22.4 Source-Route-Transparent-Bridging

SRT-Bridges verbinden Realisierungen des Transparent-Bridging- und des SRB-Algorithmus. SRT-Bridges verwenden das *Routing-Information-Anzeige*(RII)-Bit, um zwischen Frames auf SRB-Basis und Frames mit Transparent-Bridging zu unterscheiden. Ist das RII-Bit gleich 1, ist ein RIF im Frame vorhanden, und die Bridge verwendet den SRB-Algorithmus. Ist das RII-Bit gleich 0, ist kein RIF im Frame vorhanden, und die Bridge verwendet Transparent-Bridging.

Wie bei den Translational-Bridges stellen SRT-Bridges keine perfekte Lösung für die Probleme des Bridging zwischen verschiedenen Medien dar. SRT-Bridges müssen immer noch die Ethernet/Token-Ring-Inkompatibilitäten lösen, die bereits beschrieben wurden. SRT-Bridging erfordert wahrscheinlich eine Aufrüstung der Hardware der SRBs, um die erhöhte Belastung durch die Analyse jedes Pakets zu meistern. Es können auch Software-Upgrades von SRBs notwendig werden. Weiterhin müssen in Umgebungen mit gemischten SRT-Bridges, transparente Bridges, und SRBs die gewählten Source-Routen alle verfügbaren SRT-Bridges und SRBs überspannen. Die Ergebnispfade sind möglicherweise Spanning-Tree-Pfaden von transparenten Bridges wesentlich unterlegen. Letztendlich verlieren gemischte SRB/SRT-Bridging-Netzwerke die Vorteile des SRT-Bridging, so daß die Benutzer sich gezwungen sehen, eine komplette Umstellung zum SRT-Bridging mit beträchtlichen Kosten durchzuführen. Immerhin erlaubt SRT-Bridging die Koexistenz zweier zueinander nicht kompatibler Umgebungen und die Kommunikation zwischen SRB- und Transparent-Bridging-Endknoten.

KAPITEL 23

Source-Route-Bridging (SRB)

23.1 Grundlagen

Der Source-Route-Bridging-(SRB-)Algorithmus wurde von IBM entwickelt und dem IEEE-802.5-Komitee als Mittel für das Bridging zwischen allen LANs vorgelegt. Seit der Erstvorlage hat IBM dem IEEE-802-Komitee einen neuen Bridging-Standard vorgelegt: das Source-Route-Transparent-(SRT-)-Bridging. Das SRT-Bridging beseitigt Probleme reiner SRBs vollständig und schlägt vor, für die beiden LAN-Bridges-Typen transparente Bridges und SRT-Bridges zu verwenden. Obwohl SRT-Bridging unterstützt wurde, sind SRBs noch im breiten Einsatz. SRT wird in Kapitel 22, »Mixed-Media Bridging« behandelt. Dieses Kapitel behandelt den grundlegenden SRB-Frame-Forwarding-Algorithmus und beschreibt SRB-Frame-Felder.

23.2 SRB-Algorithmus

SRBs werden so bezeichnet, weil sie davon ausgehen, daß die komplette Quelle-zu-Ziel-Route in allen LAN-übergreifenden Frames plaziert wird, die von der Quelle versendet werden. SRBs speichern die Frames und leiten sie so weiter, wie durch die Route im entsprechenden Frame-Feld vorgegeben wurde. Bild 23.1 zeigt ein Beispiel für ein SRB-Netzwerk.

In Bild 23.1 ist davon auszugehen, daß Host X einen Frame an Host Y senden will. Anfangs ist Host X nicht bekannt, ob sich Host Y im gleichen oder in einem anderen LAN befindet. Host X sendet einen Test-Frame, um das festzustellen. Wenn dieser Frame ohne ein positives Anzeichen dafür zu Host X zurückkehrt, daß Host Y ihn erhalten hat, geht Host X davon aus, daß sich Host Y in einem entfernten Segment befindet.

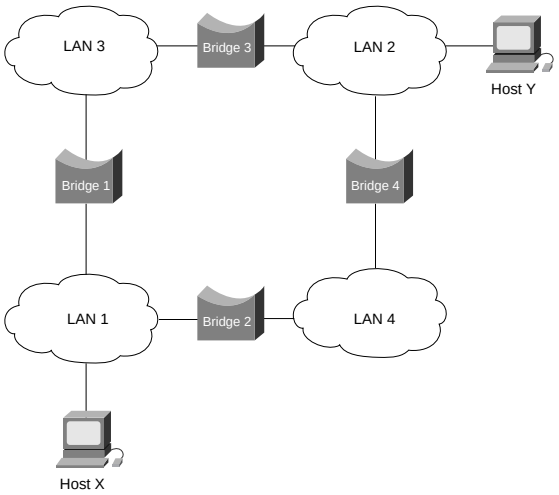


Bild 23.1: Ein SRB-Netzwerk enthält LANs und Bridges

Host X sendet einen *Explorer*-Frame, um den genauen Ort von Host Y zu bestimmen. Jede Bridge, die den Explorer-Frame erhält (in diesem Beispiel Bridges 1 und 2), kopiert den Frame zu allen ausgehenden Anschlüssen. Die Explorer-Frames werden mit der Routen-Information ergänzt, während sie durch das Netzwerk reisen. Wenn die Explorer-Frames von Host X Host Y erreichen, antwortet Host Y auf jeden einzeln und verwendet dabei die angesammelten Routen-Informationen. Nach Empfang aller Antwort-Frames wählt Host X nach vorgegebenen Kriterien einen Pfad aus.

Im Beispiel in Bild 23.1 führt dieser Prozeß zu zwei Routen:

– LAN 1 zu Bridge 1 zu LAN 3 zu Bridge 3 zu LAN 2
– LAN 1 zu Bridge 2 zu LAN 4 zu Bridge 4 zu LAN 2

Host X muß eine dieser beiden Routen auswählen. Die IEEE-802.5-Spezifikation legt kein Kriterium für die Auswahl der Route durch Host X fest, gibt aber einige Empfehlungen, zu denen auch die folgenden gehören:

– Erster empfangener Frame

– Mit minimaler Anzahl von Hops antworten

– Nach größter erlaubter Frame-Größe antworten

– Verschiedene Kombinationen der genannten Kriterien

In den meisten Fällen wird der Pfad aus dem ersten empfangenen Frame verwendet.

Nach Auswahl einer Route wird diese in Frames verpackt, die als *Routing-Information-Field* (RIF) an Host Y gehen. Ein RIF wird nur in die Frames eingefügt, die für andere LANs bestimmt sind. Das Vorhandensein von Routing-Informationen im Frame wird durch Setzen des hochwertigsten Bit innerhalb des Source-Adressenfelds angezeigt, welches als *Routing Information Indicator* (RII)-Bit bezeichnet wird.

23.3 Frame-Format

Die Struktur des IEEE-802.5-RIF wird in Bild 23.2 gezeigt.

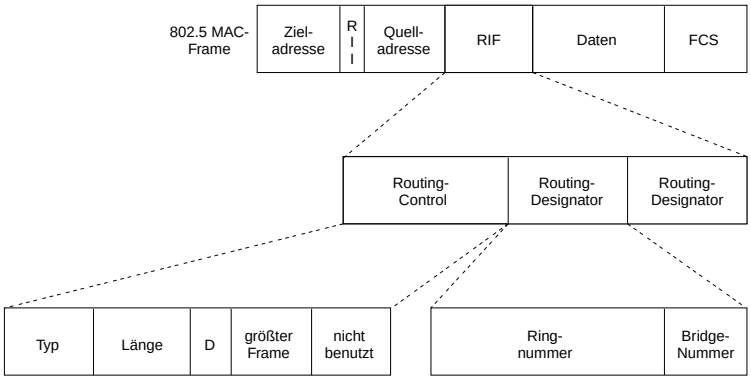


Bild 23.2: Ein IEEE-802.5-RIF

Der in Bild 23.2 gezeigte RIF enthält zwei Hauptfelder: *Routing Control* und *Routing Designator*. Diese Felder werden in der folgenden Zusammenfassung beschrieben.

23.3.1 Routing-Control-Feld

Das Routing-Control-Feld enthält zwei Teilfelder: Typ, Länge, D-Bit und größter Frame. Die Felder werden in der folgenden Liste zusammengefaßt:

- *Typ* – Enthält drei mögliche Typen von Routing-Controls:
  - Specifically Routed – Wird verwendet, wenn der Quellknoten die Route im RIF-Header vorgibt. Die Bridges vermitteln den Frame entsprechend dem (den) Route-Designator-Feld(ern).
  - All Paths Explorer – Wird verwendet, um einen entfernten Knoten zu finden. Die Route wird gesammelt, während der Frame das Netzwerk durchquert. Bridges ergänzen den Frame mit ihrer Bridge-Nummer und der Ring-Nummer, an die der Frame weitergeleitet wird. (Die erste Bridge ergänzt auch die Ring-Nummer des ersten Rings.) Das Ziel empfängt so viele Frames, wie es Routen zu diesem Ziel gibt.
  - Spanning-Tree-Explorer – Wird verwendet, um einen entfernten Knoten zu finden. Nur Bridges im Spanning-Tree leiten den Frame weiter, wobei sie ihn mit ihrer Bridge-Nummer und der angeschlossenen Ring-Nummer ergänzen, an die er weitergeleitet wird. Der Spanning-Tree-Explorer reduziert die Anzahl der Frames, die im Verlauf des Suchprozesses versandt wurden.
- *Länge* – Gibt die Gesamtlänge des RIF in Byte an. Der Wert kann zwischen 2 und 30 Byte liegen.
- *D-Bit* – Zeigt und steuert die Richtung (vorwärts oder rückwärts) der Übertragung des Frame. Das D-Bit legt fest, ob Bridges die Ring-Nummer/Bridge-Nummer-Kombinationen der Route-Designatoren von rechts nach links (vorwärts) oder links nach rechts (rückwärts) lesen.
- *Größter Frame* – Gibt die maximale Frame-Größe an, die auf einer festgelegten Route verarbeitet werden kann. Die Quelle setzt einen Anfangswert für die maximale Frame-Größe ein, den die Bridges verringern können, wenn sie die angeforderte Größe nicht erreichen können.

23.3.2 Routing-Designator-Felder

Jedes Routing-Designator-Feld enthält zwei Teilfelder:

- *Ring-Nummer (12 Bit)* – Der zugeordnete Wert muß innerhalb des durch Bridges verbundenen Netzwerks einmalig sein.
- *Bridge-Nummer (4 Bit)* – Der zugeordnete Wert folgt der Ring-Nummer. Diese Nummer muß nicht einmalig sein, solange sie nicht parallel zu einer anderen Bridge liegt, die zwei Ringe verbindet.

Bridges ergänzen den Frame mit ihrer Bridge-Nummer und der Ring-Nummer, an die der Frame weitergeleitet wird. (Die erste Bridge ergänzt auch die Ring-Nummer des ersten Rings.)

Routen sind wechselnde Folgen von Ring- und Bridge-Nummern, die mit Ring-Nummern beginnen und enden. Ein einzelner RIF kann mehr als ein Routing-Designator-Feld enthalten. Das IEEE gibt ein Maximum von 14 Routing-Designator-Feldern vor (d.h. maximal 13 Bridges oder Hops, weil die letzte Bridge-Nummer immer gleich 0 ist).

Bis vor kurzem legte IBM ein Maximum von acht Routing-Designator-Feldern fest (maximal sieben Bridges oder Hops), und die meisten Bridge-Hersteller folgten dieser Vorgabe. Neuere IBM-Bridge-Software kann auf neuen LAN-Adaptern bis zu 13 Hops unterstützen.

KAPITEL 24

Transparent-Bridging

24.1 Grundlagen

Transparent-Bridges wurden zuerst durch die Digital Equipment Corporation (Digital) in den frühen 80er Jahren entwikkelt. Digital stellte diese Ausarbeitungen dem Institute of Electrical and Electronic Engineers (IEEE) zur Verfügung, welches diese in den Standard IEEE 802.1. aufnahm. Transparente Bridges sind in Ethernet/IEEE-802.3-Netzwerken sehr verbreitet. Dieses Kapitel gibt einen Überblick über die Handhabung des Verkehrs und von Protokoll-Komponenten durch Transparent-Bridges.

24.2 Transparent-Bridging-Betrieb

Transparente-Bridges werden so bezeichnet, weil ihr Vorhandensein und ihr Betrieb für Netzwerk-Hosts transparent ist. Wenn transparente Bridges eingeschaltet werden, erlernen sie die Topologie des Netzwerks, indem sie die Quellenadressen aller eingehenden Frames der angeschlossenen Netzwerke analysieren. Wenn beispielsweise eine Bridge einen Frame erkennt, der auf Leitung 1 von Host A ankommt, schließt die Bridge daraus, daß Host A über das Netzwerk erreicht werden kann, das an Leitung 1 angeschlossen ist. Im Verlauf dieses Vorgangs bauen transparente Bridges eine Tabelle auf, wie etwa die in Bild 24.1.

Bild 24.1: Transparente Bridges bauen eine Tabelle auf, die die Erreichbarkeit des Host angibt

Host-Adresse	Netzwerknummer
15	1
17	1
12	2
13	2
18	1
9	1
14	3
.	.
.	.
.	.

Die Bridge verwendet ihre Tabelle als Basis für die Weiterleitung des Datenverkehrs. Wenn die Bridge an einer ihrer Schnittstellen einen Frame empfängt, sucht sie dessen Zieladresse in ihrer internen Tabelle. Enthält die Tabelle eine Zuordnung zwischen der Zieladresse und einem anderen Anschluß als dem, von dem der Frame empfangen wurde, wird der Frame über den angegebenen Anschluß weitergeleitet. Wird keine Zuordnung gefunden, wird der Frame an alle Anschlüsse außer dem Eingangsanschluß versandt. Broadcasts und Multicasts werden ebenfalls auf diese Art verteilt.

Transparente Bridges isolieren den Intrasegment-Datenverkehr sehr wirkungsvoll und reduzieren damit den Datenverkehr in den einzelnen Segmenten. Dadurch werden die Antwortzeiten des Netzwerks für den Benutzer sichtbar verbessert. Der Umfang der Datenverkehrsreduzierung und Verbesserung der Antwortzeiten ist vom Volumen des Datenverkehrs zwischen den Segmenten im Verhältnis zum gesamten Datenverkehr abhängig sowie weiterhin vom Volumen des Broadcast- und Multicast-Datenverkehrs.

24.2.1 Bridging-Loops

Ohne die Existenz eines Bridge-to-Bridge-Protokolls versagt der Transparent-Bridge-Algorithmus für den Fall, daß mehrere Pfade aus Bridges und Local Area Networks (LANs) zwischen zwei beliebigen LANs im Netzwerk existieren. Bild 24.2 stellt eine solche Bridging-Loop dar.

Gehen wir davon aus, daß Host A einen Frame an Host B sendet. Beide Bridges empfangen den Frame und schließen daraus korrekterweise, daß Host A sich im Netzwerk 2 befindet. Nachdem Host B zwei Kopien des Host-A-Frame empfangen hat, empfangen beide Bridges unglücklicherweise den Frame noch einmal an ihren Netzwerk-1-Schnittstellen, weil in Broadcast-LANs alle Hosts alle Meldungen empfangen. In einigen Fällen ändern die Bridges ihre internen Tabellen, um anzuzeigen, daß sich Host A im Netzwerk 1 befindet. Wenn dem so ist, empfangen bei einer Antwort von Host B auf den Frame von Host A beide Bridges diese Antwort und verwerfen sie, weil ihre Tabellen angeben, daß der Zielhost (Host A) sich im selben Netzwerksegment befindet, wie die Quelle des Frame.

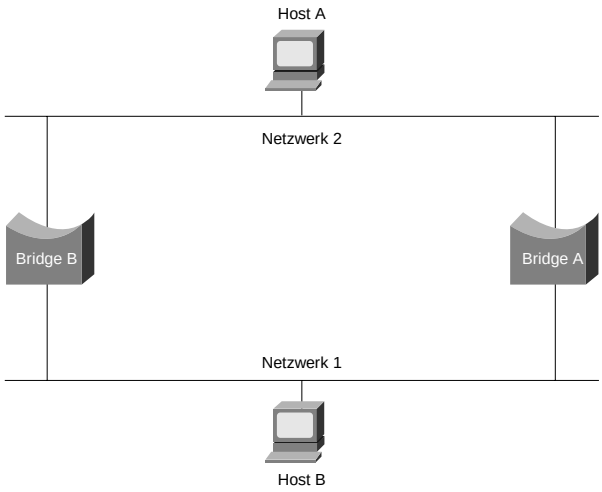


Bild 24.2: Bridging-Loops können zu ungenauer Weiterleitung und Informationsaufnahme in Transparent-Bridging-Umgebungen führen

Zusätzlich zu grundlegenden Verbindungsproblemen stellt die Multiplizierung von Broadcast-Meldungen in Netzwerken mit Schleifen ein ernsthaftes, potentielles Netzwerk-Problem dar. Wir beziehen uns wieder auf Bild 24.2 und nehmen an, daß der erste Frame von Host A ein Broadcast ist. Beide Bridges leiten die Frames immer wieder weiter, verwenden dabei die gesamte verfügbare Bandbreite des Netzwerks und blockieren so die Übertragung anderer Pakete in beiden Segmenten.

Eine Topologie mit Schleifen, wie in Bild 24.2 gezeigt, kann ebenso nützlich wie potentiell gefährlich sein. Eine Schleife ermöglicht mehrere Pfade durch ein Netzwerk. Netzwerke mit mehreren Wegen von der Quelle zum Ziel können durch die höhere Flexibilität der Topologie eine höhere gesamte Fehlertoleranz im Netzwerk erreichen.

24.2.2 Spanning-Tree-Algorithmus (STA)

Der Spanning-Tree-Algorithmus (STA) wurde durch die Digital Equipment Corporation, ein Ethernet-Hersteller, entwikkelt, um die Vorteile von Schleifen zu erhalten und die Probleme zu umgehen. Der Digital-Algorithmus wurde später durch das IEEE-802-Komitee überarbeitet und in der IEEE-802.1d-Spezifikation veröffentlicht. Der Digital-Algorithmus und der Algorithmus nach IEEE 802.1d sind nicht kompatibel.

Der STA legt eine schleifenfreie Untermenge der Netzwerk-Topologie fest, indem er die Bridge-Anschlüsse, die bei Aktivierung Schleifen bilden würden, in einen Standby-(Blocking-)-Zustand versetzt. Gesperrte Bridge-Anschlüsse können im Falle eines primären Verbindungsausfalls aktiviert werden und so einen neuen Weg durch das Netzwerk bilden.

Der STA verwendet einen Ansatz aus der Graphentheorie als Grundlage für die Herstellung von schleifenfreien Untermengen der Netzwerk-Topologie. Die Graphentheorie besagt u.a. folgendes:

> Für jeden verbundenen Graphen aus Knoten und Kanten, die Knotenpaare verbinden, stellt ein Spanning tree aus Kanten die Verbindungen des Graphen her, er enthält aber keine Schleifen.

Bild 24.3 zeigt, wie der STA Schleifen entfernt. Der STA ruft jede Bridge für die Vergabe eines eindeutigen Identifikators auf. Üblicherweise besteht dieser Identifikator aus einer der *Media Access Control* (MAC)-Adressen der Bridge und einer

Priorität. Allen Anschlüssen in allen Bridges wird weiterhin ein eindeutiger Identifikator (innerhalb dieser Brücke), der üblicherweise durch seine eigene MAC-Adresse gebildet wird, zugewiesen. Abschließend wird jedem Bridge-Anschluß ein Pfadkostenwert zugeordnet, der die Kosten für die Übertragung eines Frame in ein LAN über diesen Anschluß darstellt. In Bild 24.3 sind die Pfadkosten auf den Leitungen eingetragen, die die Bridges verlassen. Pfadkosten werden üblicherweise als Vorgabewerte gesetzt, können aber durch den Netzwerk-Administrator manuell verändert werden.

Als erster Schritt bei der Berechnung eines Spanning-Tree wird die *Root-Bridge* (Wurzel-Bridge) ausgewählt, welche die Bridge mit dem Bridge-Identifikator mit dem kleinsten Wert ist. In Bild 24.3 ist Bridge 1 die Root-Bridge. Als nächstes wird der *Root-Anschluß* an allen anderen Bridges bestimmt. Der Root-Anschluß einer Bridge ist der Anschluß, über den die Root-Bridge mit den geringsten Gesamtpfadkosten erreicht werden kann. Dieser Wert wird als *Root-Pfadkosten* bezeichnet.

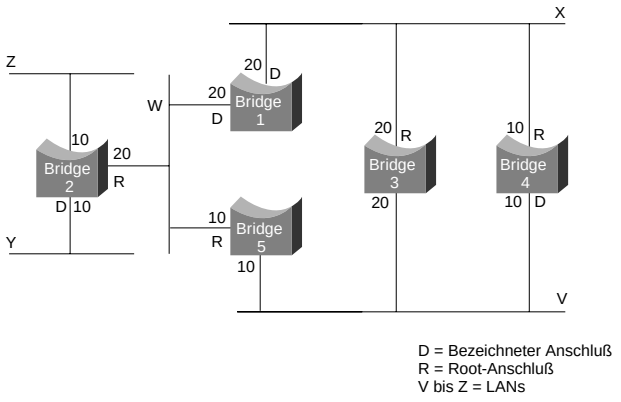


Bild 24.3: STA-basierte Bridges verwenden bezeichnete und Root-Anschlüsse, um Loops zu beseitigen

Abschließend werden bezeichnete Bridges und deren bezeichnete Anschlüsse bestimmt. Eine bezeichnete Bridge ist die Bridge in jedem LAN, welche die minimalen Root-Pfadkosten gewährleistet. Diese Bridge ist die einzige, welche Frames an und von dem LAN weiterleiten darf, für das sie als bezeichnete Bridge eingesetzt ist. Ein bezeichneter Anschluß eines

LAN ist der Anschluß, welcher es mit der bezeichneten Bridge verbindet.

In manchen Fällen können zwei oder mehr Bridges die gleichen Root-Pfadkosten haben. In Bild 24.3 können z.B. sowohl Bridge 4 als auch 5 Bridge 1 (die Root-Bridge) Pfadkosten von 10 erreichen. In diesem Fall werden die Bridge-Identifikatoren noch einmal verwendet, diesmal zur Festlegung der bezeichneten Bridges. Der LAN-V-Anschluß von Bridge 4 wird über den LAN-V-Anschluß von Bridge 5 ausgewählt.

Mit dieser Methode werden alle Bridges bis auf eine direkt mit jedem LAN verbundene Bridge entfernt, wodurch alle Schleifen zwischen zwei LANs entfernt werden. Der STA entfernt außerdem Schleifen mit mehr als zwei LANs, erhält aber dabei die Connectivity. Bild 24.4 zeigt das Ergebnis der Anwendung des STA auf das Netzwerk aus Bild 24.3. Bild 24.4 erklärt die Baum-Topologie. Ein Vergleich dieser Abbildung mit der Abbildung vor der Erstellung des Baums zeigt, daß der STA die Anschlüsse von Bridge 3 und Bridge 5 an das LAN-V in den Standby-Modus versetzt hat.

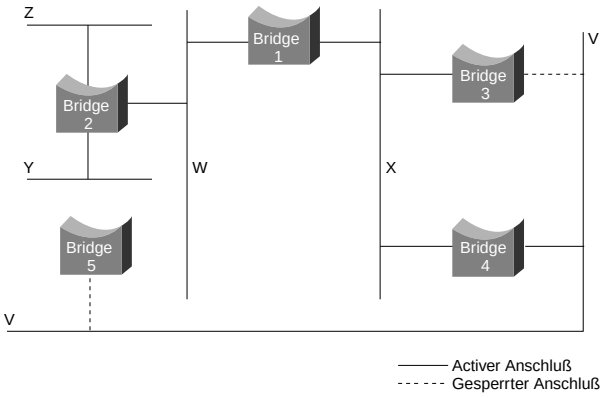


Bild 24.4: Erstellt eine Loop-freie Baumtopologie. Ein STA-basiertes Transparent-Bridge-Netzwerk

Diese Baumberechnung wird beim Einschalten der Bridge und bei Erkennung von Änderungen der Topologie ausgeführt. Die Berechnung erfordert eine Kommunikation zwischen den Spanning-Tree-Bridges, die mit Konfigurationsmeldungen realisiert wird (diese werden gelegentlich als *bridge protocol data units* oder *BPDUs* bezeichnet). Konfigurationsmeldungen ent-

halten Informationen, welche die Bridge identifizieren, die als Root angesehen wird (Root-Identifikator), und die Entfernung von der sendenden Bridge zur Root-Bridge (Root-Pfadkosten). Konfigurationsmeldungen enthalten außerdem die Bridge- und Anschluß-Identifikatoren der sendenden Bridge sowie das Alter der Informationen in der Konfigurationsmeldung.

Bridges tauschen in regelmäßigen Abständen Konfigurationsmeldungen aus (üblich sind zwischen ein und vier Sekunden). Wenn eine Bridge ausfällt (wodurch sich die Topologie ändert), stellen die benachbarten Bridges den Ausfall der Konfigurationsmeldungen fest und veranlassen eine Neuberechnung des Baums.

Alle Entscheidungen zur Topologie der Transparent-Bridges fallen lokal. Konfigurationsmeldungen werden zwischen benachbarten Bridges ausgetauscht. Innerhalb der Netzwerk-Topologie existiert keine zentrale Autorität oder Verwaltung.

24.3 Frames-Format

Transparente Bridges tauschen Konfigurationsmeldungen und Topologie-Änderungsmeldungen aus. Zum Einrichten einer Netzwerk-Topologie werden zwischen Bridges Konfigurationsmeldungen ausgetauscht. Topologie-Änderungs-Meldungen werden nach Änderungen der Topologie versendet, um einen Aufruf des STA zu veranlassen.

Bild 24.5 verdeutlicht das IEEE-802.1d-Konfigurationsmeldungsformat.

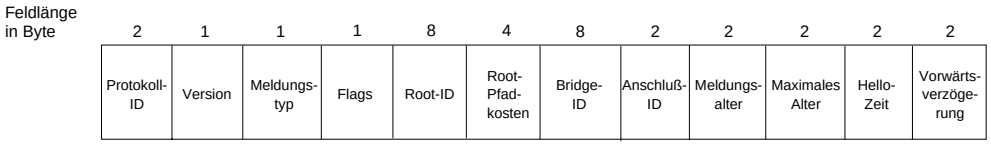


Bild 24.5: Die Transparent-Bridge-Konfigurationsmeldung setzt sich aus zwölf Feldern zusammen

Es folgen die Felder der Konfigurationsmeldung der Transparent-Bridge:

- *Protokoll-ID* – enthält den Wert Null.
- *Version* – enthält den Wert Null.
- *Meldungstyp* – enthält den Wert Null.
- *Flag* – enthält 1 Byte, von dem nur die ersten Bits verwendet werden. Das Topology-Change-(TC-)Bit weist auf eine Änderung der Topologie hin. Das Topology-Change-Acknowledgment-(TCA-)Bit wird als Anerkennung des Empfangs einer Konfigurationsmeldung mit gesetztem TC-Bit gesetzt.
- *Root-ID* – Identifiziert die Root-Bridge durch Angabe ihrer 2 Byte langen Priorität gefolgt von der 6-Byte langen ID.
- *Root-Pfadkosten* – Enthält die Pfadkosten der Bridge, welche die Konfigurationsmeldung an die Root-Bridge gesendet hat.
- *Bridge-ID* – Identifiziert die Priorität und ID der Bridge, von der die Meldung stammt.
- *Anschluß-ID* – Identifiziert den Anschluß, von dem die Konfigurationsmeldung stammt. Dieses Feld ermöglicht das Auffinden und Behandeln von Schleifen, die durch mehrere angeschlossene Bridges gebildet werden.
- *Meldungsalter* – Gibt die Zeit seit Absenden der Konfigurationsmeldung, auf der diese Konfigurationsmeldung basiert, durch die Root an.
- *Max. Alter* – Gibt an, wann die aktuelle Konfigurationsmeldung gelöscht werden sollte.
- *Hello-Zeit* – Gibt die Zeitspanne zwischen Root-Bridge-Konfigurationsmeldungen an.
- *Vorwärtsverzögerung* – Gibt die Zeitdauer an, die Bridges nach einer Änderung der Topologie warten sollen, bevor sie einen neuen Zustand herstellen. Wenn eine Bridge zu früh wechselt, sind möglicherweise nicht alle Netzwerk-Verbindungen bereit, ihren Zustand zu ändern, was zu Schleifen führen kann.

Die Topologie-Änderungsmeldungen enthalten nur 4 Byte. Dazu gehört ein *Protokoll-ID*-Feld, das den Wert Null enthält, ein *Version*-Feld, das ebenfalls den Wert Null enthält, und ein *Meldungstyp*-Feld, das den Wert 128 enthält.

Kapitel 25: AppleTalk
Kapitel 26: DECnet
Kapitel 27: IBM Systems Network Architecture (SNA)
Kapitel 28: Internet-Protokolle
Kapitel 29: NetWare-Protokolle
Kapitel 30: Protokolle der Open System Interconnection (OSI)
Kapitel 31: Banyan VINES
Kapitel 32: Xerox Network Systems (XNS)

TEIL 5

Netzwerk-Protokolle

Teil 5, »Netzwerk-Protokolle«, gibt einen Überblick über die grundlegenden Komponenten der heute am weitesten verbreiteten Netzwerk-Protokolle. In den einzelnen Kapiteln werden folgende Themen behandelt:

AppleTalk – Dieses Kapitel behandelt die AppleTalk-Protokolle.

DECnet – Dieses Kapitel behandelt die Protokolle DECnet Phase IV und DECnet/OSI (DECnet Phase V).

IBM Systems Network Architectures (SNA) – Dieses Kapitel faßt die Grundlagen des IBM-SNA-Protokolls hinsichtlich der Komponenten und der Funktionen einschließlich hierarchischer und Peer-basierter Umgebungen zusammen.

Internet Protocols – Dieses Kapitel beleuchtet die üblichen Internet-Protokolle, zu denen IP, TCP, UDP, ICMP und ARP gehören, und gibt einen Überblick zu bekannten Protokollen der Anwendungsschicht.

NetWare Protocols – Dieses Kapitel behandelt die NetWare-Protokolle, insbesondere von IPX und SAP.

Open Systems Interconnection (OSI) – Dieses Kapitel beschreibt die Funktionen und Eigenschaften der ISO-OSI-Protokolle.

Banyan VINES – In diesem Kapitel wird die Protokoll-Familie Banyan VINES beschrieben.

Xerox Network Systems (XNS) – In diesem Kapitel werden die XNS-Protokolle beschrieben.

KAPITEL 25

AppleTalk

25.1 Background

AppleTalk besteht aus einer Reihe von Übertragungsprotokollen, die in den frühen 80er Jahren von Apple Computer zusammen mit den Macintosh-Rechnern entwickelt wurden. Zweck von AppleTalk war es, vielen Benutzern die gemeinsame Verwendung von Ressourcen wie Dateien und Druckern zu ermöglichen. Geräte, die Ressourcen zur Verfügung stellen, nennt man *Server*, während man bei Geräten, die diese Ressourcen nutzen (wie etwa dem Macintosh-Rechner eines Benutzers) von *Clients* spricht. AppleTalk ist also eine der frühen Implementierungen einer verteilten Client-Server-Netzwerk-Architektur.

AppleTalk wurde mit einer transparenten Netzschnittstelle ausgestattet. Das heißt, der Austausch zwischen Client-Rechnern und Netzwerk-Servern erfordert wenig Aufwand von Seiten des Benutzers. Außerdem laufen die tatsächlichen Operationen der AppleTalk-Protokolle für den Benutzer unsichtbar ab. Er sieht also nur das Ergebnis dieser Vorgänge. Es gibt zwei Versionen von AppleTalk: AppleTalk Phase 1 und AppleTalk Phase 2.

AppleTalk Phase 1, die erste Spezifikation von AppleTalk, wurde in den frühen 80er Jahren ausschließlich für den Einsatz in lokalen Arbeitsgruppen entwickelt. Daher ist Phase 1 in zweierlei Hinsicht eingeschränkt. Ihre Netzwerk-Segmente können nicht mehr als 127 Hosts und 127 Server enthalten,

und sie unterstützt nur nicht erweiterte Netzwerke. Erweiterte und nicht erweiterte Netzwerke werden später in diesem Kapitel noch ausführlich beschrieben.

AppleTalk Phase 2, die zweite, verbesserte Implementierung von AppleTalk, wurde für den Gebrauch in größeren Verbundnetzen entwickelt. Phase 2 behebt die Einschränkungen von Phase 1 und enthält darüber hinaus einige Verbesserungen. Mit Phase 2 können in einem einzigen AppleTalk-Netzwerk-Segment 253 Hosts oder Server miteinander kombiniert werden. Außerdem werden sowohl nicht erweiterte Netzwerke als auch erweiterte Netzwerke unterstützt.

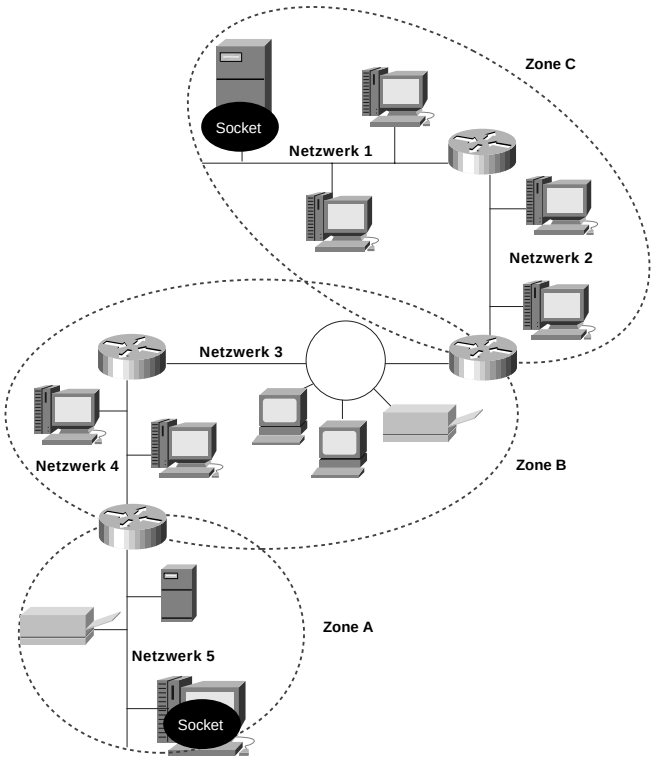


Bild 25.1: Das AppleTalk-Verbundnetz besteht aus hierarchisch angeordneten Komponenten

25.2 AppleTalk-Netzwerk-Komponenten

AppleTalk-Netzwerke sind hierarchisch aufgebaut. Die Basis eines AppleTalk-Netzwerks bilden vier Grundkomponenten: *Sockets, Knoten, Netzwerke* und *Zonen*. Bild 25.1 veran-

schaulicht den hierarchischen Aufbau dieser Komponenten in einem AppleTalk-Verbundnetz. In den folgenden Abschnitten werden sie im einzelnen erläutert.

25.2.1 Sockets

Ein AppleTalk-Socket ist eine einzelne, adressierbare Stelle innerhalb eines AppleTalk-Knotens. Er ist der logische Punkt, an dem die AppleTalk-Software der oberen Schicht abläuft und das Datagram Delivery Protocol (DDP) der Netzwerk-Schicht agiert. Diese Prozesse in der oberen Schicht nennt man Socket Clients. Socket Clients bestehen aus einem oder mehreren Sockets, die dazu verwendet werden, Datenpakete zu senden und zu empfangen. Sockets können statisch oder dynamisch zugewiesen werden. Statisch zugewiesene Sockets sind für den Gebrauch von Protokollen oder anderen Prozessen vorgesehen. Dynamisch zugewiesene Sockets werden den Socket Clients auf Anforderung von DDP zugewiesen. Ein AppleTalk-Knoten kann bis zu 254 verschiedene Socketnummern enthalten. Bild 25.2 zeigt das Verhältnis zwischen den Sockets in einem AppleTalk-Knoten und DDP in der Netzwerk-Schicht.

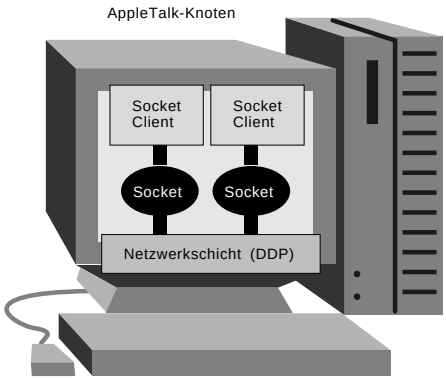


Bild 25.2: Socket Clients senden und empfangen mit Sockets Datenpakete

25.2.2 Knoten

Ein AppleTalk-Knoten ist ein Gerät, das mit dem AppleTalk-Netzwerk verbunden ist. Dieses Gerät kann ein Macintosh-Rechner, ein Drucker, ein IBM-PC, ein Router oder ein ähnliches Gerät sein. In jedem AppleTalk-Knoten laufen zahlreiche Software-Prozesse ab, die man Sockets nennt. Wie vorher

schon dargelegt, besteht die Funktion dieser Sockets darin, die Software-Prozesse zu identifizieren, die in dem Gerät ablaufen. Jeder Knoten in einem AppleTalk-Netzwerk gehört zu einem einzigen Netzwerk und einer bestimmten Zone.

25.2.3 Netzwerke

Ein AppleTalk-Netzwerk besteht aus einem einzelnen, logischen Kabel und mehreren angeschlossenen Knoten. Das logische Kabel besteht entweder aus einem einzelnen oder mehreren physischen Kabeln, die über Bridges und Router verbunden werden. AppleTalk-Netzwerke können erweitert oder nicht erweitert sein. Beides wird in den folgenden Abschnitten kurz beschrieben.

Nicht erweiterte Netzwerke

Ein nicht erweitertes AppleTalk-Netzwerk ist ein physisches Netzwerk-Segment, dem nur eine einzige Netzwerk-Nummer zugewiesen wird, die zwischen 1 und 1024 liegen kann. Netzwerk 100 und Netzwerk 562 zum Beispiel sind beides gültige Netzwerk-Nummern in einem nicht erweiterten Netzwerk. Jede Knotennummer in einem lokalen Netzwerk muß einmalig sein, und auf einem einzelnen nicht erweiterten Netzwerk-Segment kann nicht mehr als eine AppleTalk-Zone konfiguriert sein. (Eine Zone ist eine logische Gruppe von Knoten und Netzwerken.) AppleTalk Phase 1 unterstützt nur nicht erweiterte Netzwerke, aber in der Regel werden nicht erweiterte Netzwerk-Konfigurationen in modernen Netzwerken nicht mehr eingesetzt, weil sie durch erweiterte Netze ersetzt wurden. Bild 25.3 zeigt ein lokales AppleTalk-Netzwerk.

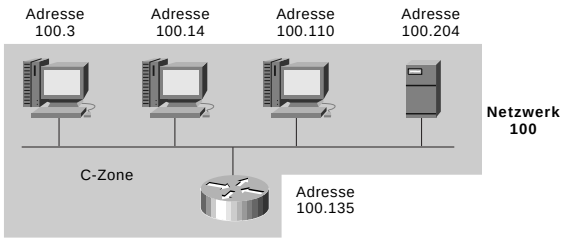


Bild 25.3: Einem lokalen Netzwerk wird nur eine Netzwerk-Nummer zugewiesen

Erweiterte Netzwerke

Ein erweitertes AppleTalk-Netzwerk ist ein physisches Netzwerk-Segment, dem mehrere Netzwerk-Nummern zugewiesen werden können. Diese Konfiguration wird Kabelbereich genannt. AppleTalk-Kabelbereiche können eine einzige Netzwerk-Nummer anzeigen oder mehrere aufeinanderfolgende. Die Kabelbereiche Netzwerk 3-3 (unitär) und Netzwerk 3-6 sind zum Beispiel beide in einem erweiterten Netzwerk gültig. Genau wie bei anderen Protokollarten, wie etwa TCP/IP und IPX, muß jede Kombination aus Netzwerk-Nummer und Knotennummer in einem erweiterten Netzwerk einmalig sein, sowie auch ihre Adresse für die Identifizierung einmalig sein muß. Bei erweiterten Netzwerken können mehrere AppleTalk-Zonen auf einem einzigen Netzwerk-Segment konfiguriert sein, und die Knoten eines erweiterten Netzwerks können zu jeder Zone gehören, die an dieses Netz angeschlossen ist. Konfigurationen erweiterter Netze haben Konfigurationen nicht erweiterter in der Regel ersetzt. Bild 25.4 zeigt ein erweitertes Netz.

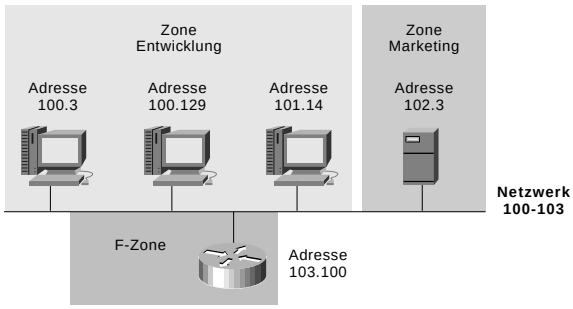


Bild 25.4: Einem erweiterten Netz können mehrere Netzwerk-Nummern zugewiesen werden

25.2.4 Zonen

Eine AppleTalk-Zone ist eine logische Gruppe von Knoten oder Netzwerken, die erstellt wird, wenn der Administrator das Netzwerk konfiguriert. Die Knoten und Netzwerke müssen physisch nicht zusammenliegen, um zur selben AppleTalk-Zone zu gehören. Bild 25.5 zeigt ein AppleTalk-Verbundnetz, das aus drei nicht benachbarten Zonen besteht.

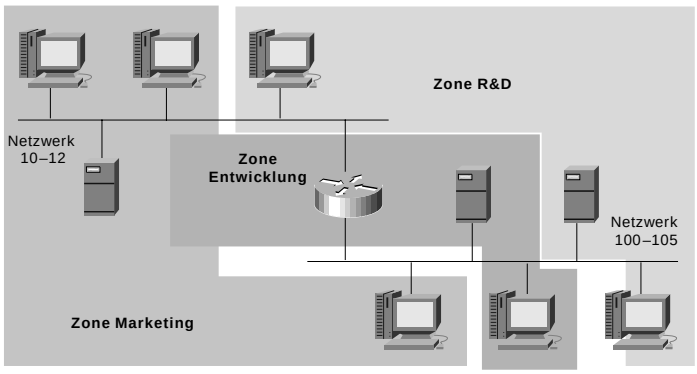


Bild 25.5: Knoten oder Netzwerke derselben Zone müssen physisch nicht zusammenliegen

25.3 Bitübertragungs- und Sicherungsschichten von AppleTalk

Wie auch bei anderen gebräuchlichen Protokollarten, wie etwa TCP/IP und IPX, wird auch in der AppleTalk-Architektur der Medienzugriff durch Protokolle aus den unteren Schichten, wie Ehernet, Token Ring und FDDI, bestimmt. Es gibt in der AppleTalk-Protokollsuite vier Hauptimplementierungen für den Medienzugriff: EtherTalk, LocalTalk, TokenTalk und FDDITalk.

Diese Implementierungen der Sicherungsschicht übernehmen Übersetzungen von Adressen und weitere Funktionen, die es den proprietären AppleTalk-Protokollen erlauben, über standardmäßige Schnittstellen, wie IEEE 802.3 (mit Hilfe von EtherTalk), Token Ring/IEEE 802.5 (mit TokenTalk) und FDDI (mit FDDITalk), zu kommunizieren. Darüber hinaus implementiert AppleTalk seine eigene Netzwerk-Schnittstelle, LocalTalk. Bild 25.6 veranschaulicht, wie die Implementierungen für den Medienzugriff von AppleTalk mit dem OSI-Schichtenmodell zusammenhängen.

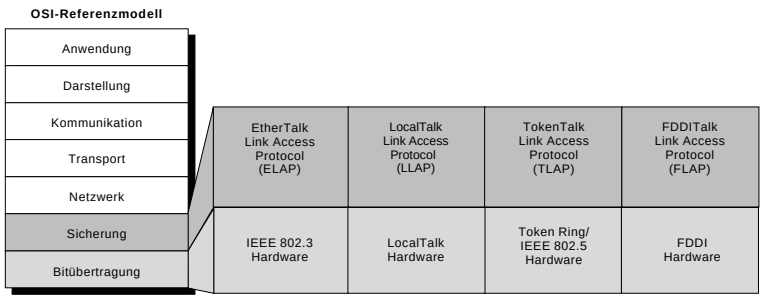


Bild 25.6: Der Medienzugriff von Apple Talk liegt in den unteren beiden Schichten des OSI-Schichtenmodells

25.3.1 EtherTalk

EtherTalk erweitert die Sicherungsschicht, damit die AppleTalk-Protokollsuite mit dem Standard IEEE 802.3 arbeiten kann. EtherTalk-Netzwerke sind genau wie IEEE-802.3-Netzwerke aufgebaut, unterstützen dieselben Geschwindigkeiten und Segmentlängen und auch dieselbe Anzahl von aktiven Netzwerk-Knoten. So kann AppleTalk in Tausenden von Ethernet-basierten Netzwerken implementiert werden, die heute im Einsatz sind. Die Kommunikation zwischen den AppleTalk-Protokollen aus den oberen Schichten und den Ethernet-Protokollen wird vom EtherTalk Link-Access Protocol (ELAP) geregelt.

EtherTalk Link-Access Protocol

Das *EtherTalk Link-Access Protocol* (ELAP) regelt das Zusammenspiel zwischen den proprietären AppleTalk-Protokollen und der standardmäßigen IEEE-802.3-Sicherungsschicht. Die AppleTalk-Protokolle der oberen Schichten erkennen die standardmäßigen IEEE-802.3-Hardware-Adressen nicht, weshalb ELAP die *Address-Mapping Table (AMT)*, die vom *AppleTalk Address-Resolution Protocol (AARP)* erstellt wird, verwendet, um Übertragungen korrekt zu adressieren.

ELAP regelt das Zusammenspiel der AppleTalk-Protokolle der oberen Schichten und der Sicherungsschicht, indem es die Daten in die Protokolleinheiten der 802.3-Sicherungsschicht einkapselt. ELAP führt drei Stufen Kapselung durch, wenn es DDP-Pakete übermittelt:

- Subnetwork-Access Protocol (SNAP) Header
- IEEE 802.2 Logical Link-Control (LLC) Header
- IEEE 802.3 Header

Dieser Prozeß der Kapselung, den ELAP durchführt, wird in den folgenden Abschnitten genauer beschrieben.

Datenübermittlung mit ELAP

ELAP verwendet einen vorgegebenen Prozeß, um Daten über ein physisches Medium zu übermitteln. Zunächst empfängt ELAP ein DDP-Paket, das eine Übermittlung anfordert. Dann findet es die Protokolladresse, die in dem DDP-Header spezi-

fiziert ist und überprüft die AMT, um die entsprechende IEEE-802.3-Hardware-Adresse zu finden. ELAP setzt daraufhin drei verschiedene Header vor das DDP-Paket, beginnend mit den SNAP- und 802.2-LLC-Headern. Der dritte Header ist der IEEE-802.3-Header. Wenn der Header vor das Paket gesetzt wird, wird die Hardware-Adresse aus der AMT in das Zieladreßfeld plaziert. Das Endergebnis, ein IEEE-802.3-Frame, wird an das physische Medium übergeben, damit er an sein Ziel übertragen wird.

25.3.2 LocalTalk

LocalTalk, eine proprietäre Implementierung der Sicherungsschicht, die von Apple Computer für die AppleTalk-Protokollsuite entwickelt wurde, war als kostengünstige Netzwerk-Lösung konzipiert, mit der lokale Arbeitsgruppen verbunden werden könnten. In Apple-Produkte ist LocalTalk-Hardware eingebaut, die leicht über die preiswerten Twisted-Pair-Kabel vernetzt werden kann. LocalTalk-Netzwerke sind in einer Bustypologie aufgebaut, das heißt, sie sind miteinander in Serie verbunden. Netzwerk-Segmente sind auf eine Reichweite von 300 Metern mit maximal 32 aktiven Knoten beschränkt, und mehrere LocalTalk-Netzwerke können durch Router oder andere zwischengeschaltete Geräte verbunden werden. Für die Kommunikation zwischen dem Sicherungsschicht-Protokoll LocalTalk und den Protokollen der oberen Schichten sorgt das LocalTalk Link-Access Protocol (LLAP).

LocalTalk Link-Access Protocol

Das *LocalTalk Link-Access Protocol* (LLAP) ist ein Protokoll für den Medienzugriff, das in LocalTalk-Netzwerken verwendet wird, um bestmögliche und fehlerfreie Auslieferung von Frames zwischen AppleTalk-Knoten zu gewährleisten. Das bedeutet, daß die Auslieferung von Datagrammen nicht von der LLAP garantiert wird. Diese Funktion wird nur von Protokollen der höheren Schichten der AppleTalk-Architektur ausgeübt. LLAP regelt den Knotenzugriff zum physischen Medium und erfaßt die Knotenadressen der Sicherungsschicht dynamisch.

Regulierung des Knotenzugriffs zum physischen Medium

LLAP realisiert den Medienzugriff mit einer Methode, die *carrier-sense multiple access, collision avoidance* (CSMA/CA) genannt wird, bei der Knoten die Verbindung daraufhin prüfen, ob sie gerade benutzt wird. Die Verbindung muß für einen gewissen Zeitraum frei sein, bevor der Knoten mit der Datenübertragung beginnen kann. LLAP nutzt einen Datenaustausch namens *Handshake*, um Kollisionen zu vermeiden (also gleichzeitige Übertragung von zwei oder mehr Knoten). Ein erfolgreicher Handshake zwischen Knoten reserviert die Verbindung effektiv zu ihrem Gebrauch. Wenn zwei Knoten einen Handshake gleichzeitig übermitteln, kollidieren die Übertragungen. In diesem Fall werden beide Übertragungen beschädigt, so daß die Pakete abgelegt werden. Der Handshake-Austausch ist nicht vollendet, und die sendenden Knoten folgern, daß es eine Kollision gab. Wenn es zur Kollision kommt, bleibt das Gerät für einen willkürlichen Zeitraum außer Betrieb und versucht dann die Übertragung erneut. Dieser Prozeß ähnelt dem Zugriffsmechanismus, der bei der Ethernet-Technologie verwendet wird.

Erfassung von Knotenadressen

LLAP erfaßt die Knotenadressen der Sicherungsschicht dynamisch. Dieser Prozeß erlaubt es, daß der Sicherungsschicht eine einzelne Adresse zugewiesen wird, ohne daß diese Adresse dem Knoten permanent zugewiesen wird. Wenn ein Knoten hochgefahren wird, weist LLAP ihm ein zufällig ausgewähltes Identifizierungszeichen (Knoten-ID) zu. Die Einmaligkeit dieser Knoten-ID wird von der Übermittlung eines speziellen Pakets bestimmt, das an die zufällig gewählte Knoten-ID adressiert ist. Wenn der Knoten eine Antwort auf dieses Paket erhält, ist die Knoten-ID nicht einmalig. Also wird dem Knoten eine andere zufällig gewählte Knoten-ID zugewiesen, und er sendet ein weiteres Paket, das an diesen Knoten adressiert ist, bis keine Antwort mehr kommt. Erhält der erfaßte Knoten bei der ersten Abfrage keine Antwort, führt er einige weitere Versuche durch. Gibt es nach diesen Versuchen immer noch keine Antwort, wird die Knoten-ID als einmalig angesehen, und der Knoten verwendet diese Knoten-ID als seine Sicherungsschichtadresse.

25.3.3 TokenTalk

TokenTalk erweitert die Sicherungsschicht dahingehend, daß es der AppleTalk-Protokollsuite erlaubt wird, auf einer standardmäßigen IEEE-802.5/Token-Ring-Implementierung zu arbeiten. TokenTalk-Netzwerke sind genauso aufgebaut wie IEEE-802.5/Token-Ring-Netzwerke und unterstützen die gleichen Geschwindigkeiten und die gleiche Anzahl von aktiven Netzwerk-Knoten. Für die Kommunikation zwischen den Protokollen der Sicherungsschicht, die mit Token Ring verwendet werden, und den Protokollen der oberen Schichten sorgt das TokenTalk Link-Access Protocol (TLAP).

TokenTalk Link-Access Protocol

Das *TokenTalk Link-Access Protocol* (TLAP) regelt die Zusammenarbeit zwischen proprietären AppleTalk-Protokollen und der standardmäßigen IEEE-802.5-Sicherungsschicht. Die AppleTalk-Protokolle der oberen Schichten erkennen die standardmäßigen IEEE-802.5-Hardware-Adressen nicht, weswegen TLAP die AMT, die von der AARP gepflegt wird, verwendet, um Übermittlungen korrekt zu adressieren. TLAP führt drei Stufen der Kapselung durch, wenn es DDP-Pakete übermittelt.

- Subnetwork-Access Protocol (SNAP) Header
- IEEE 802.2 Logical Link-Control (LLC) Header
- IEEE 802.5 Header

TLAP-Datenübertragungsprozeß

Die TLAP-Datenübertragung verwendet mehrere Schritte, um Daten über ein physisches Medium zu senden. Wenn TLAP ein DDP-Paket empfängt, das eine Übermittlung anfordert, stellt es die Protokolladresse fest, die im DDP-Header spezifiziert ist, um dann in der AMT nach der entsprechenden IEEE-802.5/Token-Ring-Hardware-Adresse zu suchen. Dann setzt TLAP drei verschiedene Header vor das DDP-Paket, wobei es mit den SNAP- und 802.2-LLC-Headern beginnt. Wenn der dritte Header, IEEE 802.5/Token Ring, vor das Paket gesetzt wird, wird die Hardware-Adresse aus der AMT in das Zieladreßfeld eingefügt. Das Endergebnis, ein IEEE-802.5/Token-

Ring-Frame, wird dem physischen Medium zur Übertragung an das Ziel übergeben.

25.3.4 FDDITalk

FDDITalk erweitert die Sicherungsschicht dahingehend, daß die AppleTalk-Protokollsuite auf einer standardmäßigen ANSI-FDDI-Implementierung arbeiten kann. FDDITalk-Netzwerke sind genau wie FDDI-Netzwerke aufgebaut und unterstützen die gleichen Geschwindigkeiten und die gleiche Anzahl von aktiven Netzwerk-Knoten.

FDDITalk Link-Access Protocol

Das *FDDITalk Link-Access Protocol* (FLAP) regelt die Zusammenarbeit zwischen den proprietären AppleTalk-Protokollen und der standardmäßigen FDDI-Sicherungsschicht. Die AppleTalk-Protokolle der oberen Schichten erkennen die standardmäßigen FDDI-Hardware-Adressen nicht, weswegen FLAP die AMT, die von der AARP gepflegt wird, verwendet, um Übermittlungen korrekt zu adressieren. FLAP führt drei Stufen der Kapselung durch, wenn es DDP-Pakete übermittelt.

- Subnetwork-Access Protocol (SNAP) Header
- IEEE 802.2 Logical Link-Control (LLC) Header
- FDDI Header

FLAP-Datenübertragungsprozeß

Wie TLAP führt auch FLAP einen mehrstufigen Prozeß durch, um Daten über ein physisches Medium zu übermitteln. Wenn FLAP ein DDP-Paket empfängt, das eine Übertragung anfordert, stellt es die Protokolladresse fest, die in dem DDP-Header spezifiziert ist, und sucht dann in der AMT nach der entsprechenden FDDI-Hardware-Adresse. Dann setzt FLAP drei verschiedene Header vor das DDP-Paket, wobei es mit den SNAP- und 802.2-LLC-Headern beginnt. Wenn der dritte Header, der FDDI-Header, vor das Paket gesetzt wird, wird die Hardware-Adresse aus der AMT in das Zieladreßfeld eingefügt. Das Endergebnis, ein FDDI-Frame, wird dem physischen Medium zur Übermittlung an das Ziel übergeben.

25.4 Netzwerk-Adressen

AppleTalk verwendet Adressen, um Geräte in einem Netzwerk zu identifizieren und zu finden, wie es ähnlich auch in jenen Prozessen geschieht, die in so gängigen Protokollen wie TCP/IP und IPX ablaufen. Diese Adressen, die, wie im folgenden Abschnitt beschrieben, dynamisch zugewiesen werden, setzen sich aus drei Elementen zusammen.

– *Netzwerk-Nummer* – Ein 16-Bit-Wert, der ein spezifiziertes AppleTalk-Netzwerk identifiziert (erweitert oder nicht erweitert).

– *Knotennummer* – Ein 8-Bit-Wert, der einen bestimmten AppleTalk-Knoten identifiziert, der an dem spezifizierten Netzwerk hängt.

– *Socketnummer* – Eine 8-Bit-Zahl, die einen spezifischen Socket identifiziert, der auf einem Netzwerk-Knoten läuft.

AppleTalk-Adressen werden gewöhnlich als Dezimalwerte geschrieben, die durch Punkte getrennt sind. Zum Beispiel bedeutet 10.1.50 Netzwerk 10, Knoten 1, Socket 50. Dies kann auch mit 10.1, Socket 50 ausgedrückt werden. Bild 25.7 zeigt das Adreßformat von AppleTalk-Netzwerken.

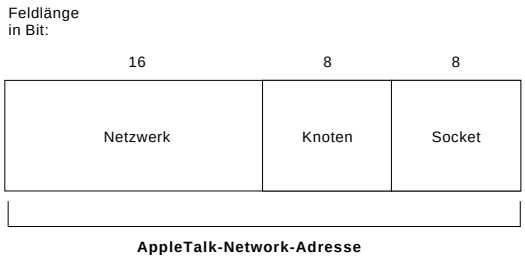


Bild 25.7: Die AppleTalk-Netzwerk-Adresse besteht aus drei Zahlen

25.4.1 Zuweisung von Netzwerk-Adressen

Eine der einzigartigen Eigenschaften von AppleTalk ist die dynamische Natur der Geräteadressen. Es ist nicht nötig, eine statische Adresse für ein AppleTalk-Gerät zu definieren. Die Adressen werden AppleTalk-Knoten dynamisch zugewiesen, wenn sie erstmalig an ein Netzwerk angeschlossen werden.

Wenn der Knoten eines AppleTalk-Netzwerks hochgefahren wird, erhält er eine provisorische Netzwerk-Schichtadresse. Der Netzwerk-Teil der provisorischen Adresse (die ersten 16 Bit) wird dem *Bootbereich* entnommen, der aus reservierten Netzwerk-Adressen besteht (Werte von 65280 bis 65534). Der Knotenteil (die nächsten 8 Bit) der provisorischen Adresse wird zufällig ausgewählt.

Über das *Zone-Information Protocol (ZIP)* kommuniziert der Knoten mit einem Router, der an das Netzwerk angeschlossen ist. Dieser Router stellt dem Netzwerk, mit dem der Knoten verbunden ist, den gültigen Kabelbereich zur Verfügung. Als nächstes wählt der Knoten eine gültige Netzwerk-Nummer aus dem Kabelbereich, den der Router bereitstellt, und eine zufällige Knotennummer. Durch einen Rundspruch wird festgestellt, ob die ausgewählte Adresse von einem anderen Knoten verwendet wird.

Wird die Adresse nicht verwendet (wenn also kein anderer Knoten innerhalb eines vorgegebenen Zeitraums auf den Rundspruch reagiert), ist dem Knoten erfolgreich eine Adresse zugewiesen worden. Benutzt ein anderer Knoten die Adresse, beantwortet er die Anfrage mit der Meldung, daß die Adresse in Gebrauch sei. Der neue Knoten muß eine andere Adresse wählen und den Vorgang wiederholen, bis er eine Adresse wählt, die nicht in Gebrauch ist.

25.5 AppleTalk Address-Resolution Protocol (AARP)

Das AppleTalk Address-Resolution Protocol (AARP) ist ein Netzwerkschicht-Protokoll der AppleTalk-Protokollsuite, das AppleTalk-Netzwerk-Adressen mit Hardware-Adressen verknüpft. AARP-Dienste werden auch von anderen AppleTalk-Protokollen verwendet. Wenn ein AppleTalk-Protokoll zum Beispiel Daten zu übermitteln hat, spezifiziert es eine Netzwerk-Adresse für das Ziel. Es ist Aufgabe von AARP, die Hardware-Adresse zu finden, die mit dem Gerät verbunden ist, das diese Netzwerk-Adresse benutzt.

AARP verwendet einen Frage-Antwort-Prozeß, um die Hardware-Adressen von anderen Netzwerk-Knoten zu erfahren. Da

AARP ein medienabhängiges Protokoll ist, variiert die Methode, wie Hardware-Adressen abgefragt werden, je nachdem, welche Implementierung der Sicherungsschicht verwendet wird. Normalerweise wird ein Rundspruch an alle AppleTalk-Knoten im Netzwerk gesendet.

25.5.1 Address-Mapping Table

Jeder AppleTalk-Knoten enthält eine Address-Mapping Table (AMT/Adreßtabelle), in der Hardware-Adressen mit entsprechenden Netzwerk-Adressen aufgelistet sind. Jedesmal, wenn das AARP eine Kombination von Netzwerk- und Hardware-Adresse zuweist, wird das in der AMT festgehalten.

Mit der Zeit steigt die Wahrscheinlichkeit, daß ein Eintrag in der AMT ungültig geworden ist. Deswegen wird jeder Eintrag in der AMT mit einem Timer verknüpft. Wenn AARP ein Paket empfängt, das den Eintrag vergleicht oder ändert, wird der Timer zurückgesetzt.

Läuft der Timer ab, wird der Eintrag aus der AMT gelöscht. Das nächste Mal, wenn ein AppleTalk-Protokoll mit diesem Knoten kommunizieren will, muß eine andere AARP-Abfrage übermittelt werden, um die Hardware-Adresse festzustellen.

25.5.2 Address Gleaning

In bestimmten Implementierungen werden eingehende DDP-Pakete auf die Hardware- und Netzwerk-Adressen des Ausgangsknotens überprüft. DDP kann diese Information dann in die AMT eintragen. Dies ist eine Methode, mit der Geräte, wie Router, Workstations oder Server, Hardware in einem AppleTalk-Netzwerk finden können.

Dieses Verfahren, Adressen aus eingehenden Paketen zu erhalten, nennt man *Address Gleaning*. Address Gleaning wird selten verwendet, aber in manchen Situationen kann es die Anzahl von AARP-Abfragen, die übermittelt werden müssen, reduzieren.

25.5.3 AARP-Operation

Das AppleTalk-Address-Resolution Protocol (AARP) ordnet Hardware-Adressen Netzwerk-Adressen zu. Muß ein AppleTalk-Protokoll Daten senden, übergibt es die Netzwerk-Adresse des Empfangsknotens an das AARP. Es ist Aufgabe des AARP, die Hardware-Adresse zu liefern, die mit dieser Netzwerk-Adresse verknüpft ist.

Das AARP überprüft die AMT darauf, ob die Netzwerk-Adresse bereits mit einer Hardware-Adresse verknüpft ist. Sind die Adressen bereits verknüpft, wird die Hardware-Adresse dem anfragenden AppleTalk-Protokoll übergeben, das sie dazu benutzt, mit dem Empfänger zu kommunizieren. Sind die Adressen nicht verknüpft, übermittelt AARP einen Rundspruch, damit der Knoten, der die fragliche Netzwerk-Adresse benutzt, seine Hardware-Adresse liefert.

Wenn die Anfrage den Knoten, der die Netzwerk-Adresse benutzt, erreicht, antwortet er mit seiner Hardware-Adresse. Wenn es keinen Knoten mit dieser spezifischen Netzwerk-Adresse gibt, wird keine Antwort gesendet. Nach einer festgelegten Anzahl von weiteren Versuchen, nimmt das AARP an, daß die Protokolladresse nicht in Gebrauch ist, und sendet eine Fehlermeldung an das anfragende AppleTalk-Protokoll. Trifft eine Antwort ein, wird die Hardware-Adresse in der AMT mit der Netzwerk-Adresse verknüpft. Die Hardware-Adresse wird dann dem anfragenden AppleTalk-Protokoll übergeben, das sie benutzt, um mit dem Empfangsknoten zu kommunizieren.

25.6 Datagram-Delivery Protocol (DDP) im Überblick

Das Datagram-Delivery Protocol (DDP) ist das primäre Netzwerkschicht-Routingprotokoll der AppleTalk-Protokollsuite, das einen leistungsstarken, verbindungsunabhängigen Datagramm-Dienst zwischen AppleTalk-Sockets bietet. Wie auch bei Protokollen wie TCP werden keine virtuellen Schaltwege oder Verbindungen zwischen zwei Geräten aufgebaut. Die Funktion der Ablieferung wird dagegen von Protokollen der oberen Schichten der AppleTalk-Protokollsuite gewährleistet.

Diese Protokolle der oberen Schichten werden später in diesem Kapitel beschrieben.

DDP hat hauptsächlich zwei Funktionen: Übermittlung und Empfang von Paketen.

- *Übermittlung von Paketen* – DDP erhält Daten von Socket Clients, erstellt einen DDP-Header, indem es die passenden Empfangsadressen verwendet, und leitet das Paket an das Protokoll der Sicherungsschicht.
- *Empfang von Paketen* – DDP erhält Frames von der Sicherungsschicht, überprüft den DDP-Header auf die Zieladresse, und leitet das Paket an den Zielsocket.

DDP pflegt den Kabelbereich des lokalen Netzwerks und die Netzwerk-Adresse eines Routers, der mit dem lokalen Netzwerk über jeden AppleTalk-Knoten verbunden ist. Außer diesen Informationen muß ein AppleTalk-Router mit Hilfe des Routing-Table Maintenance Protocol (RTMP) eine Routingtabelle führen.

25.6.1 DDP-Übertragungsverfahren

DDP arbeitet ähnlich wie andere Routingprotokolle. Pakete werden an der Quelle adressiert, an die Sicherungsschicht geleitet und ans Ziel übermittelt. Wenn DDP Daten von einem Protokoll der oberen Schichten erhält, entscheidet es, ob Quell- und Zielknoten im selben Netzwerk liegen, indem es die Netzwerk-Nummer der Empfangsadresse überprüft. Liegt die Netzwerk-Nummer des Ziels innerhalb des Kabelbereichs des lokalen Netzwerks, wird das Paket in einen DDP-Header verkapselt und zur Übermittlung an den Empfangsknoten an die Sicherungsschicht übergeben. Liegt die Netzwerk-Nummer des Ziels nicht innerhalb des Kabelbereichs des lokalen Netzwerks, wird das Paket in einen DDP-Header verkapselt und an die Sicherungsschicht übergeben, damit sie es an einen Router übermittelt. Zwischengeschaltete Router leiten die Pakete mit Hilfe ihrer Routingtabellen an das Zielnetzwerk weiter. Wenn das Paket einen Router erreicht, der mit dem Zielnetzwerk verbunden ist, wird es an den Zielknoten übermittelt.

25.7 AppleTalk-Transportschicht

Die AppleTalk-Transportschicht implementiert zuverlässige Datentransferdienste im Verbundnetz, die für die oberen Schichten transparent sind. Zu den Funktionen der Transportschicht gehören typischerweise die Ablaufsteuerung, das Multiplexing, die Verwaltung von virtuellen Schaltwegen, die Fehlerkontrolle und -beseitigung.

Es gibt fünf Hauptimplementierungen in der Transportschicht der AppleTalk-Protokollsuite:

- Routing-Table Maintenance Protocol (RTMP)
- Name-Binding Protocol (NBP)
- AppleTalk Update-Based Routing Protocol (AURP)
- AppleTalk Transaction Protocol (ATP)
- AppleTalk Echo Protocol (AEP)

Jede dieser Protokollimplementierungen wird im folgenden kurz vorgestellt.

25.7.1 Routing-Table Maintenance Protocol (RTMP) im Überblick

Das *Routing-Table Maintenance Protocol (RTMP)* ist ein Transportschicht-Protokoll der AppleTalk-Protokollreihe, das die Routingtabellen der AppleTalk-Router erstellt und pflegt.

RTMP basiert auf dem Routing-Information Protocol (RIP), und wie RIP benutzt RTMP den Hop-Zähler als Routinghilfe. Mit dem Hop-Zähler wird die Anzahl der Router oder sonstiger zwischengeschalteter Knoten festgestellt, die ein Paket durchlaufen muß, um vom Quellnetzwerk zum Zielnetzwerk zu gelangen.

RTMP-Routingtabellen

RTMP erstellt und pflegt Routingtabellen der AppleTalk-Router. Diese Routingtabellen enthalten für jedes Netzwerk, das ein Paket erreichen kann, einen Eintrag.

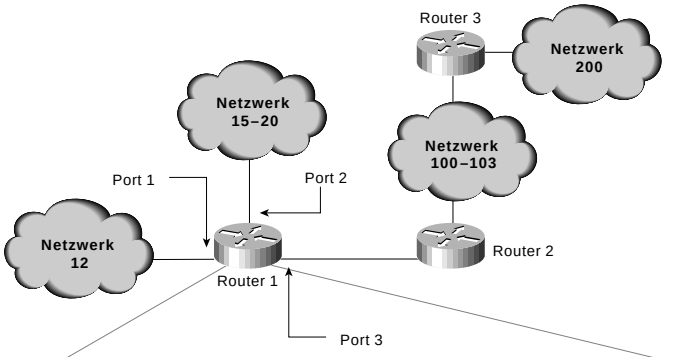
Router tauschen regelmäßig Routinginformationen aus, um sicherzustellen, daß die Routingtabelle eines jeden Routers, die

wichtigsten Informationen enthält, und daß die Informationen im Verbundnetz übereinstimmen. Eine RTMP-Routingtabelle enthält folgende Informationen von den Zielnetzwerken, die der Router kennt:

- Netzwerk-Kabelbereich des Zielnetzwerks
- Entfernung zum Zielnetzwerk in Hops
- Routerport, der zum Zielnetzwerk führt
- Adresse des nächsten Hop-Routers
- Aktueller Eintrag in der Routingtabelle (gut, zweifelhaft oder schlecht)

Bild 25.8 zeigt eine typische RTMP-Routingtabelle.

Bild 25.8: Eine RTMP-Routingtabelle enthält Informationen über jedes Zielnetzwerk, das der Router kennt



RTMP-Routingtabelle

Netzwerk-Kabelbereich	Entfernung	Port	nächster Hop	Status des Eintrags
12	0	1	0	gut
15–20	0	2	0	gut
100–103	1	3	Router 2	gut
200	2	3	Router 2	gut

25.7.2 Name-Binding Protocol (NBP) im Überblick

Das *Name-Binding Protocol (NBP)* ist ein Transportschicht-Protokoll der AppleTalk-Protokollsuite, das die Adressen, die in unteren Schichten verwendet werden, mit AppleTalk-Namen verknüpft. Socket Clients innerhalb der AppleTalk-

Knoten heißen *Network-Visible Entities (NVEs)*. Eine NVE ist eine adressierbare Ressource des Netzwerks, etwa ein Drukkerdienst, die über das Verbundnetz zugänglich ist. NVEs werden Zeichenfolgen zugewiesen, die Entitätsnamen genannt werden. NVEs haben auch eine Zone und verschiedene Attribute, die Entitätstypen, die mit ihnen verknüpft sind.

Es gibt zwei Hauptgründe dafür, Entitätsnamen statt Adressen in den oberen Schichten zu verwenden. Erstens werden die Netzwerk-Adressen den Knoten dynamisch zugewiesen, ändern sich also regelmäßig. Entitätsnamen erlauben dem Benutzer, Netzwerk-Ressourcen und -Dienste, etwa Datenserver, immer auf die gleiche Weise anzusprechen. Zweitens sind Vorgänge in den unteren Schichten für den Benutzer leichter verständlich, wenn Namen statt Adressen verwendet werden, um Ressourcen und Dienste anzusprechen.

Name Binding

Als *Name Binding* bezeichnet man den Vorgang, wenn NVE-Entitätsnamen mit Netzwerk-Adressen verknüpft werden. Jeder AppleTalk-Knoten verknüpft die Namen seiner NVEs mit ihren Netzwerk-Adressen in einer Namentabelle. Die Gesamtheit aller Namentabellen in den Verbundnetz-Knoten nennt man *Namensverzeichnis*, eine Datenbank aller Verknüpfungen zwischen Namen und Adressen. Name Binding wird durchgeführt, wenn ein Knoten erstmals hochgefahren wird, oder kurz bevor auf die genannte Entität zugegriffen wird (dynamisch).

NBP übt die folgenden vier Funktionen aus: Namensuche, Namenerkennung, Namenbestätigung und Namenlöschung. Die *Namensuche* wird verwendet, um die Netzwerk-Adresse einer NVE zu erfahren, bevor auf die Dienste dieser NVE zugegriffen wird. NBP überprüft das Namenverzeichnis auf die Verknüpfung zwischen Name und Adresse. Die *Namenregistrierung* erlaubt es dem Knoten, seine Namentabelle zu erstellen. NBP bestätigt, daß der Name nicht verwendet wird und fügt dann die Verknüpfung von Namen und Adresse in die Tabelle ein. *Namenbestätigung* wird verwendet, um abzugleichen, ob eine Verknüpfung, die durch die Namensuche zustande kam, noch zutrifft. *Namenlöschung* wird verwendet, um einen Eintrag aus der Namentabelle zu streichen, zum Beispiel, wenn der Knoten abgeschaltet wird.

25.7.3 AppleTalk Update-Based Routing Protocol (AURP)

Das *AppleTalk Update-Based Routing Protocol* (AURP) ist ein Transportschichtprotokoll der AppleTalk-Protokollsuite, mit dem zwei oder mehr AppleTalk-Verbundnetze über ein *Transmission-Control Protocol/Internet Protocol (TCP/IP)-N*etzwerk verbunden werden können, damit ein AppleTalk-WAN entsteht. AURP verkapselt Pakete in einem User-Datagram-Protocol-(UDP-)Header und ermöglicht es, daß sie transparent über ein TCP/IP-Netzwerk gesendet werden. Eine AURP-Implementierung besteht aus zwei Komponenten: externe Router und AURP-Tunnel.

Externe Router verbinden ein lokales AppleTalk-Verbundnetz mit einem AURP-Tunnel. Sie wandeln AppleTalk-Daten und Routinginformationen in AURP um und führen die Kapselung und Entkapselung des AppleTalk-Verkehrs durch. Ein externer Router fungiert als AppleTalk-Router im lokalen Netzwerk und als Endknoten im TCP/IP-Netzwerk. Wenn externe Router sich erstmalig mit einem AURP-Tunnel verbinden, tauschen sie mit anderen externen Routern Routinginformationen aus. Danach senden externe Router Routinginformationen nur unter den folgenden Umständen:

- Wenn ein Netzwerk der Routingtabelle hinzugefügt oder gelöscht wird
- Wenn die Entfernung zu einem Netzwerk geändert wird
- Wenn eine Änderung im Pfad zu einem Netzwerk den externen Router veranlaßt, über sein lokales Verbundnetz anstatt über den Tunnel auf dieses Netzwerk zuzugreifen, oder über den Tunnel anstatt über das lokale Verbundnetz

Ein AURP-Tunnel fungiert als einzelne, virtuelle Datenverbindung zwischen entfernten AppleTalk-Verbundnetzen. Auf dem Weg zwischen externen Routern kann jede Anzahl von physischen Knoten liegen, aber diese Knoten sind für die AppleTalk-Netzwerke transparent. Es gibt zwei AURP-Tunnelarten: *Point-to-Point-Tunnel* und *Mehrpunkttunnel*. Ein Point-to-Point-AURP-Tunnel verbindet nur zwei externe Router. Ein AURP-Mehrpunkttunnel verbindet drei oder mehr externe Router. Auch gibt es zwei Arten von Mehrpunkttunnel. Ein

voll integrierter Mehrpunkttunnel ermöglicht es allen angeschlossenen externen Routern, sich gegenseitig Pakete zu schicken. Bei einem partiell integrierten Mehrpunkttunnel, erkennen ein oder mehr externe Router nur ein paar, aber nicht alle anderen externen Router. Bild 25.9 zeigt ein AppleTalk-LAN, das mit einem Point-to-Point-AURP-Tunnel verbunden ist.

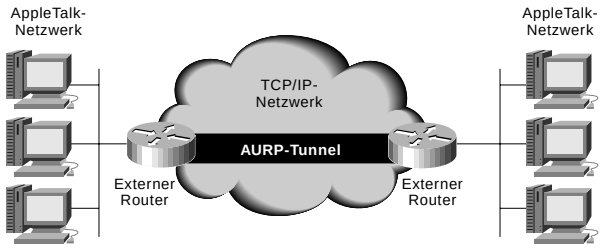


Bild 25.9: Ein AURP-Tunnel ist eine virtuelle Verbindung zwischen entfernten Netzwerken

AURP-Kapselung

Beim Austausch von Routinginformationen oder Daten über einen AURP-Tunnel müssen AppleTalk-Pakete von RTMP, ZIP und (in der Cisco-Implementierung) Enhanced IGRP in AURP umgewandelt werden. Die Pakete werden dann für den Transport durch das TCP/IP-Netzwerk in User-Datagram-Protocol (UDP-)Header verkapselt. Umwandlung und Kapselung werden von externern Routern vorgenommen, die die AppleTalk-Routinginformationen oder Datenpakete erhalten, die an ein entferntes AppleTalk-Verbundnetz gesendet werden müssen. Der externe Router wandelt die Pakete in AURP-Pakete um, die dann in UDP-Header verkapselt und in den Tunnel (also an das TCP/IP-Netzwerk) geschickt werden.

Das TCP/IP-Netzwerk behandelt die Pakete wie normalen UDP-Verkehr. Der entfernte externe Router empfängt die UDP-Pakete und entnimmt dem UDP-Header Informationen. Die AURP-Pakete, egal ob Routinginformationen oder Datenpakete, werden dann wieder in ihr ursprüngliches Format umgewandelt. Enthalten die AppleTalk-Pakete Routinginformationen, aktualisiert der externe Zielrouter seine Routingtabellen entsprechend. Enthalten die Pakete Daten, die für einen AppleTalk-Knoten im lokalen Netzwerk bestimmt sind, wird der Datenstrom an die entsprechende Schnittstelle gesendet.

25.7.4 AppleTalk Transaction Protocol (ATP)

Das *AppleTalk Transaction Protocol* (ATP) ist ein Transportschicht-Protokoll in der AppleTalk-Protokollsuite, das die Interaktion zwischen zwei AppleTalk-Sockets regelt. Eine Interaktion besteht aus Anfrage und Antwort, die zwischen den beteiligten Socket Clients ausgetauscht werden.

Der anfragende Socket Client sendet seine Anfrage, in der er den empfangenden Client auffordert, eine Aufgabe zu erledigen. Beim Empfang der Anfrage führt der Client die gewünschte Aufgabe aus und liefert in seiner Antwort die angeforderte Information. Beim Übermitteln von Anfrage und Antwort übt ATP die wichtigsten Funktionen der Transportschicht aus, einschließlich Bestätigung und erneute Übertragung, Sequentialisieren der Pakete, Segmentierung und neu Zusammensetzen.

Verschiedene Protokolle der Kommunikationssteuerschicht laufen über ATP, einschließlich des *AppleTalk Session Protocol* (ASP) und des *Printer-Access Protocol* (PAP). Diese beiden AppleTalk-Protokolle der oberen Schichten werden weiter unten in diesem Kapitel erläutert.

Antwortende Geräte verhalten sich unterschiedlich, je nachdem, welcher von den beiden Interaktionsdiensten verwendet wird: *At-Least-Once*-(ALO-) oder *Exactly-Once*-(XO-)Interaktionen. ALO-Interaktionen werden verwendet, wenn die Anfrage bei der Wiederholung unverändert bleibt. Wenn die Anwort einer Interaktion verlorengeht, wird die Quelle ihre Anfrage wiederholen. Dies wirkt sich auf die Protokolloperationen nicht nachteilig aus, weil die Anfrage bei der Wiederholung unverändert bleibt. XO-Interaktionen werden verwendet, wenn die Wiederholung der Anfrage die Protokolloperationen negativ beeinflussen kann. Die empfangenden Geräte führen eine Liste jeder kürzlich erhaltenen Interaktion, so daß doppelte Anfragen nicht mehr als einmal bearbeitet werden.

25.7.5 AppleTalk Echo Protocol (AEP)

Das *AppleTalk Echo Protocol* (AEP) ist ein Transportschicht-Protokoll der AppleTalk-Protokollsuite, das Pakete erzeugt, die die Erreichbarkeit von Netzwerkknoten überprüfen. AEP kann in jeden AppleTalk-Knoten implementiert werden und

trägt die statisch zugewiesene Socketnummer 4 (der sogenannte Echo Socket).

Um die Erreichbarkeit eines bestimmten Knotens zu prüfen, wird ein AEP-Anfragepaket an das DDP der Quelle übergeben. DDP adressiert das Paket entsprechend und verzeichnet im Typenfeld, daß es sich bei dem Paket um eine AEP-Anfrage handelt. Erreicht das Paket das Ziel, prüft DDP das Typenfeld und merkt, daß eine AEP-Anfrage vorliegt. Das Paket wird kopiert, in eine AEP-Antwort umgewandelt (durch die Veränderung eines Felds) und an den Ausgangsknoten zurückgeschickt.

25.8 AppleTalk-Protokolle der oberen Schichten

AppleTalk implementiert Dienste in der Kommunikationssteuer-, der Darstellungs- und der Anwendungsschicht des OSI-Schichtenmodells. Vier Hauptimplementierungen der Kommunikationssteuerschicht sind in der AppleTalk-Protokollreihe enthalten. (Die Kommunikationssteuerschicht erstellt, verwaltet und beendet Kommunikationsvorgänge zwischen Einheiten der Darstellungsschicht).

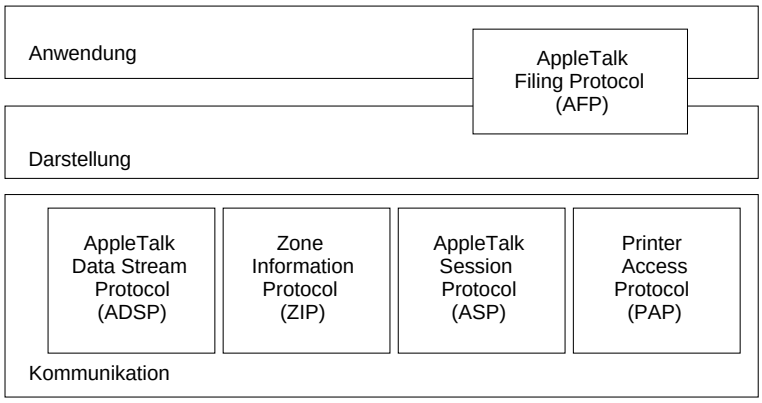
Kommunikationsvorgänge bestehen aus Frage- und Antwortdiensten, die zwischen Anwendungen ablaufen, die sich auf verschiedenen Geräten des Netzwerks befinden. Diese Fragen und Antworten werden von Protokollen gesteuert, die in der Kommunikationssteuerschicht implementiert sind.

Zu den Protokollimplementierungen der Kommunikationssteuerschicht, die von AppleTalk unterstützt werden, gehören das *AppleTalk Data-Stream Protocol (ADSP)*, das *Zone-Information Protocol (ZIP)*, das *AppleTalk Session Protocol (ASP)* und das *Printer-Access Protocol (PAP)*.

Das *AppleTalk Filing Protocol (AFP)* ist in den Darstellungs- und Anwendungsschichten der AppleTalk-Protokollsuite implementiert. Normalerweise bietet die Darstellungsschicht eine Reihe von Kodierungs- und Umwandlungsfunktionen, die auf Daten der Anwendungsschicht bezogen werden. Die Anwendungsschicht interagiert mit Software-Anwendungen (die außerhalb des OSI-Schichtenmodells liegen), die eine Kommuni-

kationskomponente einfügen. Zu den Funktionen der Anwendungsschicht gehören typischerweise die Erkennung von Kommunikationspartnern, das Feststellen, ob Ressourcen verfügbar sind, und die Synchronisierung der Kommunikation. Bild 25.10 veranschaulicht, wie die oberen Schichten der AppleTalk-Protokollreihe mit dem OSI-Schichtenmodell verknüpft sind.

Bild 25.10: Apple-Talk-Protokolle befinden sich in drei der oberen Schichten des OSI-Modells



25.8.1 AppleTalk Data-Stream Protocol (ADSP)

Das *AppleTalk Data-Stream Protocol (ADSP)* ist ein Protokoll der Kommunikationssteuerschicht in der AppleTalk-Protokollsuite, das die Vollduplex-Kommunikation zwischen zwei AppleTalk-Sockets aufbaut und aufrechterhält. ADSP stellt sicher, daß Daten korrekt aneinandergereiht und Pakete nicht dupliziert werden. ADSP implementiert ebenso einen Mechanismus der Ablaufsteuerung, der es dem Empfänger erlaubt, die Übermittlungen der Quelle zu verlangsamen, indem die Größe des Empfangsfensters reduziert wird. ADSP setzt direkt auf DDP auf.

25.8.2 Zone Information Protocol (ZIP)

Das *Zone-Information Protocol (ZIP)* ist ein Protokoll der Kommunikationssteuerschicht der AppleTalk-Protokollsuite, das die Verknüpfungen zwischen Netzwerk-Nummer und Zonennamen in AppleTalk-Routern pflegt. ZIP wird hauptsächlich von AppleTalk-Routern verwendet. Aber auch andere

Netzwerk-Knoten benutzen ZIP beim Hochfahren, um ihre Zone auszuwählen. ZIP pflegt in jedem Router eine *zone-information table* (ZIT). ZITs sind Listen, die spezifischen Netzwerk-Nummern einen oder mehr Zonennamen zuordnen. Jede ZIT listet für jedes Netzwerk im Verbundnetz Netzwerk-Nummern mit den entsprechenden Zonen auf. Bild 25.11 zeigt eine gängige ZIT.

Netzwerk-nummer	Zonen
10	Marketing
20-25	Dokumentation, Schulung
50	Finanzen
100-120	Entwicklung
100-120	Administration

Bild 25.11: Die Zoneninformationstabelle hilft bei der Identifikation von Zonen

25.8.3 AppleTalk Session Protocol (ASP)

Das *AppleTalk Session Protocol (ASP)* ist ein Protokoll der Kommunikationssteuerschicht der AppleTalk-Protokollsuite, das Sitzungen zwischen AppleTalk-Clients und -Servern aufbaut und überwacht. ASP erlaubt es einem Client, eine Sitzung zu einem Server aufzubauen und ihm Befehle zu übermitteln. Es können mehrere Client-Sitzungen zu einem einzigen Server gleichzeitig bestehen. ASP nutzt viele der Dienste, die Protokolle der unteren Schichten, wie ATP und NBP, zur Verfügung stellen.

25.8.4 Printer-Access Protocol (PAP) im Überblick

Das *Printer-Access Protocol (PAP)* ist ein Protokoll der Kommunikationssteuerschicht in der AppleTalk-Protokollsuite, das es Client-Workstations erlaubt, Verbindungen mit Servern, vor allem Druckern, aufzubauen. Eine Sitzung zwischen der Client-Workstation und einem Server wird aufgebaut, wenn die Workstation eine Sitzung mit einem bestimmten Server anfordert. PAP verwendet NBP, um die Netzwerk-Adresse des angeforderten Servers zu erfahren und öffnet dann die Verbindung zwischen Client und Server. Daten werden mit Hilfe von ATP zwischen Client und Server ausgetauscht. Wenn die Kommunikation vollendet ist, beendet PAP die Verbindung. Server mit PAP-Implementierung unterstützen mehrere Verbindungen mit Clients gleichzeitig. Dies ermöglicht es einem Druckserver zum Beispiel, Aufgaben für verschiedene Workstations zur gleichen Zeit zu bearbeiten.

25.8.5 AppleTalk Filing Protocol (AFP)

Das *AppleTalk Filing Protocol (AFP)* ermöglicht es AppleTalk-Workstations, Dateien im Netzwerk gemeinsam zu benutzen. AFP erledigt Aufgaben in den Darstellungs- und Anwendungsschichten der AppleTalk-Protokollsuite. Dieses Protokoll gewährt die Transparenz im Netzwerk, indem es Benutzern erlaubt, entfernt gespeicherte Daten genauso zu handhaben wie lokal gespeicherte. AFP verwendet die Dienste, die von ASP, ATP und AEP bereitgestellt werden.

25.9 AppleTalk-Protokollreihe

Bild 25.12 zeigt, wie die gesamte AppleTalk-Protokollsuite mit dem OSI-Schichtenmodell zusammenhängt.

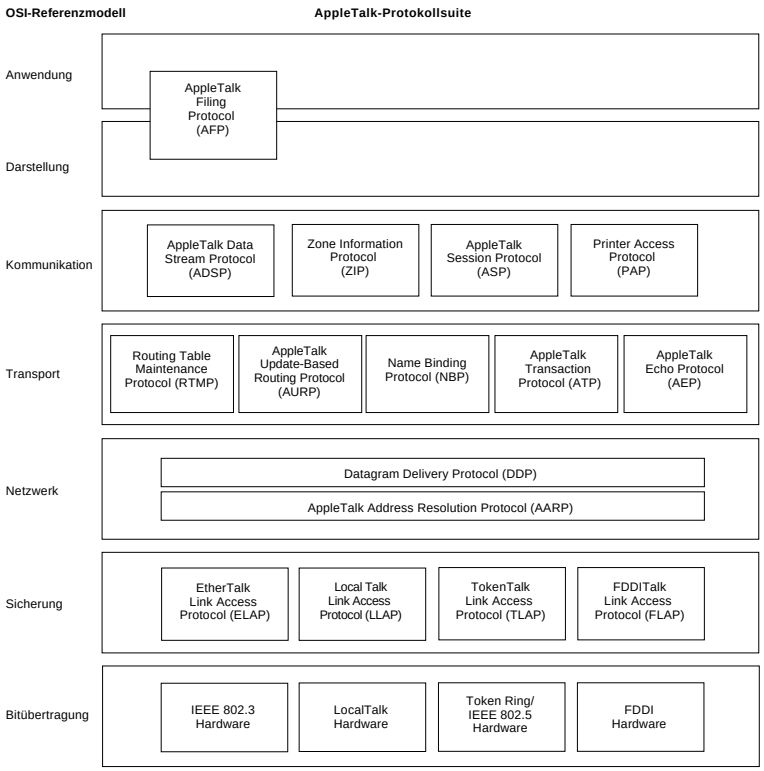


Bild 25.12: Die AppleTalk-Protokollsuite ist mit jeder Schicht des OSI-Modells verknüpft

25.9.1 Format von DDP-Paketen

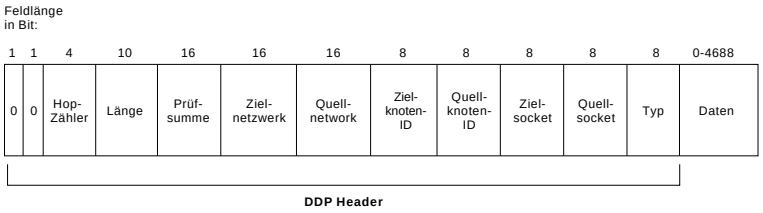
Die folgenden Beschreibungen erläutern die Felder, aus denen die DDP-Pakete bestehen. Diese Pakete haben zwei mögliche Formen:

– *Kleines DDP-Paket* – Das kleine Format wird nur für Übertragungen zwischen zwei Knoten im selben Netzwerk-Segment eines lokalen Netzwerks verwendet. In neuartigen Netzwerken wird dieses Format selten benutzt.

– *Erweitertes DDP-Paket* – Das erweiterte Format wird für Übermittlungen zwischen Knoten mit verschiedenen Netzwerk-Nummern (in einem nicht erweiterten Netzwerk) und für jegliche Übermittlung in einem erweiterten Netzwerk verwendet.

Bild 25.13 zeigt das Format des erweiterten DDP-Pakets:

Bild 25.13: Ein erweitertes DDP-Paket besteht aus 13 Feldern



Die Felder des erweiterten DDP-Pakets, die in Bild 25.13 gezeigt werden, werden im folgenden zusammengefaßt.

- *Hop-Zähler* – Zählt die Anzahl von zwischengeschalteten Geräten, die das Paket durchlaufen hat. Am Ausgangspunkt wird dieses Feld auf Null gesetzt. Jeder zwischengeschaltete Knoten, den das Paket durchläuft, erhöht den Wert um eins. Die maximale Anzahl von Hops ist 15.
- *Länge* – Zeigt die Gesamtlänge des DDP-Pakets in Byte.
- *Prüfsumme* – Enthält eine Prüfsumme, mit der Fehler aufgespürt werden sollen. Wird keine Prüfsumme angegeben, werden die Bits in diesem optionalen Feld auf Null gesetzt.
- *Zielnetzwerk* – Zeigt die 16-Bit-Nummer des Zielnetzwerks an.
- *Quellnetzwerk* – Zeigt die 16-Bit-Nummer des Quellnetzwerks an.
- *Zielknoten ID* – Zeigt die 8-Bit-ID des Zielknotens an.
- *Quellknoten ID* – Zeigt die 8-Bit-ID des Quellknotens an.
- *Zielsocket* – Zeigt die 8-Bit-Nummer des Zielsocket an.
- *Quellsocket* – Zeigt die 8-Bit-Nummer des Quellsocket an.
- *Typ* – Zeigt das Protokoll der oberen Schichten an, zu dem die Informationen im Datenfeld gehören.
- *Daten* – Enthält Daten aus einem Protokoll der oberen Schichten.

KAPITEL 26

DECnet

26.1 Background

DECnet besteht aus einer Reihe von Kommunikationsprodukten, einschließlich einer Protokollsuite, die von der Digital Equipment Corporation (Digital) entwickelt und gefördert wurde. Mit der ersten Version von DECnet, die 1975 herauskam, konnten zwei verbundene PDP-11-Minicomputer miteinander kommunizieren. Digital ist in den darauffolgenden Jahren dazu übergegangen, auch nicht proprietäre Protokolle zu unterstützen, DECnet bleibt jedoch das wichtigste von Digitals Netzwerk-Produkten. Dieses Kapitel liefert eine Übersicht über die DECnet-Protokollsuite, Digitals Netzwerk-Architektur und die die Verwaltung des Datenverkehrs von DECnet.

Bild 26.1 zeigt ein DECnet-Verbundnetz mit Routern, die zwei LANS verbinden, an die Workstations und VAXs angeschlossen sind:

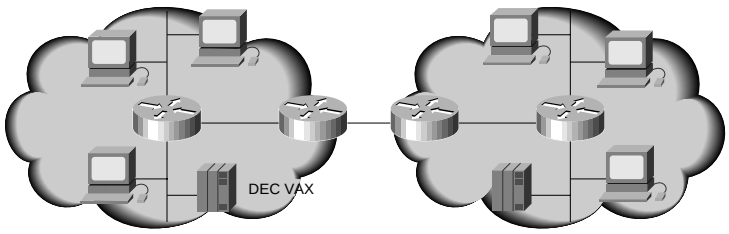


Bild 26.1: In einem Verbundnetz, das auf DECnet basiert, verbinden Router Workstations und VAXs

Verschiedene Versionen von DECnet kamen auf den Markt. Die erste erlaubte es zwei direkt verbundenen Minicomputern, miteinander zu kommunizieren.

Die folgenden Versionen erweiterten die Funktionalität von DECnet, indem die Unterstützung von zusätzlichen proprietären und Standardprotokollen ermöglicht und gleichzeitig die Kompatibilität mit der jeweils vorhergehenden Version gewährleistet wurde. Das bedeutet, daß die Protokolle abwärtskompatibel sind. Vor allem zwei Versionen von DECnet befinden sich heute im Einsatz: DECnet Phase IV und DECnet/OSI.

DECnet Phase IV ist die meistgebrauchte Version von DECnet. Die neueste ist aber DECnet/OSI. DECnet Phase IV basiert auf Phase IV der Digital Network Architecture (DNA) und unterstützt proprietäre Digital-Protokolle und andere proprietäre und Standardprotokolle. DECnet Phase IV ist abwärtskompatibel zu DECnet Phase III, ihrer Vorgängerversion.

DECnet/OSI (auch DECnet Phase V genannt) ist abwärtskompatibel zu DECnet Phase IV und die jüngste Version von DECnet. Diese Version basiert auf DECnet/OSI DNA. DECnet/OSI unterstützt einige OSI-Protokolle, mehrere proprietäre DECnet-Protokolle und andere proprietäre und Standardprotokolle.

26.2 DECnet Phase IV Digital Network Architecture (DNA)

Die *Digital Network Architecture (DNA)* ist eine verständliche, geschichtete Netzwerk-Architektur, die eine Vielzahl von proprietären und Standardprotokollen unterstützt. Phase IV DNA ähnelt der Architektur, die im OSI-Schichtenmodell beschrieben ist. Wie auch das OSI-Schichtenmodell, setzt sich Phase-IV-DNA aus Schichten zusammen, wobei spezifische Schichtenfunktionen Dienste für die darüberliegenden Protokollschichten bereitstellen und von den darunterliegenden abhängen. Anders als das OSI-Modell besteht Phase-IV-DNA aber aus acht Schichten. Bild 26.2 zeigt, wie die acht Schichten von Phase-IV-DNA mit dem OSI-Schichtenmodell korrespondieren.

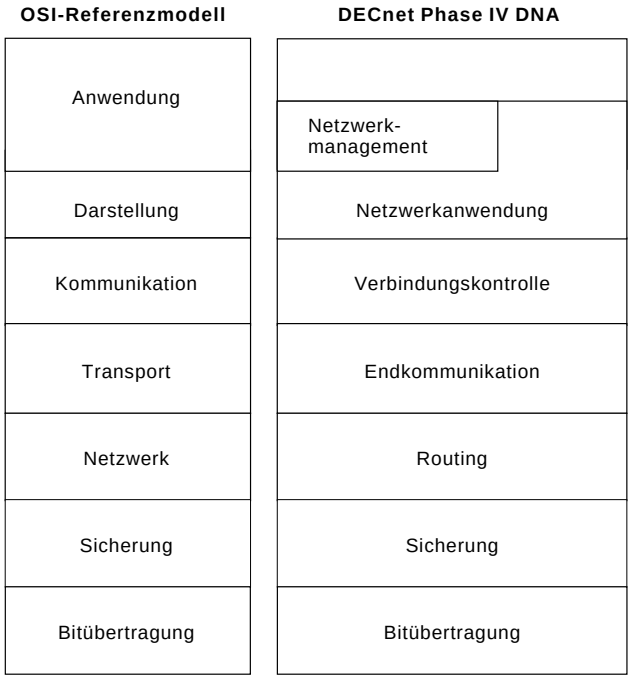


Bild 26.2: Phase IV besteht aus acht Schichten, die sich am OSI-Modell orientieren

Der folgende Abschnitt beschreibt Funktionalität und Aufgabe jeder dieser Schichten und zeigt die Ähnlichkeiten zwischen der Phase-IV-DNA-Architektur und dem OSI-Schichtenmodell auf.

26.2.1 Die Schichten von Phase-IV-DNA

Die DECnet Phase IV DNA definiert, wie in Bild 26.2 gezeigt, ein Acht-Schichten-Modell. Die *Benutzerschicht* ist die Netzwerkschnittstelle des Benutzers, die ihm über eine Kommunikationskomponente Dienste und Programme bereitstellt. Die Benutzerschicht entspricht ungefähr der OSI-Anwendungsschicht. Die *Netzwerk-Managementschicht* ist die Benutzerschnittstelle zu Informationen des Netzwerk-Managements. Diese Schicht arbeitet mit allen unteren Schichten von DNA zusammen und entspricht grob der OSI-Anwendungsschicht. Die *Netzwerk-Anwendungsschicht* stellt verschiedene Netzwerk-Anwendungen bereit, wie zum Beispiel den Fernzugriff auf Dateien und den Zugriff auf ein virtuelles Terminal. Diese Schicht entspricht ungefähr den Darstellungs- und Anwendungsschichten von OSI. Die *Verbindungskontrollschicht* re-

gelt logische Verbindungen zwischen Endknoten und entspricht ungefähr der OSI-Kommunikationssteuerschicht. Die *Endkommunikationsschicht* regelt die Ablaufsteuerung, die Segmentierung und das Wiederzusammensetzen und entspricht ungefähr der OSI-Transportschicht. Die *Routingschicht* erledigt das Routing und andere Funktionen und entspricht ungefähr der OSI-Netzwerk-Schicht. Die *Sicherungsschicht* reguliert die physischen Netzwerk-Kanäle und entspricht der OSI-Sicherungsschicht. Die *Bitübertragungschicht* kontrolliert Hardware-Schnittstellen und entscheidet über elektrische und mechanische Funktionen des physischen Mediums. Diese Schicht entspricht der OSI-Bitübertragungsschicht.

26.2.2 Phase-IV-DECnet-Adressierung

DECnet-Adressen gehen nicht aus den physischen Netzwerken hervor, mit denen die Knoten verbunden sind. Statt dessen findet DECnet Hosts, indem es Adreßpaare aus Bereich und Knoten benutzt. Der Wert eines Bereichs liegt zwischen 1 und einschließlich 63. Eine Knotenadresse kann zwischen 1 und einschließlich 1023 liegen. Also kann jeder Bereich 1023 Knoten haben, und es können ungefähr 65000 Knoten in einem DECnet-Netzwerk adressiert werden. In einem Bereich können viele Router liegen, und ein einziges Kabel kann eine Vielzahl von Bereichen unterstützen. Wenn also ein Knoten verschiedene Netzschnittstellen hat, benutzt er dieselbe Bereich/Knoten-Adresse für alle Schnittstellen. Bild 26.3 zeigt ein DECnet-Netzwerk mit verschiedenen adressierbaren Entitäten.

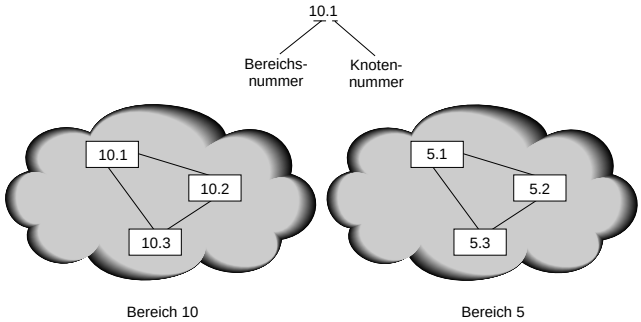


Bild 26.3: DECnet findet Hosts durch Bereich/Knoten-Adreßpaare

DECnet-Hosts verwenden keine vom Hersteller zugewiesenen Media Access Control (MAC)-Schichtenadressen. Statt dessen werden die Netzwerk-Adressen in die MAC-Adressen nach einem Algorithmus eingebettet, der die Bereichsnummern mit 1024 multipliziert und dem Produkt die Knotennummer hinzufügt. Die 16-Bit-Dezimaladresse, die daraus hervorgeht, wird in eine Hexadezimalzahl umgewandelt und an die Adresse AA00.0400 in umgekehrter Reihenfolge angehängt, das niederstwertige Byte zuerst. Aus der DECnet-Adresse 12.75 wird zum Beispiel 12363 (dezimal), was 304B entspricht (hexadezimal). Nachdem diese Adresse in umgekehrter Reihenfolge an dem standardardmäßigen DECnet MAC-Adressenvorspann angehängt ist, lautet die Adresse AA00.0400.4B30.

26.3 DECnet/OSI Digital Network Architecture (DNA)

Die DECnet/OSI (DECnet Phase V) DNA ist der Architektur, die im OSI-Schichtenmodell festgelegt ist, sehr ähnlich. DECnet Phase V verwendet einen Schichtenansatz, der bei der Unterstützung von Protokollsuiten der oberen Schichten ein hohes Maß an Flexibilität erreicht. Wie im folgenden Abschnitt beschrieben, kann DECnet OSI tatsächlich eine Vielzahl von Protokollsuiten unterstützen.

26.3.1 DECnet/OSI-DNA-Implementierungen

Die DECnet/OSI-DNA definiert ein Schichtenmodell, das drei Protokollsuiten implementiert: OSI, DECnet und Transmission Control Protocol/Internet Protocol (TCP/IP). Die OSI-Implementierung von DECnet/OSI entspricht dem OSI-Modell mit seinen sieben Schichten und unterstützt viele der standardmäßigen OSI-Protokolle. Die Digital-Implementierung von DECnet/OSI gewährt die Abwärtskompatibilität zu DECnet Phase IV und unterstützt viele proprietäre Digital-Protokolle. Die TCP/IP-Implementierung von DECnet/OSI unterstützt die TCP/IP-Protokolle der unteren Schichten und ermöglicht die Übermittlung von DECnet-Daten über TCP-Transportprotokolle. Bild 26.4 zeigt die drei DECnet/OSI-Implementierungen:

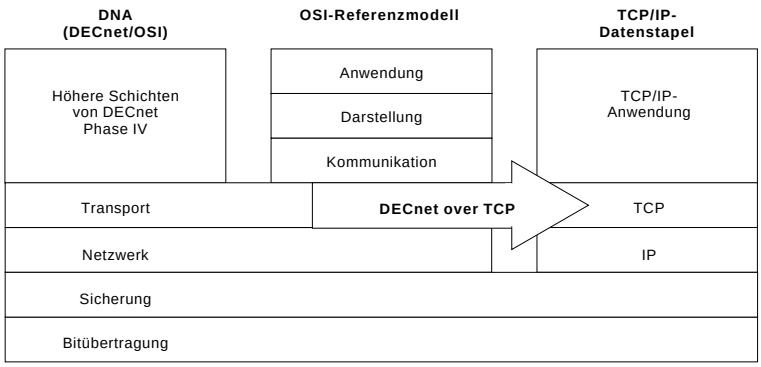


Bild 26.4: OSI, DECnet und TCP werden allesamt von DECnet/OSI DNA unterstützt

26.4 DECnet-Medienzugriff

DECnet Phase IV und DECnet/OSI unterstützen verschiedenste Implementierungen des Medienzugriffs in der Bitübertragungs- und Sicherungsschicht. Dies führte zu einer relativ breiten Akzeptanz von DECnet in der Netzwerk-Industrie. Wie in den folgenden Abschnitten erläutert wird, unterstützen sowohl DECnet Phase IV als auch Phase V viele der heute gängigen Technologien der Bitübertragungs- und Sicherungsschicht.

In der Bitübertragungsschicht unterstützen DECnet Phase IV und DECnet/OSI die populärsten physischen Implementierungen, wie etwa Ethernet/IEEE 802.3, Token Ring/IEEE 802.5, und Fiber-Distributed Data Interface (FDDI). Außerdem unterstützt DECnet/OSI Frame Relay und X.21bis.

In der Sicherungsschicht unterstützen DECnet Phase IV und DECnet/OSI IEEE 802.2 Logical Link Control (LLC), Link-Access Procedure, Balanced (LAPB), Frame Relay und High-Level Data-Link Control (HDLC). Ebenso unterstützen sowohl DECnet Phase IV als auch DECnet/OSI die proprietären Digital-Sicherungsschichtprotokolle, Digital Data Communications Message Protocol (DDCMP), die Point-to-Point- und Konferenzverbindungen, Voll- und Halbduplex-Kommunikation über synchrone und asynchrone Kanäle wie auch Fehlerbehebung, Sequenzialisieren und Verwaltung.

26.5 DECnet-Routing

DECnet-Routing wird in der Routingschicht der DNA in DECnet Phase IV und in der Netzwerkschicht des OSI-Modells in DECnet/OSI durchgeführt. Die Routingimplementierungen in DECnet Phase IV und DECnet/OSI sind ähnlich.

DECnet-Phase-IV-Routing ist durch das DECnet Routing Protocol (DRP) implementiert, bei dem es sich um ein relativ einfaches und effektives Protokoll handelt, dessen Hauptfunktion darin besteht, die optimale Strecke durch ein DECnet-Phase-IV-Netzwerk zu finden. Bild 26.5 zeigt an einem DECnet-Netzwerk, wie das Routing in einem DECnet-Phase-IV-Netzwerk abläuft.

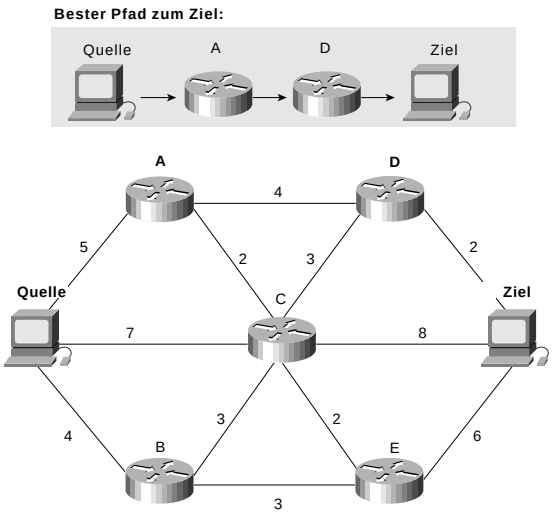


Bild 26.5: Die DRP findet die optimale Strecke durch ein DECnet-Phase-IV-Netzwerk

Die Routingentscheidungen im DECnet basieren auf Kosten, eine beliebige Maßeinheit, die von Netzwerk-Administratoren zugewiesen wird und mit der verschiedene Pfade durch ein Verbundnetz verglichen werden. Die Kosten hängen unter anderem von der Etappenzählung und der Medienbandbreite ab. Je geringer die Kosten, desto besser der Pfad. Treten Netzfehler auf, verwendet DRP den Kostenwert, um den besten Pfad zu jedem Ziel neu zu bestimmen.

DECnet/OSI-Routing ist von den standardmäßigen OSI-Routingprotokollen (ISO 8473, ISO 9542 und ISO 10589) und DRP implementiert. Genaue Informationen zu den OSI-Routingprotokollen finden Sie in Kapitel 39, »Open Systems Interconnection (OSI) Routing Protocol«.

26.6 DECnet-Endkommunikationsschicht

DECnet Phase IV unterstützt ein einziges Transportprotokoll in der DNA-Endkommunikationsschicht, das Network-Services Protocol (NSP).

26.6.1 Network-Services Protocol

Das *Network-Services Protocol (NSP)* ist ein proprietäres, verbindungsorientiertes Endkommunikationsprotokoll von Digital, das Verbindungen zwischen Knoten aufbaut und beendet, Nachrichten fragmentiert und wieder zusammensetzt und Fehler behebt.

NSP regelt auch zwei Arten von Ablaufsteuerung: einen Start/Stop-Mechanismus, bei dem der Empfänger dem Sender mitteilt, wann er die Datenübermittlung beenden und wiederaufnehmen soll, und ein komplexeres Verfahren, bei dem der Empfänger dem Sender mitteilt, wie viele Nachrichten er annehmen kann.

26.7 DECnet/OSI-Transportschicht

DECnet/OSI unterstützt NSP, drei standardmäßige OSI-Transportprotokolle und das Transmission Control Protocol (TCP).

DECnet/OSI unterstützt die Transportprotokollklassen (TP) 0, TP2 und TP4. TP0 ist das einfachste verbindungsorientierte OSI-Transportprotokoll. Von den klassischen Funktionen der Transportschicht führt es nur die Segmentierung und das Wiederzusammensetzen durch. Das heißt, daß TP0 die maximale Größe einer Protocol Data Unit (PDU) feststellt, die von zugrundeliegenden Subnetzen unterstützt wird, und das zu transportierende Paket in kleinere Einheiten aufteilt, die für die Übertragung im Netzwerk nicht zu groß sind. TP2 kann

Datenströme über einen einzigen virtuellen Schaltkreis multiplexen und demultiplexen. Durch diese Fähigkeit ist TP2 besonders geeignet für Public Data Networks (PDNs), wo jeder virtuelle Schaltkreis anderen Belastungen unterliegt. Wie TP0 und TP1, segmentiert auch TP2 PDUs und setzt sie wieder zusammen, während TP3 die Funktionen von TP1 und TP2 kombiniert. TP4, das populärste OSI-Transportprotokoll, ist dem TP der Internet-Protokollsuite ähnlich und basiert auch tatsächlich auf diesem Modell. Zusätzlich zu den TP3-Funktionen bietet TP4 einen zuverlässigen Transportdienst und geht von einem Netzwerk aus, in dem Probleme nicht erkannt werden.

Request For Comments (RFC) 1006 und RFC 1006 Extensions definieren eine Implementierung der OSI-Transportschichtprotokolle auf TCP. RFC 1006 definiert die Implementierung des OSI Transport Protocol Klasse 0 (TP0) auf TCP. RFC 1006 Extensions definiert die Implementierung des Transport Protocol Klasse 2 (TP2) auf TCP.

26.8 Die oberen Schichten von DECnet Phase IV

Die DNA von DECnet Phase IV spezifiziert vier obere Schichten, in denen Interaktionsdienste, Netzwerk-Managementfunktionen, Datentransfer und Verbindungskontrolle bereitgestellt werden. Man spricht von Benutzer-, Netzwerk-Management-, Netzwerk-Anwendungs- und Verbindungskontrollschicht. Die oberen Schichten der DECnet-Phase-IV-Architektur werden in den folgenden Abschnitten genauer beschrieben.

26.8.1 Benutzerschicht

Die DNA-Benutzerschicht unterstützt Benutzerdienste und Programme, die mit den Anwendungen des Benutzers interagieren. Der Endbenutzer arbeitet direkt mit diesen Anwendungen, und die Anwendungen benutzen die Dienste und Programme, die von der Benutzerschicht bereitgestellt werden.

26.8.2 Netzwerk-Managementschicht

Das Netzwerk-Managementprotokoll, das in DECnet-Netzwerken häufig verwendet wird, ist Digitals proprietäres Network Information and Control Exchange (NICE)-Protokoll. NICE ist ein Kommando/Antwort-Protokoll. Kommandos, die eine Handlung verlangen, werden an einen verwalteten Knoten oder Prozeß ausgegeben. Antworten, in Form von Handlungen, werden von diesen Knoten oder Prozessen geliefert. NICE führt verschiedene Funktionen des Netzwerk-Managements durch und kann dazu benutzt werden, das Betriebssystem eines lokalen Systems in ein Fernsystem umzuwandeln oder es einem unbedienten Fernsystem zu ermöglichen, seinen Speicherinhalt an das lokale System zu übergeben. Protokolle, die NICE verwenden, können bestimmte Eigenschaften des Netzwerks untersuchen und ändern. NICE unterstützt eine Protokolleinrichtung, die automatisch wichtige Ereignisse im Netzwerk, wie etwa Änderungen der Anordnung oder des Zustands von Schaltkreisen, aufspürt. NICE unterstützt Funktionen, die das Testen von Hardware und Ringleitungen zwischen Knoten erleichtern.

Bestimmte Netzwerkmanagement-Funktionen können auf das *Maintenance Operations Protocol (MOP)* zugreifen, einer Sammlung von Funktionen, die unabhängig von den DNA-Schichten zwischen Netzwerk-Management- und Sicherungsschicht arbeiten können. Dies erlaubt den Zugriff auf Knoten, die sich in einem Zustand befinden, in dem nur Dienste der Sicherungsschicht erhältlich oder ausführbar sind.

26.8.3 Netzwerk-Anwendungsschicht

Das Data-Access Protocol (DAP), ein proprietäres Protokoll von Digital, wird von DECnet Phase IV in der Netzwerk-Anwendungsschicht verwendet. DAP unterstützt den Fernzugriff auf Dateien und den Dateitransfer – Dienste also, die von Anwendungen der Netzwerk-Management- und der Benutzerschicht beansprucht werden. Andere proprietäre Digital-Protokolle, die auf der Netzwerk-Anwendungsschicht operieren sind MAIL, das den Austausch von Mailnachrichten erlaubt, und CTERM, das den Fernzugriff auf Terminals ermöglicht.

26.8.4 Verbindungskontrollschicht

Das Session-Control Protocol (SCP) ist ein Verbindungskontrollschicht-Protokoll von DECnet Phase IV, das mehrere Funktionen ausübt. SCP fordert vor allem logische Verbindungen von Endgeräten an, erhält von Endgeräten Anfragen nach logischen Verbindungen, übersetzt Namen in Adressen und beendet logische Verbindungen.

26.9 Die oberen Schichten von DECnet/OSI

Die DECnet/OSI DNA basiert auf dem OSI-Schichtenmodell. DECnet/OSI unterstützt in jeder der oberen Schichten zwei Protokollsuites: die OSI-Protokolle und die DECnet-Phase-IV-Protokolle (um die Abwärtskompatibilität zu gewährleisten). DECnet/OSI unterstützt Funktionalitäten in den Anwendungs-, Darstellungs- und Kommunikationssteuerschichten.

26.9.1 Anwendungsschicht

DECnet/OSI realisiert die standardmäßigen Implementierungen der OSI-Anwendungsschichten, so wie standardmäßige Prozesse der Anwendungsschicht wie *Common Management-Information Protocol (CMIP)* und *File Transfer, Access and Management (FTAM)*. DECnet/OSI unterstützt ebenso alle Protokolle, die von DECnet Phase IV in der Benutzer- und Netzwerk-Managementschicht der DNA implementiert sind, wie zum Beispiel das *Network Information and Control Exchange (NICE)*-Protokoll.

Die OSI-Anwendungsschicht enthält sowohl wirkliche Anwendungen als auch Application Service Elements (ASEs). ASEs ermöglichen einfache Kommunikation der Anwendungen mit den unteren Schichten. Die drei wichtigsten ASEs sind Association Control Service Element (ACSE), Remote Operations Service Element (ROSE) und das Reliable Transfer Service Element (RTSE). ACSE verbindet Anwendungsnamen miteinander, damit Anwendungen untereinander kommunizieren können. ROSE implementiert einen wählbaren Frage/Antwort-Mechanismus, der ähnliche Fernoperationen ermöglicht wie Remote Procedure Calls (RPCs). RTSE hilft bei der kor-

rekten Auslieferung, indem es die Handhabung der Kommunikationssteuerschicht vereinfacht.

26.9.2 Darstellungsschicht

DECnet/OSI realisiert alle standardmäßigen OSI-Implementierungen der Darstellungsschicht. DECnet/OSI unterstützt ebenso alle Protokolle, die DECnet Phase IV in der Netzwerk-Anwendungsschicht der DNA implementiert. Das wichtigste davon ist das *Data-Access Protocol (DAP)*.

Die OSI-Darstellungsschicht ist gewöhnlich nur ein Durchgangsprotokoll für Informationen benachbarter Schichten. Obwohl viele Leute glauben, Abstract Syntax Notation 1 (ASN.1) sei das OSI-Protokoll der Darstellungsschicht, wird ASN.1 verwendet, um Datenformate in einem maschinenunabhängigen Format auszudrücken. Dies ermöglicht die Kommunikation zwischen Anwendungen verschiedener Computersysteme auf eine Weise, die für die Anwendungen transparent ist.

26.9.3 Verbindungskontrollschicht

DECnet/OSI realisiert alle Implementierungen der standardmäßigen OSI-Kommunikationssteuerschicht. Ebenso unterstützt DECnet/OSI alle Protokolle, die von DECnet Phase IV in der Verbindungskontrollschicht der DNA implementiert sind. Das primäre Verbindungskontrollschicht-Protokoll ist das *Session-Control Protocol (SCP)*. Das OSI-Kommunikationssteuerschicht-Protokoll verwandelt die Datenströme, die von den unteren vier Schichten geliefert werden, in Sitzungen, indem es verschiedene Kontrollmechanismen realisiert. Zu diesen Mechanismen gehören Berechnungen, Verbindungskontrolle und das Aushandeln von Sitzungsparametern. Die Verbindungskontrolle wird durch ein Sendezeichen realisiert, das zum Kommunizieren berechtigt. Das Sendezeichen kann angefordert werden, und einem ES kann das Vorrecht eingeräumt werden, das Sendezeichen unterschiedlich zu verwenden.

Bild 26.6 zeigt die vollständigen DECnet-Phase-IV- und DECnet/OSI-Protokollsuiten, der Implementierung von DECnet/OSI auf TCP.

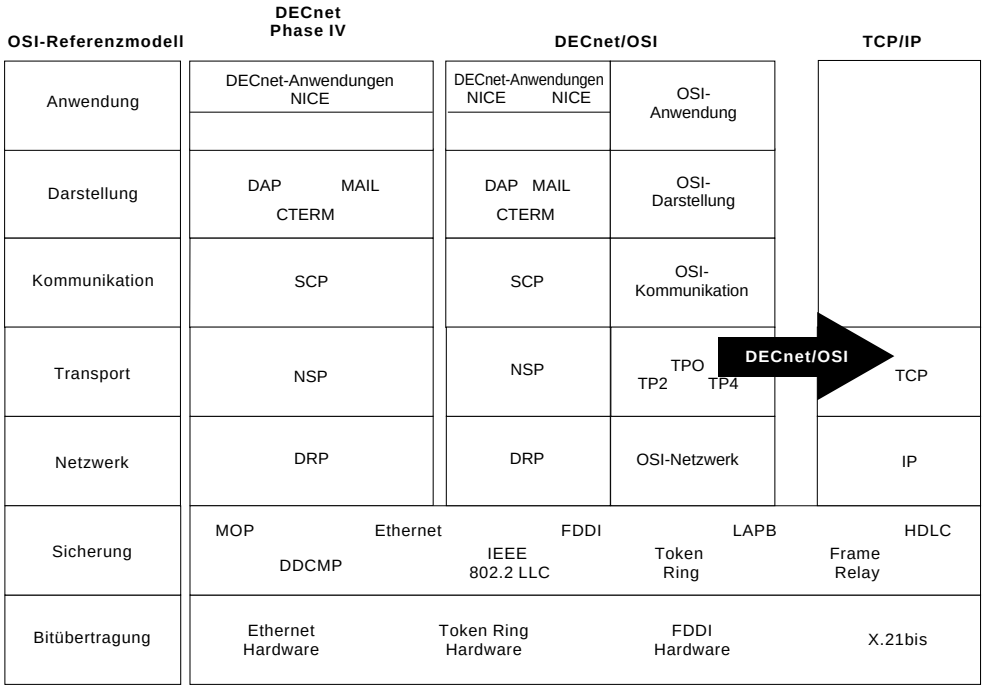


Bild 26.6: DECnet Phase IV und DECnet/OSI unterstützen die gleichen Spezifikationen von Sicherungs- und Bitübertragungensschicht

KAPITEL 27

IBM Systems Network Architecture (SNA)

27.1 Background

IBM-Netzwerke bestehen heute im wesentlichen aus zwei verschiedenen Architekturen, die einen mehr oder weniger gemeinsamen Ursprung haben. Bevor es die gegenwärtigen Netzwerke gab, bestimmte IBMs *Systems Network Architecture (SNA)* die Netzwerk-Landschaft, so daß man häufig von traditioneller oder ursprünglicher SNA spricht.

Als PCs, Workstations und Client/Server-Architekturen aufkamen, kam IBM dem Bedarf an einer Peer-to-Peer-Netzwerkstrategie mit der Entwicklung von *Advanced Peer-to-Peer Networking (APPN)* und *Advanced Program-to-Program Computing (APPC)* entgegen.

Obwohl viele der ursprünglichen Technologien der auf Großrechnern basierenden SNA auf APPN-basierte Netzwerke übertragen wurden, gibt es große Unterschiede. Dieses Kapitel beschreibt die unterschiedlichen IBM-Netzwerk-Umgebungen. Zunächst werden die ursprünglichen SNA-Umgebungen aufgezeigt, später die *AAPN* erläutert. Das Kapitel schließt mit der Beschreibung der Basic-Information Unit (BIU) und der Path-Information Unit (PIU) von IBM.

IBM-basierte Routingstrategien werden in einem eigenen Kapitel behandelt. Einzelheiten über Routingprotokolle von IBM finden Sie in Kapitel 35, »IBM Systems Network Architecture (SNA) Routing«.

27.2 Traditionelle SNA-Umgebungen

SNA wurde in den 70er Jahren mit einer Gesamtstruktur entwickelt, die dem OSI-Schichtenmodell ähnelt. In der SNA dient ein Großrechner, der mit der *Advanced Communication Facility/Virtual Telecommunication Access Method* (ACF/VTAM) arbeitet, als Hub eines SNA-Netzwerks. ACF/VTAM baut alle Verbindungen auf und aktiviert oder deaktiviert Ressourcen. In dieser Umgebung sind Ressourcen explizit vordefiniert, so daß kein Rundspruchverkehr benötigt und der Aufwand an Headern minimiert wird. Die zugrundeliegende Architektur und die Hauptkomponenten der traditionellen SNA-Vernetzung werden in den folgenden Abschnitten zusammengefaßt

27.2.1 IBM-SNA-Architektur

Die Bestandteile des IBM-SNA-Modells sind dem OSI-Schichtenmodell sehr ähnlich. Die folgenden Beschreibungen zeigen die Rolle, die die einzelnen SNA-Komponenten beim Aufbau von Verbindungen zwischen SNA-Entitäten spielen.

- *Datenübermittlungssteuerung (DLC)* – Definiert verschiedene Protokolle, wie etwa das *Synchronous Data-Link Control* (SDLC)-Protokoll für die hierarchische Kommunikation und das Token Ring Network Communication Protocol für die LAN-Kommunikation zwischen Geräten.
- *Pfadsteuerung* – Führt viele Funktionen der OSI-Netzwerk-Schicht aus, einschließlich des Routing und der *Segmentation And Reassembly (SAR)* von Datenpaketen.
- *Übertragungssteuerung* – Stellt einen zuverlässigen Dienst für die Kommunikation zwischen Endpunkten bereit und ermöglicht das Chiffrieren und Dechiffrieren.
- *Datenflußsteuerung* – Regelt Anfrage/Antwort-Prozesse, entscheidet, wer kommunizieren darf, faßt Nachrichten zusammen und unterbricht auf Anforderung den Datenfluß.
- *Darstellungsdienste* – Spezifiziert Algorithmen der Datenumwandlung, die Daten von einem Format in ein anderes übersetzen, koordiniert die gemeinsame Verwendung von Ressourcen und synchronisiert Arbeitsvorgänge.

– *Transaktionsdienste* – Stellt Anwendungsdienste in Form von Programmen bereit, die verteilte Prozeß- oder Managementdienste realisieren.

SNA definiert für die Physikalische Schnittstelle keine spezifischen Protokolle. Es wird erwartet, daß die Physikalische Schnittstelle über andere Standards realisiert wird.

Bild 27.1 zeigt, wie diese Elemente von IBMs SNA-Modell mit dem ISO/OSI-Netzwerkmodell zusammenhängen.

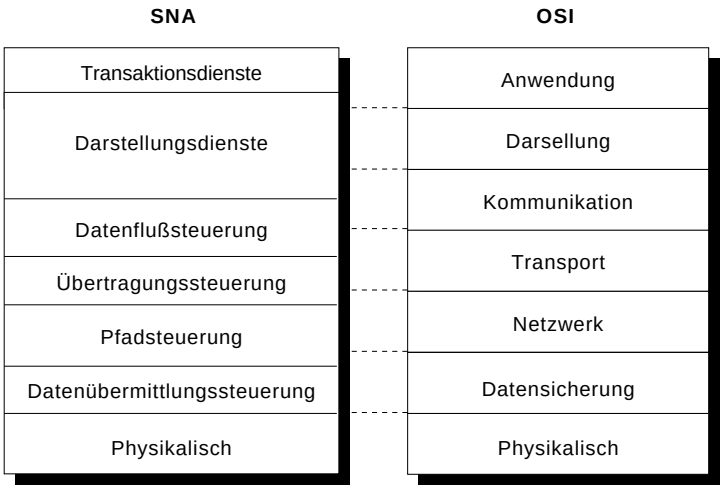


Bild 27.1: IBM-SNA entspricht den sieben Stufen des OSI-Modells

Eine Haupteinrichtung, die im SNA-Netzwerk-Modell definiert wird, ist das *Path-Control Network*, das Informationen zwischen SNA-Knoten bewegen und die Kommunikation im Verbundnetz zwischen Knoten in verschiedenen Netzwerken erleichtern soll. Die Umgebung des Path-Control Network verwendet Funktionen, die von der Wege- und Datenübermittlungsteuerung bereitgestellt werden. Das Path-Control Network ist eine Untereinheit von IBMs Transportnetzwerk.

27.2.2 Physische Entitäten von IBM SNA

Traditionelle physische Entitäten der SNA haben eine der vier folgenden Formen: *Hosts, Kommunikationssteuereinheiten, Aufbausteuereinheiten* und *Terminals*. Die Hosts der SNA

kontrollieren das gesamte oder einen Teil des Netzwerks, führen Berechnungen und Programme aus, ermöglichen den Datenbankzugriff und das Netzwerk-Management und stellen Verzeichnisdienste bereit. (Ein Beispiel für einen Host in einer traditionellen SNA-Umgebung ist ein S/370-Großrechner.) Die Kommunikationssteuereinheiten verwalten das physische Netzwerk und überwachen die Kommunikationsverbindungen. Insbesondere sollen die Kommunikationssteuereinheiten – auch *Front-End Processors* (FEPs) genannt – Daten durch ein traditionelles SNA-Netzwerk routen. (Ein Beispiel für eine Kommunikationssteuereinheit ist ein 3745.) Aufbausteuereinheiten werden gewöhnlich *Clustercontroller* genannt. Diese Geräte kontrollieren Input- und Output-Operationen angeschlossener Geräte wie etwa Terminals. (Ein Beispiel für eine Aufbausteuereinheit ist ein 3174.) Terminals, auch Workstations genannt, stellen die Benutzerschnittstelle des Netzwerks dar. (Ein typisches Beispiel ist ein 3270). Abbildung 27.2 zeigt all diese physischen Entitäten eingebunden in ein SNA-Netzwerk-Diagramm.

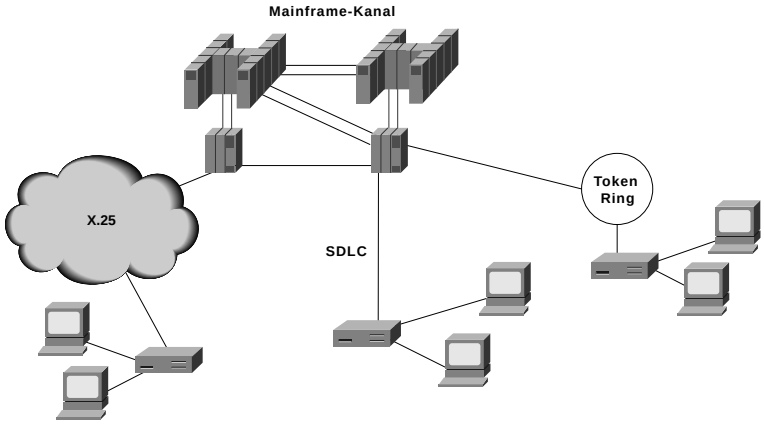


Bild 27.2: Physische Entitäten der SNA können vier Formen haben

27.2.3 Datenübermittlungssteuerung von IBM SNA

Die SNA-Datenübermittlungssteuerungschicht unterstützt diverse Medien, von denen jedes darauf ausgelegt ist, Zugriff auf Geräte und Benutzer mit unterschiedlichen Anforderungen bereitzustellen. Zu den von SNA unterstützten Medienarten

gehören unter anderem Großrechnerkanäle, SDLC, X.25 und Token Ring.

Ein standardmäßiger SNA-Großrechnerkanal bietet einen parallelen Datenkanal, der die Datenübertragungstechnik *Direct-Memory-Access* (DMA) verwendet. Ein Großrechnerkanal verbindet IBM-Hosts miteinander und über mehradrige Kabel mit Kommunikationscontrollern. Jedes Kabel kann dabei bis zu mehreren 100 Metern lang sein. Ein standardmäßiger Großrechnerkanal kann Daten mit Geschwindigkeiten zwischen 3 und 4,5 Mbyte/s übertragen.

IBMs Enterprise-Systems-Connection-Großrechnerumgebung (ESCON) erlaubt höhere Datendurchsätze und kann größere physische Entfernungen überbrücken. Im allgemeinen übermittelt ESCON Daten mit 18 Mbyte/s und unterstützt Standleitungen und Übermittlungen über mehrere Kilometer. Um höhere Datenraten und größere Entfernungen zu ermöglichen, verwendet ESCON Lichtwellenleiter als Netzwerk-Medium.

SDLC wird häufig in SNA-Netzen angewandt, um Kommunikations- und Verbindungsaufbaucontroller zu verbinden und um Daten über Telefonleitungen zu versenden.

X.25-Netze wurden lange Zeit für WAN-Verbindungen implementiert. Im allgemeinen liegt ein X.25-Netz zwischen zwei SNA-Knoten und wird wie eine einzelne Leitung behandelt. SNA implementiert X.25 als Zugriffsprotokoll, und SNA-Knoten werden im X.25-Netz als benachbart betrachtet. Um SNA-Knoten über ein X.25-basiertes WAN zu verbinden, benötigt SNA die Funktionen des DLC-Protokolls, die X.25 nicht bietet. Um die Lücke zu schließen, werden verschiedene spezielle DLC-Protokolle eingefügt, wie etwa der Physical Services Header, Qualified Logical Link Control (QLLC) und Enhanced Logical Link Control (ELLC).

Token-Ring-Netze sind die primäre SNA-DLC-Methode, um den Medienzugriff auf LAN-basierte Geräte zu gewähren. Token Ring, wie es von IBM unterstützt wird, ist praktisch identisch mit dem IEEE-802.5-Verbindungszugriffs-Protokoll, das unter IEEE 802.2 Logical Link Control Type 2 (LLC2) läuft.

Zusätzlich zu der Basisreihe von Medientypen fügte IBM die Unterstützung für diverse andere weit verbreitete Medien hinzu, wie zum Beispiel IEEE 802.3/Ethernet, Fiber-Distributed Data Interface (FDDI) und Frame Relay.

Abbildung 27.3 zeigt, wie die verschiedenen Medien im allgemeinen in ein SNA-Netz eingebunden sind.

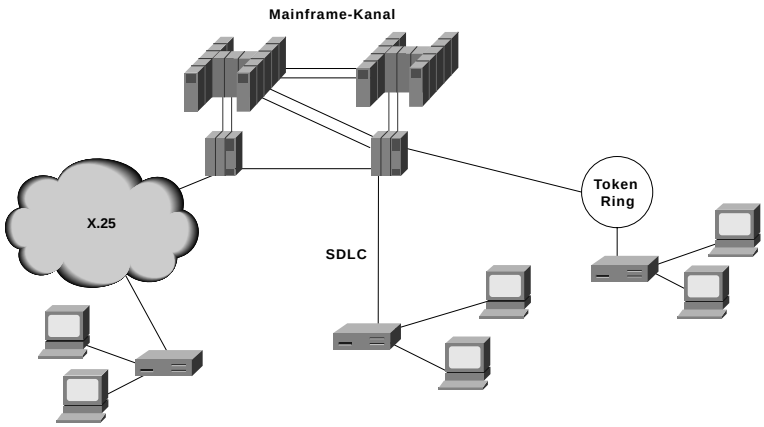


Bild 27.3: SNA unterstützt mittlerweile eine Vielzahl von Medien

27.2.4 IBM Network-Addressable Units (NAUs)

SNA definiert drei grundlegende Network-Addressable Units (NAUs): *Logical Units, Physical Units* und *Control Points*. Jede spielt eine wichtige Rolle beim Aufbau von Verbindungen zwischen Systemen in einem SNA-Netzwerk.

Logical Units (LUs) fungieren als Anbindungen für den Endbenutzer in ein SNA-Netz. LUs stellen Benutzern den Zugriff auf Netzressourcen bereit und verwalten die Übermittlungen von Informationen zwischen Endbenutzern.

Physical Units (PUs) überwachen und kontrollieren angeschlossene Netzwerk-Verbindungen und andere Netzressourcen, die mit einem bestimmten Knoten verbunden sind. PUs werden in Hosts mit SNA-Zugriffsmethoden implementiert, wie etwa der Virtual Telecommunication Access Method (VTAM). Auch werden PUs von Network Control Programs (NCPs) in Kommunikationscontroller implementiert.

Control Points (CPs) verwalten SNA-Knoten und deren Ressourcen. CPs unterscheiden sich gewöhnlich dadurch von PUs, daß sie entscheiden, welche Handlung durchgeführt werden muß, während PUs Handlungen bewirken. Ein Beispiel für ein CP ist der System Services Control Point (SSCP) der SNA. Ein SSCP kann der CP sein, der im PU-5-Knoten liegt, oder ein SSCP, wie er unter einer SNA-Zugriffsmethode wie VTAM implementiert ist.

27.2.5 IBM SNA-Knoten

Traditionelle SNA-Knoten gehören einer von zwei Kategorien an: *Subareaknoten* oder *Peripherieknoten*. SNA-Subareaknoten bieten alle Netzwerk-Dienste, einschließlich Knotenrouting und das Zuordnen von lokalen und netzwerkweiten Adressen. Es besteht keine Beziehung zwischen SNA-Knotentypen und tatsächlichen physischen Geräten. Zwei Subareaknoten sind von besonderem Interesse: *Knotentyp 4* und *Knotentyp 5*.

Knotentyp 4 (T4) liegt normalerweise in einem Kommunikationscontroller, etwa einem 3745. Ein Beispiel für einen T4-Knoten ist ein NCP, der Daten routet und den Fluß zwischen einem Front-End-Prozessor und anderen Netzressourcen kontrolliert.

Knotentyp 5 (T5) liegt gewöhnlich in einem Host, etwa einem S/370-Großrechner. Ein Beispiel für einen T5-Knoten ist die VTAM, die sich in einem IBM-Großrechner befindet. Eine VTAM kontrolliert den logischen Datenfluß durch das Netzwerk, stellt die Schnittstelle zwischen Anwendungssystemen und einem Netzwerk zur Verfügung und schützt die Anwendungssysteme vor nicht autorisiertem Zugriff.

SNA-Peripherieknoten verwenden nur lokale Adressierungen und kommunizieren über Subareaknoten mit anderen Knoten. Knotentyp 2 (T2) ist gewöhnlich der interessanteste Peripherieknoten, obwohl SNA auch einen Peripherieknoten vom Knotentyp 1 spezifiziert. T2 liegt im allgemeinen in intelligenten Terminals (wie einem 3270) oder in Verbindungsaufbaucontrollern (wie einem 3174). Knotentyp 1 (T1) ist heute veraltet, aber wenn er implementiert ist, liegt er in nicht intelligenten Terminals. Abbildung 27.4 zeigt die verschiedenen SNA-Knoten und ihre Beziehungen zueinander.

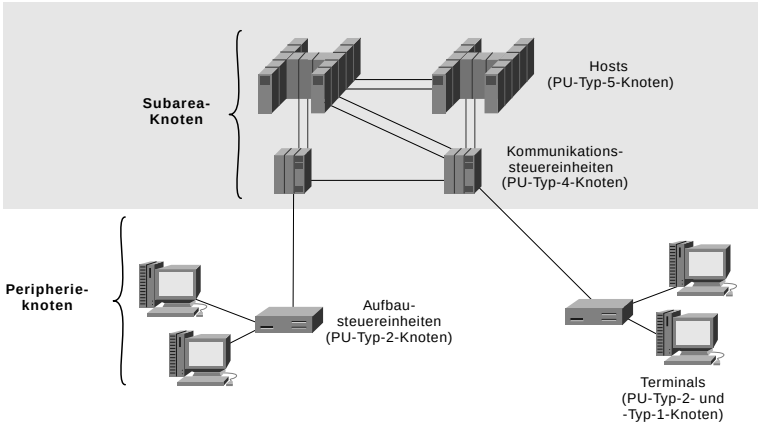


Bild 27.4: Peripherieknoten kommunizieren über Subarea-Knoten mit anderen Knoten

27.3 IBM Peer-to-Peer-Netzwerke

Wechselnde Anforderungen an Netzwerke und an die Kommunikation veranlaßten IBM, viele der Grundeigenschaften der SNA zu überprüfen und weiterzuentwickeln. Das Aufkommen von neuen Einheiten (wie etwa Router) für Peer-to-Peer-Netzwerke bewirkte eine Reihe von weitreichenden Veränderungen der SNA. Der Betrieb von Netzwerken mit gleichrangigen Geräten hängt innerhalb der SNA von verschiedenen Netzwerk-Komponenten ab, die von IBM entwickelt wurden.

Advanced Peer-to-Peer Networking (APPN) ist die zweite Generation von IBMs SNA. Mit der Entwicklung von APPN wandelte IBM die SNA von einer hierarchischen, großrechnerzentrierten Umgebung in eine Netzwerk-Umgebung um, die auf gleichrangigen Geräten beruht. Herzstück der APPN ist eine IBM-Architektur, die die Kommunikation zwischen gleichrangigen Geräten, Verzeichnisdienste und das Routing zwischen zwei oder mehr APPC-Systemen, die nicht direkt verbunden sind, unterstützt.

27.3.1 APPN-Komponenten

Außer der APPN-Umgebung spezifiziert diese Art der SNA drei zusätzliche wichtige Netzwerk-Konzepte: *Logical Units* (LUs), *Advanced Program-to-Program Computing* (APPC) und den *Knotentyp* 2.1. Jedes spielt eine bedeutende Rolle

beim Kommunikationsaufbau zwischen SNA-Geräten innerhalb eines SNA-basierten Verbundnetzes, das aus gleichrangigen Geräten besteht.

Logical Unit (LU) 6.2 regelt die Kommunikation zwischen gleichrangigen Geräten in einer SNA-Umgebung. Außerdem unterstützt LU 6.2 die allgemeine Kommunikation zwischen Programmen in einer verteilten Verarbeitungsumgebung und zwischen ähnlichen und unähnlichen Knotentypen. APPC versetzt SNA-Anwendungen in die Lage, direkt mit SNA-Anwendungen gleichrangiger Geräte zu kommunizieren, und es stellt eine Reihe von Programmiervereinbarungen und -protokollen zur Verfügung, die LU 6.2 implementieren. Knoten vom Typ 2.1 (T2.1) sind logische Entitäten, die die direkte Kommunikation zwischen Peripherieknoten, die T2.1 unterstützen, ermöglichen. Die T2.1-Entität erleichtert die Kommunikation über Punkt-zu-Punkt-Verbindungen, weil sie den Datentransport zwischen gleichrangigen Geräten unterstützt, der in der APPC vorgesehen ist. Außerdem enthält ein T2.1 einen Peripheral Node Control Point (PNCP), der die traditionellen Funktionen von Physical Unit (PU) und Control Point (CP) verbindet.

27.3.2 Knotenarten von IBM APPN

Unter APPN findet die Kommunikation zwischen gleichrangigen Geräten über verschiedene exakt definierte Knotentypen statt. Diese Knoten kann man in drei Grundtypen einteilen: *Low-Entry Nodes* (LENs), *End Nodes* (ENs) und *Network Nodes* (NNs).

Der Low-Entry-Network-Knoten (LEN) ist ein Peer-to-Peer-Knoten aus der Zeit vor APPN. Ein LEN-Knoten nimmt in einem APPN-Netz die Dienste eines benachbarten Network Node (NN) in Anspruch. Der CP des LEN-Knotens verwaltet lokale Ressourcen, baut mit dem benachbarten Knoten aber keine CP-zu-CP-Sitzung auf. Bevor eine Sitzung aufgebaut werden kann, müssen die Sitzungspartner für den LEN-Knoten definiert werden, und der LEN-Knoten muß für den benachbarten NN, der ihm Dienste zur Verfügung stellt, definiert werden.

Ein End Node (EN) enthält einen Teil der vollen APPN-Unterstützung. Ein End Node greift über einen benachbarten NN auf das Netz zu und benutzt die Routingdienste desselben benachbarten NN. Um im Netzwerk zu kommunizieren, baut ein EN eine CP-zu-CP-Sitzung mit einem benachbarten NN auf und benutzt diese Sitzung, um Ressourcen zu registrieren, Verzeichnisdienste und Routinginformationen anzufordern. Ein Network Node (NN) enthält die vollständige APPN-Funktionalität. Der CP in einem NN verwaltet die Ressourcen des NN wie auch der angeschlossenen ENs und LENs. Außerdem baut der CP in einem NN CP-zu-CP-Sitzungen mit benachbarten ENs und NNs auf und pflegt die Netzwerk-Topologie und die Verzeichnisdatenbanken, die erstellt und aktualisiert werden, indem Informationen benachbarter NNs und ENs dynamisch gesammelt werden.

Abbildung 27.5 zeigt, wo diese Gerätetypen in einer APPN-Umgebung liegen könnten.

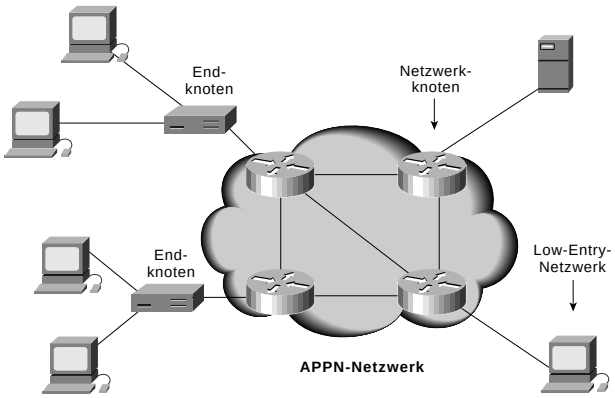


Bild 27.5: APPN unterstützt verschiedene genau definierte Knotenarten

27.3.3 IBM APPN-Dienste

Die grundlegenden APPN-Dienste lassen sich in vier Gruppen einteilen: Konfiguration, Verzeichnis, Topologie und Routing- und Verbindungsdienste. Jeder wird in den folgenden Abschnitten beschrieben.

Die Konfigurationsdienste von IBM APPN

Die Konfigurationsdienste der APPN sind dafür zuständig, Verbindungen zum APPN-Netzwerk zu aktivieren. Zur Aktivierung der Verbindung gehören der Aufbau einer Verbindung, der Aufbau einer Sitzung und die Auswahl einer Angrenzungsoption.

Die Anschlußphase einer Verbindungsaktivierung ermöglicht den erstmaligen Aufbau einer Kommunikation zwischen Knoten. Diese erstmalige Kommunikation beinhaltet den Austausch von Charakteristika und das Aufbauen von Rollen, wie etwa primär im Gegensatz zu sekundär. Der Verbindungsaufbau wird durch die Übermittlung von *Exchange Identification* Typ-3-Frames (XID3) zwischen Knoten erreicht.

Während des Sitzungsaufbaus werden CP-zu-CP-Sitzungen mit Hilfe eines benachbarten EN oder NN eingeleitet. Jeder Knoten muß mindestens ein Paar einer CP-zu-CP-Sitzung mit einem benachbarten Knoten aufbauen. Ein EN kann höchstens ein Paar einer CP-zu-CP-Sitzung aufbauen, kann aber mit mehr als einem NN verbunden werden. Zwischen NNs können Paare von CP-zu-CP-Sitzungen mit allen benachbarten Knoten oder einer Untermenge benachbarter Knoten aufgebaut werden. Die Mindestanforderung ist ein einzelnes Sitzungspaar zu einem benachbarten Knoten, der eine korrekte Aktualisierung der Topologie sichert.

Ob Knoten innerhalb der APPN benachbart sind, wird mit Hilfe einer CP-zu-CP-Sitzung entschieden. Es gibt zwei konfigurierbare Optionen, mit denen entschieden wird, ob Knoten benachbart sind. Ein Knoten kann als benachbart mit einem einzelnen Knoten spezifiziert werden oder als logisch benachbart mit jedem möglichen benachbarten Knoten. Welche Angrenzungsoption für eine bestimmte Situation ausgewählt wird, hängt von den gegebenen Verbindungsanforderungen des Netzwerks ab. Die Reduzierung von CP-zu-CP-Sitzungen, die einzelne benachbarte Knoten nach sich ziehen, kann den Netzwerk-Overhead, der durch Aktualisierungen der Topologie zustande kommt, wie auch die Anzahl der Puffer, die benötigt werden, um diese Aktualisierungen zu verteilen, reduzieren. Andererseits wird, wenn man die Anzahl benachbarter Knoten reduziert, mehr Zeit benötigt, um Router zu synchronisieren.

Die Verzeichnisdienste von IBM APPN

Verzeichnisdienste sollen Netzwerk-Geräten helfen, Dienstanbieter ausfindig zu machen. Sie sind zwingend nötig, um Sitzungen zwischen Endbenutzern aufzubauen. Die Verzeichnisdienste der APPN fordern jeden NN auf, ein Verzeichnis der lokalen Ressourcen sowie ein Netzwerk-Verzeichnis zu führen, das Endbenutzer mit den NNs verbindet, die Dienste zur Verfügung stellen. Dann wird aus der Gesamtheit der individuellen Netzwerk-Verzeichnisse der NNs ein verteilter Verzeichnisdienst zusammengestellt. Dieser Abschnitt erläutert die APPN-Datenbanken, die Verwaltung der Knotenverzeichnisdienste und die Rolle eines zentralen Verzeichnisdienstes.

Die lokalen und Netzwerk-Verzeichnis-Datenbanken unterstützen drei Arten von Diensteinträgen: *konfigurierte Einträge, registrierte Einträge* und *versteckte Einträge*. Konfigurierte Datenbankeinträge sind gewöhnlich lokale Netzwerk-Knoten mit niedrigen Einträgen, die konfiguriert werden müssen, weil keine CP-zu-CP-Sitzung, über die Informationen ausgetauscht werden, aufgebaut werden kann. Andere Knoten können konfiguriert werden, um den Rundspruchverkehr, der beim Erkennungsprozeß entsteht, zu reduzieren. Registrierte Einträge sind lokale Ressourceneinträge, über die ein Endknoten seinen entsprechenden Netzwerkknoten-Server informiert, wenn CP-zu-CP-Sitzungen aufgebaut werden. Ein NN fügt einen registrierten Eintrag in sein lokales Verzeichnis ein. Versteckte Datenbankeinträge sind Verzeichniseinträge, die als Sitzungsanforderungen gebildet und von einem NN empfangen werden. Der Benutzer kann die Gesamtzahl erlaubter versteckter Einträge vorgeben und somit die Speicheranforderungen regulieren.

Die Abwicklung des Endknoten-Verzeichnisdienstes umfaßt mehrere Schritte. Ein EN sendet zunächst eine LOCATE-Anforderung an den NN, der den Netzwerk-Dienst bereitstellt. Als nächstes werden die lokalen und Netzwerk-Verzeichnis-Datenbanken durchsucht, um festzustellen, ob der empfangende Endbenutzer bereits bekannt ist. Ist der empfangende Endbenutzer bekannt, wird eine einzelne LOCATE-Anforderung verschickt, um sicherzustellen, daß er gegenwärtig erreichbar ist. Wird der empfangende Endbenutzer in den existierenden Datenbanken nicht gefunden, sendet der NN eine LOCATE-

Anforderung an benachbarte ENs, um festzustellen, ob es sich bei dem empfangenden Endbenutzer um eine lokale Ressource handelt. Handelt es sich nicht um ein lokales Ziel, sendet der NN eine Rundspruch-LOCATE-Anforderung an alle benachbarten NNs zur Verbreitung über das Netzwerk. Wenn der NN, der Netzwerk-Dienste für den empfangenden Endbenutzer bereitstellt, die Ressource des Endbenutzers lokalisiert, wird eine Nachricht, die anzeigt, daß das Ziel gefunden wurde, an den Ausgangs-NN zurückgeschickt. Zum Schluß cachen Ausgangs- und Empfangs-NN die Information.

Verzeichnisdienste für LEN-Knoten werden von einem Proxydienstprozeß durchgeführt. Zuerst sendet ein LEN-Knoten eine Verbindungssitzung-Anforderung (BIND) für angeschlossene Ressourcen. Diese unterscheidet sich von den LOCATE-Anforderungen, die von ENs gesendet werden. Um einen Verzeichnisdienst zu erhalten, muß ein NN einen Proxydienst für einen LEN-Knoten leisten. Wenn ein Proxydienst-NN mit einem LEN-Knoten verbunden wird, sendet der NN eine Rundspruch-LOCATE-Anforderung, die der LEN-Knoten benötigt.

Bei ACF/VTAM gibt es gewöhnlich einen zentralen Verzeichnisdienst, der helfen soll, LOCATE-Rundsprüche zu reduzieren. Diese Art von Datenbank kann als zentrales Verzeichnis für ein gesamtes Netzwerk dienen, da sie konfigurierte, registrierte und versteckte Einträge enthält. Bei einem zentralen Verzeichnisdienst sendet ein NN einen LOCATE-Rundspruch an den zentralen Verzeichnisserver, der dann die zentrale Datenbank durchsucht und bei Bedarf einen Rundspruch sendet.

Topologie- und Routingdienste von IBM APPN

In einer APPN-Netzwerk-Topologie, werden Netzwerk-Knoten über *Transmission Groups (TGs)* verbunden. Jede TG besteht aus einer einzigen Verbindung, und alle NNs unterhalten eine Netzwerk-Topologie-Datenbank, in der alle NNs und TGs des Netzwerks verzeichnet sind. Transmission Groups werden in Kapitel 35, »IBM Systems Network Architecture (SNA) Routing« behandelt.

Die Topologiedatenbank eines Netzwerks wird mit den Informationen, die sie vom *Topology-Database Update (TDU)* erhält, aktualisiert. Die Nachrichten des TDU fließen bei CP-zu-

CP-Sitzungen immer dann, wenn Änderungen im Netzwerk auftreten, wenn etwa ein Knoten oder eine Verbindung aktiv oder inaktiv werden, wenn Überlastungen auftreten, oder wenn Ressourcen begrenzt sind.

Die Netzwerk-Topologie-Datenbank enthält Informationen, die zum Errechnen von Routen mit einer bestimmten *Class of Service (COS)* verwendet werden. Diese Informationen beinhalten Vernetzung und Zustand von NNs und TGs und NN- und TG-Eigenschaften, wie etwa die Kapazität von TGs.

Der APPN-Routingdienst verwendet Informationen aus Verzeichnis- und Topologiedatenbanken, um eine auf COS basierende Route festzulegen. Die Entscheidung über die Route beginnt, wenn ein Endknoten zum ersten Mal eine Sitzungsanforderung von einer logischen Einheit erhält. Eine LOCATE-Anforderung wird von dem EN an seinen NN geschickt, um Zielinformationen einzuholen und eine Route durch das Netzwerk zu erhalten. Der NN legt dann die Eigenschaften fest, die mit der angeforderten Dienststufe einhergehen. Die festgelegten Eigenschaften werden dann mit den Eigenschaften einer jeden TG und eines jeden NN im Netzwerk verglichen, und alle Routen, die die spezifizierten Kriterien erfüllen, werden als geeignet versteckt. Jeder EN, NN und TG im Verbundnetz wird eine Wertigkeit zugewiesen, die auf COS-Eigenschaften wie Kapazität, Kosten, Sicherheit und Verzögerung basiert. Diese Eigenschaften können auch benutzerdefiniert sein. Schließlich wird ein Pfad mit geringen Kosten ausgewählt, indem die Wertigkeiten summiert werden, die die Routingkriterien erfüllen.

Sitzungsdienste von IBM APPN

Auf die Routenerstellung folgt der APPN-Sitzungsaufbau, der je nach Knotentyp anders abläuft. Ist der Endbenutzer, von dem die Verbindung ausgeht, an einen EN angeschlossen, schickt der mit dem Ziel-EN benachbarte NN eine LOCATE-Antwort, die die Ortsbestimmung des Ziels und die Route enthält, an den Quell-EN zurück. Der Quell-EN schickt ein BIND an eine Sitzungsroute. Wenn die Verbindung von dem Endbenutzer ausgeht, ist er mit einem LEN-Knoten verbunden, der ein BIND an seinen benachbarten NN schickt. Der

benachbarte NN wandelt den LEN-BIND in ein APPN-BIND und schickt ein BIND an den Sitzungspfad.

Ein BIND ist ein spezifischer Typ von Anforderungsnachricht, die eine LU einer anderen LU sendet. Ein BIND enthält die Route, die für eine Sitzung verwendet wird. Es spezifiziert NNs und TGs, einen einzelnen Sitzungsidentifier für jede TG, die Übermittlungspriorität für die Sitzung und Fensterinformationen, die lernfähige Stufensteuerung unterstützen, um den Verkehr im Netzwerk zu begrenzen.

27.4 Das Format der Basic Information Unit (BIU)

IBM-SNA-NAUs benutzen Basic Information Units (BIUs), um Anforderungen und Antworten auszutauschen. Abbildung 27.6 zeigt das Format der BIU.

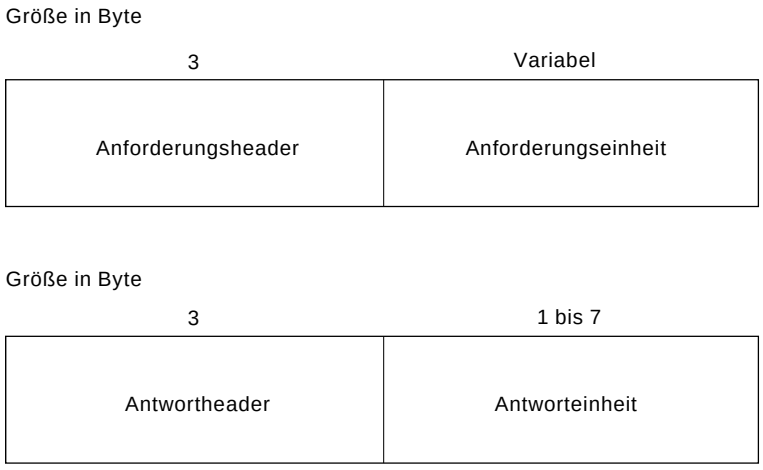


Bild 27.6: Eine Basic Information Unit (BIU) kann eine Anforderung oder eine Antwort sein

27.4.1 Die Felder der BIU

Die folgenden Beschreibungen der Felder fassen den Inhalt der BIU, wie in Abbildung 27.6 gezeigt, zusammen:

– *Anforderungsheader* – Kennzeichnet den Datentyp in der entsprechenden Anforderungseinheit. Dieser Header enthält Informationen über das Datenformat und spezifiziert

das Protokoll für die Sitzung. Informationen von Anforderungsheadern benutzen nur NAUs.

- *Anforderungseinheit* – Enthält entweder Daten des Endbenutzers oder SNA-Kommandos. Daten von Endbenutzern werden in Datenanforderungseinheiten versendet. SNA-Kommandos werden in Kommandoanforderungseinheiten versendet, die das Netzwerk kontrollieren und Informationen enthalten, die unter Endbenutzern ausgetauscht werden.
- *Antwortheader* – Kennzeichnet den Datentyp der entsprechenden Antworteinheit. Das Anforderung/Antwort-Indikatorbit unterscheidet einen Antwortheader von einem Anforderungsheader. Eine empfangende NAU zeigt an, ob die Antwort, die zum Sender der Anforderung zurückgeschickt wird, positiv oder negativ ist, wozu sie das *Response-Type Indicator*-Bit (RTI) in den Antwortheader setzt.
- *Antworteinheit* – Enthält Informationen über die Anforderung und zeigt entweder eine positive oder eine negative Antwort an. Positive Antworten auf Kommandoanforderungen enthalten gewöhnlich eine 1 bis 3 Byte große Antworteinheit, die die Kommandoanforderung identifiziert. Positive Antworten auf Datenanforderungen enthalten Antwortheader, aber keine Antworteinheit

  Negative Antworteinheiten sind zwischen 4 und 7 Byte lang und werden immer mit negativer Antwort zurückgeschickt. Eine empfangende NAU schickt eine negative Antwort immer an die anfordernde NAU zurück, wenn eine dieser drei Bedingungen gegeben ist:

  - Der Sender verletzt das SNA-Protokoll
  - Der Empfänger versteht die Übermittlung nicht
  - Etwas Unvorhergesehenes, zum Beispiel ein Pfadfehler, tritt ein

  Wird eine negative Antwort übermittelt, enthalten die ersten vier Byte einer Antworteinheit Daten, die erklären, warum die Anforderung inakzeptabel ist. Die empfangende NAU sendet bis zu drei zusätzliche Bytes, die die abgelehnte Anforderung identifizieren.

27.5 Das Format der Path Information Unit (PIU)

Die Path Information Unit (PIU) ist eine SNA-Nachrichteneinheit, die von Pfadkontrollelementen gebildet wird, indem einer BIU ein Übermittlungsheader hinzugefügt wird. Abbildung 27.7 zeigt das Format der PIU.

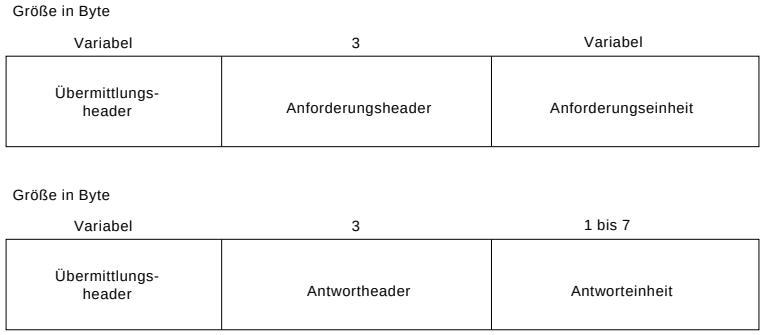


Bild 27.7: Die Anforderungen und Antworten der Path Information Unit (PIU) bestehen aus je drei Feldern

27.5.1 Die Felder der PIU

Die folgenden Felderbeschreibungen fassen den Inhalt der PIU, wie in Abbildung 27.7 gezeigt, zusammen:

– *Übermittlungsheader* – Routet Nachrichteneinheiten durch das Netzwerk. Dieser Header enthält Routinginformationen für die traditionelle SNA-Subareavernetzung. Formate der Übermittlungsheader werden durch den Typ der *Format Identification* (FID) unterschieden. Die Pfadkontrolle verwendet die FID-Typen, um Daten über SNA-Knoten zu routen.

  Drei FID-Typen sind in PIUs implementiert:

  – FID0 routet Daten zwischen benachbarten Subareaknoten für nicht-SNA-Geräte. FID0 wurde durch den FID4-Bitsatz abgelöst, der anzeigt, ob es sich um ein SNA-Gerät handelt oder nicht.

  – FID1 routet Daten zwischen benachbarten Subareaknoten, wenn einer oder beide Knoten explizite oder virtuelle Routingprotokolle nicht unterstützen.

  - FID2 routet Daten zwischen Subarea-Grenzknoten und einem benachbarten Peripherieknoten oder zwischen benachbarten Knoten vom Typ 2.1.

  Im allgemeinen wird der Übermittlungsheader verwendet, um Daten zwischen benachbarten Subareaknoten zu routen, wenn beide Subareaknoten explizite und virtuelle Routingprotokolle unterstützen.

- *Anforderungsheader* – Kennzeichnet den Datentyp in den entsprechenden Anforderungseinheiten. Dieser Header liefert Informationen über das Datenformat und spezifiziert das Sitzungsprotokoll. Informationen aus dem Anforderungsheader verwenden nur NAUs.

- *Anforderungseinheit* – Enthält entweder Daten der Endbenutzer oder SNA-Kommandos. Daten der Endbenutzer werden in Datenanforderungseinheiten versendet. SNA-Kommandos werden in Kommandoanforderungseinheiten versendet, die das Netzwerk kontrollieren und Informationen enthalten, die zwischen Endbenutzern ausgetauscht werden.

- *Antwortheader* – Kennzeichnet den Datentyp der entsprechenden Antworteinheit. Das Anforderung/Antwort-Indikatorbit unterscheidet einen Anforderungsheader von einem Antwortheader. Eine empfangende NAU zeigt an, ob die Antwort, die zum Anforderungssender zurückgeschickt wird, positiv oder negativ ist, indem sie das RTI-Bit in den Antwortheader setzt.

- *Antworteinheit* – Enthält Informationen über die Anforderung und zeigt entweder eine positive oder negative Antwort an. Positive Anworten auf Kommandoanforderungen enthalten gewöhnlich eine 1 bis 3 Byte lange Antworteinheit, die die Kommandoanforderung identifiziert. Positive Antworten auf Datenanforderungen enthalten Antwortheader, aber keine Antworteinheit.

  Negative Antworteinheiten sind 4 bis 7 Byte lang und werden immer mit negativer Antwort zurückgeschickt. Eine empfangende NAU schickt eine negative Antwort an die anfordernde NAU zurück, wenn eine von drei Bedingungen gegeben ist: Der Sender verletzt das SNA-Protokoll, ein

Empfänger versteht die Übermittlung nicht, oder etwas Unvorhergesehenes, ein Pfadfehler etwa, tritt ein.

Wird eine negative Antwort übermittelt, enthalten die ersten 4 Byte einer Antworteinheit Daten, die erklären, warum die Anforderung inakzeptabel ist. Die empfangende NAU sendet bis zu 3 zusätzliche Bytes, die die abgelehnte Anforderung identifizieren.

KAPITEL 28

Internet-Protokolle

28.1 Background

Die Internet-Protokolle sind die populärsten offenen (nicht proprietären) Protokollsuiten, da man mit ihnen über jegliche Art von verbundenen Netzwerken kommunizieren kann und sie sich somit gleichermaßen für LAN- und WAN-Kommunikation eignen. Die Internet-Protokolle bestehen aus einer Reihe von Kommunikationsprotokollen, von denen die beiden bekanntesten das Transmission Control Protocol (TCP) und das Internet Protocol (IP) sind. Die Internet-Protokollsuite enthält nicht nur Protokolle der unteren Schichten (wie TCP und IP), sondern sie spezifiziert auch gängige Anwendungen wie E-Mail, Terminalemulation und Dateiübertragung. Dieses Kapitel bietet eine detaillierte Einführung in die Spezifikationen, aus denen sich die Internet-Protokolle zusammensetzen. Es werden die IP-Adressierungen und die wichtigsten Protokolle der oberen Schichten besprochen, die das Internet verwendet. Die spezifischen Routingprotokolle werden im 6. Teil »Routingprotokolle« gesondert geschildert.

Internet-Protokolle wurden erstmalig Mitte der 70er Jahre entwickelt, als bei der Defense Advanced Research Projects Agency (DARPA) das Interesse an einem Paketvermittlungsnetz aufkam, das die Kommunikation zwischen unterschiedlichen Computersystemen an Forschungseinrichtungen erleichtern würde. Mit dem Ziel von heterogenen Verbindungen veranlaßte die DARPA die Forschung an der Stanford University und bei Bolt, Beranek und Newman (BBN). Ergebnis dieser

Bemühungen war die Internet-Protokollsuite, die in den späten 70ern fertiggestellt wurde.

Später wurde TCP/IP Berkeley Software Distribution (BSD)-UNIX beigefügt. Es wurde schließlich die Grundlage, auf der das Internet und das World Wide Web (WWW) basieren.

Die Dokumentation der Internet-Protokolle (einschließlich neuer oder überarbeiteter Protokolle) und ihre Grundsätze sind in technischen Berichten spezifiziert, die Request For Comments (RFCs) heißen. Sie werden publiziert und von der Internet-Gemeinde diskutiert und analysiert. Verbesserungen der Protokolle werden in den neuen RFCs veröffentlicht. Um den Anwendungsbereich der Internet-Protokolle zu veranschaulichen, stellt Bild 28.1 einige Protokolle aus der Internet-Protokollsuite ihren Entsprechungen der OSI-Schichten gegenüber. Dieses Kapitel befaßt sich mit den grundlegenden Elementen und Operationen dieser und anderer wichtiger Internet-Protokolle.

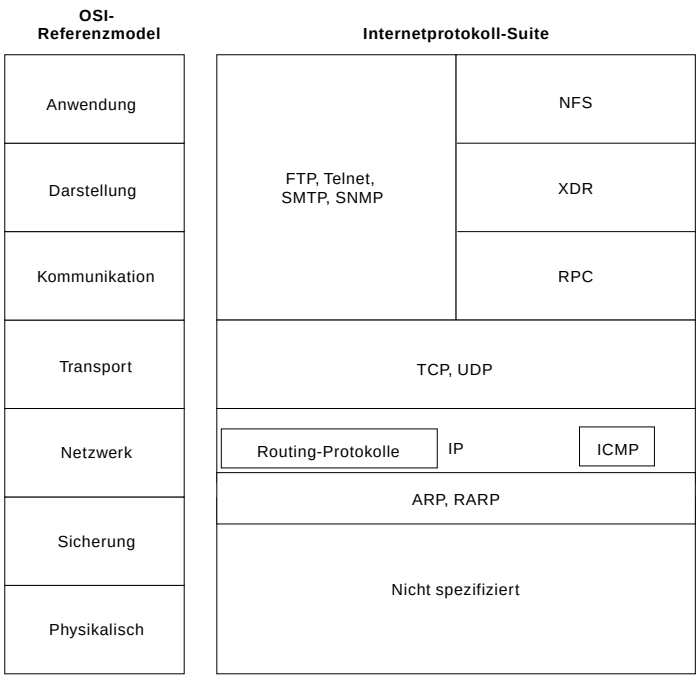


Bild 28.1: Internet-Protokolle decken die ganze Bandbreite des OSI-Schichtenmodells ab

28.2 Internet-Protokoll (IP)

Das Internet Protocol (IP) ist ein Netzwerkschichtprotokoll (Schicht 3), das Adressierungsinformationen sowie einige Kontrollinformationen enthält, mit denen Pakete geroutet werden können. IP ist in RFC 791 dokumentiert und stellt das primäre Netzwerkschichtprotokoll der Internet-Protokollsuite dar. Zusammen mit dem Transmission Control Protocol (TCP) bildet IP das Herzstück der Internet-Protokolle. IP hat zwei Hauptaufgaben: Es soll verbindungslose, effiziente Auslieferung von Datagrammen über ein Verbundnetz gewähren sowie die Fragmentierung und das Wiederzusammensetzen von Datagrammen erledigen, damit Datenleitungen mit verschiedenen Größen der Maximum Transmission Units (MTU) unterstützt werden können.

28.2.1 Das Format des IP-Pakets

Ein IP-Paket enthält verschiedene Typen von Informationen, die in Bild 28.2 gezeigt werden.

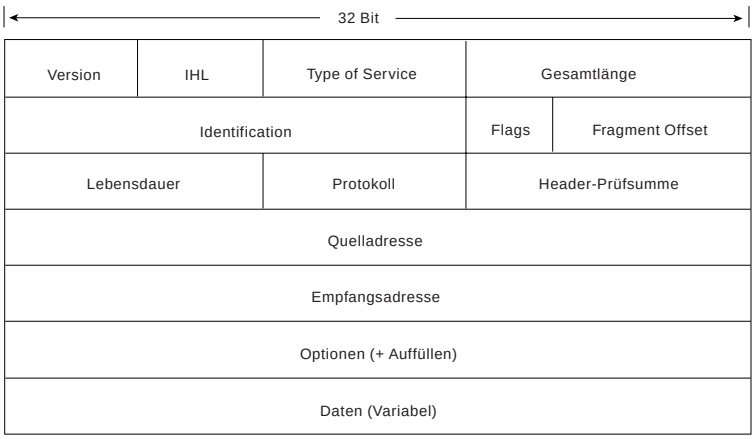


Bild 28.2: Ein IP-Paket umfaßt zehn Felder

Im folgenden werden die Felder des IP-Pakets, wie in Bild 28.2 gezeigt, erläutert:

- *Version* – Gibt die Version von IP an, die verwendet wird.
- *IP Header Length (IHL)* – Gibt die Länge des Datagrammheader in 32-Bit-Worten an.
- *Type-of-Service* – Spezifiziert, wie ein Protokoll der oberen Schichten ein Datagramm behandelt wissen will, und weist Datagrammen verschiedene Dringlichkeitsstufen zu.
- *Gesamtlänge* – Spezifiziert die Länge des gesamten IP-Pakets, einschließlich der Daten und Header, in Byte.
- *Identifizierung* – Enthält eine ganze Zahl, die das Datagramm identifiziert. Dieses Feld wird benutzt, um Datagrammfragmente zusammenzusetzen.
- *Flags (Kennzeichen)* – Besteht aus einem 3-Bit-Feld, von dem die beiden rechten Bits (die niederwertigsten) die Fragmentierung kontrollieren. Das rechte Bit legt fest, ob das Paket fragmentiert werden kann. Das mittlere Bit gibt an, ob es sich bei dem Paket um das letzte Fragment einer Reihe fragmentierter Pakete handelt. Das dritte oder höchstwertige Bit wird nicht verwendet.
- *Fragment Offset* – Gibt die Position der Daten des Fragments in bezug auf den Anfang der Daten im ursprünglichen Datagramm an, so daß es beim Empfänger möglich ist, das Ausgangsdatagramm wieder richtig zusammenzusetzen.
- *Lebensdauer* – Enthält einen Zähler, der herunterläuft, bis das Paket bei Null schließlich abgelegt wird. So wird verhindert, daß das Paket endlos herumkreist.
- *Protokoll* – Gibt an, welches Protokoll der oberen Schichten das eingehende Paket erhält, sobald die IP-Prozesse ausgeführt sind.
- *Headerprüfsumme* – Hilft, die Unversehrtheit des IP-Header sicherzustellen.

- *Quelladresse* – Spezifiziert die sendenden Knoten.
- *Empfangsadresse* – Spezifiziert den empfangenden Knoten.
- *Optionen* – Erlaubt IP, verschiedene Optionen wie etwa Sicherheit zu unterstützen.
- *Daten* – Enthält Informationen für obere Schichten.

28.2.2 IP-Adressierung

Wie bei jedem anderen Netzwerkschichtprotokoll, richtet sich auch das Schema der IP-Adressierung nach dem Prozeß, mit dem IP-Datagramme durch ein Verbundnetz geroutet werden. Jede IP-Adresse besteht aus spezifischen Komponenten und hat ein Grundformat. Diese IP-Adressen können unterteilt und dazu verwendet werden, Adressen für Subnetze zu erstellen, was später in diesem Kapitel noch ausführlicher beschrieben wird.

Jedem Host in einem TCP/IP-Netz wird eine eindeutige, logische 32-Bit-Adresse zugewiesen, die aus zwei Hauptteilen besteht: der Netzwerk-Nummer und der Host-Nmmer. Die Netzwerk-Nummer bezeichnet das Netzwerk und muß vom Internet Network Information Center (InterNIC) zugewiesen werden, sofern das Netzwerk ein Teil des Internet ist. Ein Internet Service Provider (ISP) kann Blöcke von Netzwerk-Adressen vom InterNIC erwerben und diesen Adreßbereich nach Bedarf zuweisen. Die Rechnernummer kennzeichnet einen Rechner in einem Netzwerk und wird vom lokalen Netzwerk-Administrator zugewiesen.

IP-Adreßformat

Die 32-Bit-IP-Adresse wird in Gruppen zu 8 Bits unterteilt, durch Punkte getrennt und im Dezimalformat (der sogenannten *Punktnotation*) dargestellt. Jedes Bit in dem Oktett hat eine binäre Wertigkeit (128, 64, 32, 16, 8, 4, 2, 1). Der minimale Wert für ein Oktett beträgt 0, der maximale Wert 255. Bild 28.3 zeigt das Grundformat einer IP-Adresse.

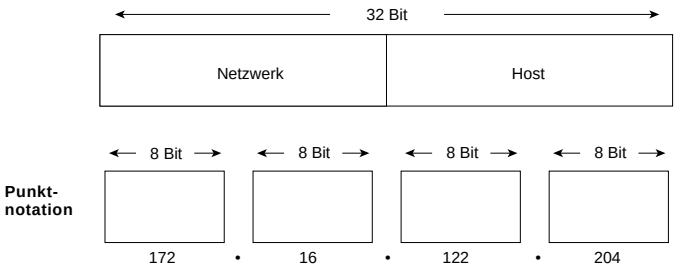


Bild 28.3: Eine IP-Adresse besteht aus 32 Bits, die in vier Oktette unterteilt sind

28.2.3 IP-Adreßklassen

Die IP-Adressierung unterstützt fünf verschiedene Adreßklassen: A, B, C, D und E. Nur die Klassen A, B und C sind für den kommerziellen Gebrauch verfügbar. Die linken (höchstwertigen) Bits geben die Netzwerk-Klasse an. Tabelle 28.1 informiert über die fünf IP-Adreßklassen:

IP-Adreß-klasse	**Format**	**Zweck**	**Höchst-wertige(s) Bit**	**Adreßbereich**	**Anzahl der Bit für Netzwerk/ Host**	**Max. Hosts**
A	N.H.H.H	Wenige Großunter-nehmen	0	1.0.0.0 bis 126.0.0.0	7/24	16777214[2] ($2^{24} - 2$)
B	N.N.H.H	Mittelgroße Organisa-tionen	1, 0	128.1.0.0 bis 191.254.0.0	14/16	65543 ($2^{16} - 2$)
C	N.N.N.H	Relativ kleine Or-ganisatio-nen	1, 0, 0	192.0.1.0 bis 223.255.254.0	22/8	254 ($2^{8} - 2$)
D	N/A	Multicast-Gruppen	1, 1, 1, 0	224.0.0.0 bis 239.255.255.255	N/A (nicht für kom-merzielle Zwecke)	N/A
E	N/A	Experimen-tell	1, 1, 1, 1	240.0.0.0 bis 254.255.255.255	N/A	N/A

Tabelle 28.1: Die wichtigsten Informationen über die fünf IP-Adreßklassen

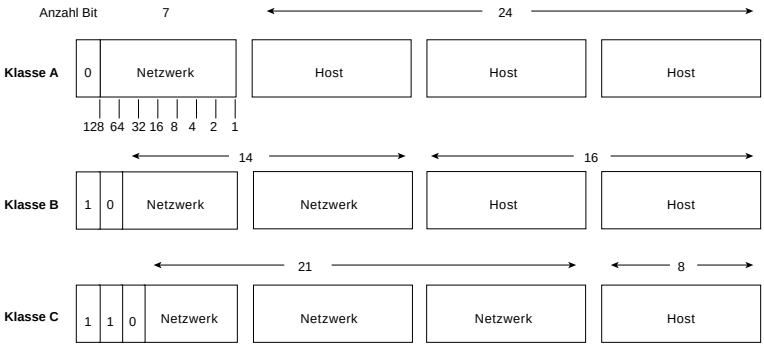


Bild 28.4: Die IP-Adreßformate A, B, und C sind für kommerzielle Zwecke verfügbar

Bild 28.4 zeigt das Format der kommerziellen IP-Adreßklassen. (Beachten Sie das linke Bit der jeweiligen Klasse.)

Die Adreßklasse kann leicht festgestellt werden, wenn man das erste Oktett der Adresse betrachtet und den Wert einem Klassenbereich der folgenden Übersicht zuordnet. In der IP-Adresse 172.31.1.2 zum Beispiel ist das erste Oktett 172. Da 172 zwischen 128 und 191 liegt, ist 172.31.1.2 eine Adresse der Klasse B. Bild 28.5 faßt den Bereich möglicher Werte für das erste Oktett einer jeden Adreßklasse zusammen.

Adreß-klasse	erstes Oktett in Dezimalnotation	höchstwertiges Bis
Klasse A	1 – 126	0
Klasse B	128 – 191	10
Klasse C	192 – 223	110
Klasse D	224 – 239	1110
Klasse E	240 – 254	1111

Bild 28.5: Für das erste Oktett einer Adreßklasse gibt es einen bestimmten Wertebereich

IP Subnet-Adressierung

IP-Netzwerke können in kleinere Netzwerke unterteilt werden, die Subnetzwerke oder Subnetze heißen. Die Einteilung in Subnetze hat für den Administrator diverse Vorteile, wie mehr Flexibilität, effizienterer Gebrauch von Netzwerk-Adressen

und die Möglichkeit, Rundspruchverkehr zu verwenden (ein Rundspruch durchläuft keinen Router).

Subnetze werden lokal verwaltet. So sieht die Außenwelt ein Unternehmen als einzelnes Netzwerk und bekommt keine genauen Einblicke in die interne Unternehmensstruktur.

Jede Netzwerk-Adresse kann auf viele Subnetze verteilt werden. Zum Beispiel sind 172.16.1.0, 172.16.2.0, 172.16.3.0 und 172.16.4.0 allesamt Subnetze im Netzwerk 171.16.0.0. (Jede 0 im Hostteil einer Adresse spezifiziert das gesamte Netzwerk.)

IP-Subnetzmasken

Eine Subnetzadresse wird gebildet, indem Bits aus dem Hostfeld »geliehen« und dem Subnetzfeld zugeschrieben werden. Die Anzahl geliehener Bits variiert und wird von der Subnetzmaske spezifiziert. Bild 28.6 zeigt, wie Bits aus dem Hostadreßfeld geliehen werden, um ein Subnetzadreßfeld zu erstellen.

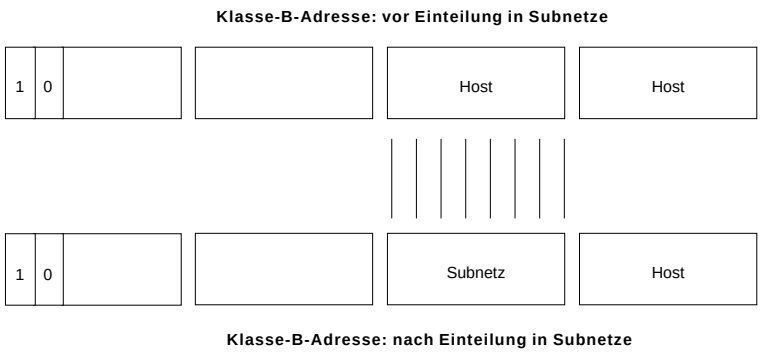


Bild 28.6: Um das Subnetzadreßfeld zu erstellen, werden dem Host-Adressenfeld Bits entliehen

Subnetzmasken verwenden dasselbe Format und dieselbe Darstellungstechnik wie IP-Adressen. Die Subnetzmasken enthalten eine binäre 1 in jedem Bit, das das Netzwerk spezifiziert, und eine binäre 0 in jedem Bit, das das Host-Feld spezifiziert. Bild 28.7 zeigt ein Beispiel für eine Subnetzmaske.

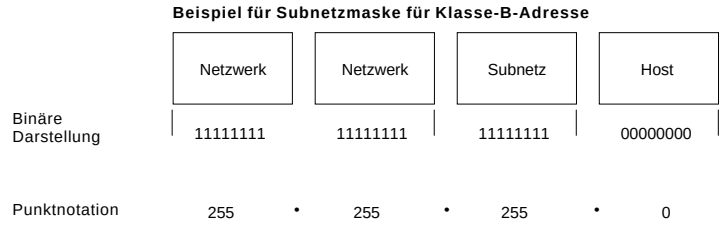


Bild 28.7: Eine Subnetzmaske besteht aus binären Nullen und Einsen

Die Bits der Subnetzmasken sollten den höchstwertigen (linken) Bits des Host-Feldes entstammen, wie in Bild 28.8 gezeigt. Einzelheiten zu Subnetzmasken der Klassen B und C folgen. Adressen der Klasse A werden in diesem Kapitel nicht besprochen, weil sie im allgemeinen in Subnetzen auf 8 Bit begrenzt sind.

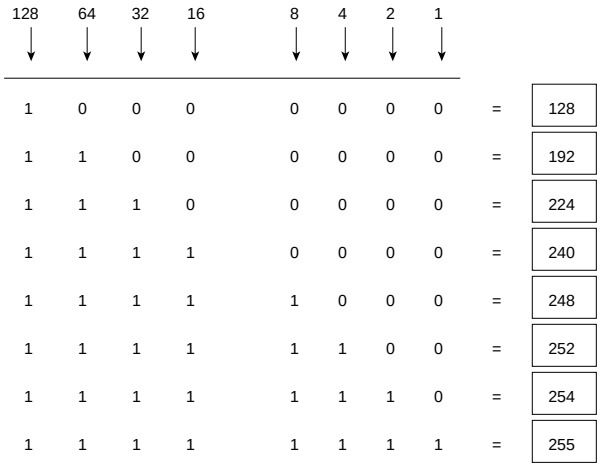


Bild 28.8: Die Bits der Subnetzmaske entstammen den höchstwertigen Bits der Host-Felder

Es gibt verschiedene Typen von Subnetzmasken für Subnetze der Klassen B und C.

Die Standardsubnetzmaske für eine Adresse der Klasse B, die kein Subnetz hat, ist 255.255.0.0, während die Subnetzmaske für die Klasse-B-Adresse 171.16.0.0, die 8 Bits eines Subnetzes spezifiziert, 255.255.255.0 ist. Das liegt daran, daß 8 Bits für den Aufbau eines Subnetzes oder 2^8–2 (1 für die Netzwerk-Adresse und 1 für die Rundspruchadresse) = 254 mögliche Subnetze mit 2^8–2 = 254 Hosts pro Subnetz bedeuten.

Die Subnetzmaske für eine Adresse der Klasse C 192.168.2.0, die fünf Bits für das Subnetz spezifiziert, ist 255.255.255.248. Mit fünf verfügbaren Bits sind 2^5–2 = 30 Subnetze mit 2^3–2 = 6 Hosts pro Subnetz möglich.

Die Charts in Tabelle 28.2 und Tabelle 28.3 können bei der Planung von Klasse-B- und Klasse-C-Netzen verwendet werden, um die erforderliche Anzahl von Subnetzen und Hosts und die geeignete Subnetzmaske herauszufinden.

Tabelle 28.2: Chart der Subnetze der Klasse B

Anzahl der Bits	Subnetzmaske	Anzahl der Subnetze	Anzahl der Hosts
2	255.255.192.0	2	16382
3	255.255.224.0	6	8190
4	255.255.240.0	14	4094
5	255.255.248.0	30	2046
6	255.255.252.0	62	1022
7	255.255.254.0	126	510
8	255.255.255.0	254	254
9	255.255.255.128	510	126
10	255.255.255.192	1022	62
11	255.255.255.224	2046	30
12	255.255.255.240	4094	14
13	255.255.255.248	8190	6
14	255.255.255.252	16382	2

Tabelle 28.3: Chart für Subnetze der Klasse C

Anzahl der Bits	Subnetzmaske	Anzahl der Subnetze	Anzahl der Hosts
2	255.255.255.192	2	62
3	255.255.255.224	6	30
4	255.255.255.240	14	14
5	255.255.255.248	30	6
6	255.255.255.252	62	2

Wie werden Subnetzmasken verwendet, um die Netzwerk-Nummer festzustellen?

Der Router führt einen bestimmten Prozeß durch, um die Netzwerk-Adresse, oder genauer, die Subnetzadresse festzustellen. Als erstes entnimmt der Router dem eingehenden Paket die IP-Zieladresse und findet die interne Subnetzmaske heraus. Dann führt er eine logische *AND*-Operation durch, um die Netzwerk-Nummer zu erhalten. Das bewirkt, daß der Host-Teil der IP-Zieladresse herausgenommen wird, während die Zielnetzwerk-Nummer bestehen bleibt. Dann schlägt der Router die Zielnetzwerk-Nummer nach und verknüpft sie mit einer abgehenden Schnittstelle. Schließlich leitet er den Frame an die IP-Zieladresse. Die Besonderheiten der logischen AND-Operationen werden in den folgenden Abschnitten besprochen.

Logische AND-Operationen

Drei Grundregeln bestimmen die AND-Verknüpfung zwischen zwei Binärzahlen. Erstens: Eine AND-Verknüpfung zwischen 1 und 1 ergibt 1. Zweitens: Eine AND-Verknüpfung zwischen 1 und 0 ergibt 0. Drittens: Eine AND-Verknüpfung zwischen 0 und 0 ergibt 0. Tabelle 28.4 veranschaulicht die Regeln für logische AND-Operationen.

Input	Input	Output
1	1	1
1	0	0
0	1	0
0	0	0

Tabelle 28.4: Regeln für logische AND-Operatoren

Es gibt zwei einfache Merksätze für logische AND-Operationen: AND-Verknüpfungen zwischen 1 und 1 ergeben den Ausgangswert, und logische AND-Verknüpfungen von 0 mit irgendeiner anderen Zahl ergeben 0.

Bild 28.9 zeigt, was passiert, wenn die IP-Zieladresse und die Subnetzmaske mit AND verknüpft werden: Die Subnetznummer, die der Router verwendet, um das Paket weiterzuleiten, bleibt bestehen.

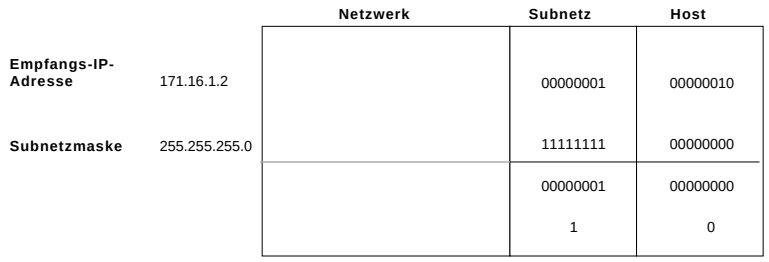


Bild 28.9: Bei einer AND-Verknüpfung zwischen der IP-Zieladresse und der Subnetzmaske ergibt sich die Subnetzwerk-Nummer

28.3 Address-Resolution Protocol (ARP) im Überblick

Wenn zwei Maschinen in einem Netzwerk kommunizieren sollen, müssen sie die physischen oder MAC-Adressen der anderen Maschine kennen. Durch einen Rundspruch der Address-Resolution-Protokolle (ARPs) kann ein Host die MAC-Schichtenadressen, die einer bestimmten IP-Netzwerkschichtadresse entsprechen, herausfinden.

Nach dem Erhalt einer MAC-Schichtenadresse erstellen IP-Geräte einen ARP-Cache, um kürzlich empfangene Verknüpfungen zwischen IP- und MAC-Adressen zu speichern, wodurch sie einen Rundspruch von ARPs vermeiden, wenn sie erneut mit einem Gerät Kontakt aufnehmen wollen. Antwortet das Gerät nicht innerhalb einer vorgegebenen Zeit, wird der Cache-Eintrag versenkt.

Zusätzlich wird das Reverse Address-Resolution Protocol (RARP) verwendet, um MAC-Schichtenadressen IP-Adressen zuzuordnen. RARP, die logische Umkehrung von ARP, kann von plattenlosen Workstations verwendet werden, die ihre IP-Adressen beim Booten nicht kennen. RARP wird von einem RARP-Server unterstützt, der Tabellen der Verknüpfungen von MAC-Schichten- und IP-Adressen enthält.

28.4 Internet-Routing

Geräte für das Internet-Routing wurden traditionell Gateways genannt. In der heutigen Terminologie bezieht sich der Begriff Gateway besonders auf ein Gerät, das Übersetzungen von

Anwendungsschichtprotokollen zwischen Geräten durchführt. Interne Gateways sind Geräte, die diese Protokollfunktionen zwischen Maschinen und Netzwerken durchführen, die von der gleichen Stelle verwaltet werden, wie es im internen Unternehmensnetzwerk der Fall ist. Hier spricht man von autonomen Systemen. Externe Gateways führen Protokollfunktionen zwischen unabhängigen Netzwerken aus.

Router im Internet sind hierarchisch organisiert. Router, die für den Informationsaustausch in autonomen Systemen verwendet werden, nennt man interne Router. Sie benutzen eine Reihe von Interior Gateway Protocols (IGPs), um ihren Zweck zu erfüllen. Ein Beispiel für ein IGP ist das Routing Information Protocol (RIP).

Router, die Informationen zwischen autonomen Systemen bewegen, nennt man externe Router Sie verwenden externe Gateway-Protokolle, um Informationen zwischen autonomen Systemen auszutauschen. Ein Beispiel für ein externes Gateway-Protokoll ist das Border Gateway Protocol (BGP).

Bestimmte Routingprotokolle, einschließlich BGP und RIP, werden in eigenen Kapiteln im sechsten Teil dieses Buches vorgestellt.

28.4.1 IP-Routing

IP-Routingprotokolle arbeiten dynamisch. Dynamisches Routing verlangt, daß die Software in Routinggeräten Leitwege in regelmäßigen Abständen automatisch berechnet. Dies unterscheidet sich vom statischen Routing, bei dem Router vom Netzwerk-Administrator eingerichtet werden und sich so lange nicht verändern, bis der Administrator dies tut.

Eine IP-Routingtabelle, die aus Paaren von Zieladresse und der nächsten Etappe besteht, wird verwendet, um das dynamische Routing zu ermöglichen. Ein Eintrag in dieser Tabelle würde zum Beispiel wie folgt interpretiert: Um zum Netzwerk 172.31.0.0 zu gelangen, sende das Paket über die Ethernetschnittstelle 0 (E0) aus.

Das IP-Routing legt fest, daß Daten ein Verbundnetz etappenweise durchqueren. Beim Beginn der Reise ist nicht die

ganze Route bekannt. Statt dessen wird bei jedem Halt das nächste Ziel berechnet, indem die Zieladresse im Datagramm mit dem Eintrag in der Routingtabelle des aktuellen Knotens abgeglichen wird.

Der Einfluß des Knotens auf den Routingprozeß beschränkt sich darauf, daß er das Paket aufgrund seiner internen Informationen weiterleitet. Der Knoten überwacht nicht, ob das Paket an sein Ziel gelangt, und IP sendet keine Fehlermeldung an den Ausgangspunkt zurück, wenn beim Routing etwas Ungewöhnliches auftritt. Diese Aufgabe bleibt einem anderen Internet-Protokoll vorbehalten: dem Internet Control-Message Protocol (ICMP), das im folgenden Abschnitt besprochen wird.

28.5 Internet Control-Message Protocol (ICMP)

Das *Internet Control-Message Protocol (ICMP)* ist ein Netzwerkschicht-Internet-Protokoll, das es Nachrichtenpaketen ermöglicht, dem Ausgangspunkt Fehler und andere Informationen, die das IP-Paket betreffen, zu melden. ICMP ist in RFC 792 dokumentiert.

28.5.1 ICMP-Nachrichten

ICMPs erstellen verschiedene Arten nützlicher Nachrichten, wie zum Beispiel »Ziel nicht erreichbar«, Echoanforderung und -antwort, »Umleiten«, »Zeit abgelaufen« und Routerbekanntgabe und -anfrage. Kann eine ICMP-Nachricht nicht ausgeliefert werden, wird keine zweite erstellt. Dadurch wird ein endloser Fluß von ICMP-Nachrichten vermieden.

Wenn ein Router eine ICMP-Ziel nicht erreichbar«-Nachricht sendet, bedeutet das, daß der Router nicht in der Lage ist, das Paket an sein Ziel zu schicken. Der Router legt das Originalpaket dann ab. Es gibt zwei Möglichkeiten, warum ein Ziel nicht erreichbar ist. Meistens hat der Quellhost eine nicht existierende Adresse angegeben. Manchmal hat aber auch der Router keinen Leitweg zum Ziel.

Es gibt vier Arten von »Ziel nicht erreichbar«-Nachrichten: »Netzwerk nicht erreichbar«, »Host nicht erreichbar«,

»Protokoll nicht erreichbar« und »Port nicht erreichbar«. *»Netzwerk nicht erreichbar«* bedeutet zumeist, daß ein Fehler beim Routing oder der Adressierung eines Pakets aufgetreten ist. *»Host nicht erreichbar«* weist gewöhnlich auf einen Fehler bei der Auslieferung hin, wie etwa bei einer falschen Subnetzmaske. *»Protokoll nicht erreichbar«* geht im allgemeinen darauf zurück, daß das Ziel das Protokoll der oberen Schichten, das im Paket spezifiziert wurde, nicht unterstützt. *»Port nicht erreichbar«* weist darauf hin, daß der TCP-Socket oder -Port nicht verfügbar ist.

Eine ICMP-Echoanforderung, die durch das Ping-Kommando erstellt wird, wird von jedem Rechner geschickt, um die Erreichbarkeit von Knoten im Verbundnetz zu prüfen. Die ICMP-Echoantwortnachricht zeigt an, daß der Knoten erreicht werden kann.

Eine ICMP-»Umleiten«-Nachricht wird von einem Router zum Quellhost geschickt, um ein effizienteres Routen zu bewirken. Dennoch leitet der Router das Originalpaket an das Ziel weiter. Durch das Umleiten von ICMP bleiben Routingtabellen klein, weil es genügt, die Adresse von nur einem Router zu kennen, auch wenn dieser Router nicht den besten Pfad bietet. Auch nach dem Erhalt einer ICMP-»Umleiten«-Nachricht benutzen manche Geräte weiterhin weniger effiziente Routen.

Eine ICMP-»Zeit abgelaufen«-Nachricht wird von dem Router geschickt, wenn das Lebensdauerfeld (angegeben in Hops oder Sekunden) eines IP-Pakets Null erreicht. Das Lebensdauerfeld verhindert, daß Pakete immer weiter durch das Verbundnetz kreisen, wenn das Verbundnetz eine Routingschleife enthält.

28.5.2 ICMP Router-Discovery Protocol (IDRP)

IDRP verwendet Routerbekanntgabe- und Routeranfrage-Nachrichten, um die Adressen von Routern in direkt verbundenen Subnetzen zu erfahren. Jeder Router sendet von jeder seiner Schnittstellen in regelmäßigen Abständen Routerbekanntgaben. Die Hosts entdecken dann die Routeradressen von den angeschlossenen Subnetzen, indem sie diese Nachrichten abhören. Hosts können Routeranfragen benutzen, um Be-

kanngaben unmittelbar abzufragen, anstatt auf eine nicht angeforderte Nachricht zu warten.

Gegenüber anderen Methoden, Adressen benachbarter Router zu finden, bietet IRDP verschiedene Vorteile. Vor allem benötigt es keine Hosts, um Routingprotokolle zu erkennen, und es muß auch nicht manuell vom Netzwerk-Administrator konfiguriert werden.

Routerbekanntgabe-Nachrichten ermöglichen es Hosts zwar, benachbarte Router zu finden, sagen aber nichts darüber aus, welcher Router sich am besten eignet, ein bestimmtes Ziel zu erreichen. Wenn ein Host einen einfachen First-Hop-Router verwendet, um ein Ziel zu erreichen, erhält er eine »Umleiten«-Nachricht, die eine bessere Möglichkeit aufzeigt.

28.6 Transmission-Control Protocol (TCP)

TCP bietet die zuverlässige Datenübermittlung in einer IP-Umgebung. TCP entspricht der Transportschicht (Schicht 4) des OSI-Basisreferenzmodells. Zu den Diensten von TCP gehören Datenstromtransfer, Verläßlichkeit, effiziente Ablaufsteuerung, Vollduplexbetrieb und Multiplexing.

Mit dem Datenstromtransfer liefert TCP einen unstrukturierten Strom von Bytes, die durch Sequenznummern identifiziert werden. Dieser Dienst kommt Anwendungen zugute, weil sie Daten nicht in Blöcke zerlegen müssen, bevor sie diese an TCP weiterreichen. Statt dessen gruppiert TCP die Bytes in Segmente und übergibt sie zur Auslieferung an IP.

TCP bietet Zuverlässigkeit durch eine verbindungsorientierte Paketauslieferung zwischen Endpunkten in einem Verbundnetz. Dabei werden Bytes mit einer Bestätigungsnummer versehen, die dem Ziel das nächste Byte anzeigen, das die Quelle erwartet. Nicht bestätigte Bytes werden innerhalb eines gewissen Zeitraums erneut übermittelt. Der Zuverlässigkeitsmechanismus von TCP erleichtert Geräten den Umgang mit verlorenen, verspäteten, doppelten oder falsch gelesenen Paketen. Über einen Zeitschaltmechanismus können Geräte verlorene Pakete aufspüren und eine erneute Übermittlung anfordern.

TCP bietet effiziente Ablaufsteuerung. Das heißt, wenn die Bestätigungen an die Quelle zurückgeschickt werden, gibt der

empfangende TCP-Prozeß die höchste Sequenznummer an, die er empfangen kann, ohne daß seine internen Puffer überlaufen.

Vollduplex-Betrieb bedeutet, daß TCP-Prozesse gleichzeitig senden und empfangen können.

Multiplexing bedeutet schließlich, daß bei TCP gleichzeitig zahlreiche Unterhaltungen in den oberen Schichten über eine einzige Verbindung ablaufen können.

28.6.1 TCP-Verbindungsaufbau

Um einen zuverlässigen Transportdienst zu nutzen, müssen TCP-Hosts miteinander eine verbindungsorientierte Sitzung aufbauen. Die Verbindung wird durch den »three-way handshake« aufgebaut.

Ein three-way handshake synchronisiert die beiden Enden einer Verbindung, wobei beide Seiten sich auf eine Anfangssequenznummer einigen können. Dieser Mechanismus garantiert auch, daß beide Seiten bereit sind, Daten zu übermitteln, und wissen, daß die andere Seite ebenfalls dazu bereit ist. Dies ist notwendig, damit keine Pakete während des Sitzungsaufbaus oder nach der Beendigung der Sitzung übermittelt werden. Jeder Host wählt eine Sequenznummer nach dem Zufallsprinzip aus, die dazu verwendet wird, Bytes in dem Strom, die er sendet und empfängt, aufzuspüren. Dann wird der three-way handshake auf die folgende Weise durchgeführt:

Der erste Host (Host A) leitet eine Verbindung ein, indem er ein Paket mit der Anfangssequenznummer (X) und einem SYN-Bitsatz, der die Anforderung darstellt, sendet. Der zweite Host (Host B) erhält die SYN, stellt die Sequenznummer X fest und antwortet mit einer Bestätigung der SYN (mit ACK = X + 1). Host B fügt seine eigene Anfangssequenznummer hinzu (SEQ = Y). Ein ACK = 20 bedeutet, daß der Host die Bytes 0 bis 19 empfangen hat und als nächstes Byte 20 erwartet. Diese Technik wird *Forward Acknowledgment* genannt. Host A bestätigt alle Bytes, die Host B mit einer Übermittlungsbestätigung geschickt hat, die das nächste Byte zeigt, das Host A erwartet (ACK = Y + 1).

Dann kann der Datentransfer beginnen.

28.6.2 Positive Acknowledgment and Retransmission (PAR)

Ein einfaches Transportprotokoll könnte Zuverlässigkeits- und Ablaufsteuerungstechniken realisieren, bei denen die Quelle ein Paket sendet, einen Timer startet und auf eine Bestätigung wartet, bevor ein neues Paket abgeschickt wird. Geht keine Bestätigung ein, bevor der Timer abläuft, sendet die Quelle das Paket erneut. Solch eine Technik wird *Positive Acknowledgment And Retransmission* genannt.

Indem PAR jedem Paket eine Sequenznummer zuordnet, versetzt es Hosts in die Lage, verlorene oder doppelte Pakete aufzuspüren, welche durch Netzwerk-Verzögerungen verursacht wurden, die zu einer verfrühten Neuübermittlung führen. Die Sequenznummern werden in den Bestätigungen, die dann überprüft werden können, zurückgeschickt.

PAR gebraucht die Bandbreite wenig effizient, denn ein Host muß hier auf eine Bestätigung warten, bevor er ein neues Paket sendet, und außerdem kann nur ein Paket auf einmal gesendet werden.

28.6.3 TCP Sliding Window

Ein *TCP Sliding Window* bietet einen effizienteren Gebrauch der Netzwerk-Bandbreite als PAR, weil es Hosts in die Lage versetzt, viele Bytes oder Pakete zu senden, ohne vorher auf eine Bestätigung zu warten.

Bei TCP gibt der Empfänger die aktuelle Fenstergröße eines jeden Pakets an. Da TCP eine Bytestromverbindung zur Verfügung stellt, werden Fenstergrößen in Byte angegeben. Das heißt, ein Fenster ist die Anzahl von Datenbytes, die der Sender verschicken darf, bevor er auf eine Bestätigung warten muß. Die anfänglichen Fenstergrößen werden beim Verbindungsaufbau angezeigt, doch können sie während des Datentransfers variieren, um die Ablaufsteuerung zu gewährleisten. Die Fenstergröße Null bedeutet zum Beispiel: »Sende keine Daten«.

Bei einem Sliding-Window-Vorgang in TCP kann der Sender zum Beispiel eine Bytesequenz (von 1 bis 10 numeriert) an einen Empfänger verschicken, der die Fenstergröße 5 hat. Der

Sender würde dann ein Fenster um die ersten 5 Bytes erstellen und sie zusammen verschicken. Dann würde er auf eine Bestätigung warten.

Der Empfänger würde mit ACK = 6 antworten und damit anzeigen, daß er die Bytes 1 bis 5 erhalten hat und als nächstes Byte 6 erwartet. Im selben Paket würde der Empfänger anzeigen, daß seine Fenstergröße 5 beträgt. Dann würde der Sender sein Sliding Window fünf Bytes nach rechts versetzen und die Bytes 6 bis 10 übermitteln. Der Empfänger antwortet in diesem Fall mit ACK = 11, was bedeutet, daß er als nächstes Byte 11 erwartet. In diesem Paket könnte der Empfänger auch mitteilen, daß seine Fenstergröße 0 beträgt (zum Beispiel, weil seine internen Puffer voll sind). In diesem Fall kann der Sender keine Bytes mehr verschicken, bis der Empfänger ein anderes Paket mit einer Fenstergröße über 0 übermittelt.

28.6.4 TCP-Paketformat

Bild 28.10 zeigt die Felder und das Format eines TCP-Pakets.

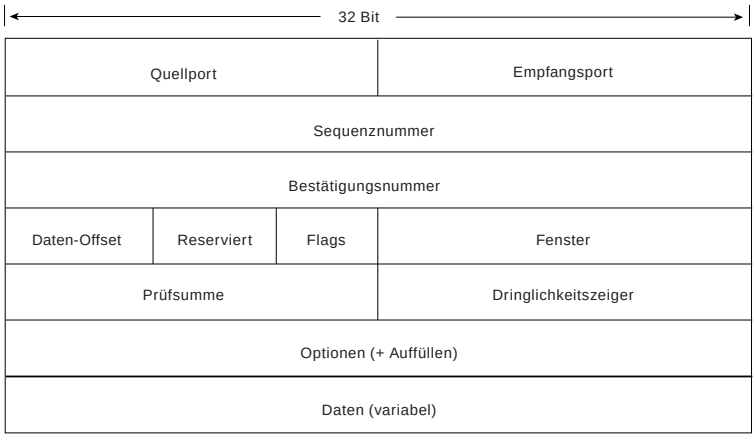


Bild 28.10: Ein TCP-Paket wird aus zwölf Feldern gebildet

28.6.5 Beschreibung der TCP-Paketfelder

Die folgenden Beschreibungen erläutern die TCP-Paketfelder, die in Bild 28.10 gezeigt werden:

– *Quellport und Empfangsport* – Gibt die Punkte an, an denen Quell- und Empfangsprozesse der oberen Schichten TCP-Dienste erhalten.

- *Sequenznummer* – Gibt gewöhnlich die Nummer an, die dem ersten Datenbyte der Nachricht zugewiesen wurde. In der Phase des Verbindungsaufbaus kann dieses Feld auch dazu benutzt werden, eine Anfangssequenznummer zu kennzeichnen, die in der aufkommenden Übermittlung verwendet wird.
- *Bestätigungsnummer* – Enthält die Sequenznummer des nächsten Datenbytes, das der Sender des Pakets erwartet.
- *Daten-Offset* – Zeigt die Nummer eines 32-Bit-Worts im TCP-Header.
- *Reserviert* – Bleibt für zukünftigen Gebrauch reserviert.
- *Flags* – Trägt eine Vielzahl von Kontrollinformationen, einschließlich der SYN- und ACK-Bits, die zum Verbindungsaufbau benutzt werden, und des FIN-Bits, das für die Beendigung der Verbindung verwendet wird.
- *Fenster* – Spezifiziert die Größe des Empfangsfensters des Senders (das heißt, den Pufferbereich, der für eingehende Daten verfügbar ist).
- *Prüfsumme* – Gibt an, ob der Header beim Transport beschädigt wurde.
- *Dringlichkeitszeiger* – Zeigt auf das dringendste Datenbyte im Paket.
- *Optionen* – Spezifiziert verschiedene TCP-Optionen.
- *Daten* – Enthält Informationen der oberen Schichten.

28.7 User Datagram Protocol (UDP)

Das User Datagram Protocol (UDP) ist ein verbindungsloses Transportschichtprotokoll (Schicht 4), das zur Internet-Protokollfamilie gehört. UDP ist im Prinzip eine Schnittstelle zwischen IP und Prozessen der oberen Schichten. UDP-Protokollports unterscheiden die vielen Anwendungen, die auf einem einzigen Gerät laufen, voneinander.

Anders als TCP fügt UDP IP keine Funktionen wie Zuverlässigkeit, Ablaufsteuerung oder Fehlerbehebung hinzu. Da UDP

eine einfache Struktur hat, enthalten UDP-Header weniger Bytes und beanspruchen weniger Netzwerk-Overhead als TCP.

UDP ist nützlich in Situationen, in denen die Zuverlässigkeitsmechanismen von TCP nicht erforderlich sind, wenn also ein Protokoll der höheren Schichten die Fehlerbehebung und Ablaufsteuerung erledigen kann.

UDP ist das Transportprotokoll für verschiedene, sehr bekannte Anwendungsschichtprotokolle wie etwa Network File System (NFS), Simple Network-Management Protocol (SNMP), Domain Name System (DNS) und Trivial File-Transfer Protocol (TFTP).

Das Format des UDP-Pakets enthält vier Felder, die in Bild 28.11 dargestellt sind. Es sind Quell- und Zielport-, Längen- und Prüfsummenfelder.

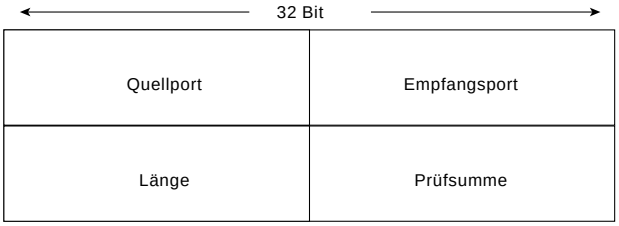


Bild 28.11: Ein UDP-Paket besteht aus vier Feldern

Quell- und Empfangsport enthalten die 16 Bit langen UDP-Protokollportnummern, die verwendet werden, um Datagramme zu demultiplexen, damit Anwendungsschichtprozesse empfangen werden können. Ein Längenfeld spezifiziert die Länge des UDP-Headers und der Daten. Die Prüfsumme stellt eine (optionale) Unversehrtheitskontrolle von UDP-Headern und Daten bereit.

28.8 Die Anwendungsschichtprotokolle der Internet-Protokolle

Die Internet-Protokollsuite beinhaltet zahlreiche Anwendungsschichtprotokolle, die eine Vielzahl von Anwendungen, wie etwa die folgenden, repräsentieren:

– *File Transfer Protocol (FTP)* – Bewegt Dateien zwischen Geräten

- *Simple Network-Management Protocol (SNMP)* – Meldet in der Hauptsache ungewöhnliche Netzwerk-Bedingungen und setzt Netzwerk-Schwellenwerte
- *Telnet* – Dient als Terminalemulationsprotokoll
- *X Windows* – Dient als verteiltes Fenstersystem und grafische Benutzeroberfläche, die für die Kommunikation zwischen X-Terminals und UNIX-Workstations verwendet wird
- *Network File System (NFS), External Data Representation (XDR) und Remote Procedure Call (RPC)* – Arbeiten gemeinsam, um den transparenten Zugriff auf entfernte Netzwerk-Ressourcen zu gewähren
- *Simple Mail Transfer Protocol (SMTP)* – Stellt E-Mail-Dienste bereit
- *Domain Name System (DNS)* – Übersetzt die Namen der Netzwerk-Knoten in Netzwerk-Adressen

Tabelle 28.5 listet diese Protokolle der höheren Schichten mit den Anwendungen, die sie unterstützen, auf.

Tabelle 28.5: Protokolle oberer Schichten und ihre Anwendungen

Anwendung	**Protokolle**
Dateitransfer	FTP
Netzwerkmanagement	SNMP
Terminalemulation	Telnet
Verteiler Dateidienst	NFS, XDR, RPC, X Windows
Elektronische Post	SMTP
Namensvertriebsservice	DNS

KAPITEL 29

NetWare-Protokolle

29.1 Hintergrund

NetWare ist ein Netzwerk-Betriebssystem, das transparenten Fernzugriff auf Dateien und zahlreiche andere Netzwerk-Dienste bereitstellt, zum Beispiel die gemeinsame Nutzung von Druckern und die Unterstützung verschiedener Anwendungen wie E-Mail und Datenbankzugriff. NetWare spezifiziert die oberen fünf Schichten des OSI-Schichtenmodells und läuft somit mit praktisch jedem Medienzugriffsprotokoll (Schicht 2). Außerdem arbeitet NetWare quasi auf allen Computersystemen, vom PC bis zum Großrechner. Dieses Kapitel beschreibt die wichtigsten Kommunikationsprotokolle, die NetWare unterstützen.

NetWare wurde in den frühen 80er Jahren von Novell, Inc., entwickelt und eingeführt. Es ging aus dem Xerox Network System (XNS) hervor, das von der Xerox Corporation in den späten 70ern geschaffen wurde. Clients (manchmal auch *Workstations* genannt) rufen Dienste, wie etwa Datei- und Druckerzugriff von Servern ab.

NetWares Client-Server-Architektur unterstützt den Fernzugriff, der den Benutzern durch Fernabfrage zugänglich ist. Eine Fernabfrage beginnt, wenn das lokale Computerprogramm, das auf dem Client läuft, eine Abfrage an den Remote

Server sendet. Der Server bearbeitet dann die Fernabfrage und schickt die gewünschte Information an den lokalen Client.

Bild 29.1 zeigt die NetWare-Protokollsuite, die Medienzugriffsprotokolle, auf denen NetWare läuft, und das Verhältnis zwischen den NetWare-Protokollen und dem OSI-Schichtenmodell. Dieses Kapitel erläutert die Elemente und Operationen dieser Protokollkomponenten.

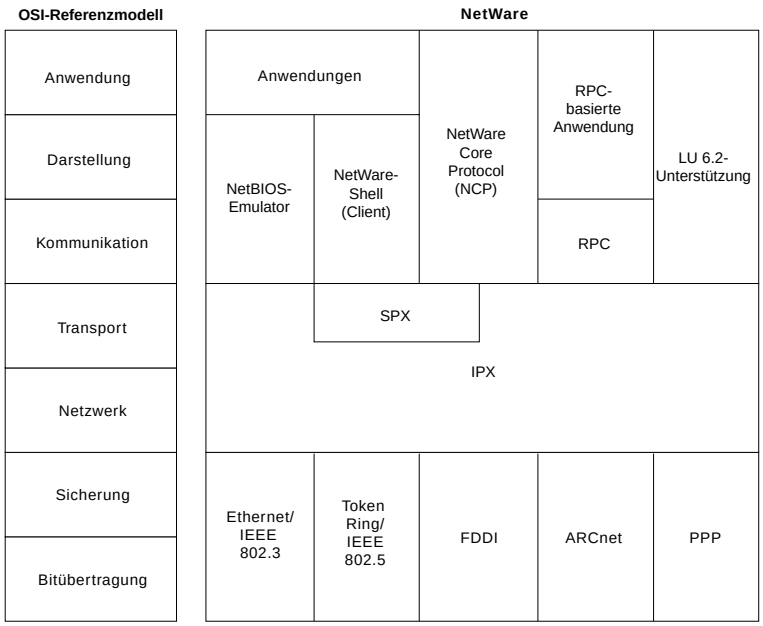


Bild 29.1: Die NetWare-Protokollsuite liegt in allen OSI-Schichten

29.2 NetWare Medienzugriff

Die NetWare-Protokollsuite unterstützt verschiedene Medienzugriffsprotokolle (Schicht 2), wie etwa Ethernet/IEEE 802.3, Token Ring/IEEE 802.5, Fiber-Distributed Data Interface (FDDI) und das Point-to-Point-Protokoll (PPP). Bild 29.2 hebt die Bandbreite von NetWares Medienzugriffsunterstützung hervor.

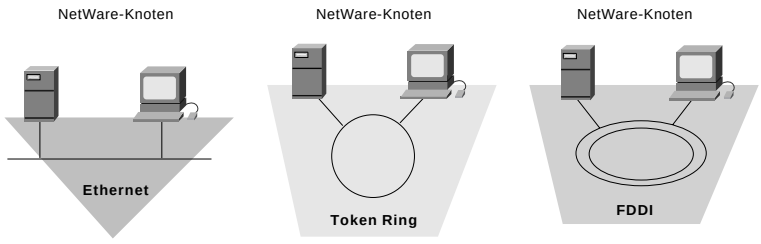


Bild 29.2: NetWare unterstützt die gängigsten Medienzugriffsprotokolle

29.3 Internetwork Packet Exchange (IPX) im Überblick

Das Internetwork Packet Exchange (IPX) ist das ursprüngliche Netzwerkschichtprotokoll (Schicht 3) von NetWare und wird verwendet, um Pakete durch ein Verbundnetz zu routen. IPX ist ein verbindungsloses, auf Datagrammen basierendes Netzwerk-Protokoll und ähnelt als solches dem Internet-Protokoll, das man in TCP/IP-Netzwerken findet.

IPX nutzt die Dienste eines dynamischen Vektorfernroutingprotokolls (Routing-Information Protocol [RIP]) oder eines Verbindungszustandsroutingprotokolls (NetWare Link-State Protocol [NLSP]). IPX RIP sendet alle 60 Sekunden Routingaktualisierungen. Um über den besten Routingpfad zu entscheiden, verwendet IPX RIP ein »Tick« als Maß, das im Prinzip die Verzögerung ist, die bei der Verwendung einer bestimmten Länge zustandekommt. Ein Tick ist ein Achtzehntel einer Sekunde. Im Falle von zwei Pfaden mit gleicher Tickzählung verwendet IPX RIP die Etappenzählung als entscheidendes Kriterium. (Wenn ein Paket einen Router durchläuft, ist eine Etappe vollendet.) IPXs RIP ist mit RIP-Implementierungen anderer Netzwerk-Umgebungen nicht kompatibel.

Wie sonstige Netzwerk-Adressen, müssen auch die Netzwerk-Adressen von Novell IPX einmalig sein. Diese Adressen werden hexadezimal angegeben und bestehen aus zwei Teilen: einer Netzwerk-Nummer und einer Knotennummer. Die IPX-Netzwerk-Nummer, die vom Netzwerk-Administrator zugewiesen wird, ist 32 Bit lang. Die Knotennummer, die gewöhnlich die Media-Access-Control-Adresse (MAC) für eine der Netzwerk-Interfacekarten (NIC) im System ist, ist 48 Bit lang.

Indem IPX MAC-Adressen als Knotennummern verwendet, ermöglicht es dem System, Knoten zu senden, um vorherzusehen, welche MAC-Adresse bei einer Datenverknüpfung benutzt werden muß. (Da der Host-Teil der IP-Netzwerk-Adresse nichts mit der MAC-Adresse zu tun hat, müssen IP-Knoten das Address-Resolution Protocol [ARP] anwenden, um die MAC-Adresse des Ziels festzustellen.)

29.4 IPX-Kapselungsarten

Novell NetWare IPX unterstützt mehrere Verfahren der Kapselung auf einer einzigen Routerschnittstelle, vorausgesetzt, es werden mehrere Netzwerknummern zugewiesen. Die Kapselung ist das Verfahren, bei dem Informationen aus oberen Schichten zusammen mit Daten in einen Frame gepackt werden. NetWare unterstützt die vier folgenden Arten der Kapselung:

- *Novell Proprietary* – Novell Proprietary, auch »802.3 raw« oder Novell Ethernet_802.3 genannt, ist das Grundverfahren, das Novell für die Kapselung benutzt. Es enthält ein IEEE-802.3-Längenfeld, aber keinen IEEE-802.2-(LLC-)-Header. Der IPX-Header folgt unmittelbar auf das 802.3-Längenfeld.
- *802.3* – Ist das standardmäßige IEEE-802.3-Frameformat, auch Novell_802.2, 802.3 genannt.
- *Ethernet Version 2* – Heißt auch Ethernet-II oder ARPA. Ethernet Version 2 beinhaltet den standardmäßigen Ethernet-Version-2-Header, der aus Ziel- und Quelladreßfeldern gefolgt von einem EtherType-Feld besteht.
- *SNAP* – Wird auch als Ethernet-SNAP bezeichnet. SNAP erweitert den IEEE-802.2-Header, indem es einen Typencode hinzufügt, der jenem ähnlich ist, der in der Ethernet-Version-2-Spezifizierung definiert wird.

Bild 29.3 veranschaulicht diese Arten der Kapselung.

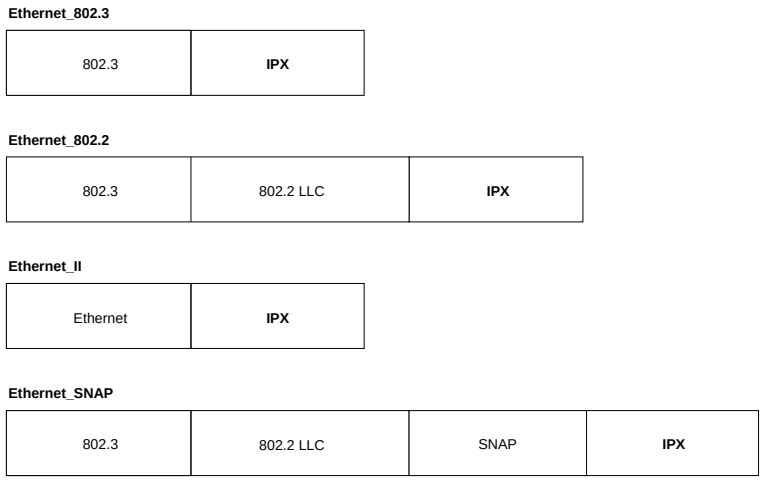


Bild 29.3: Es gibt vier Arten der Kapselung

29.5 Service-Advertisement Protocol (SAP)

Das *Service-Advertisement Protocol (SAP)* ist ein IPX-Protokoll, über das Netzwerk-Ressourcen wie Dateiserver- und Printserver-Adressen und die Dienste bekannt gegeben werden. Alle 60 Sekunden werden Bekanntmachungen über SAP gesendet. Dienste werden mit einer Hexadezimalzahl gekennzeichnet, die SAP-Identifier genannt wird (zum Beispiel 4 = Dateiserver und 7 = Printserver).

Eine SAP-Operation beginnt, wenn Router SAPs anhören und eine Tabelle mit allen bekannten Diensten und den dazugehörigen Netzwerk-Adressen erstellen. Dann senden Router alle 60 Sekunden ihre SAP-Tabelle aus. Novell-Clients können eine Anforderung abschicken, in der sie einen bestimmten Datei-, Print- oder Gateway-Dienst verlangen. Der lokale Router antwortet auf die Anforderung mit der Netzwerk-Adresse des verlangten Dienstes, und der Client kann dann den Dienst direkt ansprechen.

SAP hat die Vormachtstellung in heutigen Netzwerken, die auf NetWare 3.11 und früheren Versionen basieren. In Novell-4.0-Netzwerken kommt es jedoch seltener zum Einsatz, weil hier Workstations Dienste über einen NetWare Directory Services (NDS)- Server in Anspruch nehmen können. Dennoch ist SAP für die Workstations auch in NetWare-4.0-Netzen noch erforderlich, damit sie beim Booten einen NDS-Server finden.

29.5.1 SAP-Filter

Mit den SAP-Identifiern können SAP-Bekanntmachungen in dem Ein-/Ausgabe-Anschluß eines Routers oder einem spezifischen Router herausgefiltert werden. SAP-Filter sparen Bandbreite des Netzwerks ein und sind besonders nützlich in großen Novell-Installationen, in denen es Hunderte von SAP-Diensten gibt.

Im allgemeinen ist der Gebrauch von SAP-Filtern für Dienste empfehlenswert, die nicht für ein bestimmtes Netzwerk reserviert sind. Zum Beispiel brauchen entfernte Standorte wahrscheinlich keine SAP-Bekanntmachungen über Printdienste, die sich an einer zentralen Stelle befinden. Ein SAP-Outputfilter an der zentralen Stelle (empfehlenswert) oder ein SAP-Inputfilter, der den SAP-Identifier für einen Printserver an dem entlegenen Standort benutzt, hält den Router davon ab, Printdienste mit den SAP-Aktualisierungen zu verschicken.

29.6 NetWare-Transportschicht

Das *Sequenced Packet-Exchange (SPX)*-Protokoll ist das gängigste NetWare-Transportprotokoll in der Schicht 4 des OSI-Schichtenmodells. SPX setzt in der NetWare-Protokollsuite auf IPX auf. SPX ist ein verläßliches, verbindungsorientiertes Protokoll, das den Datagrammdienst von IPX, NetWares Netzwerkschichtprotokoll (Schicht 3), ergänzt. SPX ging aus dem Sequenced Packet Protocol (SPP) von Xerox Networking Systems (XNS) hervor. Novell bietet auch die Unterstützung des Internet-Protokolls in Form des User Datagram Protocol (UDP) an. IPX-Datagramme werden zum Transport durch ein IP-basiertes Verbundnetz in UDP/IP-Köpfe verkapselt.

29.7 NetWare-Protokolle und Dienste der oberen Schichten

NetWare unterstützt eine Vielzahl von Protokollen der oberen Schichten, wie zum Beispiel *NetWare Shell, NetWare Remote Procedure Call, NetWare Core Protocol* und *Network Basic Input/Output System*.

Die NetWare Shell steuert Clients (von der NetWare-Gemeinde oft Workstations genannt) und fängt Input/Output (I/O)-Aufrufe von Anwendungen ab, um festzustellen, ob sie zur Vollendung Netzwerk-Zugriff benötigen. Wenn die Anforderung der Anwendung den Netzwerk-Zugriff erfordert, verpackt die NetWare Shell die Anforderung und sendet sie zur Verarbeitung und Netzwerk-Übermittlung an Software der unteren Schichten. Erfordert die Anfrage der Anwendung keinen Netzwerk-Zugriff, wird die Anforderung an die lokalen I/O-Ressourcen geleitet. Client-Anwendungen wissen nicht, ob ein Netzwerk-Zugriff erforderlich ist, um den Aufruf einer Anwendung zu vollenden.

NetWare Remote Procedure Call (NetWare RPC) ist ein weiterer, allgemeinerer Nachsendemechanismus, der der NetWare Shell von Novell vom Konzept her ähnelt.

Das NetWare Core Protocol (NCP) besteht aus einer Reihe von Serverroutinen, die den Zweck haben, Anfragen von Anwendungen zu bedienen, die zum Beispiel von der NetWare Shell kommen. Zu den Diensten, die NCP bereitstellt, gehören Dateizugriff, Druckerzugriff, Namenverwaltung, Abrechnungen, Sicherheit und Dateisynchronisierung.

NetWare unterstützt auch das *Network Basic Input/Output System* (NetBIOS), die Spezifizierung der Schnittstelle der Kommunikationssteuerschicht von IBM und Microsoft. NetWares NetBIOS-Emulierungssoftware ermöglicht es, daß Programme, die für die standardmäßige NetBIOS-Schnittstelle geschrieben wurden, auf NetWare-Systemen laufen.

29.7.1 NetWare-Dienste der Anwendungsschicht

Zu den NetWare-Diensten der Anwendungsschicht gehören *NetWare Message-handling Service (NetWare MHS), Btrieve, NetWare Loadable Modules (NLMs),* und *IBM Logical Unit (LU) 6.2 Network-Addressable Units (NAUs)*. NetWare MHS ist ein Nachrichtensystem, das den Transport von E-Mails ermöglicht. Btrieve ist Novells Implementierung des Binary-Tree-Datenbankzugriffmechanismus (btree). NLMs sind Add-on-Module, die an ein NetWare-System angeschlossen werden. NLMs, die es derzeit von Novell und Drittanbietern gibt, beinhalten alternative Protokollstapel, Kommunikationsdienste

und Datenbankdienste. In Verbindung mit IBM LU 6.2 NAU erlaubt NetWare Peer-to-Peer-Verbindungen und den Informationsaustausch über IBM-Netzwerke. NetWare-Pakete werden zum Transport durch ein IBM-Netzwerk in LU-6.2-Paket verkapselt.

29.8 IPX-Paketformat

Das IPX-Paket ist die Grundeinheit des Netzbetriebs von Novell NetWare. Bild 29.4 zeigt das Format eines NetWare-IPX-Pakets.

29.4: Ein NetWare-IPX-Paket besteht aus elf Feldern

Die folgenden Definitionen erläutern die IPX-Paketfelder, die in Bild 29.4 gezeigt werden:

- *Prüfsumme* – Gibt an, daß die Prüfsumme nicht gebraucht wird, wenn dieses 16-Bit-Feld auf 1s (FFFF) gesetzt wird.
- *Paketlänge* – Gibt die Länge eines kompletten IPX-Datagramms in Byte an. IPX-Pakete können bis zur Größe der Media Maximum Transmission Unit (MTU) jede Länge haben (es ist keine Paketfragmentierung erlaubt).

- *Transportkontrolle* – Zeigt die Anzahl von Routern, durch die das Paket gelaufen ist. Erreicht dieser Wert 16, wird das Paket abgelegt, weil davon ausgegangen wird, daß es zu einer Routingschleife kommen könnte.
- *Pakettyp* – Bestimmt, welches Protokoll der oberen Schichten die Informationen des Pakets erhalten soll. Häufig enthält es einen dieser beiden Werte:
  - 5: Gibt Sequenced Packet-Exchange (SPX) an.
  - 17: Gibt NetWare Core Protocol (NCP) an.
- *Zielnetzwerk, Zielknoten* und *Zielsocket* – Spezifiziert die Zielinformationen.
- *Quellnetzwerk, Quellknoten* und *Quellsocket* – Spezifiziert die Quellinformationen.
- *Daten der oberen Schichten* – Enthält Informationen über Prozesse der oberen Schichten.

KAPITEL 30

Protokolle der Open System Interconnection (OSI)

30.1 Hintergrund

Die Protokollsammlung für die Open System Interconnection (OSI) umfaßt eine Vielzahl von Standardprotokollen, die auf dem OSI-Referenzmodell basieren. Diese Protokolle sind Bestandteil eines internationalen Programms zur Entwicklung von Datennetzwerkprotokollen und anderen Normen, die das Zusammenarbeiten von Geräten mehrerer Anbieter ermöglichen. Das OSI-Programm erwuchs aus der Notwendigkeit für internationale Netzwerk-Normen und wurde entworfen, um die Kommunikation zwischen Hardware- und Software-Systemen unabhängig von Unterschieden in der zugrundeliegenden Architektur zu ermöglichen.

Die OSI-Spezifikationen wurden von zwei internationalen Standardisierungsorganisationen erdacht und umgesetzt: der International Organisation for Standardization (ISO) und der International Telecommunication Union-Telecommunication Standardization Sector (ITU-T). Dieses Kapitel liefert eine Zusammenfassung der OSI-Protokollsammlung und veranschaulicht deren Zusammenhang mit dem allgemeinen OSI-Referenzmodell.

30.2 OSI-Netzwerkprotokolle

In der Bild 30.1 ist die gesamte OSI-Protokollsammlung sowie deren Beziehung zu den Schichten des OSI-Referenzmodells dargestellt. Jede Komponente dieser Protokollsammlung wird

in diesem Kapitel kurz erläutert. Auf die OSI-Routing-Protokolle wird im Kapitel 39, »Routing-Protokolle für die Open Systems Interconnection (OSI)«, genauer eingegangen.

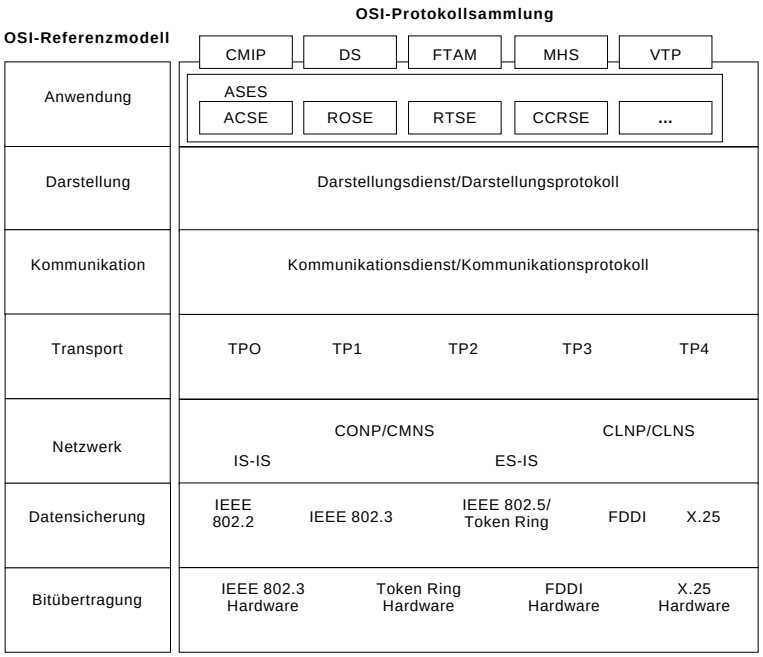


Bild 30.1: Die OSI-Protokollsammlung führt eine Abbildung auf jede Schicht des OSI-Referenzmodells durch

30.2.1 Physikalische Bitübertragungsschicht und Datensicherungsschicht von OSI

Die OSI-Protokollsammlung unterstützt zahlreiche Standardprotokolle für den Medienzugriff in der physikalischen Bitübertragungsschicht und der Datensicherungsschicht. Die große Vielfalt der von der OSI-Protokollsammlung unterstützten Protokolle für den Medienzugriff gestattet neben OSI ein einfaches Bestehen anderer Protokollsammlungen auf denselben Netzwerk-Medien. Zu den unterstützten Protokollen für den Medienzugriff gehören IEEE 802.2 LLC, Token Ring/IEEE 802.5, Fiber Distributed Data Interface (FDDI) sowie X.25.

30.2.2 OSI-Netzwerkschicht

Die OSI-Protokollsammlung bestimmt zwei Routing-Protokolle für die Netzwerkschicht: Endsystem-zu-Zwischensystem (ES-IS) und Zwischensystem-zu-Zwischensystem (IS-IS). Zusätzlich implementiert die OSI-Protokollsammlung zwei Arten von Netzdiensten: den verbindungslosen sowie den verbindungsbezogenen Dienst.

Normen zu den OSI-Schichten

Zusätzlich zu den Normen, welche die Protokolle und Dienste für die OSI-Netzwerkschicht festlegen, beschreiben folgende Dokumente andere Spezifikationen der OSI-Netzwerkschicht:

- ISO 8648 – Diese Norm definiert die interne Organisation der Netzwerkschicht (IONL), welche die Netzwerkschicht in drei voneinander getrennte Unterschichten unterteilt, um verschiedene Arten von Subnetzen zu unterstützen.
- ISO 8348 – Diese Norm definiert die Adressierung der Netzwerkschicht und beschreibt die von der OSI-Netzwerkschicht bereitgestellten verbindungsbezogenen und verbindungslosen Dienste.
- ISO TR 9575 – Diese Norm beschreibt den Rahmen, die Konzepte und die Terminologie, die im Zusammenhang mit den OSI-Routing-Protokollen verwendet werden.

Verbindungsloser Netzdienst von OSI

Der verbindungslose Netzdienst von OSI ist mittels des Connectionless Network Protocol (CLNP) und des Connectionless Network Service (CLNS) implementiert. CLNP und CLNS sind in der ISO-Norm 8473 beschrieben.

Bei CLNP handelt es sich um ein OSI-Protokoll für die Netzwerkschicht, das Daten und Fehlerangaben aus darüberliegenden Schichten über verbindungslose Anschlüsse überträgt. CLNP stellt die Schnittstelle zwischen dem Connectionless Network Service (CLNS) und den darüberliegenden Schichten bereit.

CLNS stellt mittels CLNP Dienste der Netzwerkschicht für die Transportschicht zur Verfügung.

CLNS baut weder eine Verbindung auf noch beendet er diese, da die Verbindungspfade für jedes über ein Netzwerk übermittelte Paket unabhängig voneinander bestimmt werden. Damit steht er im Gegensatz zum Connection-Mode Network Service (CMNS).

Außerdem sorgt CLNS für eine bestmögliche Zustellung, was nichts anderes heißt, als daß es keine Garantie dafür gibt, daß Daten nicht verloren gehen, beschädigt, in die falsche Reihenfolge gebracht oder vervielfältigt werden. Für die Fehlererkennung und -korrektur verläßt CLNS sich auf Protokolle für die Transportschicht.

Verbindungsbezogener Netzdienst von OSI

Der verbindungsbezogene Netzdienst von OSI ist mittels des Connection-Oriented Network Protocol (CONP) und des Connection-Mode Network Service (CMNS) implementiert.

Bei CONP handelt es sich um ein OSI-Protokoll für die Netzwerkschicht, das Daten und Fehlerangaben aus darüberliegenden Schichten über verbindungsbezogene Anschlüsse überträgt. CONP basiert auf dem X.25 Packet-Layer Protocol (PLP) und ist in der ISO-Norm 8208, »X.25 Packet-Layer Protocol for DTE«, beschrieben.

CONP stellt die Schnittstelle zwischen CMNS und den darüberliegenden Schichten bereit. Es handelt sich dabei um einen Dienst der Netzwerkschicht, der als Schnittstelle zwischen der Transportschicht und CONP fungiert und in der ISO-Norm 8878 beschrieben ist.

CMNS führt Funktionen durch, die mit der expliziten Einrichtung von Verbindungspfaden zwischen kommunizierenden Transportschichteinheiten verbunden sind. Zu diesen Funktionen gehören der Aufbau, die Aufrechterhaltung und die Beendigung der Verbindung. Weiterhin stellt CMNS im Gegensatz zu CLNS einen Mechanismus bereit, der eine bestimmte Dienstqualität (*Quality of Service* = QoS) anfordert.

Adressierung der Netzwerkschicht

Die Adressierung der OSI-Netzwerkschicht ist durch zwei Arten von hierarchischen Adressen implementiert: Network

Service Access Point-Adressen (NSAP) und Network Entity Titles (NET).

Bei einem Network Service Access Point handelt es sich um einen vorgestellten Punkt an der Grenze zwischen den Netzwerk- und Transportschichten. Der NSAP stellt denjenigen Ort dar, an dem der Transportschicht die OSI-Netzdienste zur Verfügung gestellt werden. Jeder Transportschichteinheit wird ein einzelner NSAP zugewiesen, der in einem OSI-Netzwerk mittels NSAP-Adressen individuell angesprochen wird.

Im Bild 30.2 ist der Aufbau der OSI-NSAP-Adresse dargestellt, die einzelne NSAPs ausweist.

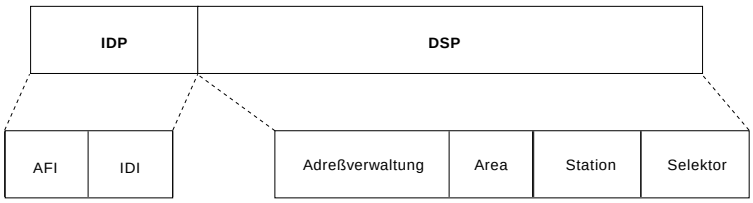


Bild 30.2: Jeder Transportschicht-Einheit wird eine OSI-NSAP-Adresse zugewiesen

NSAP-Adreßfelder

Es gibt zwei NSAP-Adreßfelder: den *Initial Domain Part* (IDP) und den *Domain-Specific Part* (DSP).

Das IDP-Feld ist in zwei Bestandteile unterteilt: den *Authority Format Identifier* (AFI) und den *Initial Domain Identifier* (IDI). Der AFI stellt Informationen über den Aufbau und den Inhalt des IDI- und des DSP-Felds bereit, beispielsweise darüber, ob der IDI von variabler Länge ist und ob der DSP auf die Dezimal- oder Binärnotation zurückgreift. Der IDI gibt diejenige Einheit an, die dem DSP-Anteil der NSAP-Adresse Werte zuweisen kann.

Das Adreßfeld DSP ist von der für die Verwaltung zuständigen Authority in vier Bestandteile unterteilt worden. Das Feld *Address Administration* ermöglicht eine weitergehende Verwaltung der Adressierung, indem ein zweiter Authority-Identifier hinzugefügt und die Adreßverwaltung an eine Subauthority delegiert wird. Das Feld *Area* bezeichnet den spezifischen Bereich innerhalb einer Domain und wird für Routingzwecke verwendet. Das Feld *Station* bezeichnet eine spezifische Station innerhalb eines Bereichs und wird ebenfalls für Routing-

zwecke verwendet. Das Feld *Selector* stellt den spezifischen n-Selektor innerhalb einer Station bereit und wird wie schon die übrigen Felder für Routingzwecke verwendet. Der reservierte n-Selektor 00 bezeichnet die Adresse wie ein Network Entity Title (NET).

NSAPs von Endsystemen

Ein OSI-Endsystem (ES) verfügt häufig über mehrere NSAP-Adressen; eine für jede der enthaltenen Transportschichteinheiten. Ist dies der Fall, so unterscheidet sich die NSAP-Adresse jeder Transportschicht-Einheit üblicherweise nur im letzten Byte (*n-Selektor* genannt). Das Bild 30.3 stellt die Beziehungen zwischen einer Transportschichteinheit, dem NSAP und dem Netzdienst dar.

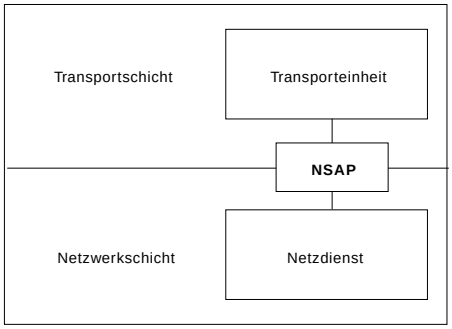


Bild 30.3: Der NSAP stellt eine Verbindung zwischen einer Transportschichteinheit und einem Netzdienst bereit

Ein Network Entity Title (NET) wird verwendet, um die Netzwerkschicht eines Systems zu identifizieren, ohne das System mit einer bestimmten Transportschichteinheit zu verknüpfen (wie es eine NSAP-Adresse tut). NETs sind für die Adressierung von Zwischensystemen (IS) wie beispielsweise Router hilfreich, die keine Schnittstelle mit der Transportschicht besitzen. Ein IS kann über einen einzigen NET oder über mehrere NETs verfügen, wenn es an mehreren Bereichen oder Domains beteiligt ist.

30.2.3 OSI-Protokolle für die Transportschicht

Die OSI-Protokollsammlung implementiert zwei Arten von Diensten für die Transportschicht: verbindungsbezogene Transportdienste und verbindungslose Transportdienste.

In der OSI-Protokollsammlung gibt es fünf verbindungsbezogene Protokolle für die Transportschicht, und zwar vom Transportprotokoll der Klasse 0 (TP0) bis zum Transportprotokoll der Klasse 4 (TP4). Ein verbindungsloser Transportdienst wird lediglich vom Transportprotokoll der Klasse 4 unterstützt.

Das Transportprotokoll der Klasse 0 (TP0), das einfachste OSI-Transportprotokoll, sorgt für eine Aufteilung und das Wiederzusammensetzen der Daten. Für TP0 ist ein verbindungsbezogener Netzdienst erforderlich.

Das Transportprotokoll der Klasse 1 (TP1) führt eine Aufteilung und das Wiederzusammensetzen durch und bietet eine einfache Fehlerbehebung. TP1 sorgt für eine Reihe von Protokolldateneinheiten (*Protocol Data Unit* = PDU) und überträgt die PDUs erneut bzw. stellt erneut eine Verbindung her, falls eine übermäßige Anzahl von PDUs nicht bestätigt wird. Für TP1 ist ein verbindungsbezogener Netzdienst erforderlich.

Das Transportprotokoll der Klasse 2 (TP2) sorgt für eine Aufteilung und das Wiederzusammensetzen sowie für das Multiplexing und Demultiplexing der Datenströme über eine einzelne virtuelle Verbindung. Für TP2 ist ein verbindungsbezogener Netzdienst erforderlich.

Das Transportprotokoll der Klasse 3 (TP3) bietet eine einfache Fehlerbehebung, sorgt für eine Aufteilung und das Wiederzusammensetzen und führt weiterhin das Multiplexing und Demultiplexing der Datenströme über eine einzelne virtuelle Verbindung durch. TP3 sorgt für eine Reihe von PDUs und überträgt diese erneut bzw. stellt erneut eine Verbindung her, falls eine übermäßige Anzahl nicht bestätigt wird. Für TP3 ist ein verbindungsbezogener Netzdienst erforderlich.

Das Transportprotokoll der Klasse 4 (TP4) bietet eine einfache Fehlerbehebung, sorgt für eine Aufteilung und das Wiederzusammensetzen und unterstützt das Multiplexing und Demultiplexing der Datenströme über eine einzelne virtuelle Verbindung. TP4 sorgt für eine Reihe von PDUs und überträgt diese erneut bzw. stellt erneut eine Verbindung her, falls eine übermäßige Anzahl nicht bestätigt wird. TP4 bietet sowohl verbindungsbezogenen als auch verbindungslosen Netzdiensten zuverlässige Transportdienste und Funktionen. Es basiert auf

dem Transmission Control Protocol (TCP) der Sammlung von Internet Protocols (IP) und stellt das einzige OSI-Protokoll dar, das verbindungslose Netzdienste unterstützt.

30.2.4 OSI-Protokolle für die Kommunikationsschicht

Die Implementierung der Kommunikationsschicht in der OSI-Protokollsammlung besteht aus einem Kommunikationsprotokoll und einem Kommunikationsdienst (*Session Service*). Das Kommunikationsprotokoll ermöglicht es Benutzern des Kommunikationsdienstes, sich mit dem Kommunikationsdienst zu verständigen. Bei dem Benutzer des Kommunikationsdienstes handelt es sich um eine Einheit, welche die Dienste der Kommunikationsschicht anfordert. Solche Anforderungen erfolgen an Session Service Access Points (SSAPs), die einen Zugriff auf den Kommunikationsdienst ermöglichen. Benutzer des Kommunikationsdienstes werden durch eine SSAP-Adresse eindeutig identifiziert. Im Bild 30.4 sind die Beziehungen zwischen dem Benutzer des Kommunikationsdienstes, dem SSAP, dem Kommunikationsprotokoll und dem Kommunikationsdienst dargestellt.

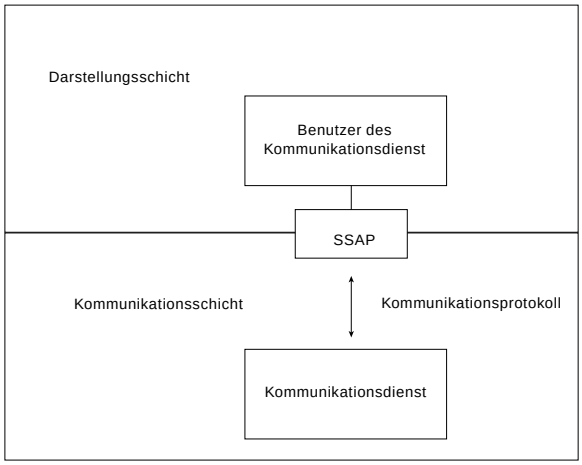


Bild 30.4: Funktionen der Kommunikationsschicht stellen den Funktionen der Darstellungsschicht ihre Dienste über einen SSAP zur Verfügung

Der Kommunikationsdienst stellt Benutzern des Dienstes vier grundlegende Dienste zur Verfügung. Erstens baut er Verbindungen zwischen Benutzern des Kommunikationsdienstes auf und ab und synchronisiert den Datenaustausch zwischen diesen. Zweitens führt er verschiedene Verhandlungen für den

Einsatz von Tokens für die Kommunikationsschicht durch, die der Benutzer des Kommunikationsdienstes besitzen muß, um mit dem Austausch beginnen zu können. Drittens fügt er Synchronisationsstellen in die übermittelten Daten ein, mit denen es möglich ist, die Arbeitssitzungen nach dem Auftreten von Fehlern oder Unterbrechungen wiederherzustellen. Und schließlich ermöglicht er es Benutzern des Kommunikationsdienstes, eine Arbeitssitzung zu unterbrechen und später zu einem bestimmten Zeitpunkt fortzusetzen.

Der Kommunikationsdienst ist in der ISO-Norm 8326 und der ITU-T-Empfehlung X.215 definiert. Das Kommunikationsprotokoll ist in der ISO-Norm 8327 und der ITU-T-Empfehlung X.225 definiert. Eine verbindungslose Fassung des Kommunikationsprotokolls ist in der ISO-Norm 9548 definiert.

30.2.5 OSI-Protokolle für die Darstellungsschicht

Die Implementierung der Darstellungsschicht in der OSI-Protokollsammlung besteht aus einem Darstellungsprotokoll und einem Darstellungsdienst (*Presentation Service*). Das Darstellungsprotokoll ermöglicht es Benutzern des Darstellungsdienstes, mit dem Darstellungsdienst zu kommunizieren.

Bei einem Benutzer des Darstellungsdienstes handelt es sich um eine Einheit, welche die Dienste der Darstellungsschicht anfordert. Solche Anforderungen finden an Presentation Service Access Points (PSAPs) statt, die einen Zugriff auf den Darstellungsdienst ermöglichen. Benutzer des Darstellungsdienstes werden durch eine PSAP-Adresse eindeutig identifiziert.

Der Darstellungsdienst handelt die Übertragungssyntax aus und übersetzt die Daten in die bzw. aus der Übertragungssyntax für die Benutzer des Darstellungsdienstes, die die Daten in verschiedener Syntax darstellen. Der Darstellungsdienst wird von zwei Benutzern des Dienstes verwendet, um sich über die zu verwendende Übertragungssyntax zu verständigen. Sobald man sich auf eine Übertragungssyntax geeinigt hat, müssen die Einheiten des Darstellungsdienstes die vom Benutzer des Dienstes stammenden Daten in die richtige Übertragungssyntax übersetzen.

Der Darstellungsdienst ist in der ISO-Norm 8822 und der ITU-T-Empfehlung X.216 definiert. Das Darstellungsprotokoll ist in der ISO-Norm 8823 und der ITU-T-Empfehlung X.226 definiert. Eine verbindungslose Fassung des Darstellungsprotokolls ist in der ISO-Norm 9576 definiert.

30.2.6 OSI-Protokolle für die Anwendungsschicht

Die Implementierung der Anwendungsschicht in der OSI-Protokollsammlung besteht aus verschiedenen Anwendungseinheiten. Bei einer Anwendungseinheit handelt es sich um denjenigen Teil eines Anwendungsprozesses, der für den Einsatz der OSI-Protokollsammlung von Bedeutung ist. Eine Anwendungseinheit setzt sich aus dem Benutzerelement und einem Element des Anwendungsdienstes (*Application Service Element* = ASE) zusammen.

Bei dem Benutzerelement handelt es sich um denjenigen Teil einer Applikationseinheit, der auf ASEs zurückgreift, um die für den Anwendungsprozeß notwendige Kommunikation zu betreiben. Bei dem ASE handelt es sich um denjenigen Teil einer Applikationseinheit, der Benutzerelementen – und somit auch Anwendungsprozessen – Dienste zur Verfügung stellt. ASEs stellen weiterhin Schnittstellen für die darunterliegenden OSI-Schichten bereit. Im Bild 30.5 sind die Zusammensetzung eines einzelnen Anwendungsprozesses (bestehend aus der Anwendungseinheit, dem Benutzerelement und den ASEs) sowie dessen Beziehungen zum PSAP und dem Darstellungsdienst dargestellt.

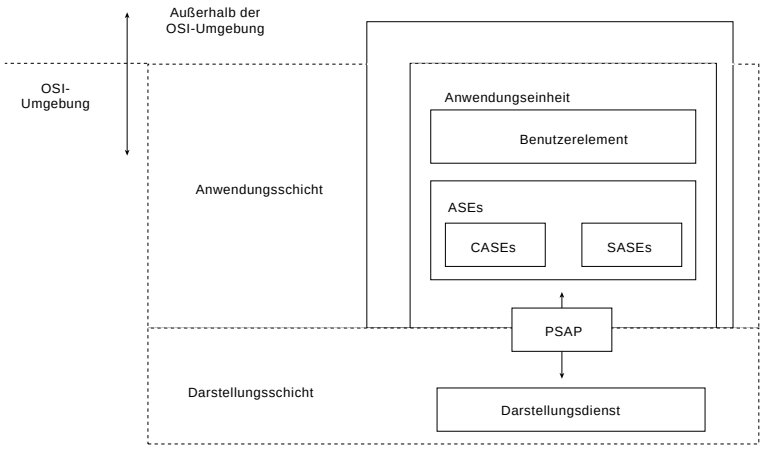


Bild 30.5: Ein Anwendungsprozeß beruht auf dem PSAP und dem Darstellungsdienst

Ein ASE läßt sich einer der folgenden beiden Klassifizierungen zuordnen: Entweder handelt es sich um ein allgemeines (*Common Application Service Element* = CASE) oder um ein bestimmtes Element eines Anwendungsdienstes (*Specific Application Service Element* = SASE). Beide können in einer einzigen Anwendungseinheit vorkommen.

Allgemeine Elemente eines Anwendungsdienstes (CASEs)

Bei CASEs handelt es sich um ASEs, die von einer großen Vielfalt von Anwendungsprozessen verwendete Dienste zur Verfügung stellen. Häufig macht eine einzige Anwendungseinheit von mehreren CASEs Gebrauch. Die folgenden vier CASEs sind in der OSI-Spezifikation definiert:

- *Association Control Service Element (ACSE)* – Erstellt als Vorbereitung für eine Kommunikation von Anwendung zu Anwendung Verknüpfungen zwischen zwei Anwendungseinheiten.
- *Remote Operations Service Element (ROSE)* – Implementiert einen Request-Response-Mechanismus, der verschiedene entfernte Operationen über eine durch das ACSE aufgebaute Anwendungsverknüpfung zuläßt.
- *Reliable Transfer Service Element (RTSE)* – Ermöglicht es ASEs, Nachrichten zuverlässig zu übertragen, wobei die Transparenz komplexer Einrichtungen tieferliegender Schichten erhalten bleibt.
- *Commitment, Concurrence and Recovery Service Elements (CCRSE)* – Koordiniert Dialoge zwischen mehreren Anwendungseinheiten.

Bestimmte Elemente eines Anwendungsdienstes (SASEs)

Bei SASEs handelt es sich um ASEs, die Dienste zur Verfügung stellen, welche nur von einem bestimmten Anwendungsprozeß verwendet werden, beispielsweise bei der Dateiübertragung, beim Datenbankzugriff oder beim Befehlseingang.

OSI-Protokolle für Anwendungsprozesse

Bei einem Anwendungsprozeß handelt es sich um dasjenige Element einer Anwendung, das die Schnittstelle zwischen der Anwendung selber und der OSI-Anwendungsschicht bereitstellt. Zu den standardmäßigen OSI-Anwendungsprozessen gehören folgende:

- *Common Management Information Protocol (CMIP)* – Führt Funktionen der Netzwerkverwaltung durch, die den Austausch von Verwaltungsinformationen zwischen Endsystemen (ES) und Verwaltungsrechnern ermöglichen. CMIP ist in der ITU-T-Empfehlung X.700 spezifiziert und ähnelt von der Funktion her dem Simple Network Management Protocol (SNMP) und NetView.
- *Directory Services (DS)* – Dient als verteiltes Verzeichnis, das innerhalb von OSI-Netzwerken für die Identifikation und Adressierung von Knoten verwendet wird. DS ist in der ITU-T-Empfehlung X.500 spezifiziert.
- *File Transfer, Access and Management (FTAM)* – Stellt einen Dateiübertragungsdienst und Einrichtungen für einen verteilten Dateizugriff bereit.
- *Message Handling System (MHS)* – Stellt unter Verwendung von ansammelnden und weiterleitenden Diensten einen Übermittlungsmechanismus für elektronische Messaging-Anwendungen und andere Anwendungen bereit.
- *Virtual Terminal Protocol (VTP)* – Stellt eine Terminalemulation bereit, die es einem Computersystem ermöglicht, einem entfernten Endsystem als direkt angebundenes Terminal zu erscheinen.

KAPITEL 31

Banyan VINES

31.1 Hintergrund

Banyan Virtual Integrated Network Service (VINES) implementiert ein verteiltes Netzwerkbetriebssystem, das auf einer proprietären Protokollfamilie basiert, die von den Protokollen des Xerox Network System (XNS) der Xerox Corporation abgeleitet ist. VINES greift auf eine Client-Server-Architektur zurück, in der die Clients bestimmte Dienste wie beispielsweise den Zugriff auf Dateien und Drucker von den Servern anfordern. In diesem Kapitel erfolgt eine Zusammenfassung der VINES-Kommunikationsprotokolle. Der VINES-Protokollstack ist im Bild 31.1 dargestellt.

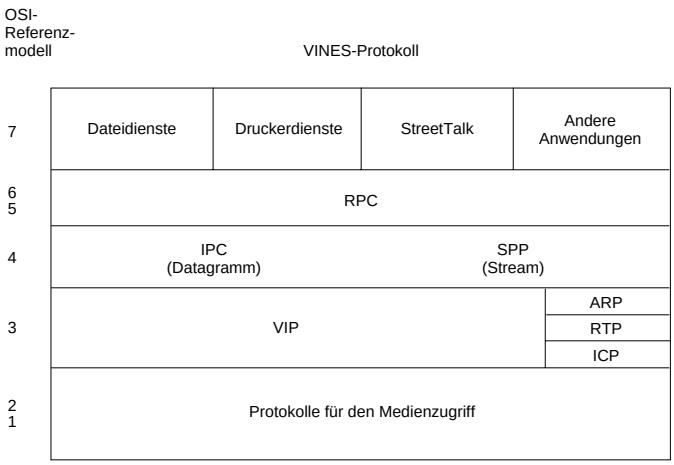


Bild 31.1: Der VINES-Protokollstack setzt sich aus fünf voneinander getrennten Schichten zusammen

31.2 Medienzugriff

Die unteren beiden Schichten des VINES-Stack sind durch eine Vielzahl bekannter Mechanismen für den Zugriff auf Medien implementiert, zu denen High Level Data Link Control (HDLC), X.25, Ethernet und Token-Ring zählen.

31.3 Netzwerkschicht

VINES verwendet das *VINES Internetwork Protocol* (VIP), um Schicht-3-Handlungen (einschließlich Netzwerkrouting) auszuführen. Weiterhin unterstützt VINES sein eigenes *Address Resolution Protocol* (ARP), seine eigene, *Routing Table Protocol* (RTP) genannte Fassung des *Routing Information Protocol* (RIP) sowie das *Internet Control Protocol* (ICP), das eine Exceptionbehandlung und besondere Informationen für den Aufwand beim Routing bereitstellt. ARP-, ICP- und RTP-Pakete werden in einem VIP-Header gekapselt.

31.3.1 VINES Internetwork Protocol (VIP)

Bei den Adressen der VINES-Netzwerkschicht handelt es sich um Einheiten zu 48 Bit, die sich auf Anteile für das Netzwerk (32 Bit) und das Subnetz (16 Bit) verteilen. Die Netzwerknummer wird besser als eine Server-Nummer bezeichnet, da sie direkt vom *Schlüssel* (einem Hardware-Modul, das eine eindeutige Nummer sowie die Software-Optionen für diesen Server ausweist) des Servers abgeleitet ist. Der Anteil einer VINES-Adresse für das Subnetz wird besser als eine Host-Nummer bezeichnet, da er für die Identifizierung von Hosts in einem VINES-Netzwerk verwendet wird. Im Bild 31.2 ist das VINES-Adreßformat dargestellt.

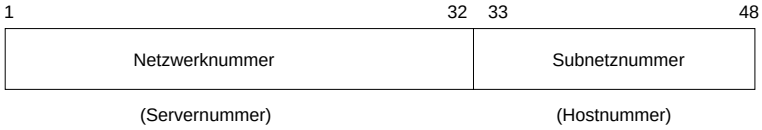


Bild 31.2: Eine VINES-Adresse setzt sich aus einer Netzwerk-Nummer und einer Subnetznummer zusammen

Die Netzwerk-Nummer identifiziert ein logisches VINES-Netzwerk, das als ein Baum mit zwei Ebenen dargestellt wird, wobei ein *Dienstknoten* die Wurzel bildet. Dienstknoten, bei

denen es sich üblicherweise um Server handelt, stellen Dienste zum Auflösen von Adressen und zum Routing für *Clients* bereit, welche die Blätter des Baums darstellen. Der Dienstknoten weist Clients VIP-Adressen zu.

Sobald ein Client eingeschaltet wird, überträgt er einen Request an Server, und alle Server, die diesen Request mitbekommen, reagieren darauf. Der Client greift das erste Response-Paket (Antwort/Reaktion) auf und ersucht um eine Subnetz- bzw. Hostadresse von diesem Server. Der Server antwortet mit einer Adresse, die aus seiner eigenen (von seinem Schlüssel abgeleiteten) Netzwerk-Adresse zusammen mit einer von ihm gewählten Subnetzadresse besteht. Subnetzadressen für Clients werden üblicherweise bei 80001H beginnend aufeinanderfolgend zugewiesen. Die Subnetzadresse des Servers ist immer 1. Im Bild 31.3 ist der Vorgang der VINES-Adreßermittlung dargestellt.

Die dynamische Adreßzuweisung ist in der Wirtschaft nicht einzigartig (auch AppleTalk geht auf diese Weise vor), aber sie ist sicherlich nicht so verbreitet wie die statische Adreßzuweisung.

Da Adressen exklusiv von einem bestimmten Server ausgewählt werden (dessen Adresse aufgrund des Hardware-Schlüssels einzigartig ist), ist die Gefahr einer mehrfachen Adreßvergabe äußerst gering. Hierbei kann man von Glück sprechen, weil eine mehrfache Adreßvergabe für das *Internet Protocol* (IP) und andere Netzwerke möglicherweise verheerende Folgen nach sich ziehen kann.

Im VINES-Netzwerk-Schema stellen alle Server mit mehreren Schnittstellen im wesentlichen Router dar. Clients wählen immer ihren eigenen Server als Router für den ersten Hop (Sprung), selbst dann, wenn ein anderer Server auf demselben Kabel eine bessere Route zum letztendlichen Ziel bereitstellt. Clients können von anderen Routern erfahren, indem sie Redirect-Nachrichten von ihrem eigenen Server erhalten. Da Clients beim Routing für den ersten Hop auf ihren Server angewiesen sind, unterhalten VINES-Server Routing-Tabellen, um ihnen beim Ausfindigmachen entfernter Knoten behilflich zu sein.

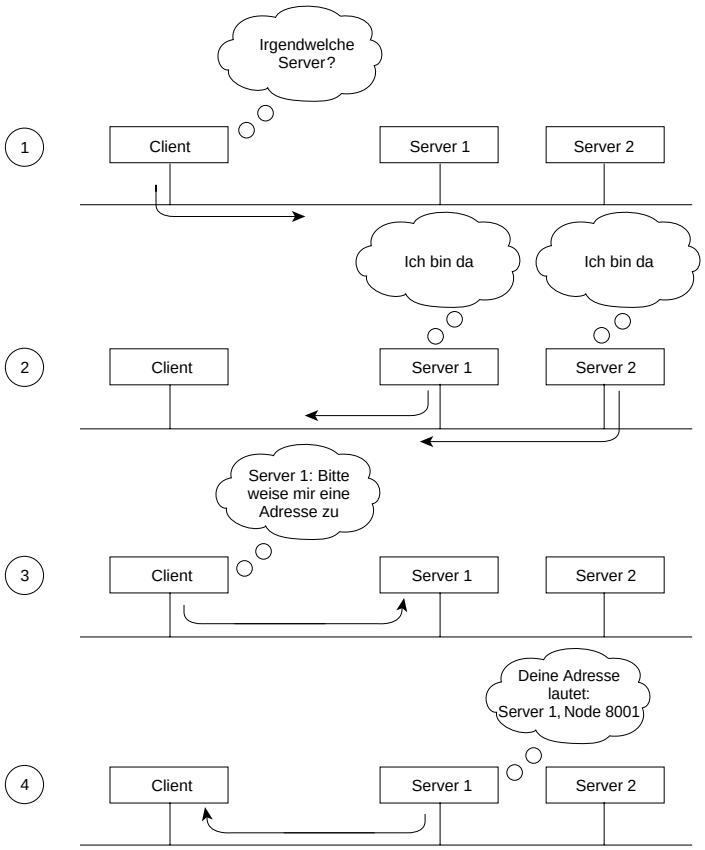


Bild 31.3: VINES durchläuft für die Ermittlung einer Adresse vier Schritte

VINES-Routing-Tabellen bestehen aus Host/Aufwand-Paaren, wobei der *Host* einem erreichbaren Netzknoten und *Aufwand* einer zum Erreichen des Knotens benötigten Verzögerung (in Millisekunden) entspricht. RTP unterstützt VINES-Server dabei, benachbarte Clients, Server und Router ausfindig zu machen.

Regelmäßig machen Clients sowohl die Adresse ihrer Netzwerkschicht als auch ihrer MAC-Schicht mit der Entsprechung eines Hello-Pakets bekannt, mit dem angezeigt wird, daß der Client noch immer in Betrieb ist und für das Netzwerk bereitsteht. Die Server wiederum verschicken regelmäßig Routing-Updates an andere Server, um andere Router auf geänderte Knotenadressen und Änderungen in der Netzwerk-Topologie aufmerksam zu machen.

Wenn ein VINES-Server ein Paket erhält, prüft er nach, ob das Paket für einen anderen Server bestimmt ist oder ob es sich um einen Broadcast handelt. Wenn der aktuelle Server das Ziel ist, behandelt der Server den Request entsprechend. Wenn ein anderer Server das Ziel ist, schickt der Server das Paket entweder direkt weiter (wenn es sich bei dem Server um einen Nachbarn handelt) oder er leitet es an den nächsten Server weiter. Wenn es sich bei dem Paket um einen Broadcast handelt, überprüft der Server, ob das Paket über die günstigste Route angekommen ist. Falls nicht, wird das Paket verworfen. Falls ja, wird das Paket über alle Schnittstellen außer derjenigen weitergeschickt, an der es angekommen ist. Dieses Vorgehen hilft dabei, die Anzahl von Broadcaststürmen zu verringern, die in anderen Netzwerk-Umgebungen ein übliches Problem darstellen. Der VINES-Routing-Algorithmus ist im Bild 31.4 dargestellt.

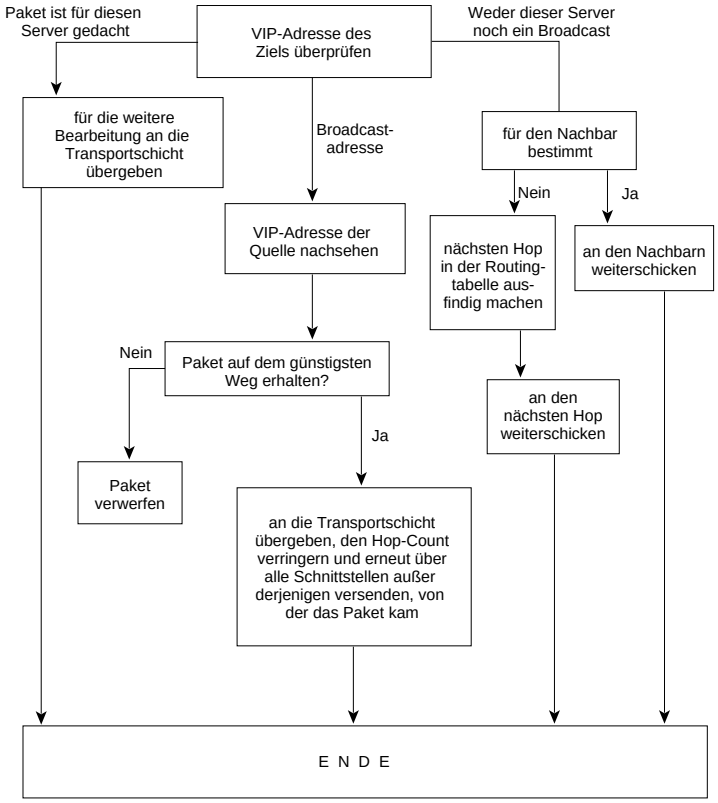


Bild 31.4: Der VINES-Routing-Algorithmus legt den geeigneten Pfad zu einem Ziel fest

In Bild 31.5 ist das Format des VIP-Pakets dargestellt.

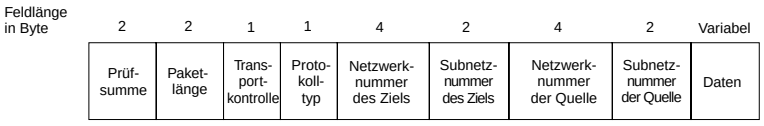


Bild 31.5: Ein VIP-Paket setzt sich aus neun Einzelfeldern zusammen

Zu den Feldern eines VIP-Pakets gehören eine *Prüfsumme*, *Paketlänge*, *Transportkontrolle*, *Protokollart*, *Netzwerknummer des Ziels*, *Subnetznummer des Ziels*, *Netzwerknummer der Quelle* und die *Subnetznummer der Quelle*.

Das Feld *Prüfsumme* wird verwendet, um Beschädigungen am Paket festzustellen. Das Feld *Paketlänge* gibt den Umfang des gesamten VIP-Pakets an.

Das Feld *Transportkontrolle* setzt sich aus mehreren Teilfeldern zusammen. Wenn es sich bei dem Paket um ein Broadcast-Paket handelt, dann gibt es zwei Teilfelder: *Klasse* (Bit 1 bis 3) und *Hop-Count* (Bit 4 bis 7). Wenn es sich nicht um ein Broadcast-Paket handelt, dann stehen vier Teilfelder zur Verfügung: *Fehler*, *Metrik*, *Redirect* und *Hop-Count*. Das Teilfeld *Klasse* gibt die Art des Knotens an, der den Broadcast empfangen soll. Zu diesem Zweck werden Knoten entsprechend der Art des Knotens und der Verbindung, in welcher der Knoten gefunden wurde, in verschiedene Kategorien unterteilt. Durch die Angabe der Art von Knoten für den Empfang der Broadcasts reduziert das Teilfeld *Klasse* die durch Broadcasts verursachten Störungen. Das Teilfeld *Hop-Count* steht für die Anzahl von Hops (Routern), die das Paket durchlaufen mußte. Das Teilfeld *Fehler* gibt an, ob das ICP-Protokoll ein Exception-Notification-Paket an die Paketquelle schicken soll, falls sich ein Paket als unzustellbar erweist. Das Teilfeld *Metrik* wird von einer Transporteinheit auf eins gesetzt, sofern sie etwas über den Routingaufwand für das Bewegen von Paketen zwischen einem Dienstknoten und einem Nachbarn erfahren muß. Das Teilfeld *Redirect* gibt an, ob der Router gegebenenfalls ein Redirect erzeugen soll.

Das Feld *Protokollart* gibt das Protokoll der Netzwerk- oder Transportschicht an, für welches das Metric- oder das Exception-Notification-Paket bestimmt ist.

Schließlich stellen die Felder *Netzwerknummer des Ziels*, *Subnetznummer des Ziels*, *Netzwerknummer der Quelle* und *Subnetznummer der Quelle* alle eine VIP-Adresse bereit.

31.3.2 Routing Table Protocol (RTP)

RTP stellt Informationen über die Netzwerk-Topologie zur Verfügung. Sowohl von Clients als auch von Dienstknoten werden regelmäßig Routing-Update-Pakete übertragen. Diese Pakete informieren Nachbarn über das Vorhandensein eines Knotens und zeigen außerdem an, ob es sich bei dem Knoten um einen Client oder einen Dienstknoten handelt. Dienstknoten schließen in jedem Routing-Update-Paket eine Auflistung sämtlicher bekannter Netzwerke sowie die Aufwandsfaktoren mit ein, die mit dem Erreichen jener Netzwerke verbundenen sind.

Es werden zwei Tabellen geführt: eine *Tabelle aller bekannten Netzwerke* und eine *Nachbartabelle*. Für Dienstknoten enthält die Tabelle aller bekannten Netzwerke mit Ausnahme des eigenen Netzwerks des Dienstknotens einen Eintrag für jedes bekannte Netzwerk. Jeder Eintrag besteht aus einer Netzwerknummer, einer Routing-Metrik und einem Zeiger auf den Eintrag für den nächsten Hop im Netzwerk aus der Tabelle der Nachbarn. Die Tabelle der Nachbarn enthält für jeden benachbarten Dienstknoten und Clientknoten jeweils einen Eintrag. Die Einträge bestehen aus einer Netzwerknummer, einer Subnetznummer, dem für das Erreichen dieses Knotens verwendeten Protokoll für den Medienzugriff (beispielsweise Ethernet), einer LAN-Adresse (sofern es sich bei dem den Nachbar anbindenden Medium um ein Local Area Network handelt) und einer Metrik für den Nachbarn.

RTP spezifiziert vier Pakettypen: *Routing-Update*, *Routing-Request*, *Routing-Response* und *Routing-Redirect*. Routing-Updates werden regelmäßig abgesetzt, um das Vorhandensein von Nachbarn einer Einheit mitzubekommen. Routing-Requests werden von Einheiten ausgetauscht, wenn sie die Netzwerk-Topologie schnell erfahren müssen. Routing-Response-Pakete enthalten Informationen über die Topologie und werden von Dienstknoten verwendet, um auf Routing-Request-Pakete zu reagieren. Ein Routing-Redirect-Paket stellt

Knoten, die ungünstige Routen verwenden, bessere Routeninformationen zur Verfügung.

RTP-Pakete besitzen einen 4 Byte großen Header, der sich aus folgenden Feldern zu je 1 Byte zusammensetzt: Das Feld *Betriebsart* gibt den Pakettyp an; das Feld *Knotenart* gibt an, ob das Paket von einem Dienstknoten oder einem Nicht-Dienstknoten stammt; das Feld *Controllerart* gibt an, ob der Controller des das RTP-Paket übermittelnden Knotens über einen Multibuffer-Controller verfügt; das Feld *Rechnerart* gibt an, ob der Prozessor des RTP-Senders schnell oder langsam ist.

Die Felder *Controllerart* und *Rechnerart* werden für das Pacing eingesetzt.

31.3.3 Address Resolution Protocol (ARP)

Das Address Resolution Protocol (ARP) verwendende Einheiten werden entweder als *Adressen auflösende Clients* oder als *Adressen auflösende Dienste* klassifiziert. Adressen auflösende Clients sind üblicherweise auf Clientknoten implementiert, wohingegen Adressen auflösende Dienste typischerweise von Dienstknoten bereitgestellt werden.

ARP-Pakete besitzen einen 8 Byte großen Header, der sich aus einem 2 Byte großen *Pakettyp*, einer 4 Byte großen *Netzwerknummer* und einer 2 Byte großen *Subnetznummer* zusammensetzt. Es gibt vier Pakettypen: ein *Query-Request* fordert einen ARP-Dienst an; mit einer *Service-Response* wird auf einen Query-Request reagiert; ein *Assignment-Request* wird an einen ARP-Dienst geschickt, um eine VINES-Netzwerk-Adresse anzufordern; und eine *Assignment-Response* wird vom ARP-Dienst als Reaktion auf den Assignment-Request verschickt. Die Felder für die Netzwerk-Nummer und die Subnetznummer sind nur in einem Assignment-Response-Paket von Bedeutung.

ARP-Clients und -Dienste implementieren folgenden Algorithmus beim Hochfahren eines Client. Zuerst überträgt der Client Query-Request-Pakete. Darauf reagiert jeder mit dem Client benachbarte Dienst mit einem Service-Response-Paket. Anschließend setzt der Client ein Assignment-Request-Paket an den ersten Dienst ab, der auf sein Query-Request-Paket ge-

antwortet hat. Der Dienst reagiert mit einem Assignment-Response-Paket, das die zugewiesene Netzwerk-Adresse enthält.

31.3.4 Internet Control Protocol (ICP)

Das Internet Control Protocol (ICP) spezifiziert die *Exception-Notification-* und *Metric-Notification*-Pakete. Exception-Notification-Pakete stellen Informationen über Exceptions der Netzwerk-Schicht zur Verfügung; Metric-Notification-Pakete enthalten Informationen über die abschließende zum Erreichen eines Client-Knotens verwendete Übermittlung.

Exception-Notifications werden verschickt, wenn ein VIP-Paket nicht richtig weitergeleitet werden kann und das Teilfeld *Fehler* im Feld *Transportkontrolle* des VIP-Headers aktiviert ist. Diese Pakete enthalten außerdem ein Feld, das die jeweilige Exception mittels ihres Fehlercodes ausweist.

ICP-Einheiten in Dienstknoten erzeugen dann Metric-Notification-Nachrichten, wenn das Teilfeld *Metrik* im Feld *Transportkontrolle* des VIP-Header aktiviert ist und die Zieladresse im Paket des Dienstknotens einen Nachbarn des Dienstknotens angibt.

31.4 Transportschicht

VINES stellt drei Dienste für die Transportschicht zur Verfügung: den *Unreliable Datagram Service*, den *Reliable Message Service* und den *Data Stream Service*.

Der *Unreliable Datagram Service* (nicht zuverlässige Datagramme versendender Dienst) verschickt Pakete, die so gut wie möglich weitergeleitet, am Ziel aber nicht bestätigt werden.

Beim *Reliable Message Service* (zuverlässige Nachrichten versendender Dienst) handelt es sich um einen virtuellen Verbindungsdienst, der für eine zuverlässige und bestätigte Zustellung von Nachrichten zwischen Netzknoten sorgt. Eine zuverlässige Nachricht kann in maximal vier VIP-Paketen übermittelt werden.

Der *Data Stream Service* (Datenstrom versendender Dienst) unterstützt einen kontrollierten Datenfluß zwischen zwei Pro-

zessen. Beim Data Stream Service handelt es sich um einen bestätigten virtuellen Verbindungsdienst, der die Übermittlung von Nachrichten mit unbegrenztem Umfang unterstützt.

31.5 Protokolle übergeordneter Schichten

Als verteiltes Netzwerk greift VINES für die Kommunikation zwischen Clients und Servern auf das *Remote Procedure Call*-Protokoll (RPC) zurück. RPC ist die Grundlage von Umgebungen mit verteilten Diensten. Das *NetRPC Protocol* (Schichten 5 und 6) stellt eine hochwertige Programmiersprache zur Verfügung, die einen Zugriff auf entfernte Dienste gestattet, der sowohl für den Benutzer als auch für die Anwendung transparent ist.

Auf der Schicht 7 stellt VINES neben Anwendungen für den Dateidienst und Druckerdienst auch *StreetTalk* zur Verfügung, das einen global konsistenten Namensdienst für ein vollständiges Internetzwerk bereitstellt.

VINES stellt außerdem für verschiedene Betriebssysteme wie beispielsweise DOS und UNIX eine integrierte Entwicklungsumgebung für Anwendungen zur Verfügung. Diese Entwicklungsumgebung ermöglicht es Fremdanbietern, sowohl Clients als auch Dienste zu entwickeln, die in der VINES-Umgebung laufen.

KAPITEL 32

Xerox Network Systems (XNS)

32.1 Hintergrund

Die Protokolle des Xerox Network Systems (XNS) wurden in den späten 1970er und frühen 1980er Jahren von der Xerox Corporation entwickelt. Sie wurden für den Einsatz mit einer Vielzahl von Kommunikationsmedien, Prozessoren und Büroanwendungen entworfen. Mehrere XNS-Protokolle ähneln dem Internet Protocol (IP) und dem Transmission Control Protocol (TCP), die von der Defense Advanced Research Projects Agency (DARPA) für das U.S. Department of Defense entwickelt wurden.

Wegen seiner Verfügbarkeit und seiner frühen Markteinführung wurde XNS von den meisten der frühen LAN-Firmen wie beispielsweise Novell, Inc., Ungermann-Bass, Inc. (jetzt ein Teil von Tandem Computers) und 3Com Corporation aufgegriffen. Seitdem hat jede dieser Firmen verschiedene Änderungen an den XNS-Protokollen vorgenommen. Novell fügte das Service Advertisement Protocol (SAP) hinzu, um Ressourcen bekanntmachen zu können und modifizierte die OSI-Protokolle der Schicht 3 (die Novell in IPX für Internetwork Packet Exchange umbenannte), damit sie mit IEEE-802.3- statt Ethernet-Netzwerken laufen. Ungermann-Bass modifizierte RIP, damit sowohl Verzögerung als auch Hop-Count unterstützt werden, und führte weitere kleinere Änderungen durch. Mit der Zeit wurde die XNS-Implementierung für die Arbeit mit PC-Netzwerken beliebter als das XNS, wie es von Xerox entworfen wurde. Dieses Kapitel liefert eine Zusammen-

fassung des XNS-Protokollstacks im Zusammenhang mit dem OSI-Referenzmodell.

32.2 Übersicht über die XNS-Hierarchie

Obwohl die Entwurfziele bei XNS mit denen des OSI-Referenzmodells übereinstimmen, unterscheidet sich das XNS-Konzept einer Protokollhierarchie doch etwas von der durch das OSI-Referenzmodell bereitgestellten wie es Bild 32.1 verdeutlicht.

Wie im Bild 32.1 dargestellt, liefert Xerox ein 5-Schichten-Modell der Paketkommunikation. Schicht 0 entspricht grob den OSI-Schichten 1 und 2, die für den Zugriff auf die Verbindung und die Beeinflussung des Bitstroms zuständig sind. Schicht 1 entspricht grob dem Anteil der OSI-Schicht 3, der den Netzwerkverkehr betrifft. Schicht 2 entspricht dem Anteil der OSI-Schicht 3, der das Routing im Internetzwerk betrifft, sowie der OSI-Schicht 4, die für die Kommunikation der Prozesse untereinander zuständig ist. Die Schichten 3 und 4 entsprechen grob den oberen beiden Schichten des OSI-Modells, die für die Strukturierung der Daten, die Interaktion von Prozeß zu Prozeß sowie die Anwendungen zuständig sind. XNS verfügt über kein Protokoll, das der OSI-Schicht 5 (die Kommunikationsschicht) entspricht.

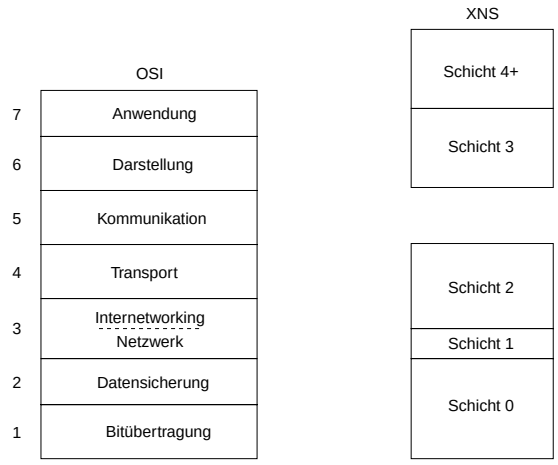


Bild 32.1: Xerox entschied sich für ein 5-Schichten-Modell der Paketkommunikation

32.3 Medienzugriff

Obwohl die XNS-Dokumentation X.25, Ethernet und High Level Data Link Control (HDLC) erwähnt, definiert XNS nicht ausdrücklich, auf welches Protokoll für die Schicht 0 sich das System bezieht. Wie viele andere Protokollsammlungen auch, läßt XNS die Frage nach dem Medienzugriff offen, wobei implizit beliebige solcher Protokolle für den Transport von XNS-Paketen über ein physikalisches Medium zugelassen sind.

32.4 Netzwerkschicht

Das XNS-Protokoll für die Netzwerkschicht heißt *Internet Datagram Protocol* (IDP). IDP führt standardmäßige Schicht-3-Funktionen durch, wozu die logische Adressierung sowie die Datagrammzustellung von Endsystem zu Endsystem über ein Internetzwerk gehören. Im Bild 32.2 ist das Format für ein IDP-Paket dargestellt.

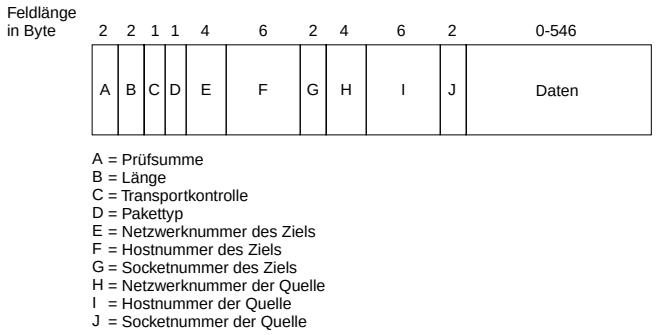


Bild 32.2: Ein IDP-Paket besteht aus elf Feldern

Im folgenden werden die im Bild 32.2 dargestellten Felder eines IDP-Pakets zusammenfassend beschrieben:

- *Prüfsumme* – Ein 16 Bit großes Feld, das beim Einschätzen der Integrität des Pakets hilft, nachdem es das Internetzwerk durchquert hat.
- *Länge* – Ein 16 Bit großes Feld, das den Gesamtumfang (einschließlich der Prüfsumme) des aktuellen Datagramms enthält.

- *Transportkontrolle* – Ein 8 Bit großes Feld, das die Teilfelder *Hop-Count* und *maximale Paketlebensdauer* (MPL) enthält. Das Teilfeld Hop-Count wird von der Quelle mit 0 initialisiert und jedesmal um eins erhöht, wenn das Datagramm einen Router durchläuft. Sobald das Feld Hop-Count den Wert 16 erreicht, wird das Datagramm in der Annahme verworfen, daß eine Schleife beim Routing aufgetreten ist. Das Teilfeld MPL enthält die maximal zur Verfügung stehende Zeit (in Sekunden), die ein Paket im Internetzwerk verbleiben kann.
- *Pakettyp* – Ein 8 Bit großes Feld, welches das Format des Datenfelds angibt.
- *Netzwerknummer des Ziels* – Ein 32 Bit großes Feld, welches das Zielnetzwerk in einem Internetzwerk eindeutig ausweist.
- *Hostnummer des Ziels* – Ein 48 Bit großes Feld, welches den Ziel-Host eindeutig ausweist.
- *Socketnummer des Ziels* – Ein 16 Bit großes Feld, welches einen Socket (Prozeß) innerhalb des Ziel-Host eindeutig ausweist.
- *Netzwerknummer der Quelle* – Ein 32 Bit großes Feld, welches das Quellnetzwerk in einem Internetzwerk eindeutig ausweist.
- *Hostnummer der Quelle* – Ein 48 Bit großes Feld, welches den Quell-Host eindeutig ausweist.
- *Socketnummer der Quelle* – Ein 16 Bit großes Feld, welches einen Socket (Prozeß) innerhalb des Quell-Host eindeutig ausweist.

IEEE-802-Adressen sind äquivalent zu Host-Nummern, so daß Hosts, die an mehr als ein IEEE-802-Netzwerk angebunden sind, über dieselbe Adresse in jedem Segment verfügen. Dadurch werden Netzwerknummern zwar redundant, sie bleiben aber für das Routing hilfreich. Bestimmte Socketnummern sind *bekannt*, was zur Folge hat, daß der Dienst fest definiert ist, der von der Software ausgeführt wird, die auf diese Nummern zurückgreift. Alle weiteren Socketnummern sind wiederverwendbar.

XNS unterstützt die Kapselung Ethernet Version 2.0 für Ethernet und drei Arten der Kapselung für Token Ring: 3Com, SubNet Access Protocol (SNAP) und Ungermann-Bass.

XNS unterstützt *Unicast*-Pakete (Punkt-zu-Punkt), *Multicast*-Pakete (Mehrpunkt-zu-Mehrpunkt) und *Broadcast*-Pakete (Punkt-zu-Mehrpunkt). Multicast- und Broadcast-Adressen werden weiterhin in *gelenkte* und *globale* Typen unterteilt. Gelenkte Multicasts stellen allen Mitgliedern einer Multicast-Gruppe in demjenigen Netzwerk Pakete zu, das in der Netzwerk-Adresse des Ziel-Multicast angegeben ist. Gelenkte Broadcasts stellen allen Mitgliedern eines angegebenen Netzwerks Pakete zu. Globale Multicasts stellen allen Mitgliedern der Gruppe im gesamten Internetzwerk Pakete zu, während globale Broadcasts allen Internetzwerk-Adressen Pakete zustellen. Ein Bit in der Hostnummer unterscheidet eine Unicast- von einer Multicast-Adresse. Im Gegensatz dazu weist ein mit Einsen gefülltes Feld für die Host-Adresse auf eine Broadcast-Adresse hin.

Um Pakete in einem Internetzwerk zu routen, greift XNS auf das dynamische Routing-Schema von RIP zurück. Derzeit stellt RIP das am häufigsten verwendete *Interior Gateway Protocol* (IGP) in der Internet-Gemeinschaft dar. Weitere Informationen über das RIP finden Sie im Kapitel 42, »Routing Information Protocol (RIP)«.

32.5 Transportschicht

Die Funktionen der OSI-Transportschicht werden durch verschiedene Protokolle implementiert. Jedes der folgenden Protokolle ist in der XNS-Spezifikation als ein Schicht-2-Protokoll beschrieben.

Das *Sequenced Packet Protocol* (SPP) bietet eine zuverlässige, verbindungsorientierte, ablaufkontrollierte Paketübertragung im Auftrag von Client-Prozessen. Von der Funktionalität her ähnelt es dem *Transmission Control Protocol* (TCP) der Internet-Protokollsammlung (IP) und dem *Transport Protocol 4* (TP4) der OSI-Protokollsammlung. Weitere Informationen zu TCP finden Sie in Kapitel 28, »Internet-Protokolle«. Weitere Informationen zu TP4 finden Sie in Kapitel 30, »Protokolle der Open System Interconnection (OSI)«.

Jedes SPP-Paket enthält eine Sequenznummer, die verwendet wird, um die Pakete zu sortieren und festzustellen, ob irgendwelche vervielfältigt worden oder verlorengegangen sind. SPP-Pakete enthalten weiterhin zwei 16 Bit große *Verbindungs-IDs*. Je eine Verbindungs-ID wird von jedem Ende der Verbindung angegeben, und zusammengenommen weisen diese beiden Verbindungs-IDs eine logische Verbindung zwischen Client-Prozessen eindeutig aus.

SPP-Pakete können nicht mehr als 576 Byte umfassen. Client-Prozesse können während des Verbindungsaufbaus den Einsatz einer anderen Paketgröße aushandeln, aber SPP definiert die Art und Weise dieses Aushandelns nicht.

Bei dem *Packet Exchange Protocol* (PEP) handelt es sich um ein Request-Response-Protokoll, das entworfen wurde, um zuverlässiger zu sein als ein einfacher Datagrammdienst (wie er beispielsweise von IDP bereitgestellt wird), aber weniger zuverlässig zu sein als SPP. PEP ähnelt von der Funktionalität her dem *User Datagram Protocol* (UDP) der Internet-Protokollsammlung. Weitere Informationen zu UDP finden Sie in Kapitel 28. PEP basiert auf Einzelpaketen und stellt eine erneute Übermittlung bereit, aber kein Erkennen vervielfältigter Pakete. Somit ist es in Anwendungen von Nutzen, in denen Request-Response-Transaktionen wiederholt werden können, ohne daß Daten beschädigt werden, oder in denen die zuverlässige Übermittlung auf einer anderen Schicht ausgeführt wird.

Das *Error Protocol* (EP) kann von jedem Client-Prozeß verwendet werden, um einen anderen Client-Prozeß darauf hinzuweisen, daß ein Netzwerkfehler aufgetreten ist. Dieses Protokoll kommt beispielsweise in Situationen zum Einsatz, in denen eine SPP-Implementierung ein vervielfältigtes Paket festgestellt hat.

32.6 Protokolle übergeordneter Schichten

XNS bietet mehrere Protokolle für die übergeordneten Schichten an. Das *Printing Protocol* stellt Druckerdienste, das *Filing Protocol* Dienste für den Dateizugriff und das *Clearinghouse Protocol* Namensdienste bereit. Jedes dieser drei Protokolle läuft auf dem *Courier Protocol* ab, das Konventionen für die

Strukturierung von Daten und für die Prozeßinteraktion bereitstellt.

XNS definiert außerdem Schicht-4-Protokolle, bei denen es sich um Anwendungsprotokolle handelt. Aber da sie nur wenig mit den eigentlichen Kommunikationsfunktionen zu tun haben, umfaßt die XNS-Spezifikation keine sachgemäßen Definitionen.

Das *Echo Protocol* der Schicht 2 wird verwendet, um die Erreichbarkeit von XNS Netzknoten zu überprüfen und um Funktionen zu unterstützen, wie sie beispielsweise von dem Befehl PING in UNIX und anderen Umgebungen bereitgestellt werden.

Kapitel 33: Border Gateway Protocol (BGP)
Kapitel 34: Enhanced IGRP
Kapitel 35: IBM Systems Network Architecture (SNA) Routing
Kapitel 36: Interior Gateway Routing Protocol (IGRP)
Kapitel 37: Internet Protocol (IP) Multicast
Kapitel 38: NetWare Link Services Protocol (NLSP)
Kapitel 39: Open Systems Interconnection (OSI) Routing-Protokoll
Kapitel 40: Open Shortest Path First (OSPF)
Kapitel 41: Resource Reservation Protocol (RSVP)
Kapitel 42: Routing Information Protocol (RIP)
Kapitel 43: Simple Multicast Routing Protocol (SMRP)

TEIL 6

Routing-Protokolle

Der Teil 6, »Routing-Protokolle«, faßt die Eigenschaften und Operationen geläufiger Routing-Protokolle zusammen. Folgende Themen werden in einzelnen Kapiteln besprochen:

Border Gateway Protocol (BGP) – Beschreibt die Operationen von BGP, einem externen Gateway-Protokoll aus der Internet Protokollsammlung (IP).

Enhanced Interior Gateway Routing Protocol (Enhanced IGRP) – Beschreibt die Eigenschaften und Operationen von Enhanced IGRP.

IBM Systems Network Architecture (SNA) Routing – Liefert Beschreibungen von Komponenten und Operationen des IBM SNA Routing in Subarea- und APPN-Routing-Umgebungen.

Interior Gateway Routing Protocol (IGRP) – Beschreibt die Eigenschaften und Operationen von IGRP.

Internet Protocol (IP) Multicast – Beschreibt das IP-Multicast und umreißt die grundlegenden Operationen verschiedener unterstützender Protokolle.

NetWare Link Services Protocol (NLSP) – Beschreibt die Komponenten, Eigenschaften und Operationen von Novell NLSP.

Open Systems Interconnection (OSI) Routing – Behandelt die Eigenschaften und Operationen der OSI-Routing-Protokolle: ES-IS, IS-IS, Integrated IS-IS und IDRP.

Open Shortest Path First (OSPF) – Behandelt die Komponenten und Operationen dieses internen Gateway-Routing-Protokolls für Link-State Routing.

KAPITEL 33

Border Gateway Protocol (BGP)

33.1 Hintergrund

Zum Routing gehören zwei grundlegende Tätigkeiten: das Ermitteln der optimalen Routingpfade und der Transport von Informationsgruppen (typischerweise als Pakete bezeichnet) über ein Internet-Netzwerk. Der Transport von Paketen über ein Internet-Netzwerk erfolgt verhältnismäßig direkt. Das Ermitteln der Pfade kann andererseits sehr kompliziert sein. Eines der Protokolle, die sich in heutigen Netzwerken mit der Aufgabe beschäftigen, Pfade zu ermitteln, ist das *Border Gateway Protocol* (BGP). Dieses Kapitel liefert eine Zusammenfassung der grundlegenden Operationen von BGP und eine Beschreibung der Komponenten des Protokolls.

BGP führt das Routing zwischen Domains eines Transmission Control Protocol/Internet Protocol-Netzwerks (TCP/IP) durch. Bei BGP handelt es sich um ein Exterior Gateway Protocol (EGP); das heißt, es führt das Routing zwischen mehreren autonomen Systemen oder Domains durch und tauscht Informationen über das Routing und die Erreichbarkeit mit anderen BGP-Systemen aus.

BGP wurde entwickelt, um seinen Vorgänger, das mittlerweile überholte *Exterior Gateway Protocol* (EGP), als das standardmäßig im globalen Internet eingesetzte externe Gateway-Routing-Protokoll zu ersetzen. BGP löst schwerwiegende Pro-

bleme mit EGP und paßt sich dem Wachstum des Internet effizienter an.

Hinweis: Bei EGP handelt es sich um ein bestimmtes Beispiel für ein externes Gateway-Protokoll (ebenfalls mit EGP abgekürzt) – die beiden sollten nicht durcheinander gebracht werden.

Im Bild 33.1 sind Core-Router dargestellt, die für das Routing des Verkehrs zwischen verschiedenen autonomen Systemen auf BGP zurückgreifen.

BGP wird in mehreren *Request For Comments* (RFCs) spezifiziert:

- RFC 1771 – Beschreibt BGP4, die aktuelle Version von BGP
- RFC 1654 – Beschreibt die erste BGP4-Spezifikation
- RFC 1105, RFC 1163 und RFC 1267 – Beschreiben die vor BGP4 liegenden Versionen von BGP

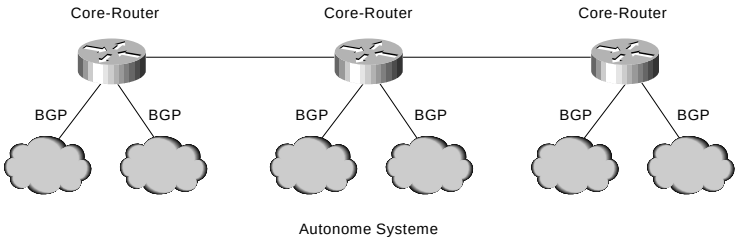


Bild 33.1: Core-Router können für das Routing des Verkehrs zwischen autonomen Systemen auf BGP zurückgreifen

33.2 BGP-Operationen

BGP führt drei Arten von Routing durch: *Interautonomous System Routing, Intra-Autonomous System Routing* und *Pass-Through Autonomous System Routing.*

Interautonomous System Routing kommt zwischen zwei oder mehr BGP-Routern aus verschiedenen autonomen Systemen vor. Peer-Router in diesem System verwenden BGP, um eine konsistente Ansicht der Topologie des Internetzwerks zu er-

halten. BGP-Nachbarn, bei denen eine Kommunikation zwischen autonomen Systemen stattfindet, müssen sich im gleichen physikalischen Netzwerk befinden. Das Internet ist ein Beispiel für eine auf diese Routing-Strategie zurückgreifende Einheit, weil es aus autonomen Systemen oder Verwaltungsdomains besteht. Viele dieser Domains stehen für verschiedene Institutionen, Firmen und andere Existenzen, die gemeinsam das Internet ausmachen. BGP wird häufig für die Ermittlung der Verbindungspfade eingesetzt, um für ein optimales Routing innerhalb des Internet zu sorgen.

Intra-Autonomous System Routing kommt zwischen zwei oder mehr BGP-Routern vor, die sich im selben autonomen System befinden. Peer-Router innerhalb desselben autonomen Systems verwenden BGP, um eine konsistente Ansicht der Topologie des Systems zu erhalten. BGP wird außerdem eingesetzt, um zu ermitteln, welcher Router als Verbindungspunkt für bestimmte externe autonome Systeme dienen soll. Wiederum stellt das Internet ein Beispiel für das Routing innerhalb eines autonomen Systems dar. Eine Organisation wie beispielsweise eine Universität kann von BGP Gebrauch machen, um für ein optimales Routing innerhalb ihrer eigenen Verwaltungsdomain oder ihres eigenen autonomen Systems zu sorgen. Das Protokoll BGP stellt Routingdienste sowohl zwischen (*inter*) autonomen Systemen als auch innerhalb (*intra*) eines autonomen Systems zur Verfügung.

Pass-Through Autonomous System Routing kommt zwischen zwei oder mehr BGP-Peer-Routern vor, die den Datenaustausch über ein autonomes System vornehmen, auf dem BGP nicht läuft. In einer Umgebung mit einem durchreichenden (*pass-through*) autonomen System stammt der BGP-Datenverkehr nicht aus dem betroffenen autonomen System und ist auch nicht für einen Knoten dieses autonomen Systems bestimmt. BGP muß dazu mit einem beliebigen innerhalb des autonomen Systems verwendeten Routing-Protokoll interagieren, um den BGP-Datenverkehr erfolgreich durch das autonome System zu transportieren. Im Bild 33.2 ist eine Umgebung mit einem durchreichenden autonomen System dargestellt:

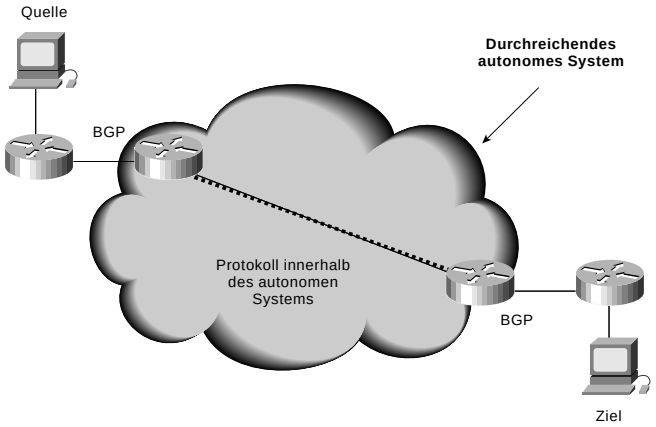


Bild 33.2: Beim Routing durch ein durchreichendes autonomes System arbeitet BGP mit einem anderen innerhalb des autonomen Systems verwendeten Routing-Protokoll zusammen

33.3 BGP-Routing

Wie jedes Routing-Protokoll unterhält BGP eine Routing-Tabelle, übermittelt Routing-Updates und gründet Entscheidungen für das Routing auf Routing-Metriken. Die vordergründige Aufgabe eines BGP-Systems besteht darin, Informationen über die Erreichbarkeit von Netzwerken, wozu auch Informationen über die Liste der Pfade zu autonomen Systemen zählen, mit anderen BGP-Systemen auszutauschen. Mit diesen Informationen läßt sich ein Diagramm der Verbindungen zwischen den autonomen Systemen erstellen, aus denen Schleifen beim Routing entfernt werden können und mit denen grundsätzliche Entscheidungen der autonomen Systemebene durchgesetzt werden können.

Jeder BGP-Router unterhält eine Routing-Tabelle, die alle durchführbaren Pfade zu einem bestimmten Netzwerk aufführt. Der Router erneuert die Routing-Tabelle jedoch nicht. Statt dessen werden von Peer-Routern erhaltene Routing-Informationen so lange zurückgehalten, bis ein inkrementelles Update erhalten wird.

BGP-Geräte tauschen ihre Routing-Informationen beim ersten Datenaustausch und nach inkrementellen Updates aus. Wenn ein Router sich zum ersten Mal an ein Netzwerk anbindet, tauschen BGP-Router ihre gesamten BGP-Routing-Tabellen miteinander aus. Wenn Änderungen an Routing-Tabellen auftreten, versenden Router entsprechend den Anteil ihrer Routing-Tabelle, der sich verändert hat. BGP-Router versenden

keine regelmäßig eingeplanten Routing-Updates, und BGP-Routing-Updates geben nur den optimalen Pfad zu einem Netzwerk bekannt.

BGP greift auf eine einzige Routing-Metrik zurück, um den besten Pfad zu einem gegebenen Netzwerk zu ermitteln. Diese Metrik besteht aus einer beliebigen Einheitszahl, die den Grad der Vorliebe für eine bestimmte Verbindung angibt. Der einer Verbindung zugeordnete Wert kann auf einer beliebigen Anzahl von Kriterien basieren, zu denen u.a. die Anzahl der mit dem Verbindungspfad durchlaufenen autonomen Systeme, die Stabilität, Geschwindigkeit, Verzögerungszeit oder Kosten gehören.

33.4 Nachrichtentypen von BGP

In der RFC 1771, »A Border Gateway Protocol 4(BGP4)«, werden vier Nachrichtentypen für BGP spezifiziert: *Open*-Nachricht, *Update*-Nachricht, *Notification*-Nachricht und *Keep-Alive*-Nachricht.

Die *Open*-Nachricht eröffnet eine BGP-Kommunikationssitzung zwischen Peers und ist die erste von jeder Seite nach dem Aufbau einer Verbindung durch ein Transportprotokoll gesendete Nachricht. Open-Nachrichten werden mittels einer vom Peer-Gerät verschickten Keep-Alive-Nachricht bestätigt, und sie müssen bestätigt worden sein, bevor Updates, Notifications und Keep-Alives ausgetauscht werden können.

Eine *Update*-Nachricht wird verwendet, um anderen BGP-Systemen Routing-Updates bereitzustellen, wodurch Routern das Erstellen einer konsistenten Ansicht der Netzwerk-Topologie ermöglicht wird. Updates werden mittels des Transmission Control Protocol (TCP) verschickt, um eine zuverlässige Zustellung zu gewährleisten. Update-Nachrichten können einen oder mehrere undurchführbare Pfade aus der Routing-Tabelle entfernen und gleichzeitig einen anderen Pfad bekanntgeben.

Die *Notification*-Nachricht wird verschickt, wenn eine Fehlerbedingung entdeckt wird. Notifications werden verwendet, um aktive Arbeitssitzungen zu schließen und um alle ange-

bundenen Router darüber zu informieren, warum die Arbeitssitzung geschlossen wird.

Die *Keep-Alive*-Nachricht teilt BGP-Peers mit, daß ein Gerät aktiv ist. Keep-Alives werden häufig genug verschickt, damit Arbeitssitzungen nicht verfallen.

33.5 BGP-Paketformate

In den folgenden Abschnitten erfolgt eine Zusammenfassung der Nachrichtentypen Open, Update, Notification und Keep-Alive sowie des grundlegenden Header-Formats von BGP. Jedes Format wird durch eine Zeichnung erläutert, und die dargestellten Felder werden definiert.

33.5.1 Header-Format

Alle Nachrichtentypen von BGP greifen auf den zugrundeliegenden Paket-Header zurück. Open-, Update- und Notification-Nachrichten verfügen über zusätzliche Felder, während für Keep-Alive-Nachrichten nur der zugrundeliegende Paket-Header Verwendung findet. Im Bild 33.3 sind die im BGP-Header verwendeten Felder dargestellt. Im folgenden Abschnitt sind die Funktionen jedes Felds zusammengefaßt.

Felder des Paket-Headers von BGP

Jedes BGP-Paket enthält einen Header, dessen vordergründige Aufgabe darin besteht, die Funktion des fraglichen Pakets auszuweisen. In den folgenden Beschreibungen werden die Funktionen jedes Felds des in dem Bild 33.3 dargestellten BGP-Header zusammengefaßt.

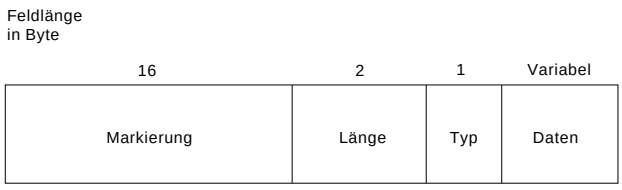


Bild 33.3: Ein BGP-Paket-Header setzt sich aus vier Feldern zusammen

– *Markierung* – Enthält einen Authentifizierungswert, den der Nachrichtenempfänger voraussagen kann.

– *Länge* – Gibt die Gesamtlänge der Nachricht in Byte an.
– *Typ* – Legt den Nachrichtentyp aus einer der folgenden Möglichkeiten fest:
  – Open
  – Update
  – Notification
  – Keep-Alive
– *Daten* – In diesem optionalen Feld sind Informationen für übergeordnete Schichten enthalten.

33.5.2 Format der Open-Nachricht

Die Open-Nachrichten von BGP setzen sich aus einem BGP-Header und zusätzlichen Feldern zusammen. Im Bild 33.4 sind die für Open-Nachrichten zusätzlich verwendeten Felder dargestellt.

Bild 33.4: Eine Open-Nachricht von BGP setzt sich aus sechs Feldern zusammen

Felder der Open-Nachricht von BGP

BGP-Pakete, deren Headerfeld *Typ* das Paket als eine Open-Nachricht ausweist, enthalten folgende Felder. Diese Felder stellen die Austauschkriterien für zwei BGP-Router bereit, damit diese eine Peer-Verbindung aufbauen können.

– *Version* – Stellt die Versionsnummer von BGP bereit, damit der Empfänger feststellen kann, ob auf ihm dieselbe Version läuft wie auf dem Sender.
– *Autonomes System* – Stellt die Nummer des sendenden autonomen Systems bereit.
– *Wartezeit* – Gibt die maximale Anzahl von Sekunden an, die ohne einen Empfang einer Nachricht verstreichen darf, bevor der Übermittler als außer Funktion betrachtet wird.

– *BGP-ID* – Stellt den BGP-Identifizierer des Senders bereit (eine IP-Adresse), der beim Hochfahren bestimmt wird und für sämtliche lokalen Schnittstellen und BGP-Peers identisch ist.

– *Länge der optionalen Parameter* – Gibt die Länge des Felds für die optionalen Parameter an (sofern vorhanden).

– *Optionale Parameter* – Enthält eine Auflistung optionaler Parameter (sofern vorhanden). Derzeit ist nur der optionale Parametertyp *Authentication Information* definiert, der sich aus folgenden beiden Feldern zusammensetzt:

  – *Authentication Code*: Gibt die Art der verwendeten Authentifizierung an.

  – *Authentication Data*: Enthält von der Authentifizierung verwendete Daten (sofern eine zum Einsatz gelangt).

33.5.3 Format der Update-Nachricht

Die Update-Nachrichten von BGP setzen sich aus einem BGP-Header und zusätzlichen Feldern zusammen. Im Bild 33.5 sind die für Update-Nachrichten zusätzlich verwendeten Felder dargestellt.

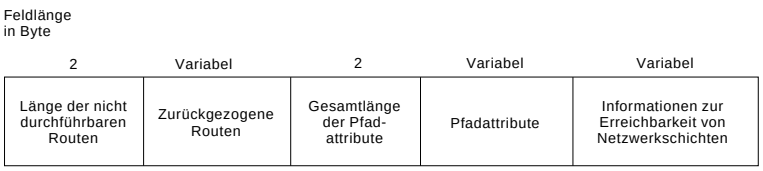


Bild 33.5: Eine Update-Nachricht von BGP setzt sich aus fünf Feldern zusammen

Felder der Update-Nachricht von BGP

BGP-Pakete, deren Headerfeld *Typ* das Paket als eine Update-Nachricht ausweist, enthalten folgende Felder. Durch das Empfangen eines Update-Pakets sind Router dazu in der Lage, bestimmte Einträge zu ihren Routing-Tabellen hinzuzufügen oder aus ihnen zu löschen, um dadurch für ein stimmiges Abbild zu sorgen. Update-Nachrichten setzen sich aus folgenden Feldern zusammen:

– *Länge der nicht durchführbaren Routen* – Zeigt entweder die Gesamtlänge des Felds *Zurückgezogene Routen* oder das Nichtvorhandensein des Felds an.

- *Zurückgezogene Routen* – Enthält eine Auflistung der IP-Adreßpräfixe für außer Dienst gestellte Routen.
- *Gesamtlänge der Pfadattribute* – Zeigt die Gesamtlänge des Felds *Pfadattribute* oder das Nichtvorhandensein des Felds an.
- *Pfadattribute* – Beschreibt die Charakteristik des mitgeteilten Pfads. Folgende Attribute sind für einen Pfad möglich:
  - *Origin*: Vorgeschriebenes Attribut, das die Herkunft (*Origin*) der Pfadinformationen definiert.
  - *AS Path*: Vorgeschriebenes Attribut, das sich aus einer Reihe von Pfadsegmenten der autonomen Systeme (*AS*) zusammensetzt.
  - *Next Hop*: Vorgeschriebenes Attribut, das die IP-Adresse desjenigen Übergangsrouter definiert, der für den nächsten Hop zum Feld *Informationen zur Erreichbarkeit von Netzwerk-Schichten* aufgeführten Zielen verwendet werden soll.
  - *Mult Exit Disc*: Optionales Attribut zum Unterscheiden (*Disc*riminate) mehrerer Übergangspunkte (*Mult*iple *Exit*points) zu einem benachbarten autonomen System.
  - *Local Pref*: Beliebiges Attribut zur Angabe des Grads der Vorliebe (*Pref*erence) für eine mitgeteilte Route.
  - *Atomic Aggregate*: Beliebiges Attribut zur Bekanntgabe von Informationen über die Auswahl von Routen.
  - *Aggregator*: Optionales Attribut, das Informationen über Gesamtrouten (*aggregate* Routes) enthält.
- *Informationen zur Erreichbarkeit von Netzwerk-Schichten* – Enthält eine Auflistung von IP-Adreßpräfixen für die mitgeteilten Routen.

33.5.4 Format der Notification-Nachricht

Im Bild 33.6 sind die für Notification-Nachrichten zusätzlich verwendeten Felder dargestellt.

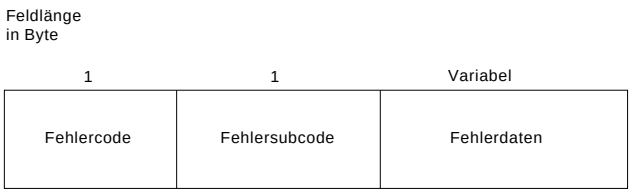


Bild 33.6: Eine Notification-Nachricht von BGP setzt sich aus drei Feldern zusammen

Felder der Notification-Nachricht von BGP

BGP-Pakete, deren Headerfeld *Typ* das Paket als eine Notification-Nachricht ausweist, enthalten folgende Felder. Dieses Paket wird verwendet, um den Peers des Herkunftsrouters eine Art von Fehlerbedingung anzuzeigen.

- *Fehlercode* – Zeigt die Art des aufgetretenen Fehlers an. Folgende Fehlertypen sind durch das Feld definiert:
  - *Message Header Error*: Zeigt ein Problem mit dem Header der Nachricht an wie beispielsweise eine unannehmbare Nachrichtenlänge, einen unannehmbaren Wert im Feld *Markierung* oder einen unannehmbaren Nachrichtentyp.
  - *Open Message Error*: Zeigt ein Problem mit einer Open-Nachricht an wie beispielsweise eine nicht unterstützte Versionsnummer, eine unannehmbare Nummer für das autonome System bzw. die IP-Adresse oder ein nicht unterstützter Authentifizierungscode.
  - *Update Message Error*: Zeigt ein Problem mit einer Open-Nachricht an wie beispielsweise eine mißgebildete Attributliste, ein Fehler in der Attributliste oder ein unzulässiges *Next Hop*-Attribut.
  - *Hold Time Expired*: Zeigt an, daß die Wartezeit abgelaufen ist, nach der ein BGP-Knoten als außer Funktion betrachtet wird.

  - *Finite State Machine Error*: Zeigt ein unberücksichtigtes Ereignis an.
  - *Cease*: Schließt eine BGP-Verbindung auf Anforderung eines BGP-Geräts bei Nichtvorhandensein schwerwiegender Fehler.
- *Fehlersubcode* – Liefert genauere Informationen über die Art des mitgeteilten Fehlers.
- *Fehlerdaten* – Enthält auf den Feldern *Fehlercode* und *Fehlersubcode* beruhende Daten. Dieses Feld wird verwendet, um die Ursache für die Notification-Nachricht festzustellen.

KAPITEL 34

Enhanced IGRP

34.1 Hintergrund

Das Enhanced Internet Gateway Routing Protocol (IGRP) stellt eine Weiterentwicklung des Vorgängers IGRP dar (siehe dazu Kapitel 36, »Internet Gateway Routing Protocol (IGRP)«). Diese Weiterentwicklung liegt in den Änderungen der Netzwerk-Arbeit und den Anforderungen verschiedener umfangreicher Internetzwerke begründet. Das Enhanced IGRP integriert die Fähigkeiten von Link-State-Protokollen in Distance-Vector-Protokolle. Es baut auf dem bei SRI International von Dr. J. J. Garcia-Luna-Aceves entwickelten *Diffusing-Update-Algorithmus* (DUAL) auf.

Enhanced IGRP ist kompatibel zu IGRP-Routern und sorgt für eine problemlose Zusammenarbeit. Ein automatischer Umwandlungsmechanismus ermöglicht die Übernahme von IGRP-Routen in Enhanced IGRP und andersherum, so daß sich Enhanced IGRP nach und nach in ein bestehendes IGRP-Netzwerk einbinden läßt. Da sich die Metriken beider Protokolle direkt ineinander übertragen lassen, sind sie so leicht miteinander zu vergleichen, als ob es sich bei ihnen um Routen handelte, die aus ihren eigenen autonomen Systemen (AS) stammten. Außerdem behandelt Enhanced IGRP IGRP-Routen als externe Routen und bietet dem Netzwerk-Administrator einen Weg, diese anzupassen.

In diesem Kapitel wird ein Überblick über die grundlegenden Operationen und Protokolleigenschaften von Enhanced IGRP gegeben.

34.2 Fähigkeiten und Attribute von Enhanced IGRP

Zu den Schlüsselfähigkeiten, die Enhanced IGRP von anderen Routing-Protokollen unterscheiden, gehören die schnelle Konvergenz, die Unterstützung von Subnetz-Masken mit variabler Länge, partiellen Updates sowie mehreren Protokollen für die Netzwerk-Schicht.

Ein Router, auf dem Enhanced IGRP läuft, speichert alle Routing-Tabellen seiner Nachbarn, so daß er sich schnell an andere Routen anpassen kann. Wenn keine geeignete Route vorhanden ist, fragt Enhanced IGRP seine Nachbarn ab, um eine andere Route zu ermitteln. Diese Abfragen setzen sich fort, bis eine andere Route ausfindig gemacht wurde.

Die Unterstützung von Subnetz-Masken variabler Länge ermöglicht es, daß Routen automatisch nach einer Netzwerk-Nummerngrenze zusammengefaßt werden. Weiterhin läßt sich Enhanced IGRP so konfigurieren, daß eine Zusammenfassung nach jeder beliebigen Bitgrenze für jede beliebige Schnittstelle erfolgen kann.

Enhanced IGRP führt keine regelmäßigen Updates durch. Statt dessen verschickt es partielle Updates nur dann, wenn sich die Metrik für eine Route ändert. Die Weiterleitung von partiellen Updates ist automatisch begrenzt, so daß nur diejenigen Router aktualisiert werden, die diese Informationen benötigen. Als eine Folge dieser beiden Fähigkeiten verbraucht Enhanced IGRP bedeutend weniger Bandbreite als IGRP.

Enhanced IGRP unterstützt AppleTalk, IP und Novell NetWare. Die Implementierung von AppleTalk verbreitet aus dem Routing Table Maintenance Protocol (RTMP) erfahrene Routen weiter. Die IP-Implementierung verbreitet von OSPF, dem Routing Information Protocol (RIP), IS-IS, dem Exterior Gateway Protocol (EGP) oder dem Border Gateway Protocol

(BGP) erfahrene Routen weiter. Die Implementierung von Novell verbreitet aus dem Novell RIP oder dem Service Advertisement Protocol (SAP) erfahrene Routen weiter.

34.3 Zugrundeliegende Prozesse und Techniken

Für eine bessere Leistung beim Routing setzt Enhanced IGRP vier Schlüsseltechniken ein, die gemeinsam für eine Unterscheidung von anderen Routing-Techniken sorgen: *Neighbor Discovery/Recovery*, *Reliable Transport Protocol* (RTP), *DUAL Finite-State Machine* und *protokollabhängige Module*.

Das *Neighbor Discovery/Recovery* wird von Routern verwendet, um dynamisch von anderen Routern aus den direkt angebundenen Netzwerken Kenntnis zu erlangen. Router müssen weiterhin feststellen können, daß ihre Nachbarn unerreichbar werden oder nicht mehr in Betrieb sind. Dieser Vorgang wird mit geringem Overhead durch das regelmäßige Verschicken kleiner Hello-Pakete erreicht. Solange ein Router diese Pakete von einem benachbarten Router empfängt, geht er davon aus, daß der Router funktioniert, und die beiden können Routing-Informationen austauschen.

Das *Reliable Transport Protocol* (RTP) ist für die garantierte, geordnete Zustellung von Enhanced-IGRP-Paketen an sämtliche Nachbarn verantwortlich. Es unterstützt eine gemischte Übertragung von Multicast- oder Unicast-Paketen. Aus Effizienzgründen werden nur bestimmte Enhanced-IGRP-Pakete zuverlässig übertragen. In einem multicastfähigen Netzwerk mit Mehrfachzugriff wie beispielsweise Ethernet ist es nicht erforderlich, Hello-Pakete zuverlässig an alle Nachbarn einzeln zu verschicken. Aus diesem Grund verschickt Enhanced IGRP ein einziges Multicast-Hello-Paket, das einen Indikator enthält, der die Empfänger darüber informiert, daß das Paket nicht bestätigt zu werden braucht. Andere Pakettypen wie beispielsweise Updates weisen im Paket darauf hin, daß ein Acknowledgement erforderlich ist. RTP verfügt über eine Vorkehrung zum schnellen Verschicken von Multicast-Paketen, wenn unbestätigte Pakete anhängig sind, was bei Verbindungen mit sich verändernder Geschwindigkeit hilft, die Konvergenzzeit kurz zu halten

DUAL Finite-State Machine verkörpert den Entscheidungsprozeß für alle Routenberechnungen durch das Verfolgen sämtlicher durch die Nachbarn bekanntgemachten Routen. DUAL greift für die Auswahl effizienter, schleifenfreier Pfade auf Entfernungsinformationen zurück und wählt Routen zum Einfügen in eine Routing-Tabelle basierend auf erreichbaren Nachfolgern aus. Bei einem *erreichbaren Nachfolger* handelt es sich um einen für das Weiterleiten von Paketen verwendeten benachbarten Router, der den günstigsten Pfad zu einem Ziel darstellt, das sich garantiert nicht in einer Routing-Schleife befindet. Sobald ein Nachbar eine Metrik ändert oder eine Änderung in der Topologie auftritt, führt DUAL eine Überprüfung auf erreichbare Nachfolger durch. Wenn einer gefunden wird, verwendet DUAL diesen, um eine unnötige Neubestimmung der Route zu vermeiden. Wenn kein erreichbarer Nachfolger vorhanden ist, Nachbarn das Ziel aber immer noch bekanntmachen, muß eine Neubestimmung (was auch als verteilte Bestimmung bezeichnet wird) erfolgen, um einen neuen Nachfolger festzulegen. Obwohl die Neubestimmung nicht prozessorintensiv ist, beeinflußt es die Konvergenzzeit, so daß es von Vorteil ist, unnötige Neubestimmungen zu vermeiden.

Protokollabhängige Module sind für diejenigen Anforderungen verantwortlich, welche die Protokolle der Netzwerk-Schicht betreffen. Das Modul IP-Enhanced IGRP ist beispielsweise für das Verschicken und Empfangen von Enhanced-IGRP-Paketen verantwortlich, die in IP gekapselt sind. Zusätzlich zeichnet IP-Enhanced IGRP dafür verantwortlich, Enhanced-IGRP-Pakete zu parsen und DUAL die neuen Informationen mitzuteilen, die empfangen worden sind. IP-Enhanced IGRP kümmert sich um das Neuverteilen der von anderen IP-Routing-Protokollen erfahrenen Routen.

34.4 Begriffe zum Routing

Enhanced IGRP beruht auf folgenden vier grundlegenden Begriffen: Nachbartabellen, Topologietabellen, Routenzustände und Routenkennzeichnungen. Jeder dieser Begriffe wird in den folgenden Zusammenfassungen behandelt.

34.4.1 Nachbartabellen

Wenn ein Router einen neuen Nachbarn entdeckt, zeichnet er die Adresse und die Schnittstelle des Nachbarn als Eintrag in der *Nachbartabelle* auf. Es gibt eine Nachbartabelle für jedes protokollabhängige Modul. Wenn ein Nachbar ein Hello-Paket verschickt, gibt er eine Wartezeit bekannt, bei der es sich um die Zeitspanne handelt, für die ein Router einen Nachbarn als erreichbar und in Betrieb behandelt. Wenn innerhalb der Wartezeit kein Hello-Paket empfangen wird, läuft die Wartezeit ab, und DUAL wird über die Änderung in der Topologie informiert.

Der Eintrag in der Nachbartabelle enthält außerdem von RTP benötigte Informationen. Es werden Sequenznummern verwendet, um Acknowledgments mit Datenpaketen in Übereinstimmung zu bringen, und die letzte vom Nachbarn erhaltene Sequenznummer wird aufgezeichnet, damit aus der Reihe tanzende Pakete entdeckt werden. Es wird eine auf Nachbarn bezogene Übertragungsliste verwendet, um Pakete für eine mögliche erneute Übertragung einzureihen. Im Tabelleneintrag werden Zeitmesser für den Hin- und Rückweg vorgehalten, um eine optimale Zeitspanne für die erneute Übertragung abzuschätzen.

34.4.2 Topologietabellen

Die *Topologietabelle* enthält alle von benachbarten Routern mitgeteilten Ziele. Die protokollabhängigen Module füllen die Tabelle auf, und die Tabelle wird von der DUAL Finite-State Machine verwendet. Jeder Eintrag in der Topologietabelle enthält die Zieladresse mit einer Auflistung der Nachbarn, die dieses Ziel mitgeteilt haben. Für jeden Nachbarn zeichnet der Eintrag die mitgeteilte Metrik auf, die der Nachbar in seiner Routing-Tabelle speichert. Distance-Vector-Protokolle müssen die wichtige Regel befolgen, daß, wenn ein Nachbar ein Ziel bekannt macht, er die Route zum Weiterleiten von Paketen verwenden muß.

Die Metrik, die der Router zum Erreichen des Ziels verwendet, ist ebenfalls mit dem Ziel verbunden. Die Metrik, die der Router in der Routing-Tabelle verwendet und anderen Routern mitteilt, ist die Summe der besten von allen Nachbarn

mitgeteilten Metriken plus dem Verbindungsaufwand zum besten Nachbarn.

34.4.3 Routenzustände

Ein Eintrag für ein Ziel in der Topologietabelle kann in einem von zwei Zuständen vorliegen: aktiv oder passiv. Ein Ziel befindet sich im passiven Zustand, falls der Router keine Neubestimmung durchführt, oder im aktiven Zustand, falls der Router eine Neubestimmung durchführt. Wenn erreichbare Nachfolger immer verfügbar sind, braucht ein Ziel niemals in den aktiven Zustand überzugehen, wodurch eine Neubestimmung vermieden wird.

Es kommt zu einer Neubestimmung, wenn ein Ziel nicht über erreichbare Nachfolger verfügt. Der Router stößt die Neubestimmung an, indem er ein Query-Paket an jeden seiner benachbarten Router verschickt. Der benachbarte Router kann ein Reply-Paket verschicken, mit dem angezeigt wird, daß er über einen erreichbaren Nachfolger für das Ziel verfügt, oder er kann ein Query-Paket verschicken, mit dem angezeigt wird, daß er sich an der Neubestimmung beteiligt. Solange sich ein Ziel im aktiven Zustand befindet, kann ein Router die Informationen der Routing-Tabelle des Ziels nicht ändern. Nachdem der Router ein Reply von jedem benachbarten Router erhalten hat, geht der Eintrag für das Ziel in der Topologietabelle wieder in den passiven Zustand über, und der Router kann einen Nachfolger auswählen.

34.4.4 Routenkennzeichnung

Enhanced IGRP unterstützt interne und externe Routen. Interne Routen stammen aus einem Enhanced IGRP AS. Daher wird ein direkt angebundenes Netzwerk, das für den Betrieb von Enhanced IGRP konfiguriert ist, als interne Route betrachtet und mit dieser Information durch das Enhanced IGRP AS weitergeleitet. Externe Routen werden von einem anderen Routing-Protokoll in Erfahrung gebracht oder befinden sich als statische Routen in der Routing-Tabelle. Diese Routen werden individuell mit der Identität ihrer Herkunft gekennzeichnet.

Externe Routen werden mit folgenden Informationen gekennzeichnet:

- Router-ID des Enhanced-IGRP-Router, der die Route weiterverteilt
- AS-Nummer des Ziels
- Konfigurierbare Administratorkennzeichnung
- ID des externen Protokolls
- Metrik aus dem externen Protokoll
- Bit-Flags für vorgegebenes Routing

Die Routenkennzeichnung ermöglicht es dem Netzwerk-Administrator, das Routing den Bedürfnissen anzupassen und eine flexible Sicherheitskontrolle zu erhalten. Die Routenkennzeichnung ist insbesondere beim Durchqueren von AS nützlich, wobei Enhanced IGRP typischerweise mit einem Routing-Protokoll zwischen Domains interagiert, das umfassendere Sicherheitsmaßnahmen implementiert, die zu einem äußerst skalierbaren, sicherheitsorientierten Routing führen.

34.5 Pakettypen von Enhanced IGRP

Enhanced IGRP greift auf folgende Pakettypen zurück: Hello und Acknowledgment, Update sowie Query und Reply.

Hello-Pakete werden für das Neighbor Discovery/Recovery als Multicast verschickt und erfordern kein Acknowledgment. Bei einem *Acknowledgment*-Paket handelt es sich um ein datenloses Hello-Paket. Acknowledgment-Pakete enthalten eine Acknowledgment-Nummer ungleich Null und werden immer unter Verwendung einer Unicast-Adresse verschickt.

Update-Pakete werden verwendet, um die Erreichbarkeit von Zielen zu übermitteln. Sobald ein neuer Nachbar entdeckt wird, werden Unicast-Pakete verschickt, so daß der Nachbar seine Topologietabelle aufbauen kann. In anderen Fällen wie beispielsweise bei Änderungen im Verbindungsaufwand erfolgen Updates als Multicast. Updates werden immer zuverlässig übermittelt.

Query- und Reply-Pakete werden verschickt, sobald ein Ziel über keine erreichbaren Nachfolger verfügt. Query-Pakete erfolgen immer als Multicast. Reply-Pakete werden als Reaktion auf Query-Pakete verschickt, um den Urheber dazu zu bringen, die Route nicht neu zu bestimmen, da erreichbare Nachfolger vorhanden sind. Reply-Pakete erfolgen als Unicast an den Urheber des Query-Pakets. Sowohl Query- als auch Reply-Pakete werden zuverlässig übertragen.

KAPITEL 35

IBM Systems Network Architecture (SNA)

35.1 Hintergrund

IBMs Netzwerk-Architektur hat sich beträchtlich weiterentwickelt, da sich das Computerwesen im allgemeinen von der Dominanz zentraler Rechnerlösungen hin zu gleichberechtigten Rechnern entwickelt hat. Heutzutage bezieht das IBM Systems Network Architecture (SNA) Routing zwei verschiedene Umgebungsarten ein, obgleich eine Reihe von Schlüsselbegriffen für alle SNA-Routing-Situationen zentral sind. In diesem Kapitel werden Funktionen und Dienste besprochen, die sowohl das SNA Subarea Routing als auch das Advanced Peer-to-Peer Networking (APPN) Routing ermöglichen. Zu den behandelten Themen gehören Sitzungsverbindungen (*Session Connections*), Übertragungsgruppen (*Transmission Groups*), explizite und virtuelle Routen sowie Dienstklassen (*Classes Of Service* = COS). Allgemeine Informationen zum traditionellen IBM SNA und APPN finden Sie in Kapitel 27, »Protokolle der IBM Systems Network Architecture (SNA)«. Das Bild 35.1 verdeutlicht die in diesem Kapitel behandelten Begriffe im Kontext einer herkömmlichen SNA-Umgebung.

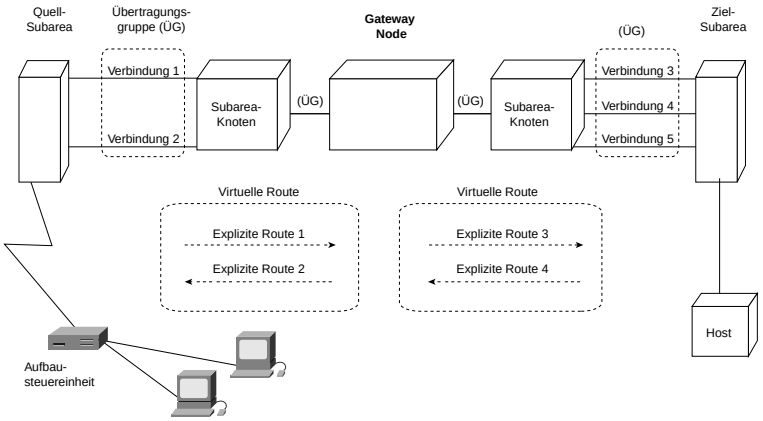


Bild 35.1: Das SNA Routing stützt sich für die Verbindung von Subarea-Einheiten auf Übertragungsgruppen

35.2 IBM-SNA-Sitzungsverbindungen

IBM-SNA-Sitzungsverbindungen werden verwendet, um Adreßräume zu überbrücken, wenn Arbeitssitzungen mehrere Adreßräume durchqueren. Es gibt drei Arten von Sitzungsverbindungen: *Übergangsfunktionen*, *SNA Network Interconnection (SNI) Gateways* und *zwischengeschaltete APPN-Routing-Funktionen*. Übergangsfunktionen befinden sich in Subarea-Knoten und bilden die Adreßräume von Subareas und Peripherie aufeinander ab. SNI Gateways dienen als Brücke zwischen SNA-Netzwerken und nehmen Daten aus dem einen Netzwerk entgegen, die sie an das entsprechende Ziel in einem anderen Netzwerk übertragen. SNI Gateways sind für am Endpunkt an das Netzwerk angehängte Einheiten transparent. Zwischengeschaltete APPN-Knoten führen das dazwischenliegende Routing in APPN-Netzwerken durch. Im Bild 35.1 können Sie die Lage einer Sitzungsverbindung in einer herkömmlichen SNA-Umgebung erkennen.

35.3 IBM-SNA-Übertragungsgruppen

Bei IBM-SNA-Übertragungsgruppen handelt es sich um logische, zwischen benachbarten IBM-SNA-Knoten hergestellte Verbindungen, die verwendet werden, um den in einer Arbeitssitzung anfallenden SNA-Datenverkehr zu übergeben. Übertragungsgruppen setzen sich aus einer oder mehreren SNA-Verbindungen und den ihnen zugewiesenen Übertragungsprioritäten zusammen. Übertragungsgruppen mit mehre-

ren Verbindungen, die eine höhere Zuverlässigkeit und Bandbreite bieten, werden verwendet, um mehrere physikalische Verbindungen in einer einzigen logischen SNA-Verbindung zu bündeln. Derartige Übertragungsgruppen werden nur zwischen T4-Knoten unterstützt. Die Sequenznummern der Übertragungsgruppen werden verwendet, um aus der Reihe tanzende Nachrichten bei jedem Hop erneut einzureihen. Für jede Übertragungsgruppe werden vier Übertragungsprioritäten unterstützt: *Low*, *Medium*, *High* und *Network-Service Traffic* (die höchste Priorität). Im Bild 35.1 sind die Beziehungen der Übertragungsgruppen hinsichtlich anderer Routing-Komponenten im Kontext einer SNA-Subarea-Routing-Umgebung zu sehen.

35.4 IBM-SNA – explizite und virtuelle Routen

Routen zwischen Subareas können entweder als *explizite* oder *virtuelle* Routen betrachtet werden. Explizite Routen sind die physikalischen Verbindungen zwischen zwei Subarea-Knoten und dienen als geordnete Verkettung von Subareas und verbindende Übertragungsgruppen. Sie sind unidirektional, und es sind zwei explizite Routen erforderlich, um einen vollduplex-fähigen Pfad einzurichten. Bei virtuellen Routen handelt es sich um zwischen zwei Subarea-Knoten eingerichtete logische Verbindungen in beide Richtungen. Eine virtuelle Route bewegt sich sowohl auf einer expliziten Route als auch auf einer entgegengesetzten expliziten Route, die demselben physikalischen Pfad folgt. Virtuelle Routen überschreiten keine Netzwerkgrenzen; statt dessen greifen sie auf eine netzwerkverbindende SNA-Sitzungsverbindung zurück, um zwei virtuelle Routen zu überbrücken. Virtuelle Routen enthalten Werte, die die Übertragungspriorität und die globale Datenflußkontrolle definieren, die durch Pacing bereitgestellt werden, wobei ein Empfänger mit ausreichendem Zwischenspeicher Pacing-Fenster für den Sender bewilligt. Jedes Pacing-Fenster ermöglicht es dem Sender, eine bestimmte Informationsmenge zu übertragen, bevor der Sender das nächste Pacing-Fenster anfordern muß. Im Bild 35.1 können Sie das Verhältnis zwischen expliziten und virtuellen Routen sowie deren Lage im Kontext einer SNA-Subarea-Routing-Umgebung erkennen.

35.5 IBM-SNA-Dienstklassen

Die IBM-SNA-Dienstklassen (*Classes Of Service* = COS) kennzeichnen die Eigenschaften des übertragenden Netzwerks einer gegebenen Arbeitssitzung. In Abhängigkeit von Benutzerbedürfnissen können in einem SNA-Netzwerk verschiedene Dienstklassen angegeben werden. Die Dienstklasse stellt den Mechanismus zum Ermitteln aller SNA-Routen bereit und beschreibt annehmbare Dienstebenen für eine Arbeitssitzung. Die Dienstklasse gibt weiterhin die Eigenschaften des Dienstes an, zu denen Reaktionszeit, Sicherheit und Verfügbarkeit zählen. Außerdem kann die Dienstklasse automatisch beim Anmelden oder (vom Benutzer) manuell beim Einrichten der Arbeitssitzung eingeführt werden. Jeder Dienstklassenname ist mit einer Reihe von virtuellen Routen verknüpft, die der gewünschten Dienstebene entsprechen. Für eine gegebene Arbeitssitzung relevante Informationen werden in Subarea- und APPN-Tabellen der Dienstklasse aufbewahrt. Die Unterschiede der Implementierung von Dienstklassen beim Subarea und APPN Routing sind in den folgenden Abschnitten zusammengefaßt.

35.5.1 Dienstklassen beim Subarea Routing

Beim Subarea Routing definiert der Benutzer die für eine bestimmte Arbeitssitzung erforderliche Dienstklassenunterstützung. Bestimmte virtuelle Routen werden auf bestimmte Dienste abgebildet, während die Eigenschaften der Dienstklasse mit den zugrundeliegenden expliziten Routen verknüpft werden. Der *System Services Control Point* (SSCP) greift auf die Dienstklassentabelle zurück, um der Pfadkontrollfunktion Informationen über virtuelle Routen und die Übertragungspriorität bereitzustellen. Im Gegenzug wählt die Pfadkontrolle eine virtuelle Route und Übertragungspriorität (*Transmission Priority* = TPRI) für den Einsatz in einer Arbeitssitzung aus. Das Format für einen Eintrag in der Dienstklassentabelle beim Subarea Routing ist im Bild 35.2 dargestellt.

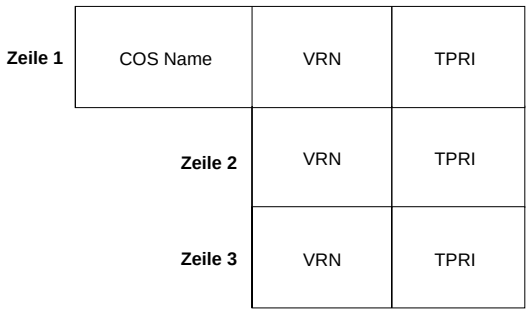


Bild 35.2: Eine Dienstklassentabelle enthält beim Subarea Routing Daten über virtuelle Routen und Übertragungsprioritäten

Einträge in der Dienstklassentabelle bestehen beim Subarea Routing aus dem *Dienstklassenname*, der *Nummer der virtuellen Route* (VRN) und der *Übertragungspriorität in der Subarea*.

Der Dienstklassenname ist eine Standardbezeichnung wie beispielsweise SEC3, der durch Konventionen geregelt ist.

Die VRN weist eine bestimmte Route zwischen Subareas aus. Bis zu acht Nummern von virtuellen Routen können zwischen zwei Subarea-Knoten zugewiesen sein. Jede virtuelle Route kann bis zu drei verschiedenen Übertragungsprioritäten zugewiesen werden, und bis zu 24 virtuelle Routen sind zwischen zwei Subareas möglich.

TPRI weist die Priorität des Datenflusses für eine Arbeitssitzung von einer logischen Einheit (*Logical Unit* = LU) zu einer anderen logischen Einheit über eine explizite Route aus. Benutzer können zwischen drei Prioritäten für jede virtuelle Route wählen: 0 (niedrigste), 1 oder 2 (höchste).

35.5.2 Dienstklassen beim APPN Routing

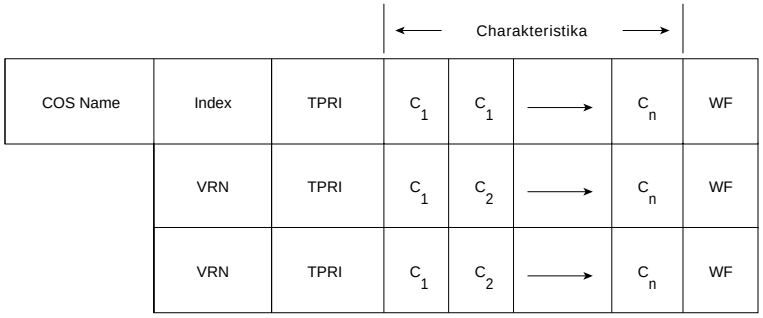
Eine Dienstklasse ist beim APPN explizit mit Parametern der Dienstklassentabelle definiert. Dienstklassen sind beim APPN feiner aufgegliedert als beim Subarea SNA. Insbesondere ermöglichen Dienstklassen es beim APPN, eine Route auf Kapazität, Aufwand, Sicherheit, Weiterleitungsverzögerung und benutzerdefinierten Eigenschaften basierend zu definieren. Der Dienst wird auf Endknoten ausgeweitet und ist nicht auf Kommunikationscontroller beschränkt wie beim Subarea SNA. Eine APPN-Dienstklasse gestattet es der Topologiedatenbank, einen Baum für jeden Dienst vorzuhalten, der alle

Routen und den jeweiligen Aufwand verfolgt. Weiterhin bietet eine APPN-Dienstklasse eine Konfigurationsmöglichkeit für die Kontrolle des Dienstklassenbäumen zugeordneten Speichers. Das Format für einen Eintrag in der Dienstklassentabelle beim APPN Routing ist im Bild 35.3 dargestellt.

Einträge in der Dienstklassentabelle bestehen beim APPN Routing aus dem *Dienstklassennamen* (COS Name), einem *Index*, der *APPN-Übertragungspriorität* (TPRI), *Charakteristika* (C1…Cn) und einem Weight Field (WF) oder *Gewichtungsfeld* der APPN-Dienstklasse.

Der Dienstklassenname ist eine Standardbezeichnung wie beispielsweise SEC3, der durch Konventionen geregelt ist.

Bild 35.3: Eine Dienstklassentabelle kann beim APPN Routing besondere Informationen über Charakteristiken und Routengewichtung enthalten



Der Eintrag im Indexfeld ermöglicht das Speichern von und Zugreifen auf berechnete Gewichtungswerte für Routenkomponenten. Dieser Eintrag zeigt auf den Eintrag im Gewichtungsarray der Dienstklasse, in dem die Gewichtungen für die Dienstklasse gespeichert sind.

APPN TPRI weist die Priorität des Datenflusses für eine Arbeitssitzung von einer logischen Einheit zu einer anderen logischen Einheit über eine explizite Route aus. Es wird nur ein TPRI-Feld für jeden Eintrag in der Dienstklassentabelle angegeben. APPN TPRI sorgt dafür, daß der über eine gegebene Arbeitssitzung mit derselben Dienstklasse erfolgende Datenverkehr sich in einem bestimmten APPN-Netzwerk mit derselben Übertragungspriorität bewegt.

Die Eigenschaften für Knoten und Übertragungsgruppe bestehen aus einer benutzerdefinierten Auflistung von Charakteristika, die für eine ausgewiesene Dienstklasse geeignet sind.

Jede Zeile definiert entweder eine Reihe von Eigenschaften für einen Knoten oder für eine Übertragungsgruppe. Die Einträge können Angaben zur Sicherheit, zu den Kosten pro Verbindungsdauer und zur verfügbaren Kapazität enthalten. Das eine Charakteristik repräsentierende Feld setzt sich aus einer Reihe zulässiger Werte zusammen.

Das Gewichtungsfeld einer APPN-Dienstklasse ermöglicht es *Routes Selection Services* (RSS), einer gegebenen möglichen Routenkomponente (Knoten oder Übertragungsgruppe) eine Gewichtung zuzuordnen. Ein RSS greift auf das Gewichtungsfeld zurück, um eine relative Vorliebe für eine bestimmte Routenkomponente zu ermitteln. Das Gewichtungsfeld kann eine Konstante oder die Bezeichnung einer Funktion enthalten, auf die der RSS bei der Gewichtungsberechnung zurückgreift.

35.6 IBM SNA – Subarea Routing

Die logischen SNA-Bereiche und die Knotenadressierung sind zwei zentrale Komponenten beim herkömmlichen Routing in SNA-Umgebungen. Dieser Abschnitt behandelt diese Themen im Kontext des herkömmlichen SNA-Netzwerk-Betriebs.

SNA-Netzwerke werden in logische Bereiche unterteilt: *Subareas* und *Domains*. Subareas bestehen aus einem Subarea-Knoten und der an diesem angebundenen Peripherie. Domains bestehen aus einem *System Services Control Point* (SSCP) und denjenigen Netzwerk-Ressourcen, die dieser kontrollieren kann. SSCPs aus unterschiedlichen Domains können miteinander kooperieren, um Ausfälle des Hostprozessors zu kompensieren. Im Bild 35.4 ist das Verhältnis zwischen Subareas und Domains im Kontext des SNA Subarea Routing dargestellt.

Knotenadressen werden als *Adressen von Subarea-* und *Peripherieknoten* kategorisiert. Adressen von Subarea-Knoten sind global und müssen im gesamten Netzwerk eindeutig sein. Diese Adressen werden einer an das Netzwerk angehängten Einheit beim Aktivieren zugewiesen. Adressen von Subarea-Knoten setzen sich im allgemeinen aus einem Anteil für die Subarea und einem Anteil für die Einheit zusammen. Alle an das Netzwerk angehängten Einheiten innerhalb einer gegebenen Subarea teilen sich dieselbe Subarea-Adresse, besitzen aber unterschiedliche Elementadressen.

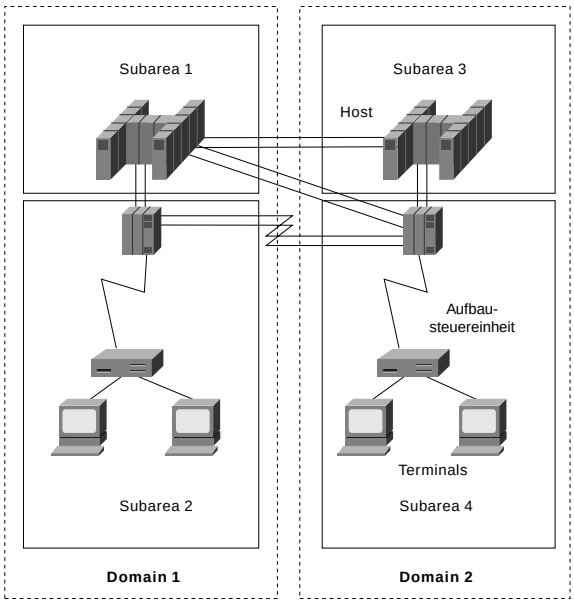


Bild 35.4: Subareas bestehen beim SNA Subarea Routing innerhalb von Domains

Adressen von Peripherieknoten, die als lokale Adressen betrachtet werden, unterscheiden sich in Abhängigkeit davon, ob es sich um einen T2- oder T2.1-Knoten handelt. T2-Adressen beziehen sich auf an das Netzwerk angehängte Einheiten und werden statisch zugewiesen, wohingegen T2.1-Adressen dynamisch für die Dauer einer Arbeitssitzung zugewiesen werden und die Arbeitssitzung statt der an das Netzwerk angehängten Einheit ausweisen. Adressen von Peripherieknoten werden auch als lokale Arbeitssitzungs-IDs bezeichnet.

35.7 IBM Advanced Peer-to-Peer Networking (APPN) Routing

Das APPN Routing erfolgt dynamisch und basiert auf einem aus den von allen APPN-Netzknoten erhaltenen Angaben berechneten Pfad mit der geringsten Gewichtung. Jeder APPN-Netzknoten ist für die Meldung von Änderungen in seiner lokalen Topologie (d.h. der Knoten selber und die angebundenen Verbindungen) verantwortlich. Topologieinformationen werden so lange übergeben, bis alle APPN-Knoten sie erhalten. Wenn ein Knoten Daten erhält, über die er bereits verfügt, beendet er das Weiterleiten der Daten an andere Knoten. Vervielfältigte Informationen werden durch eine Überprüfung

der Update-Sequenznummer erkannt. Im Bild 35.5 ist dargestellt, wie APPN-Netzknoten in das allgemeine Schema einer APPN-Umgebung mit Endknoten und niederschwelligen Netzknoten passen.

Das APPN Routing wird durch mehrere zugrundeliegende Funktionen und Fähigkeiten ermöglicht. Dazu gehören das *Node Type 2.1 Routing*, das *Dependent Logical Unit Requester/Server* (DLUR/S) *Routing*, Verbindungsnetzwerke und Übergangsknoten.

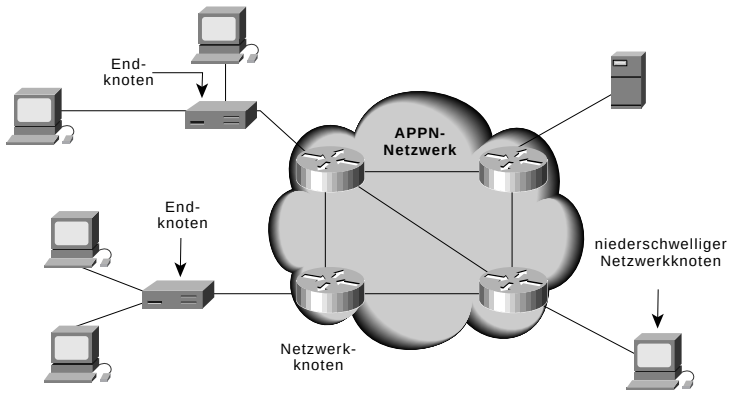


Bild 35.5: APPN-Netzknoten stellen Verbindungen mit Endknoten, niederschwelligen und anderen Netzknoten her

35.7.1 IBM APPN – Node Type 2.1 Routing

Das Node Type 2.1 Routing betrifft den Routingverkehr von einem oder mehreren APPN-Netzknoten. Es werden zwei Node-Type-2.1-Routingvorgänge unterstützt: das *Intermediate Session Routing* (ISR) und das *High Performance Routing* (HPR).

Intermediate Session Routing (ISR)

Der Vorgang des ISR bezieht BIND-Requests und -Responses von *Verbindungssitzungen* ein, die von Netzknoten zu Netzknoten weitergeleitet werden. In dieser Umgebung werden Systemverbindungen aufgebaut und anstelle von Routing-Tabellen beim APPN verwendet. Mit ISR wird eine Karte mit der ID und dem Anschluß der Arbeitssitzung von einer Seite des Kno-

tens zur anderen angelegt. Eine eindeutige Sitzungs-ID im Header der Sitzungsverbindung wird mit einer ausgehenden ID getauscht und anschließend vom passenden Anschluß verschickt.

Zu den von ISR unterstützten Eigenschaften der Subarea SNA gehören die Aufbereitung von Fehlern bei der Verbindung von Knoten zu Knoten und der Datenflußkontrolle sowie das Umlenken von Arbeitssitzungen um Netzwerk-Ausfälle herum. Die Aufbereitung von Fehlern bei der Verbindung von Knoten zu Knoten und der Datenflußkontrolle werden als redundant und unnötig angesehen, da diese Vorgänge den Durchsatz von Endsystem zu Endsystem verringern.

High Performance Routing (HPR)

Das Protokoll des *High Performance Routing* (HPR), einer Alternative zu ISR, basiert auf den beiden Schlüsselkomponenten *Rapid Transport Protocol* (RTP) und *Automatic Network Routing* (ANR). Bei RTP handelt es sich um ein zuverlässiges, verbindungsbezogenes Protokoll, das die Zustellung sicherstellt und von Endsystem zu Endsystem auftretende Netzwerk-Fehler sowie die Datenflußkontrolle behandelt. RTP erzeugt neue Routen nach einem Netzwerk-Ausfall. Bei ANR handelt es sich um einen verbindungslosen Dienst, der für quellengeleitete Dienste von Knoten zu Knoten verantwortlich ist.

Die RTP-Schicht wird nur an den Grenzen eines APPN-Netzwerks aufgerufen. Von dazwischenliegenden Knoten wird lediglich die ANR-Schicht aufgerufen. RTP-Knoten richten RTP-Verbindungen ein, um Sitzungsdaten weiterzuführen. Sämtlicher Datenverkehr für eine einzelne Sitzung bewegt sich über dieselbe RTP-zu-RTP-Verbindung und wird im Multiplex-Verfahren mit dem Datenverkehr aus anderen Arbeitssitzungen übertragen, die dieselbe Verbindung verwenden. Im Bild 35.6 ist die gesamte Architektur einer HPR-basierten Routingumgebung dargestellt.

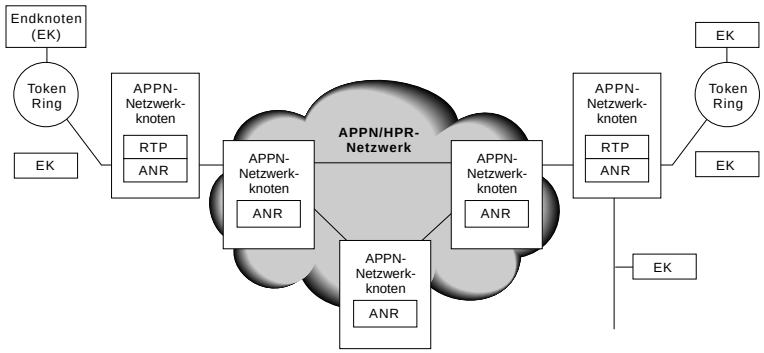


Bild 35.6: RTP wird nur von an APPN grenzenden Netzknoten unterstützt

Ein typischer HPR-Routingvorgang umfaßt mehrere Schritte. Zuerst wird unter Einsatz von ISR eine Route ausgewählt. Um eine Verbindung zwischen den angrenzenden RTP-Knoten einzurichten, wird entweder auf eine bestehende RTP-zu-RTP-Verbindung zurückgegriffen, oder es wird ein *Route Services Request* (RSR) verschickt. Die zurückgelieferte *Route Services Reply* (RSP) enthält Informationen, die einen Hin- und einen Rückpfad durch das Netzwerk anzeigen.

Pfade stellen die hin- und zurückführenden Anschlußauflistungen dar und enthalten die in jedem ANR-Knoten verwendete Anschluß-ID. Diese Auflistungen werden von jeder Nachricht mitübertragen, wodurch die Notwendigkeit für Routing-Tabellen oder Sitzungsverbindungen in den ANR-Knoten wegfällt.

HPR sorgt bei Verbindungsausfall für eine Wiederherstellung. Wenn eine Verbindung ausfällt und ein alternativer Pfad zwischen den RTP-Endpunkten für eine bestimmte Dienstklasse vorhanden ist, kann eine neue RTP-zu-RTP-Verbindung ausgewählt werden, und eine Sitzung kann ohne Unterbrechung umgeleitet werden. Wenn keine Verbindung über den neuen Pfad besteht, werden RSR- und RSP-Nachrichten verschickt, um die neue Anschlußauflistung zu erhalten. Das erneute Senden eines BIND ist nicht erforderlich, da die Arbeitssitzung nicht unterbrochen wurde.

Die Datenflußkontrolle greift in einer HRP-Umgebung auf eine Technik zurück, die als *anpassungsfähige ratenbasierte* Datenflußkontrolle bezeichnet wird. Die anpassungsfähige ratenbasierte Datenflußkontrolle überwacht und kontrolliert das Aufkommen des in das Netzwerk übertragenen Datenverkehrs. Unter ihr tauschen die sendenden und empfangenden RTP-Knoten in regelmäßigen Intervallen Nachrichten aus. Der in das Netzwerk übertragene Datenverkehr wird an die Netzwerk-Bedingungen angepaßt.

35.7.2 IBM APPN – DLUR/S Routing

Beim *Dependent Logical Unit Requester/Server* (DLUR/S) handelt es sich um eine APPN-Eigenschaft, die es gewährtem SNA-Datenverkehr ermöglicht, sich in einem APPN-Netzwerk fortzubewegen.

Unter DLUR/S wird eine Client-Server-Beziehung zwischen einem *Dependent Logical Unit Server* (DLUS) und einem *Dependent Logical Unit Requester* (DLUR) eingerichtet. Bei einem DLUS handelt es sich typischerweise um eine ACF/UTAM4.2-Einheit, bei einem DLUR typischerweise um einen Router. Ein Paar von LU-6.2-Sitzungen überträgt die gewährten SNA-Kontrollnachrichten. Die in einer APPN-Umgebung nicht identifizierten Nachrichten werden in der LU-6.2-Sitzung gekapselt. Die Nachrichten werden dann von DLUR entkapselt und an die gewährte SNA LU übergeben. Die Aufnahme der DLU-Sitzung wird anschließend an den DLUS übergeben und von ihm als gewährter Datenverkehr bearbeitet. Der DLUS verschickt eine Nachricht an den Anwendungshost, und der Anwendungshost verschickt das BIND. Schließlich bewegen sich die gewährten SNA-Daten ganz natürlich mit dem APPN-Datenverkehr fort.

35.7.3 IBM-APPN-Verbindungsnetzwerk

Bei einem IBM-APPN-Verbindungsnetzwerk handelt es sich um ein logisches Konstrukt, auf das zurückgegriffen wird, um eine direkte Verbindung zwischen APPN-Endknoten ohne den

Konfigurationsoverhead beim Definieren von direkten Verbindungen zwischen jedem Paar von Endknoten bereitzustellen. Im allgemeinen beginnt der Vorgang des Erstellens eines Verbindungsnetzwerks, sobald ein LOCATE-Request von einem Endknoten empfangen wird.

Ein Netzknoten wird dann verwendet, um das im LOCATE-Request angegebene Ziel ausfindig zu machen. Wenn der Netzknoten feststellt, daß beide Endknoten (Quelle und Ziel) am gleichen Transportmedium hängen (beispielsweise Token Ring), kommt ein *virtueller Knoten* zum Einsatz, um die beiden Endpunkte zu verbinden und ein Verbindungsnetzwerk herzustellen. Der Netzknoten definiert den Sitzungspfad als eine direkte Verbindung vom Endknoten 1 über den virtuellen Knoten zum Endknoten 2, und anschließend darf der Datenverkehr fließen.

35.7.4 IBM APPN – Übergangsknoten

Bei einem Übergangsknoten handelt es sich um eine Einheit, die es ermöglicht, mehrere APPN-Netzwerke miteinander zu verbinden. Derzeit sind Übergangsknoten nur als ACF/VTAM und OS/400 implementiert. Übergangsknoten sind für das Zusammenknüpfen von Verzeichnis- und Topologiedatenbanken für Verbindungsnetzwerke sowie für das Aufbauen von BIND-Requests zum Anzeigen gesonderter Routen in jedem Netzwerk verantwortlich.

Durch Übergangsknoten werden die Topologie- und Verzeichnisdatenbanken auf Netzknoten auf den für einzelne Subnetze anstelle des zusammengesetzten Netzwerks benötigten Umfang reduziert. Weiterhin werden das Netzwerk durchlaufende Arbeitssitzungen über den Übergangsknoten geleitet. Das Bild 35.7 verdeutlicht die Lage von Übergangsknoten (ACF/VTAM und OS/400) in einer APPN-Umgebung mit mehreren Netzwerken.

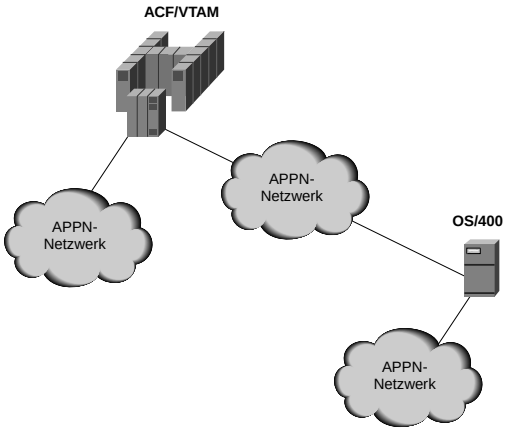


Bild 35.7: Übergangsknoten (ACF/VTAM und OS/400) können APPN-Netzwerke verbinden

KAPITEL 36

Interior Gateway Routing Protocol (IGRP)

36.1 Hintergrund

Beim *Interior Gateway Routing Protocol* (IGRP) handelt es sich um ein Mitte der 1980er Jahre von Cisco Systems, Inc. entwickeltes Routing-Protokoll. Ciscos Hauptziel bei der Erstellung von IGRP bestand darin, ein stabiles Protokoll für das Routing innerhalb eines autonomen Systems (AS) bereitzustellen.

Mitte der 1980er Jahre war das *Routing Information Protocol* (RIP) das beliebteste Routing-Protokoll zwischen autonomen Systemen. Obgleich RIP für das Routing in verhältnismäßig homogenen Internetzwerken, mit kleinem bis gemäßigtem Umfang ziemlich brauchbar war, wurden seine Grenzen durch das Wachstum der Netzwerke erreicht. Insbesondere beschränkte die niedrige Grenze des RIP für den Hop-Count (16) die Größe von Internetzwerken und seine einzige Metrik (Hop-Count) sorgte nicht gerade für eine große Flexibilität beim Routing in komplexen Umgebungen. Die Beliebtheit von Cisco-Routern und die Stabilität von IGRP brachten viele Organisationen mit großen Internetzwerken dazu, RIP durch IGRP zu ersetzen.

Ciscos erste Implementierung von IGRP lief in Netzwerken mit dem Internet Protocol (IP). IGRP wurde allerdings so entworfen, daß es in jeder beliebigen Netzwerk-Umgebung funktioniert, und schon bald portierte Cisco es für Netzwerke mit dem OSI Connectionless Network Protocol (CLNP). Anfang

der 1990er Jahre entwickelte Cisco das Enhanced IGRP, um die Effizienz im Betrieb von IGRP zu verbessern. In diesem Kapitel werden der grundlegende Entwurf und die Implementierung von IGRP beschrieben. Enhanced IGRP wird im Kapitel 34, »Enhanced IGRP« behandelt.

36.2 Eigenschaften von IGRP

Bei IGRP handelt es sich um ein auf dem *Distance-Vector*-Verfahren beruhendes Interior-Gateway-Protokoll (IGP). Auf dem Distance-Vector-Verfahren beruhende Routing-Protokolle fordern jeden Router dazu auf, ihre gesamte oder einen Teil ihrer Routing-Tabelle in regelmäßigen Abständen in einer Routing-Update-Nachricht an jeden seiner benachbarten Router zu versenden. Da sich die Routing-Informationen durch das Netzwerk hindurch stark vermehren, können Router Distanzen zu sämtlichen Knoten innerhalb des Internetzwerks berechnen.

Routing-Protokolle nach dem Distance-Vector-Verfahren werden häufig Routing-Protokollen nach dem Link-State-Verfahren entgegengestellt, die lokale Verbindungsinformationen an alle Knoten im Internetzwerk verschicken. Eine Besprechung von *Open Shortest Path First* (OSPF) und *Intermediate System-to-Intermediate System* (IS-IS), zwei beliebten Routing-Algorithmen für das Link-State-Verfahren, finden Sie in Kapitel 40, »Open Shortest Path First (OSPF)« bzw. Kapitel 39, »Open Systems Interconnection (OSI) Routing Protocol«.

IGRP verwendet eine Kombination (Vektor) von Metriken. *Verzögerungszeit des Internetzwerks*, *Bandbreite*, *Zuverlässigkeit* und *Last* gehen alle als Faktor in die Entscheidung für das Routing ein. Netzwerk-Administratoren können die Gewichtungsfaktoren für jede dieser Metriken festlegen. IGRP verwendet entweder die vom Administrator gesetzten oder die vorgegebenen Gewichtungen, um optimale Routen automatisch zu berechnen.

IGRP sieht einen großen Wertebereich für seine Metriken vor. Für Zuverlässigkeit und Last können beispielsweise Werte zwischen 1 und 255 angenommen werden; für Bandbreite können Werte angenommen werden, die für Geschwindigkeiten von 1200 bps bis zu 10 Gigabit pro Sekunde stehen, wäh-

rend für die Verzögerungszeit jeder beliebige Wert zwischen 1 und 2 hoch 24 angenommen werden kann. Große Wertebereiche für Metriken ermöglichen eine angemessene Einstellung der Metriken in Internetzwerken mit stark schwankenden Performance-Eigenschaften. Am wichtigsten dabei ist die Tatsache, daß die Metrikkomponenten in einem vom Benutzer definierbaren Algorithmus vereint sind. Als eine Folge davon können Netzwerk-Administratoren die Routenauswahl auf intuitive Weise beeinflussen.

Für eine größere Flexibilität gestattet IGRP das Routing über mehrere Pfade (*Multipath*). Zwei Leitungen mit gleicher Bandbreite können einen einzelnen Verkehrsstrom in jeder-gegen-jeden-Manier betreiben, wobei automatisch auf die zweite Leitung übergewechselt wird, wenn eine Leitung wegfällt. Mehrere Pfade können selbst dann verwendet werden, wenn sich die Metriken der Pfade unterscheiden. Wenn ein Pfad beispielsweise dreimal besser ist als ein anderer, da seine Metrik um das Dreifache niedriger ist, wird der bessere Pfad dreimal so häufig verwendet. Für mehrfache Pfade werden nur Routen verwendet, deren Metriken innerhalb eines bestimmten Bereichs der besten Route liegen.

36.2.1 Stabilitätsmerkmale

IGRP bietet eine Reihe von Merkmalen, die zu einer erhöhten Stabilität führen sollen: *Hold-Downs*, *Split-Horizons* und *Poison-Reverse-Updates*.

Hold-Downs werden eingesetzt, um reguläre Update-Nachrichten davon abzuhalten, eine Route wiedereinzusetzen, die unbrauchbar geworden sein könnte. Wenn ein Router ausfällt, entdecken dies die angrenzenden Router durch das Fehlen regulär eingeplanter Update-Nachrichten. Diese Router bestimmen dann neue Routen und verschicken Routing-Update-Nachrichten, um ihren Nachbarn die Routenänderung mitzuteilen. Diese Handlung löst eine Welle von Updates aus, die durch das Netzwerk sickern. Diese ausgelösten Updates kommen nicht sofort bei jedem Gerät im Netzwerk an; somit ist es möglich, daß ein Gerät, dem der Netzwerk-Ausfall noch mitgeteilt werden muß, eine reguläre Update-Nachricht (die anzeigt, daß eine gerade ausgefallene Route noch brauchbar ist) an ein Gerät verschickt, das den Netzwerk-Ausfall bereits mit-

bekommen hat. In diesem Fall würde das spätere Gerät falsche Routing-Informationen erhalten (und möglicherweise verbreiten). Hold-Downs teilen Routern mit, alle Änderungen für einige Zeit zurückzuhalten, die sich auf Routen auswirken könnten. Die Zeitspanne für das Hold-Down ist üblicherweise so berechnet, daß sie etwas größer ausfällt als die für eine Aktualisierung des gesamten Netzwerks mit einer Routing-Änderung benötigte Zeit.

Split-Horizons ziehen ihren Nutzen aus der Annahme, daß es niemals sinnvoll ist, Routeninformationen in die Richtung zurückzuschicken, aus der sie kamen. Das Bild 36.1 veranschaulicht die Split-Horizon-Regel. Router 1 (R1) macht anfangs bekannt, daß er über eine Route zum Netzwerk A verfügt. Es gibt keinen Grund für Router 2 (R2), diese Route in sein an R1 zurückgeschicktes Update aufzunehmen, da sich R1 näher an Netzwerk A befindet. Die Split-Horizon-Regel besagt, daß R2 diese Route aus allen an R1 gesendeten Updates streichen soll. Diese Regel hilft dabei, Routing-Scheifen zu vermeiden. Nehmen wir beispielsweise an, die Schnittstelle von R1 zum Netzwerk A fällt aus. Ohne Split-Horizons fährt R2 damit fort, R1 darüber zu informieren, daß er das Netzwerk A (über R1) erreichen kann. Wenn R1 nicht intelligent genug ist, kann er tatsächlich die Route von R2 als Alternative zu seiner fehlgeschlagenen direkten Verbindung aufgreifen und eine Routing-Schleife verursachen. Obgleich Hold-Downs solche Situationen vermeiden sollten, sind Split-Horizons in IGRP implementiert, da sie für eine zusätzliche Stabilität des Algorithmus sorgen.

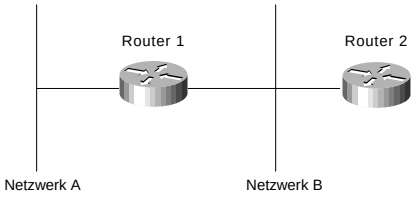


Bild 36.1: Die Split-Horizon-Regel hilft bei der Vermeidung von Routing-Schleifen

Split-Horizons sollten Routing-Schleifen zwischen benachbarten Routern vermeiden, aber Poison-Reverse-Updates sind notwendig, um größere Routing-Schleifen zu bekämpfen. Das Anwachsen von Routing-Metriken weist im allgemeinen auf Routing-Schleifen hin. Dann werden Poison-Reverse-Updates

verschickt, um die Route zu entfernen und in den Hold-Down-Zustand zu versetzen. Bei der IGRP Implementierung durch Cisco werden Poison-Reverse-Updates verschickt, falls eine Routenmetrik um den Faktor 1,1 oder größer angewachsen ist.

36.2.2 Timer

IGRP unterhält eine Reihe von Timern und Variablen, die Zeitabschnitte enthalten. Dabei handelt es sich um einen *Update*-Timer, einen *Invalid*-Timer, einen Zeitabschnitt für die *Hold-Time* und einen *Flush*-Timer. Der *Update*-Timer gibt an, wie oft Routing-Update-Nachrichten verschickt werden sollen. Der von IGRP vorgegebene Wert für diese Variable beträgt 90 Sekunden. Der *Invalid*-Timer gibt an, wie lange ein Router beim Fehlen von Routing-Update-Nachrichten für eine bestimmte Route warten soll, bevor diese Route für unzulässig erklärt wird. IGRP gibt für diese Variable das Dreifache des Update-Zeitabschnitts vor. Die Variable *Hold-Time* gibt den Zeitabschnitt für ein Hold-Down an. IGRP gibt für diese Variable das Dreifache des Update-Zeitabschnitts plus 10 Sekunden vor. Schließlich zeigt der *Flush*-Timer an, wieviel Zeit verstreichen soll, bevor eine Route aus der Routing-Tabelle gelöscht werden soll. IGRP gibt hierfür das Siebenfache des Update-Zeitabschnitts vor.

KAPITEL 37

Internet Protocol (IP) Multicast

37.1 Hintergrund

Beim Internet Protocol (IP) Multicast handelt es sich um eine Technik, die es ermöglicht, den IP-Verkehr von einer oder mehreren Quellen aus zu senden und ihn mehreren Zielen zuzustellen. Anstatt einzelne Pakete an jedes Ziel zu schicken wird ein einzelnes Paket an eine Multicast-Gruppe geschickt, die durch eine einzige IP-Zielgruppenadresse ausgewiesen ist. Das IP Multicast-Routing kam auf, weil Unicast- und Broadcast-Techniken den Anforderungen neuer Anwendungen nicht gewachsen sind. Die Adressierung beim Multicast unterstützt beispielsweise die Übertragung eines einzelnen IP-Datagramms an mehrere Hosts. Dieses Kapitel beschäftigt sich mit den wichtigsten Möglichkeiten beim Multicast-Routing. Im Bild 37.1 ist der allgemeine Aufbau einer Multicast-Umgebung dargestellt.

37.2 Internet Group Membership Protocol (IGMP)

Eine wesentliche Komponente beim IP Multicast stellt das Internet Group Membership Protocol (IGMP) dar. IGMP greift für die Erzeugung von Multicast-Gruppen auf Class-D-IP-Adressen zurück und ist in der RFC 1112 definiert. IGMP wird eingesetzt, um einzelne Hosts in einer Multicast-Gruppe dynamisch mit einer Class-D-Adresse zu registrieren. Hosts

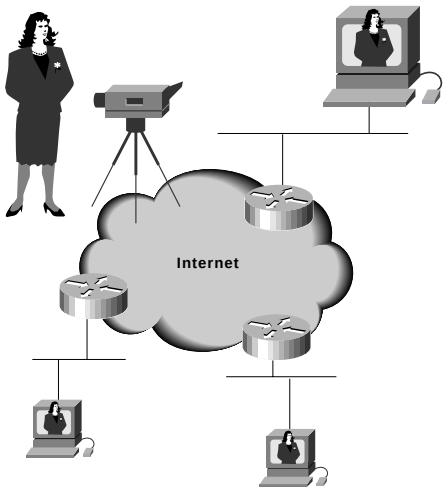


Bild 37.1: IP Multicast stellt ein Mittel zur Verfügung, um mehreren Zielen Datenverkehr zuzustellen, der eine hohe Bandbreite benötigt

stellen eine Gruppenmitgliedschaft fest, indem sie IGMP-Nachrichten verschicken, und der Datenverkehr wird an sämtliche Mitglieder dieser Multicast-Gruppe weitergeleitet. Unter IGMP achten Router auf IGMP-Nachrichten und verschicken regelmäßig Query-Pakete, um festzustellen, welche Gruppen in bestimmten LANs aktiv oder inaktiv sind. Router kommunizieren miteinander, indem sie ein oder mehrere Protokolle für den Aufbau von Multicast-Routen für jede Gruppe einsetzen.

Tabelle 37.1: Zusammenfassung der Möglichkeiten beim Multicast-Routing

Protokoll	**Notwendiges Unicast-Protokoll**	**Verbreitungsalgorithmus**
PIM (Dense-Modus)	beliebig	Reverse Path Flooding (RPF)
PIM (Sparse-Modus)	beliebig	RPF
DVMRP	internes, RIP-artiges Routing-Protokoll	RPF
MOSPF	Open Shortest Path First (OSPF)	Shortest Path First (SPF)

37.3 Protokolle für das IP Multicast-Routing

Für das Entdecken von Multicast-Gruppen und den Aufbau von Routen für jede Gruppe wird auf mehrere Routing-Protokolle zurückgegriffen. Bei diesen handelt es sich um: *Protocol-Independent Multicast* (PIM), *Distance-Vector Multicast*

Routing Protocol (DVMRP) und *Multicast Open Shortest Path First* (MOSPF). In der Tabelle 37.1 sind die Möglichkeiten für das Multicast-Routing mit den notwendigen Anforderungen an Unicast und den Verteilungsalgorithmen zusammengefaßt.

37.3.1 Protocol-Independent Multicast (PIM)

Das Protocol-Independent Multicast (PIM) wird in einem RFC-Entwurf für das Internet behandelt (die Diskussionen finden in der IETF Multicast Routing Working Group statt). Es umfaßt zwei unterschiedliche Verhaltensmodi für Umgebungen mit dichtem (*dense*) und spärlichem (*sparse*) Verkehr: den *Dense*-Modus und den *Sparse*-Modus.

Der Dense-Modus von PIM greift auf einen Vorgang der umgekehrten Pfadverteilung (*Reverse Path Flooding* = RPF) zurück, der dem DVMRP ähnelt. Es gibt allerdings dennoch Unterschiede zwischen dem Dense-Modus von PIM und DVMRP. PIM benötigt beispielsweise kein bestimmtes Unicast-Protokoll, um festzustellen, welche Schnittstelle zur Quelle eines Datenstroms zurückführt. DVMRP verwendet sein eigenes Unicast-Protokoll, während PIM auf jedes beliebige Unicast-Protokoll zurückgreift, das vom Internetzwerk verwendet wird.

Der Sparse-Modus von PIM wurde für Internetzwerke mit vielen Datenströmen und relativ wenigen LANs optimiert. Er definiert einen Treffpunkt, der anschließend als Registrierungsstelle verwendet wird, um das Routing von Paketen zu erleichtern.

Wenn ein Sender Daten übertragen will, schickt der Routerknoten des ersten Hop (hinsichtlich der Quelle) Daten an den Treffpunkt. Wenn ein Empfänger Daten erhalten will, registriert sich der Router des letzten Hop (hinsichtlich des Empfängers) beim Treffpunkt. Anschließend kann ein Datenstrom vom Sender über den Treffpunkt weiter zum Empfänger fließen. Auf dem Pfad befindliche Router optimieren den Pfad und beseitigen jeden unnötigen Hop automatisch, sogar am Treffpunkt.

37.3.2 Distance-Vector Multicast Routing Protocol (DVMRP)

Das Distance-Vector Multicast Routing Protocol (DVMRP) greift auf ein Verfahren mit umgekehrter Pfadverteilung (*Reverse Path Flooding* = RPF) zurück und wird als Basis für den Multicast Backbone (MBONE) des Internet eingesetzt. DVMRP ist in der RFC 1075 definiert und bringt einige Nachteile mit sich. Insbesondere ist DVMRP für eine schlechte Netzwerkskalierung berüchtigt, die ihre Ursache in der erneuten Verteilung (*Reflooding*) hat; dies gilt ganz besonders für Versionen, in denen kein Mechanismus implementiert ist, um sich streichen zu lassen (*Pruning*). Weiterhin wirkt sich der einfache Mechanismus für das Unicast-Routing von DVMRP auf dessen Skalierungsfähigkeiten aus.

Bei der umgekehrten Pfadverteilung (Reverse Path Flooding) verschickt ein Router nach Erhalt eines Pakets über alle Pfade (außer dem zum Ursprung zurückführenden Pfad) eine Kopie des Pakets. Anschließend verschicken Router eine Pruning-Nachricht an die Quelle zurück, um den Datenstrom anzuhalten, falls der Router an ein LAN angebunden ist, das eine bestimmte Multicast-Gruppe nicht empfangen möchte.

Auf die *erneute Verteilung* und das *DVMRP Unicast* wird bei den DVMRP-Vorgängen zur Pfadverteilung zurückgegriffen. Bei der erneuten Verteilung beliefern DVMRP-Router ein angebundenes Netzwerk regelmäßig erneut, um neue Hosts zu erreichen. Der Verteilungsmechanismus greift auf einen Algorithmus zurück, der die Verteilungshäufigkeit und die von einem neuen Multicast-Gruppenmitglied benötigte Zeit zum Empfangen des Datenstroms berücksichtigt. Dieses Verfahren ist einzigartig für DVMRP, ähnelt RIP allerdings darin, daß es auf dem Zählen der Hops basiert. Die Unicast-Umgebung von DVMRP gestattet die Verwendung eines anderen Pfads als den für den Multicast-Datenverkehr verwendeten.

37.3.3 Multicast Open Shortest Path First (MOSPF)

Bei Multicast Open Shortest Path First (MOSPF) handelt es sich um eine Erweiterung von OSPF. Allgemein gesagt, verwendet MOSPF ein Unicast-Routing-Protokoll, bei dem jedem

Router sämtliche verfügbaren Verbindungen bekannt sein müssen.

Ein MOSPF-Router bestimmt von der Quelle aus Routen zu allen möglichen Gruppenmitgliedern einer bestimmten Multicast-Gruppe. MOSPF-Router enthalten Multicast-Informationen in den Verbindungszuständen von OSPF. MOSPF bestimmt die Routen für jedes Quelle/Multicast-Gruppen-Paar, sobald der Router Verkehr für dieses Paar erhält. Dabei werden die Routen so lange im Zwischenspeicher gehalten, bis eine Änderung in der Topologie auftritt. Dann bestimmt MOSPF die Topologie erneut.

Für MOSPF haben sich mehrere Themen für die Implementierung herausgeschält und verlangen eine nähere Betrachtung. Zuerst einmal funktioniert MOSPF ausschließlich in Internetzwerken, die OSPF einsetzen. Weiterhin ist MOSPF am besten für Umgebungen mit verhältnismäßig wenigen Quelle/Gruppe-Paaren geeignet. MOSPF kann in Umgebungen, die über viele aktive Quelle/Gruppe-Paare verfügen oder instabil sind, die Route r-CPU beträchtlich beanspruchen.

KAPITEL 38

NetWare Link Services Protocol (NLSP)

38.1 Hintergrund

Beim *NetWare Link Services Protocol* (NLSP) handelt es sich um ein mit dem Link-State-Verfahren arbeitendes Routing-Protokoll von Novell, das entworfen wurde, um die mit dem IPX Routing Information Protocol (RIP) und dem begleitenden Service Advertisement Protocol (SAP) verbundenen Einschränkungen zu überwinden. NLSP basiert auf dem OSI-Intermediate-System-to-Intermediate-System-Protokoll (IS-IS) und wurde entworfen, um die ursprünglichen Routing-Protokolle RIP und SAP von Novell zu ersetzen, die entstanden sind als Internetzwerke lokal und verhältnismäßig klein waren. Solcherart sind RIP und SAP für die heutigen großen, globalen Internetzwerke nicht mehr geeignet. In diesem Kapitel erfolgt eine Zusammenfassung der Vorgänge beim Routing sowie der Protokollkomponenten von NLSP.

Verglichen mit RIP und SAP bietet NLSP ein verbessertes Routing, eine höhere Effizienz und Skalierbarkeit. Außerdem sind NLSP-basierte Router abwärtskompatibel zu RIP-basierten Routern. NLSP-basierte Router verwenden ein zuverlässiges Zustellungsprotokoll, so daß für die Zustellung garantiert wird. Darüber hinaus erleichtert NLSP bessere Entscheidungen beim Routing, da NLSP-basierte Router eine vollständige Abbildung des Netzwerks gespeichert haben und nicht nur Informationen über den nächsten Hop, wie sie RIP-basierte Router verwenden. Die Routing-Informationen werden nur dann übertragen, wenn Änderungen in der Topologie aufgetre-

ten sind und nicht alle 60 Sekunden, wie es bei RIP-basierten Routern unabhängig davon der Fall ist, ob sich die Topologie wirklich verändert hat. Weiterhin verschicken NLSP-basierte Router Updates für Dienstinformationen nur dann, wenn sich Dienste ändern und nicht alle 60 Sekunden, wie es unter SAP der Fall ist.

NLSP ist in mehrerlei Hinsicht brauchbar. Es ist insbesondere über eine WAN-Verbindung von Nutzen, da die Unterstützung der Header-Komprimierung von IPX es möglich macht, den Paketumfang zu reduzieren. NLSP unterstützt außerdem die Multicast-Adressierung, so daß Routing-Informationen nur an andere NLSP-Router geschickt werden und nicht an sämtliche Geräte, wie es bei RIP der Fall ist.

Weiterhin unterstützt NLSP den Lastenausgleich über parallele Pfade und verbessert die Verbindungsintegrität. Es überprüft regelmäßig Verbindungen auf ihre Unversehrtheit sowie die Datenintegrität der Routing-Informationen. Wenn eine Verbindung ausfällt, wechselt NLSP auf eine andere Verbindung und aktualisiert die in jedem Knoten gespeicherten Datenbanken für die Netzwerktopologie, sobald irgendwo in der Routing-Area Änderungen bei den Verbindungen auftreten.

Hinsichtlich der Skalierbarkeit kann NLSP bis zu 127 Hops unterstützen (RIP unterstützt bloß 15 Hops) und gestattet die hierarchische Adressierung von Netzknoten, wodurch Netzwerke Tausende von LANs und Servern enthalten können.

38.2 NLSP – hierarchisches Routing

NLSP unterstützt hierarchisches Routing mit Areas, Domains und Komponenten des globalen Internetzwerks. Bei einer *Area* handelt es sich um ein Ansammlung von verbundenen Netzwerken, die alle über dieselbe Area-Adresse verfügen. Eine *Domain* ist eine Ansammlung von Areas, die zur gleichen Organisation gehören. Ein globales Internetzwerk ist eine Ansammlung von Domains, die üblicherweise zu verschiedenen Organisationen gehören, aber in einer engen Beziehung zueinander stehen. Areas können miteinander verbunden werden, um Routing-Domains zu bilden, und Domains können miteinander verbunden werden, um ein globales Internetzwerk zu bilden.

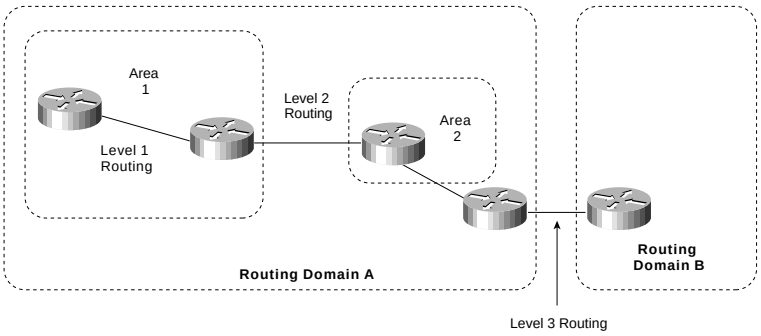


Bild 38.1: NLSP definiert drei Ebenen des Routing

38.2.1 Leistungen des hierarchischen Routing

Das hierarchische Routing vereinfacht den Ausbau eines Netzwerks, da die von jedem Router für das Weiterleiten von Paketen innerhalb einer Domain zu speichernde und zu bearbeitende Informationsmenge reduziert wird. Ein Level-1-Router braucht nur detaillierte Informationen über seine eigene Area vorzuhalten, statt Informationen über den Verbindungszustand für jeden Router und jedes Netzwerksegment in seiner Domain zu speichern. Für den Austausch von Datenverkehr mit anderen Areas braucht ein Level-1-Router lediglich den nächsten Level-2-Router ausfindig zu machen. Den Areas geben Level-2-Router bloß die Area-Adresse(n) der zu ihnen gehörenden Areas bekannt, nicht aber ihre gesamte Datenbank mit Verbindungszuständen. Level-3-Router agieren entsprechend auf der Ebene von Domains.

38.2.2 NLSP – angrenzende Umgebung

Durch den Austausch von Hello-Paketen ermittelt ein Router die Erreichbarkeit seiner Nachbarn, und auf diese Informationen greift er zurück, um seine angrenzende Umgebung aufzubauen. Bei der *angrenzenden Umgebung* handelt es sich um eine vom Router vorgehaltene Aufzeichnung über den Zustand seines Anschlusses mit einem Nachbarn und den Attributen des benachbarten Routers. Der Router speichert diese Datensätze in seiner Umgebungsdatenbank.

Die Verfahren zum Aufbau der angrenzenden Umgebung variieren in Abhängigkeit davon, ob der Router die angrenzende Umgebung über ein WAN oder ein LAN aufbaut und wartet.

Für den Aufbau der angrenzenden Router-Umgebung über ein WAN muß zuerst die zugrundeliegende vermittelnde Verbindung eingerichtet werden (weitere Einzelheiten hängen dabei vom Medium ab). Die Router tauschen anschließend mittels des IPX-WAN-Version-2-Protokolls Identitäten miteinander aus und legen bestimmte operationale Eigenschaften der Verbindung fest. Es werden Hello-Pakete ausgetauscht, und die Router aktualisieren ihre Umgebungsdatenbanken. Anschließend tauschen die Router sowohl Link-State-Pakete (LSPs), die den Zustand ihrer Verbindungen beschreiben, als auch IPX-Datenpakete über die Verbindung aus. Um eine WAN-Verbindung aufrechtzuerhalten, greift der Router für jede angrenzende Umgebung auf eine Zustandsvariable zurück, die anzeigt, ob die Verbindung aktiv oder inaktiv ist bzw. initialisiert wird. Wenn der Router innerhalb der von einem Hold-Timer angegebenen Zeit nichts von einem Nachbarn hört, generiert er eine die Inaktivität der Verbindung anzeigende Nachricht und streicht die angrenzende Umgebung.

Hello-Pakete im WAN ermöglichen es Routern, die Identitäten untereinander auszumachen, zu entscheiden, ob sie sich in derselben Routing-Area befinden, und festzustellen, ob sich andere Router und Verbindungen in Betrieb befinden. Ein Router verschickt Hello-Pakete, wenn eine Strecke zum ersten Mal eingerichtet wird, ein Timer abläuft oder sich die Inhalte des nächsten zu übertragenden Hello-Pakets von denen des zuvor von diesem System übertragenen Hello-Pakets unterscheiden (und eine oder mehrere Sekunden seit dem vorherigen Hello-Paket vergangen sind). Hello-Paket werden so lange verschickt wie eine Strecke besteht.

Aufbau einer neuen angrenzenden WAN-Umgebung

Ein typischer Anfangsvorgang zwischen zwei Routern (A und B) mit einer WAN-Verbindung beginnt mit einer *inaktiven* Verbindung. Router A schickt ein die Inaktivität der Verbindung anzeigendes Hello-Paket an Router B, der seinen Verbindungszustand auf *initialisierend* ändert. Router B schickt ein die Initialisierung anzeigendes Hello-Paket an Router A. Daraufhin ändert auch Router A seinen Verbindungszustand zu *initialisierend* und schickt ein entsprechendes Hello-Paket an Router B. Router B ändert seinen Verbindungszustand zu *aktiv* und verschickt ein Hello-Paket, das diesen neuen

Zustand anzeigt. Schließlich ändert auch der Router A seinen Verbindungszustand zu *aktiv*.

Die Wartung angrenzender Umgebungen in LANs

Wenn eine Broadcast-Strecke wie beispielsweise ein 802.3-Ethernet oder 802.5-Token-Ring auf einem Router aktiviert ist, beginnt der Router damit, Hello-Pakete zu verschicken und von anderen Routern im LAN entgegenzunehmen sowie das Auswahlverfahren für den designierten Router zu starten.

Der designierte Router stellt in der Datenbank für Verbindungszustände das LAN als Ganzes dar, trifft zugunsten des Ganzen Entscheidungen zum Routing und verursacht Link-State-Pakete für das LAN. So wird sichergestellt, daß sich der Umfang der von jedem Router zu erstellenden und zu verwaltenden Datenbank für Verbindungszustände in einem vertretbaren Rahmen hält.

Regelmäßig verschickt jeder Router ein Multicast-Hello-Paket über das LAN. Der Router mit der höchsten Priorität (ein konfigurierbarer Parameter) wird zum designierten Level-1-Router im LAN. Im Falle eines Gleichstands gewinnt der Router mit der höheren MAC-Adresse.

38.2.3 Hello-Pakete im LAN verschicken

Hello-Pakete gestatten es Routern auf der Broadcast-Strecke, die Identität der anderen Level-1-Router in derselben Routing-Area auf dieser Strecke herauszufinden. Die Pakete werden sofort, wenn irgendeine Strecke aktiviert worden ist, an eine besondere Multicast-Zieladresse verschickt. Router achten bei ankommenden Hello-Paketen auf diese Adresse.

38.3 NLSP – Vorgehen

Ein NLSP-Router bezieht bestimmte Informationen aus der Datenbank für die angrenzende Umgebung und fügt ihnen vor Ort abgeleitete Informationen hinzu. Mit diesen Informationen erstellt der Router ein Link-State-Paket (LSP), das die Verbindungszustände zu dessen direkten Nachbarn beschreibt. Alle von allen Routern der Routing-Area erzeugten LSPs ma-

chen gemeinsam die Datenbank für Verbindungszustände der Area aus.

Die NLSP-Spezifikation sieht vor, daß jeder Router eine Kopie der Datenbank für Verbindungszustände betreibt und daß diese Kopien miteinander abgeglichen werden. Die Datenbank wird durch zuverlässig weitergeleitete LSPs in der gesamten Routing-Area synchronisiert, sobald ein Router eine Änderung in der Topologie ausmacht. Zwei Methoden sorgen für eine Weiterleitung genauer Informationen über Topologieänderungen: *Verteilung* (Flooding) und *Empfangsbestätigung*.

Die Verteilung wird angestoßen, sobald ein Router eine Veränderung in der Topologie feststellt. Wenn eine solche Änderung erkannt wird, erstellt der Router ein neues LSP und überträgt es an alle seine Nachbarn. In einem WAN stellen solche LSPs gerichtete Pakete, in einem LAN Multicast-Pakete dar. Beim Empfang eines LSP verwendet der Router die Sequenznummer im Paket, um zu entscheiden, ob das Paket neuer als die aktuell in seiner Datenbank gespeicherte Fassung ist. Wenn es ein neueres LSP ist, überträgt der Router es an alle seine Nachbarn (außer auf der Strecke, über die das LSP empfangen wurde).

Das Verfahren der Empfangsbestätigung unterscheidet sich für LANs und WANs. In WANs antwortet ein LSP empfangender Router mit einem Acknowledgment-Paket. In LANs tritt kein explizites Acknowledgment auf, sondern der designierte Router verschickt im Multicast-Verfahren regelmäßig ein *Complete Sequence Number Packet* (CSNP) genanntes Paket, das alle IDs und Sequenznummern der LSPs enthält, die er in seiner Datenbank für die gesamte Area aufbewahrt. Damit wird sichergestellt, daß andere Router feststellen können, ob sie noch mit dem designierten Router synchronisiert sind oder nicht.

38.4 NLSP – hierarchische Adressierung

NLSP unterstützt ein hierarchisches Adressierungsschema. Jede Routing-Area wird von zwei 32-Bit-Größen identifiziert: einer *Netzwerkadresse* und einer *Maske*. Dieses Zahlenpaar wird als *Area-Adresse* bezeichnet. Es folgt ein Beispiel für eine hexadezimal dargestellte Area-Adresse:

– 01234500 – Bei dieser Zahl handelt es sich um die Netzwerk-Adresse für diese Routing-Area. Jede Netzwerk-Nummer innerhalb jener Area beginnt mit dem Identifizierungscode 012345.

– FFFFFF00 – Bei dieser Zahl handelt es sich um die Maske, die angibt, wie groß der auf die Area selbst entfallende Anteil und wie groß der auf einzelne Netzwerke innerhalb der Area entfallende Anteil an der Netzwerk-Adresse ist.

Im obigen Beispiel weisen die ersten 24 Bit (012345) die Routing-Area aus. Die übrigen 8 Bit werden verwendet, um einzelne Netzwerk-Nummern innerhalb der Routing-Area zu identifizieren (beispielsweise 012345AB, 012345C1, 01234511). In der Abbildung 38.2 sind die vorstehenden Adressierungsbegriffe für drei unterschiedliche Netzwerke in einer einzigen Area hervorgehoben.

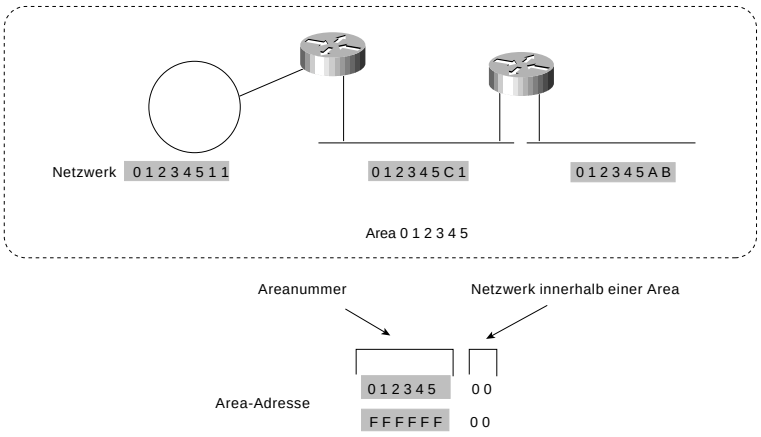


Bild 38.2: NLSP-Adressen setzen sich aus einer Netzwerk-Adresse und einer Maske zusammen

Eine Routing-Area kann bis zu drei verschiedene Area-Adressen mit unterschiedlichen Masken besitzen. Das Vorhandensein von mehr als einer Area-Adresse ermöglicht die Reorganisation der Routing-Area, ohne Operationen zu unterbrechen. Innerhalb einer Domain kann jede beliebige Kombination von Area-Adressen verwendet werden.

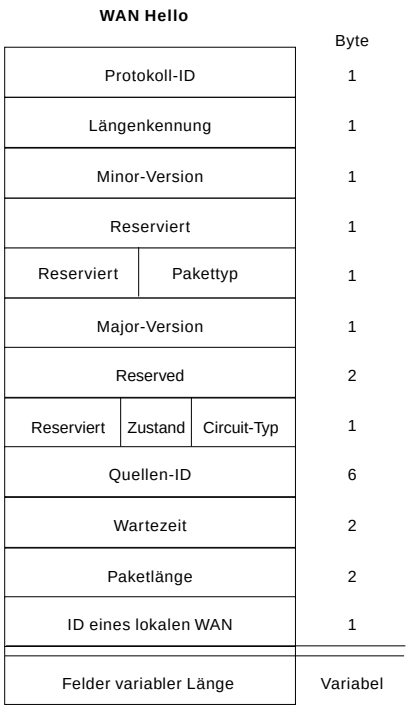
38.5 NLSP – Hello-Pakete

Unter NLSP gibt es zwei Arten von Hello-Paketen: Hello-Pakete für WANs und Hello-Pakete für Level-1-LANs.

38.5.1 Hello-Pakete für WANs

In der Abbildung 38.3 sind die Felder eines Hello-Pakets für WANs dargestellt.

Bild 38.3: Ein Hello-Paket für WANs besteht aus 14 Feldern



Felder des Hello-Pakets für WANs

Im folgenden werden alle in der Abbildung 38.3 dargestellten Felder des Hello-Pakets für WANs kurz beschrieben:

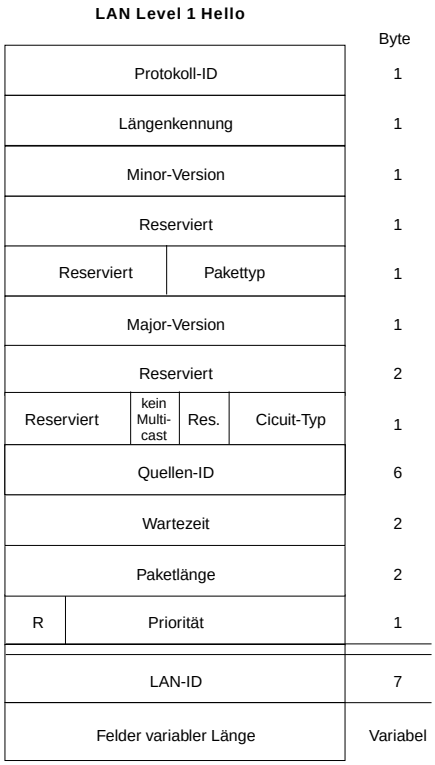
- *Protokoll-ID* – Weist die Ebene des NLSP-Routing mit der Hexadezimalzahl 0x83 aus.
- *Längenkennung* – Bestimmt die Anzahl von Bytes im festen Anteil des Headers.
- *Minor-Version* – Enthält einen möglichen Dezimalwert und wird beim Empfang ignoriert.
- *Reserviert* – Enthält keine Dezimalwerte und wird beim Empfang ignoriert.

- *Pakettyp (5 Bit)* – Enthält einen von 17 möglichen Dezimalwerten.
- *Major-Version* – Enthält einen möglichen Dezimalwert.
- *Reserviert* – Enthält keine Dezimalwerte und wird beim Empfang ignoriert.
- *Zustand (2 Bit)* – Verschickt den mit der Verbindung verknüpften Zustand des Routers (0 = Aktiv, 1 = Initialisierend, 2 = Inaktiv).
- *Circuit-Typ* – Besteht aus zwei Bits. Dieses Feld kann einen der folgenden Werte enthalten:
  - 0 = Reservierter Wert, das gesamte Paket ignorieren.
  - 1 = Nur Level-1-Routing.
  - 2 = Nur Level-2-Routing (der Sender verwendet diese Verbindung für Level 2 Routing).
  - 3 = Sowohl Level 1 als auch Level 2 (der Sender ist ein Level-2-Router und verwendet diese Verbindung für Level-1- und Level-2-Verkehr).
- *Quellen-ID* – Dient als System-ID für den sendenden Router.
- *Wartezeit* – Enthält die für den sendenden Router anzuwendende Wartezeit in Sekunden.
- *Paketlänge* – Bestimmt die Gesamtlänge des Pakets in Bytes, einschließlich des NLSP-Header.
- *ID eines lokalen WAN* – Dient als eindeutige, dieser Strecke beim Erstellen durch den Router zugewiesene ID.
- *Felder variabler Länge* – Besteht aus einer Reihe optionaler Felder.

38.5.2 Hello-Pakete für LANs

In der Abbildung 38.4 sind die Felder des Hello-Pakets für Level-1-LAN dargestellt.

Bild 38.4: Ein Hello-Paket für Level- 1-LANs besteht aus 16 Feldern



Felder des Hello-Pakets für Level-1-LANs

Im folgenden werden alle in der Abbildung 38.4 dargestellten Felder des Hello-Pakets für Level-1-LANs kurz beschrieben:

– *Protokoll-ID* – Weist die Ebene des NLSP Routing mit der Hexadezimalzahl 0x83 aus.

– *Längenkennung* – Bestimmt die Anzahl von Bytes im festen Anteil des Headers (bis einschließlich des Felds LAN-ID).

– *Minor-Version* – Enthält einen möglichen Dezimalwert und wird beim Empfang ignoriert.

- *Reserviert* – Enthält keine Dezimalwerte und wird beim Empfang ignoriert.
- *Pakettyp (5 Bit)* – Enthält einen von 15 möglichen Dezimalwerten.
- *Major Version* – Enthält einen möglichen Dezimalwert.
- *Reserviert* – Enthält keine Dezimalwerte und wird beim Empfang ignoriert.
- *Kein Multicast (1 Bit)* – Zeigt an, falls auf 1 gesetzt, daß der Paketsender keinen an eine Multicast-Adresse gerichteten Datenverkehr empfangen kann. (Zukünftige Pakete in diesem LAN müssen an die Broadcast-Adresse geschickt werden.)
- *Circuit-Typ* – Besteht aus zwei Bits. Dieses Feld kann einen der folgenden Werte enthalten:
  - 0 = Reservierter Wert, das gesamte Paket ignorieren.
  - 1 = Nur Level-1-Routing.
  - 2 = Nur Level-2-Routing (der Sender verwendet diese Verbindung für Level-2-Routing).
  - 3 = Sowohl Level 1 als auch Level 2 (der Sender ist ein Level-2-Router und verwendet diese Verbindung für Level-1- und Level-2-Verkehr).
- *Quellen-ID* – Enthält die System-ID des sendenden Router.
- *Wartezeit* – Enthält die für den sendenden Router anzuwendende Wartezeit in Sekunden.
- *Paketlänge* – Bestimmt die Gesamtlänge des Pakets in Bytes, einschließlich des NLSP-Headers.
- *R* – Enthält keine Dezimalwerte und wird beim Empfang ignoriert.
- *Priorität (7 Bit)* – Dient als mit dem designierten Router des LAN Level 1 verknüpfte Priorität. (Höhere Zahlen bedeuten eine höhere Priorität.)

- *LAN-ID* – Enthält die System-ID (6 Bytes) des designierten Router des LAN Level 1, gefolgt von einem durch diesen designierten Router zugewiesenen Feld.
- *Felder variabler Länge* – Besteht aus einer Reihe optionaler Felder.

KAPITEL 39

Open Systems Interconnection (OSI) Routing-Protokoll

39.1 Hintergrund

Die International Organization for Standardization (ISO) entwickelte eine vollständige Sammlung von Routing-Protokollen für den Einsatz in der Open-Systems-Interconnection-Protokollsammlung (OSI). Zu diesen zählen *Zwischensystem-zu-Zwischensystem* (Intermediate System-to-Intermediate System = IS-IS), *Endsystem-zu-Zwischensystem* (End System-to-Intermediate System = ES-IS) und das *Interdomain Routing Protocol* (IDRP). In diesem Kapitel werden die grundlegenden Operationen jedes dieser Protokolle behandelt.

IS-IS basiert auf der Arbeit, die bei der Digital Equipment Corporation (Digital) ursprünglich für DECnet/OSI (DECnet Phase V) erbracht wurde. IS-IS wurde ursprünglich entwickelt, um das Routing in ISO *Connectionless-Network-Protocol*-(CLNP-)Netzwerken vorzunehmen. Seither wurde eine Version entwickelt, die sowohl Netzwerke unter CLNP als auch unter dem *Internet Protocol* (IP) unterstützt. Diese Version wird üblicherweise als *Integrated IS-IS* bezeichnet (wurde aber auch *Dual IS-IS* genannt).

Die OSI-Routing-Protokolle sind in mehreren ISO-Dokumenten zusammengefaßt, zu denen auch die ISO 10589 gehört, in der die Definition für IS-IS erfolgt. Das American National Standards Institute (ANSI) X3S3.3 (Netzwerk- und Transportschichten) Komitee war die treibende Kraft hinter der ISO-

Standardisierung von IS-IS. Andere ISO-Dokumente sind die ISO 9542 (die ES-IS definiert) und ISO 10747 (die IDRP definiert).

39.1.1 OSI-Netzwerk-Terminologie

In der Welt der OSI-Netzwerke werden einige bestimmte Begriffe verwendet wie beispielsweise *Endsystem* (ES), womit jeder nicht weiterleitende Netzknoten gemeint ist, und *Zwischensystem* (Intermediate System = IS), womit ein Router gemeint ist. Diese Begriffe bilden die Grundlage für die OSI-Protokolle ES-IS und IS-IS. Das ES-IS-Protokoll ermöglicht es ESs und IS, einander zu bemerken. Das IS-IS-Protokoll sorgt für das Routing zwischen IS. Andere wichtige Begriffe für OSI-Netzwerke sind: *Area*, *Domain*, *Level-1-Routing* und *Level-2-Routing*. Bei einer Area handelt es sich um eine Gruppe aneinandergrenzender Netzwerke und angebundener Hosts, die von einem Netzwerk-Administrator oder -Verwalter als Area festgelegt wurden. Bei einer Domain handelt es sich um eine Ansammlung miteinander verbundener Areas. Routing Domains (RDs) bieten für alle in ihr befindlichen Endsysteme eine vollständige Anbindung. Mit Level-1-Routing wird das Routing innerhalb einer Level-1-Area bezeichnet, wohingegen es sich beim Level-2-Routing um das Routing zwischen Level-1-Areas handelt. Im Bild 39.1 sind die Beziehungen zwischen Areas und Domains sowie die zwischen diesen vorkommenden Routing-Level dargestellt.

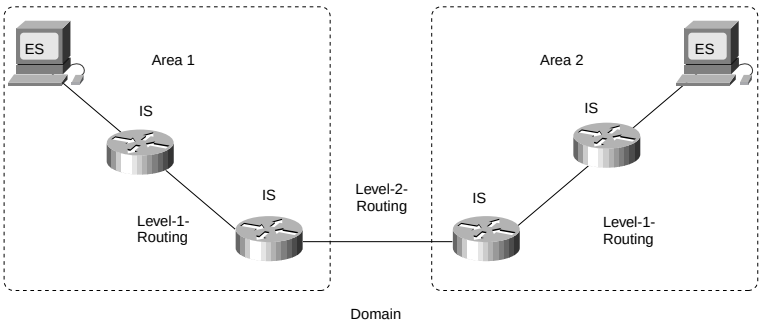


Bild 39.1: Areas kommen in einer größeren Domain vor und greifen für die Kommunikation untereinander auf Level-2-Routing zurück

39.2 Endsystem-zu-Zwischensystem (ES-IS)

Bei Endsystem-zu-Zwischensystem (ES-IS) handelt es sich um ein OSI-Protokoll, mit dem definiert wird, wie Endsysteme (Hosts) und Zwischensysteme (Router) etwas voneinander mitbekommen, ein Vorgang, der als *Konfiguration* bezeichnet wird. Die Konfiguration muß stattfinden, bevor es ein Routing zwischen ES geben kann.

ES-IS ist eher ein Protokoll zum *Entdecken* als ein Routing-Protokoll. Es unterscheidet zwischen drei verschiedenen Arten von Subnetzen: *Punkt-zu-Punkt-Subnetzen*, *Broadcast-Subnetzen* und *Subnetzen mit einer allgemeinen Topologie*. Punkt-zu-Punkt-Subnetze wie beispielsweise serielle WAN-Verbindungen stellen eine Punkt-zu-Punkt-Verbindung zwischen zwei Systemen bereit. Broadcast-Subnetze wie beispielsweise Ethernet oder IEEE 802.3 leiten eine einzelne physikalische Nachricht an alle im Subnetz befindlichen Knoten weiter. Subnetze mit einer allgemeinen Topologie wie beispielsweise X.25 unterstützen eine beliebige Anzahl von Systemen. Anders als bei Broadcast-Subnetzen wächst in einem Subnetz mit einer allgemeinen Topologie jedoch der Aufwand für eine Übertragung über *n* Wege direkt mit der Größe des Subnetzes.

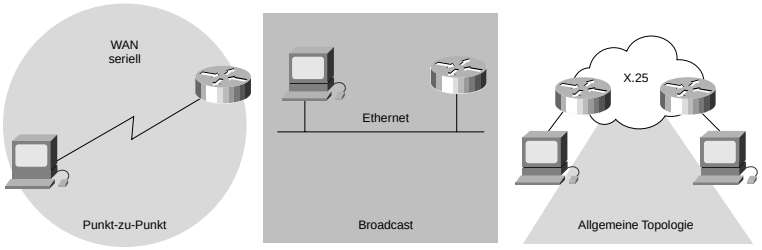


Bild 39.2: ES-IS kann in Punkt-zu-Punkt- und Broadcast-Subnetzen sowie in Subnetzen mit allgemeiner Topologie eingesetzt werden

39.2.1 ES-IS-Konfiguration

Bei der ES-IS-Konfiguration handelt es sich um denjenigen Vorgang, bei dem ES und IS einander entdecken, so daß es zum Routing zwischen ES kommen kann. Die Konfigurationsinformationen werden in regelmäßigen Abständen durch zwei Nachrichtentypen übertragen: *ES-Hello-Nachrichten* (ESH) und *IS-Hello-Nachrichten* (ISH). ESH werden von

ES generiert und an jedes IS im Subnetz geschickt. ISH werden von IS generiert und an jedes ES im Subnetz geschickt. Diese Hello-Nachrichten dienen in erster Linie dazu, die Adressen des Subnetzes und der Netzwerk-Schicht der generierenden Systeme zu übermitteln. Soweit möglich, versucht ES-IS, die Konfigurationsinformationen gleichzeitig an viele Systeme zu verschicken. In Broadcast-Subnetzen werden ES-IS Hello-Nachrichten über eine besondere, alle Endsysteme kennzeichnende Multicast-Adresse an alle IS geschickt. In Subnetzen mit einer allgemeinen Topologie überträgt ES-IS wegen des hohen Aufwands bei Multicast-Übertragungen üblicherweise keine Konfigurationsinformationen.

39.2.2 ES-IS-Adressierung

Das ES-IS-Konfigurationsprotokoll übermittelt sowohl die Adresse der OSI-Netzwerkschicht als auch des OSI-Subnetzes. Die Adresse der OSI-Netzwerkschicht weist entweder den Network Service Access Point (NSAP), der die Schnittstelle zwischen der OSI Schicht 3 und Schicht 4 darstellt, oder den Network Entity Title (NET) aus, bei dem es sich um die Einheit der Netzwerkschicht in einem OSI-IS handelt. Adressen von OSI-Subnetzen bzw. *Subnetwork Point Of Attachment Addresses* (SNPAs) sind die Stellen, an denen ein ES oder IS physikalisch an ein Subnetz angebunden ist. Die SNPA weist jedes an das Subnetz angebundene System eindeutig aus. In einem Ethernet-Netzwerk ist die SNPA beispielsweise eine 48-Bit-*Media-Access-Control*-(MAC-)Adresse. Zu den von ES-IS übertragenen Konfigurationsinformationen gehört die NSAP-zu-SNPA- oder NET-zu-SNPA-Zuordnung.

39.3 Zwischensystem-zu-Zwischensystem (IS-IS)

Bei *Zwischensystem-zu-Zwischensystem* (Intermediate System-to-Intermediate System = IS-IS) handelt es sich um ein hierarchisches Routing-Protokoll für Verbindungszustände von OSI, welches das Netzwerk mit Informationen über Verbindungszustände beliefert, um ein konsistentes Abbild der Netzwerk-Topologie zu erstellen. Um den Entwurf und Betrieb von Routern zu vereinfachen, unterscheidet IS-IS zwischen Level-1-

und Level-2-IS. Level-1-IS kommunizieren mit anderen Level-1-ISs in derselben Area. Level-2-IS sind für das Routing zwischen Level-1-IS zuständig und bilden einen Routing-Backbone zwischen den Domains. Das hierarchische Routing vereinfacht den Entwurf des Backbone, da Level-1-IS nur zu wissen brauchen, wie sie zum nächsten Level-2-IS gelangen. Das Routing-Protokoll für den Backbone kann ohne Einfluß auf das innerhalb der Area eingesetzte Routing-Protokoll ausgetauscht werden.

39.3.1 OSI – der Routing-Vorgang

Jedes ES befindet sich in einer bestimmten Area. Das OSI-Routing beginnt, sobald die ES den nächstgelegenen IS entdecken, indem sie auf ISH-Pakete achten. Wenn ein ES ein Paket an ein anderes ES schicken will, schickt es das Paket an eines der IS im direkt angebundenen Netzwerk. Der Router sieht dann nach der Zieladresse und leitet das Paket über die beste Route weiter. Wenn sich das Ziel-ES im gleichen Subnetz befindet, ist dies dem lokalen IS bekannt, da es auf ESHs achtet, so daß es das Paket entsprechend weiterleitet. Das IS kann auch eine *Redirect*-Nachricht (RD) an die Quelle zurückschicken, um ihr mitzuteilen, daß es eine direktere Route gibt. Wenn es sich bei der Zieladresse um ein ES in einem anderen Subnetz in derselben Area handelt, ist dem IS die richtige Route bekannt, so daß es das Paket entsprechend weiterleitet. Wenn es sich bei der Zieladresse um ein ES in einer anderen Area handelt, schickt das Level-1-IS das Paket an das nächstgelegene Level-2-IS. Das Weiterleiten über Level-2-IS setzt sich fort, bis das Paket ein Level-2-IS in der Ziel-Area erreicht. Innerhalb der Ziel-Area leiten IS das Paket über den besten Pfad weiter, bis das Ziel-ES erreicht ist.

Update-Nachrichten für Verbindungszustände helfen IS dabei, etwas über die Netzwerktopologie zu erfahren. Zuerst generiert jedes IS ein Update-Paket, das die ES und IS, mit denen es verbunden ist, sowie die zugehörigen Metriken angibt. Das Update wird dann an alle angrenzenden IS geschickt, die es an deren Nachbarn weiterleiten (*Flooding*) und so weiter (Sequenznummern beenden die Verbreitung und unterscheiden alte von neuen Updates). Mittels dieser Updates kann sich jedes IS eine vollständige Topologie des Netzwerks erstellen.

Wenn sich die Topologie ändert, werden neue Updates verschickt.

39.3.2 IS-IS – Metriken

IS-IS greift auf eine einzige vorgegebene Metrik mit einem maximalen Pfadwert von 1024 zurück. Die beliebige Metrik wird typischerweise von einem Netzwerk-Administrator zugewiesen. Jede einzelne Verbindung kann maximal den Wert 64 annehmen, und Pfadwerte werden durch Aufsummieren der Verbindungswerte berechnet. Maximale Metrikwerte werden in dieser Höhe gesetzt, damit die Aufgliederung einerseits verschiedene Verbindungsarten unterstützen kann, während gleichzeitig sichergestellt wird, daß der für die Bestimmung der Route eingesetzte Algorithmus für den kürzesten Pfad effektiv genug ist. IS-IS definiert weiterhin drei optionale Metriken (Aufwand): Verzögerung, Kosten und Fehler. Die Metrik des Verzögerungsaufwands steht für die auf einer Verbindung anfallende Verzögerungszeit. Die Metrik des Kostenaufwands steht für die mit der Verwendung der Verbindung verbundenen Kommunikationskosten. Die Metrik des Fehleraufwands steht für die Fehlerrate der Verbindung. IS-IS bildet diese vier Metriken auf die *Dienstqualität*-Option (Quality Of Service = QOS) im Header des CLNP-Pakets ab. IS-IS verwendet diese Zuordnung für die Berechnung von Routen durch das Internetzwerk.

IS-IS – Paketformate

IS-IS greift auf drei grundlegende Paketformate zurück: *IS-IS Hello-Pakete*, *Link-State-Pakete* (LSPs) und *Sequenznummer-Pakete* (SNPs). Jedes dieser drei IS-IS-Pakete hat ein komplexes Format mit folgenden drei unterschiedlichen logischen Bestandteilen. Der erste Teil besteht aus einem festen 8 Byte umfassenden Header, den alle drei Pakettypen gemein haben. Der zweite Teil ist ein pakettypspezifischer Anteil mit einem festen Format. Der dritte Teil ist ebenfalls pakettypspezifisch, aber von variabler Länge. Das Bild 39.3 veranschaulicht den logischen Aufbau von IS-IS-Paketen. Im Bild 39.4 sind die Felder des gemeinsamen Header von IS-IS-Paketen dargestellt.

Bild 39.3: IS-IS-Pakete setzen sich aus drei logischen Headern zusammen

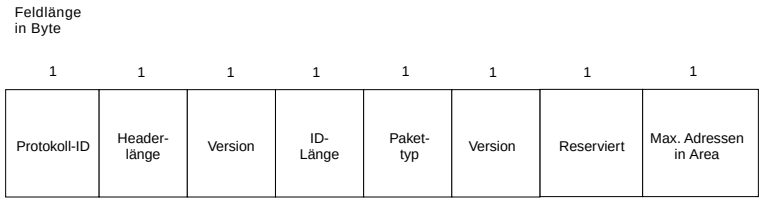


Bild 39.4: IS-IS-Pakete setzen sich aus acht Feldern zusammen

Im folgenden werden die im Bild 39.4 dargestellten Felder kurz beschrieben:

- *Protokoll-ID* – Weist das Protokoll IS-IS aus und enthält den Wert 131.
- *Header-Länge* – Enthält die feste Länge des Header. Die Länge ist immer 8 Byte, aber dennoch enthalten, damit sich IS-IS-Pakete nicht wesentlich von CLNP-Paketen unterscheiden.
- *Version* – Enthält in der aktuellen IS-IS-Spezifikation den Wert 1.
- *ID-Länge* – Gibt den Umfang des Anteils der ID an einer NSAP-Adresse an. Falls das Feld einen Wert von 1 bis 8 (einschließlich) enthält, dann erstreckt sich der Anteil der ID über diese Anzahl von Bytes. Falls das Feld den Wert 0 enthält, dann erstreckt sich der Anteil der ID über 6 Bytes. Wenn das Feld den Wert 255 (alles Einsen) enthält, dann erstreckt sich der Anteil der ID an der NSAP-Adresse über null Bytes.
- *Pakettyp* – Gibt den Typ des IS-IS-Pakets an (Hello, LSP, SNP).
- *Version* – Wird nach dem Feld Packet Type wiederholt.

– *Reserviert* – Wird vom Empfänger ignoriert und ist gleich 0.

– *Max. Adressen in Area* – Gibt die Anzahl der in dieser Area gestatteten Adressen an.

Auf den gemeinsamen Header folgend verfügt jeder Pakettyp über einen unterschiedlichen zusätzlichen festen Anteil, auf den ein variabler Anteil folgt.

39.4 Integrated IS-IS

Bei Integrated IS-IS handelt es sich um eine Version des OSI-Routing-Protokolls IS-IS, die einen einzigen Routing-Algorithmus verwendet, um mehr Protokolle für die Netzwerk-Schicht als nur CLNP zu unterstützen. Integrated IS-IS wird manchmal auch nach einer für IP- und CLNP-Netzwerke entworfenen Version als Dual IS-IS bezeichnet. Den IS-IS-Paketen werden mehrere Felder hinzugefügt, damit IS-IS zusätzliche Netzwerk-Schichten unterstützen kann. Diese Felder informieren Router über die Erreichbarkeit von Netzwerk-Adressen aus anderen Protokollsammlungen und andere von einer bestimmten Protokollsammlung benötigte Informationen. Implementierungen von Integrated IS-IS verschicken nur einen Satz Routing-Updates, was effizienter ist als zwei getrennte Implementierungen.

Integrated IS-IS stellt eine von zwei Möglichkeiten dar, mehrere Protokolle für die Netzwerkschicht auf einem Router zu unterstützen; die andere Vorgehensweise wird als *Ships-in-the-night* bezeichnet. Das Ships-in-the-night-Routing unterstützt den Einsatz völlig unterschiedlicher und getrennter Routing-Protokolle für jedes Netzwerk-Protokoll, so daß die vielen Routing-Protokolle im wesentlichen unabhängig voneinander bestehen. Dadurch ziehen die verschiedenen Arten von Routing-Informationen wie Schiffe in der Nacht vorüber. Das Integrated Routing verfügt durch von einem einzigen Routing-Protokoll berechnete Tabellen über die Fähigkeit, mehrere Protokolle für die Netzwerk-Schicht weiterzuleiten, wodurch die Mittel des Router geschont werden. Integrated IS-IS greift auf dieses Vorgehen zurück.

39.5 Interdomain Routing Protocol (IDRP)

Beim Interdomain Routing Protocol (IDRP) handelt es sich um ein OSI-Protokoll, das angibt, wie Router mit Routern in anderen Domains kommunizieren. IDRP wurde für eine nahtlose Zusammenarbeit mit CLNP, ES-IS und IS-IS entworfen. IDRP basiert auf dem Border Gateway Protocol (BGP), einem Protokoll für das Routing zwischen Domains, das seinen Ursprung in der IP-Gruppe hat. Zu den Eigenschaften von IDRP gehören:

- Unterstützung der CLNP-Dienstqualität (Quality Of Service = QOS)
- Vermeidung von Schleifen durch Verfolgen aller von einer Route durchquerten Routing Domains (RDs)
- Reduzierung von Routeninformationen und -bearbeitung durch den Einsatz von Confederations (Bündnissen), die Komprimierung von Informationen zu RD-Pfaden und anderen Mitteln
- Zuverlässigkeit durch einen eingebauten zuverlässigen Transport
- Sicherheit durch den Einsatz von kryptographischen Signaturen für jedes Paket
- Routenserver

39.5.1 IDRP – Terminologie

Mit IDRP werden einige umgebungsspezifische Begriffe eingeführt. Bei diesen handelt es sich um *Border Intermediate System* (BIS), *Routing Domain* (RD), *Routing Domain Identifier* (RDI), *Routing Information Base* (RIB) und *Confederation*. Bei einem BIS handelt es sich um ein IS, das am Routing zwischen Domains beteiligt ist und somit IDRP einsetzt. Eine RD ist eine Gruppe von ES und IS, die mit denselben Verwaltungsregeln betrieben werden und sich einen gemeinsamen Routing-Plan teilen. Bei einer RDI handelt es sich um eine eindeutige ID für eine RD. Bei einer RIB handelt es sich um eine von IDRP verwendete Routing-Datenbank, die von jedem BIS mit aus der RD und von anderen BIS empfangenen Informationen aufgebaut wird. Eine RIB enthält die von einer

bestimmten BIS für den Einsatz ausgewählten Routen. Bei einer Confederation (Bündnis) handelt es sich um eine Gruppe von RD, die den RD außerhalb der Confederation als eine einzige RD erscheint. Die Topologie der Confederation ist für RD außerhalb der Confederation nicht erkennbar. Confederations müssen ineinander verschachtelt sein und helfen dabei, den Netzwerkverkehr zu reduzieren, indem sie im Internetzwerk als Firewalls agieren. Im Bild 39.5 ist die Beziehung zwischen IDRP-Einheiten dargestellt.

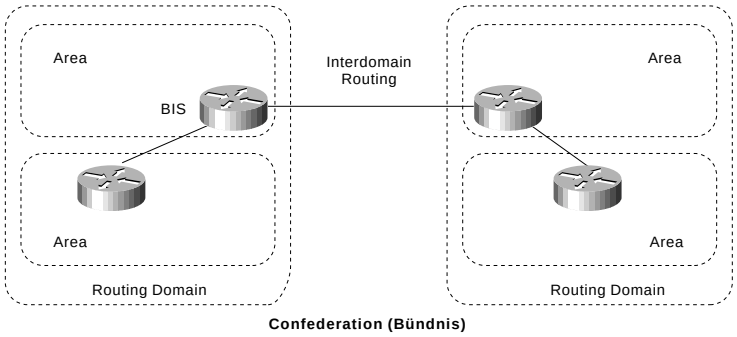


Bild 39.5: Domains kommunizieren mittels Border Intermediate Systems (BIS) miteinander

39.5.2 IDRP-Routing

Eine IDRP-Route besteht aus einer Folge von RDIs, von denen einige Confederations sein können. Jedes BIS ist so konfiguriert, daß es die RD und die Confederations kennt, zu denen es gehört. Von anderen BISs, RDs und Confederations erfährt es durch den Informationsaustausch mit jedem Nachbar. Wie beim Distance-Vector-Routing häufen sich Routen zu einem bestimmten Ziel vom Ziel ausgehend an. Nur Routen, die den lokalen Sicherheitsbestimmungen eines BIS genügen und für den Einsatz ausgewählt wurden, werden an andere BIS übergeben. Die Neubestimmung von Routen erfolgt partiell beim Auftreten eines der folgenden drei Ereignisse: Ein inkrementelles Routing-Update mit neuen Routen wird empfangen, ein Nachbar eines BIS wird deaktiviert, oder ein Nachbar eines BIS wird aktiviert.

KAPITEL 40

Open Shortest Path First (OSPF)

40.1 Hintergrund

Bei Open Shortest Path First (OSPF) handelt es sich um ein Routing-Protokoll, das von der Interior-Gateway-Protocol-(IGP-)Arbeitsgruppe der Internet Engineering Task Force (IETF) für Internet-Protocol-(IP-)Netzwerke entwickelt wurde. Die Arbeitsgruppe kam 1988 zusammen, um ein auf dem Shortest-Path-First-(SPF-)Algorithmus basierendes IGP für den Einsatz im Internet zu entwerfen. Ähnlich wie das Interior Gateway Routing Protocol (IGRP) wurde OSPF entwickelt, weil das Routing Information Protocol (RIP) Mitte der 1980er Jahre zusehends weniger in der Lage war, große, heterogene Internetzwerke zu bedienen. In diesem Kapitel werden die Routing-Umgebung OSPF, der zugrundeliegende Routing-Algorithmus sowie allgemeine Komponenten des Protokolls behandelt.

OSPF wurde von verschiedenen Forschungsvorhaben abgeleitet, zu denen der 1978 für das ARPANET (ein in den frühen 1970er Jahren von BBN entwickeltes wegweisendes, auf dem Austausch von Paketen basierendes Netzwerk) entwickelte SPF-Algorithmus nach Bolt, Beranek, Newman (BBN), Dr. Radia Perlmans Arbeit zur fehlertoleranten Verbreitung von Routing-Informationen (1988), BBNs Arbeit zum Area-Routing (1986) sowie eine frühe Fassung des OSI-Routing-Protokolls Intermediate-System-to-Intermediate-System (IS-IS) gehören.

OSPF verfügt über zwei wesentliche Eigenschaften. Erstens ist das Protokoll »offen«, was bedeutet, daß dessen Spezifikation zur Public Domain gehört. Die OSPF-Spezifikation wurde als Request For Comments (RFC) 1247 veröffentlicht. Zweitens basiert OSPF auf dem SPF-Algorithmus, der manchmal auch nach seinem Erschaffer als *Dijkstra*-Algorithmus bezeichnet wird.

Bei OSPF handelt es sich um ein Routing-Protokoll für Verbindungszustände (*Link-States*), welches das Versenden von *Link-State Advertisements* (LSAs) an alle anderen Router auf derselben Hierarchiestufe der Area erfordert. Informationen über angebundene Schnittstellen, verwendete Metriken und andere Variablen werden in die LSAs von OSPF einbezogen. Während OSPF-Router Informationen über die Verbindungszustände sammeln, verwenden sie für die Berechnung des kürzesten Pfads zum nächsten Knoten den SPF-Algorithmus.

Als Routing-Protokoll für Verbindungszustände unterscheidet sich OSPF von RIP und IGRP, bei denen es sich um Routing-Protokolle nach dem Distance-Vector-Verfahren handelt. Router, die den Distance-Vector-Algorithmus verwenden, verschikken ihre gesamten Routing-Tabellen oder Teile davon in Routing-Update-Nachrichten an ihre Nachbarn.

40.2 Routing-Hierarchie

Anders als RIP kann OSPF innerhalb einer Hierarchie betrieben werden. Die größte Einheit in der Hierarchie ist das *autonome System* (AS), bei dem es sich um eine Ansammlung von Netzwerken unter einer gemeinsamen Administration handelt, für die eine gemeinsame Routing-Strategie zum Einsatz kommt. Bei OSPF handelt es sich um ein innerhalb eines AS verwendetes (Interior Gateway) Routing-Protokoll, obgleich es dazu in der Lage ist, Routen von anderen AS zu empfangen oder an sie zu verschicken.

Ein AS kann in eine Anzahl von *Areas* unterteilt werden, bei denen es sich um Gruppen angrenzender Netzwerke und angebundener Hosts handelt. Router mit mehreren Schnittstellen können an mehreren Areas beteiligt sein. Diese als *Area Border Router* (Übergangsrouter) bezeichneten Router pflegen für jede Area eine eigene Topologie-Datenbank.

Eine Topologie-Datenbank stellt im wesentlichen eine Übersicht über Netzwerke und deren Verhältnisse zu Routern dar. Die Topologie-Datenbank enthält die von allen Routern derselben Area empfangene Ansammlung von LSAs. Da sich Router einer Area dieselben Informationen teilen, verfügen sie über identische Topologie-Datenbanken.

Der Begriff *Domain* wird manchmal verwendet, um einen Teil des Netzwerks zu bezeichnen, in dem alle Router über identische Topologie-Datenbanken verfügen. Häufig wird Domain als Synonym für AS verwendet.

Die Topologie einer Area ist für Einheiten außerhalb der Area nicht sichtbar. Durch die Abtrennung der Area-Topologien verursacht OSPF weniger Routingverkehr, als wenn das AS nicht unterteilt wäre.

Die Unterteilung in Areas sorgt für zwei unterschiedliche Arten von OSPF-Routing, die davon abhängig sind, ob sich die Quelle und das Ziel in derselben Area oder in unterschiedlichen Areas befinden. Zum Intra-Area-Routing kommt es, wenn sich Quelle und Ziel in derselben Area befinden; zum Inter-Area-Routing, wenn sie in unterschiedlichen Areas liegen.

Ein OSPF-*Backbone* ist für das Verteilen von Routing-Informationen zwischen Areas verantwortlich. Er setzt sich aus allen Übergangsroutern der Area, nicht vollständig in einer Area enthaltenen Netzwerken sowie deren angebundenen Routern zusammen. Im Bild 40.1 ist ein Beispiel für ein Internetzwerk mit mehreren Areas dargestellt.

In der Abbildung stellen die Router 4, 5, 6, 10, 11 und 12 den Backbone dar. Wenn der Host H1 in Area 3 ein Paket an den Host H2 in Area 2 schicken will, wird das Paket an den Router 13 gesendet, der es an den Router 12 weiterleitet, der das Paket seinerseits an den Router 11 schickt. Der Router 11 leitet das Paket dann über den Backbone zum Area Border Router 10, der das Paket über zwei Intra-Area-Router (Router 9 und 7) an den Host H2 weiterleitet.

Beim Backbone selber handelt es sich um eine OSPF-Area, so daß alle Backbone-Router die gleichen Prozeduren und Algorithmen für die Wartung der Routing-Informationen innerhalb des Backbone verwenden. Die Topologie des Backbone ist für

alle Intra-Area-Router ebenso unsichtbar, wie es die einzelnen Area-Topologien für den Backbone sind.

Areas können auf eine Weise definiert werden, daß der Backbone nicht zusammenhängend ist. In diesem Fall muß die Verbundenheit des Backbone über *virtuelle Verbindungen* wiederhergestellt werden. Virtuelle Verbindungen werden zwischen allen denjenigen Backbone-Routern konfiguriert, die über eine Verbindung zu einer nicht zum Backbone gehörenden Area verfügen; und sie wirken wie direkte Verbindungen.

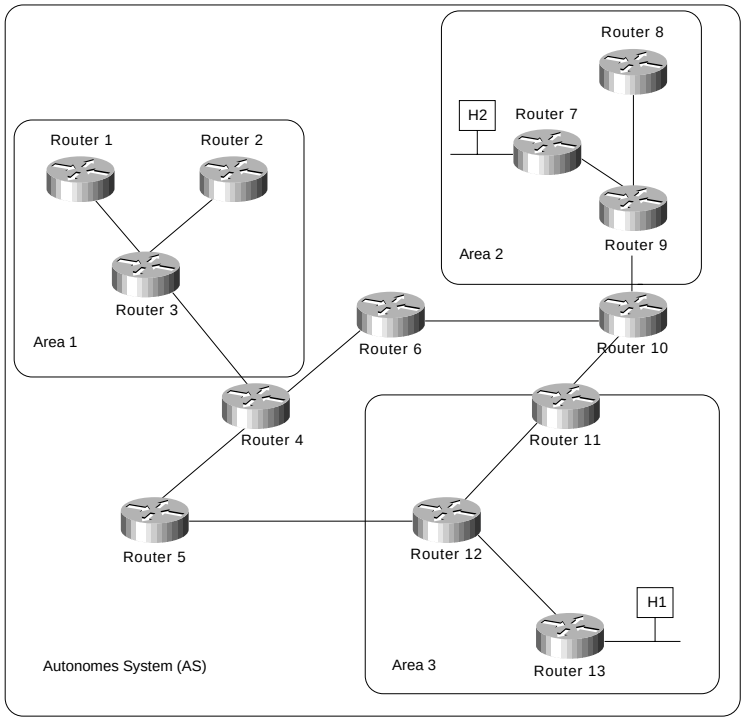


Bild 40.1: Ein OSPF-AS setzt sich aus mehreren durch Router miteinander verbundenen Areas zusammen

Übergangsknoten eines AS, auf denen OSPF läuft, erfahren durch Protokolle für externe Gateways (EGPs) wie beispielsweise das Exterior Gateway Protocol (EGP) oder das Border Gateway Protocol (BGP), aber auch durch Konfigurationsinformationen von außerhalb liegenden Routen. Weitere Informationen zu diesen Protokollen finden Sie in Kapitel 33, »Border Gateway Protocol (BGP)«.

40.3 SPF-Algorithmus

Der Routing-Algorithmus Shortest Path First (SPF) stellt die Basis beim Vorgehen von OSPF dar. Sobald ein SPF-Router angeschaltet wird, initialisiert er die Datenstrukturen seines Routing-Protokolls und wartet anschließend auf Zeichen von Protokollen darunterliegender Schichten, daß seine Schnittstellen funktionieren.

Nachdem einem Router das Funktionieren seiner Schnittstellen zugesichert worden ist, greift er auf das OSPF *Hello Protocol* zurück, um sich *Nachbarn* zu verschaffen, bei denen es sich um Router mit Schnittstellen zu einem gemeinsamen Netzwerk handelt. Der Router verschickt Hello-Pakete an seine Nachbarn und erhält deren Hello-Pakete. Daneben, daß Hello-Pakete beim Verschaffen von Nachbarn behilflich sind, fungieren sie weiterhin als Keep-Alives, wodurch Router erfahren, daß andere Router noch immer in Betrieb sind.

In *Multiaccess-Netzwerken* (Netzwerke, die mehr als zwei Router unterstützen) wählt das Hello-Protokoll einen *designierten Router* sowie einen designierten Reserve-Router aus. Unter anderem zeichnet sich der designierte Router für die Generierung von LSAs für das gesamte Multiaccess-Netzwerk verantwortlich. Designierte Router ermöglichen eine Reduktion des Netzwerk-Verkehrs und des Umfangs der Topologie-Datenbank.

Wenn die Datenbanken für Verbindungszustände von zwei benachbarten Routern miteinander synchronisiert werden, gelten die Router als direkt benachbart (*adjacent*). In Multiaccess-Netzwerken bestimmt der designierte Router, welche Router direkt benachbart sein sollen. Unter Paaren von direkt benachbarten Routern werden Topologie-Datenbanken miteinander synchronisiert. Die jeweilige Nähe untereinander steuert die Verteilung von Paketen des Routing-Protokolls, die nur aufgrund der Nachbarschaft verschickt und empfangen werden.

Jeder Router verschickt regelmäßig ein LSA, um Informationen über die direkten Nachbarn eines Routers zu liefern oder andere Router darüber zu informieren, wenn sich der Zustand eines Routers verändert. Durch einen Vergleich der eingeführten direkten Nachbarn mit den Verbindungszuständen lassen

sich ausgefallene Router schnell entdecken und läßt sich die Netzwerk-Topologie entsprechend ändern. Unter Rückgriff auf die aus LSAs generierte Topologie-Datenbank berechnet jeder Router einen Baum mit sich selber als Wurzel mit den kürzesten Wegen (Shortest Path). Dieser Baum bringt wiederum eine Routing-Tabelle hervor.

40.4 Paketformat

Alle OSPF-Pakete beginnen mit einem 24-Byte-Header, wie er im Bild 40.2 dargestellt ist.

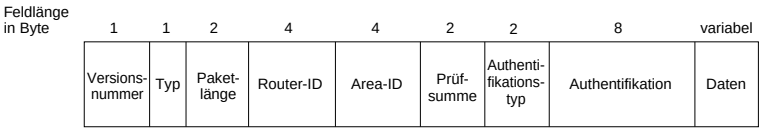


Bild 40.2: OSPF-Pakete setzen sich aus neun Feldern zusammen

Die Felder des im Bild 40.2 dargestellten Headers werden im folgenden kurz beschrieben:

- *Versionsnummer* – Weist die verwendete Version von OSPF aus.
- *Typ* – Enthält einen der folgenden Werte für den OSPF-Pakettyp:
  - Hello: Richtet Beziehungen zu Nachbarn ein und wartet sie.
  - Database-Description: Beschreibt den Inhalt der Topologie-Datenbank. Diese Nachrichten werden bei der Initialisierung einer direkten Nachbarschaft ausgetauscht.
  - Link-State-Request: Fordert Teile der Topologie-Datenbank von benachbarten Routern an. Diese Nachrichten werden ausgetauscht, nachdem ein Router festgestellt hat (durch das Untersuchen von Database-Description-Paketen), daß Teile seiner Topologie-Datenbank veraltet sind.
  - Link-State-Update: Antwortet auf ein Link-State-Request-Paket. Diese Nachrichten werden außerdem für das regelmäßige Streuen von LSAs eingesetzt. In einem

einzigen Link-State-Update-Paket können mehrere LSAs enthalten sein.

  - Link-State-Acknowledgment: Bestätigt Link-State-Update-Pakete.

- *Paketlänge* – Gibt die Länge des Pakets einschließlich des OSPF-Header in Bytes an.
- *Router-ID* – Weist die Quelle des Pakets aus.
- *Area-ID* – Weist die Area aus, zu der das Paket gehört. Alle OSPF-Pakete sind mit einer einzelnen Area verknüpft.
- *Prüfsumme* – Überprüft den gesamten Paketinhalt auf Verluste beim Übertragen.
- *Authentifizierungstyp* – Enthält die Art der Authentifizierung. Jeglicher Protokollaustausch unter OSPF ist authentifiziert. Die Art der Authentifizierung läßt sich für jede Area konfigurieren.
- *Authentifizierung* – Enthält Informationen zur Authentifizierung.
- *Daten* – Enthält gekapselte Informationen übergeordneter Schichten.

40.5 Weitere Eigenschaften von OSPF

Zu den weiteren Eigenschaften von OSPF gehören *Equal-Cost*, *Multipath-Routing* sowie auf *Type-Of-Service*-(TOS-)-Requests basierendes Routing übergeordneter Schichten. Das TOS-basierende Routing unterstützt diejenigen Protokolle übergeordneter Schichten, die bestimmte Arten von Diensten angeben können. Beispielsweise könnte eine Anwendung angeben, daß gewisse Daten dringend sind. Wenn OSPF-Verbindungen mit hoher Priorität zur Verfügung stehen, dann können diese für den Transport des dringenden Datagramms verwendet werden.

OSPF unterstützt eine oder mehrere Metriken. Wenn nur eine Metrik zum Einsatz gelangt, wird sie als beliebig betrachtet und TOS nicht unterstützt. Wenn mehrere Metriken verwendet werden, wird TOS optional durch den Einsatz einer eigenen Metrik (und damit einer eigenen Routing-Tabelle) für jede

der acht möglichen Kombinationen mit den drei IP-TOS-Bits (die Bits für *Verzögerung*, *Durchsatz* und *Zuverlässigkeit*) unterstützt. Wenn die IP-TOS-Bits beispielsweise eine geringe Verzögerung, niedrigen Durchsatz und hohe Zuverlässigkeit angeben, dann berechnet OSPF Routen zu allen Zielen, die auf diesen TOS-Angaben basieren.

IP-Subnetzmasken werden für jedes bekanntgemachte Ziel einbezogen, wodurch *Subnetzmasken variabler Länge* möglich sind. Mit Subnetzmasken variabler Länge kann ein IP-Netzwerk in viele Subnetze verschiedenen Umfangs aufgeteilt werden. Dies gibt Netzwerk-Administratoren zusätzliche Flexibilität bei der Konfiguration des Netzwerks.

KAPITEL 41

Resource Reservation Protocol (RSVP)

41.1 Hintergrund

Beim Resource Reservation Protocol (RSVP) handelt es sich um ein netzwerksteuerndes Protokoll, das es Internet-Anwendungen ermöglicht, besondere Dienstqualitäten (QOS) für ihren Datenstrom zugewiesen zu bekommen. RSVP ist kein Routing-Protokoll, sondern arbeitet mit Routing-Protokollen zusammen und installiert so etwas wie dynamische Zugriffslisten entlang der von Routing-Protokollen berechneten Routen. RSVP nimmt den Platz eines Transport-Protokolls im OSI-Modell mit den sieben Protokollschichten ein. RSVP wurde ursprünglich von Forschern der University of South California (USC) Information Sciences Institute (ISI) und Xerox Palo Alto Research Center erdacht. Die Internet Engineering Task Force (IETF) arbeitet mittlerweile an der Standardisierung durch eine RSVP-Arbeitsgruppe. In diesem Kapitel werden folgende Themen zum Betrieb von RSVP besprochen: *Datenstrom*, *Dienstqualität*, *Hochfahren einer Sitzung*, *Reservierungsmethode* und *Soft-State-Implementierung*. Im Bild 41.1 ist eine RSVP-Umgebung dargestellt.

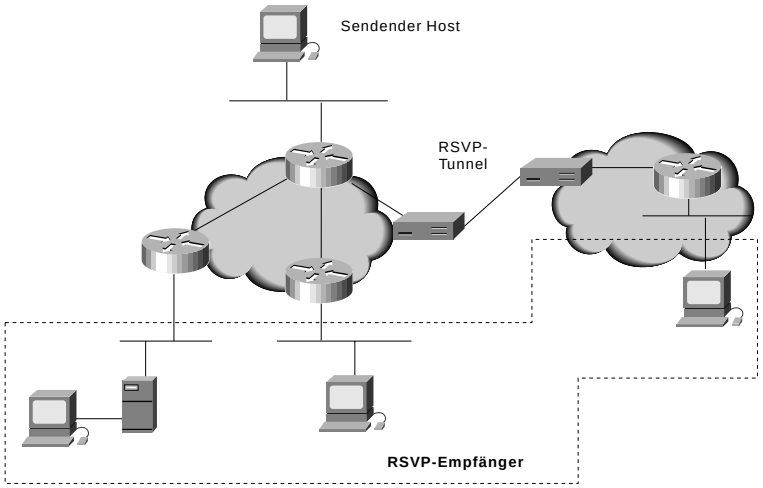


Bild 41.1: Beim RSVP werden Informationen vom Host den Empfängern über Datenströme zugestellt

41.2 RSVP – Datenströme

Beim RSVP ist ein *Datenstrom* (Data Flow) eine Abfolge von Nachrichten, die über dieselbe Quelle, dasselbe Ziel (eins oder mehrere) und dieselbe Dienstqualität verfügen. Die Anforderungen an die Dienstqualität werden durch das Netzwerk mittels einer *Datenstrom-Spezifikation* mitgeteilt, bei der es sich um eine von Hosts des Internetzwerks für das Anfordern besonderer Dienste vom Internetzwerk verwendete Datenstruktur handelt. Eine Datenstrom-Spezifikation garantiert häufig eine Umgangsmethode des Internetzwerks mit Teilen seines Host-Verkehrs.

RSVP unterstützt drei Typen von Datenverkehr: *best-effort* (so gut es geht), *rate-sensitive* (der Übertragungsrate angepaßt) und *delay-sensitive* (der Verzögerungszeit angepaßt). Die Art des zur Unterstützung dieser Typen von Datenverkehr verwendeten Datenstromdienstes hängt von der implementierten Dienstqualität ab. In den folgenden Absätzen werden diese Arten von Datenverkehr sowie die zugehörigen Dienste behandelt. Weitere Informationen hinsichtlich der Dienstqualität finden Sie im entsprechenden Abschnitt, der später in diesem Kapitel folgt.

Der best-effort Verkehr ist der übliche IP-Verkehr. Zu den Anwendungen gehören der Dateitransfer wie beispielsweise

Mail-Übertragungen, das Mounten von Festplatten, interaktive Anmeldungen und der Verkehr aufgrund von Transaktionen. Der den best-effort Verkehr unterstützende Dienst wird als *best-effort Service* bezeichnet.

Der rate-sensitive Verkehr ist dazu bereit, für eine garantierte Übertragungsrate auf bessere Zeiten zu verzichten. Beispielsweise kann für den rate-sensitive Verkehr eine Bandbreite von 100 Kbps gefordert sein. Wenn der Verkehr nun tatsächlich für längere Zeit mit 200 Kbps erfolgt, kann ein Router den Verkehr verzögern. Der rate-sensitive Verkehr ist nicht für Switched-Circuit-Netzwerke gedacht, obgleich er in der Regel mit einer Anwendung verbunden ist, die von einem Switched-Circuit-Netzwerk (wie beispielsweise ISDN) portiert wurde und in einem Datagramm-Netzwerk läuft (IP).

Ein Beispiel für eine solche Anwendung ist das H.323-Video-Conferencing, das entworfen wurde, um unter ISDN (H.320) oder ATM (H.310) zu laufen, aber im Internet zu finden ist. Die H.323-Kodierung erfolgt mit einer konstanten oder nahezu konstanten Rate und erfordert eine konstante Übertragungsrate. Der den rate-sensitive Verkehr unterstützende RSVP-Dienst wird als *guaranteed bit-rate Service* bezeichnet.

Beim delay-sensitive Verkehr handelt es sich um Datenverkehr, für den eine rechtzeitige Zustellung erforderlich ist und der seine Übertragungsrate entsprechend anpaßt. MPEG-II-Video z.B. benötigt in Abhängigkeit von den Änderungen im Bild eine Bandbreite von durchschnittlich 3 bis 7 Mbps. 3 Mbps könnten beispielsweise für ein Bild einer gestrichenen Wand ausreichen, während 7 Mbps für ein Bild von Wellen auf dem Meer erforderlich wären. Quellen für MPEG-II-Video versenden sogenannte Key- und Delta-Frames. Typischerweise beschreiben ein oder zwei Key-Frames pro Sekunde das gesamte Bild und 13 bzw. 28 Delta-Frames die Abweichungen vom Key-Frame. Delta-Frames fallen in der Regel erheblich kleiner aus als Key-Frames. Als eine Folge davon variieren die Übertragungsraten von Frame zu Frame ein wenig. Ein einzelner Frame muß jedoch innerhalb der Frame-Zeit zugestellt werden, ansonsten kann der CODEC seine Aufgabe nicht erfüllen. Für den Datenverkehr mit Delta-Frames muß eine bestimmte Priorität ausgehandelt werden. Der den delay-sensi-

tive Verkehr unterstützende RSVP-Dienst wird als *controlled-delay Service* (für Dienste in Nicht-Echtzeit) oder *predictive Service* (für Echtzeitdienste) bezeichnet.

41.2.1 RSVP – Bearbeitung von Datenströmen

RSVP-Datenströme sind üblicherweise durch *Sitzungen* gekennzeichnet, über die Datenpakete fließen. Eine Sitzung ist eine Gruppe von Datenströmen mit demselben Unicast- oder Multicast-Ziel; und RSVP behandelt jede Sitzung unabhängig voneinander. RSVP unterstützt sowohl Unicast- als auch Multicast-Sitzungen (wobei eine Sitzung aus einer Anzahl von Sendern besteht, die mit einer Anzahl von Empfängern kommunizieren), während ein Datenstrom immer von einem einzigen Sender stammt. Datenpakete in einer bestimmten Sitzung werden an dieselbe IP-Zieladresse oder eine allgemeine Zielschnittstelle geleitet. Die IP-Zieladresse kann die Gruppenadresse einer Multicast-Zustellung oder die Unicast-Adresse eines einzelnen Empfängers sein. Eine allgemeine Zielschnittstelle kann durch ein UDP/TCP-Feld für die Zielschnittstelle, ein entsprechendes Feld eines anderen Transport-Protokolls oder anwendungsspezifische Informationen definiert werden.

Die Datenverteilung erfolgt unter RSVP entweder durch Multicasts oder Unicasts. Beim Multicast-Verkehr wird auf eine Kopie jedes von einem Sender an mehrere Ziele weitergeleiteten Datenpakets zurückgegriffen. Zum Unicast-Verkehr kommt es in einer Sitzung mit einem einzigen Empfänger. Auch wenn es sich um eine Unicast-Zieladresse handelt, kann es mehrere Empfänger geben, die von einer allgemeinen Schnittstelle unterschieden werden. Außerdem kann es auch mehrere Sender für ein Unicast-Ziel geben; in diesem Fall kann RSVP Reservierungen für eine Mehrpunkt-zu-Punkt-Übertragung vornehmen.

Jeder RSVP-Sender und -Empfänger kann einem eindeutigen Internet-Host entsprechen. Ein Host kann allerdings aus mehreren logischen Sendern und Empfängern bestehen, die von allgemeinen Schnittstellen unterschieden werden.

41.3 RSVP – Dienstqualität

Unter RSVP handelt es sich bei der Dienstqualität (*Quality Of Service* = QOS) um ein in der Datenstrom-Spezifikation angegebenes Attribut, das verwendet wird, um die Art und Weise zu bestimmen, in welcher der Datenaustausch von den beteiligten Einheiten (Router, Empfänger und Sender) behandelt wird. RSVP wird eingesetzt, um die Dienstqualität sowohl des Host als auch des Router anzugeben. Hosts verwenden RSVP, um für den Datenstrom einer Anwendung einen Grad der Dienstqualität vom Netzwerk anzufordern. Router verwenden RSVP, um Anfragen nach einem Grad der Dienstqualität an andere Router entlang des Pfads oder der Pfade des Datenstroms weiterzuleiten. Dabei hält RSVP den Zustand des Hosts und des Routers aufrecht, um den angeforderten Dienst bereitzustellen.

41.4 RSVP – Hochfahren der Sitzung

Um eine RSVP-Multicast-Sitzung einzuleiten, stellt ein Empfänger mittels des Internet Group Membership Protocol (IGMP) zuerst die durch eine IP-Zieladresse angegebene Multicast-Gruppe zusammen. Im Falle einer Unicast-Sitzung übernimmt das Unicast-Routing die Rolle, die IGMP gemeinsam mit dem Protocol Independent Multicast (PIM) beim Multicasting spielt. Nachdem der Empfänger eine Gruppe zusammengestellt hat, beginnt ein potentieller Sender mit dem Senden von RSVP-Pfadnachrichten an die IP-Zieladresse. Die empfangende Anwendung erhält eine Pfadnachricht und beginnt mit dem Senden einer entsprechenden Reservation-Request-Nachricht, welche die gewünschten Datenstrom-Deskriptoren mittels RSVP angibt. Nachdem die sendende Anwendung eine Reservation-Request-Nachricht erhalten hat, beginnt der Sender mit dem Verschicken von Datenpaketen.

41.5 RSVP – Reservierungsmethode

Die Reservierungsmethode bezieht sich auf eine Menge von Kontrolloptionen, die einige unterstützte Parameter angeben. RSVP unterstützt zwei hauptsächliche Klassen von Reservierungen: *getrennte Reservierungen* (distinct) und *gemeinsame*

Reservierungen (shared). Getrennte Reservierungen richten in jeder Sitzung einen Datenstrom für jeden relevanten Sender ein. Eine gemeinsame Reservierung wird von einer Gruppe von Sendern benutzt, von denen bekannt ist, daß sie sich nicht gegenseitig stören. Im Bild 41.2 sind die Arten der getrennten und gemeinsamen Reservierungsmethoden unter RSVP im Kontext ihres Geltungsbereichs dargestellt. Jede unterstützte Kombination aus Reservierungsmethode und Geltungsbereich wird entsprechend der Darstellung beschrieben.

Bild 41.2: RSVP unterstützt sowohl getrennte als auch gemeinsame Reservierungen

Geltungsbereich	Reservierungen	
	getrennt	gemeinsam
Explizit	Fixed-Filter-Methode (FF-Methode)	Shared-Explicit-Methode (SE-Methode)
Wildcard	nicht definiert	Wildcard-Filter-Methode (WF-Methode)

41.5.1 Wildcard-Filter-Methode (WF)

Die *Wildcard-Filter*-Methode (WF) beschreibt eine gemeinsame Reservierung mit einem beliebigen Geltungsbereich. Bei einer Reservierung mit der WF-Methode wird eine einzige Reservierung erzeugt, in der Datenströme von allen auf dem Pfad liegenden Sendern zusammenkommen. Reservierungen lassen sich als eine gemeinsame Röhre vorstellen, deren Umfang unabhängig von der Anzahl der Sender der größten von allen Empfängern für diese Verbindung angeforderten Ressource entspricht. Die Reservierung verbreitet sich stromaufwärts zu allen sendenden Hosts und wird automatisch auf neue Sender ausgedehnt, sobald diese erscheinen.

41.5.2 Fixed-Filter-Methode (FF)

Die *Fixed-Filter*-Methode (FF) beschreibt eine getrennte Reservierung mit einem expliziten Geltungsbereich. Bei einer Reservierung mit der FF-Methode wird ein eigenständiger Reservation-Request für Datenpakete von einem bestimmten Sender erzeugt. Der Geltungsbereich der Reservierung ist durch eine explizite Auflistung von Sendern festgelegt. Die gesamte

Reservierung für eine gegebene Sitzung auf einer Verbindung entspricht der Summe der FF-Reservierungen für alle angefragten Sender. FF-Reservierungen, die zwar von unterschiedlichen Empfängern angefordert wurden, aber den gleichen Sender auswählen, müssen allerdings zusammengefaßt werden, um sich eine einzige Reservierung in einem gegebenen Knoten zu teilen.

41.5.3 Shared-Explicit-Methode (SE)

Die *Shared-Explicit*-Methode (SE) beschreibt eine gemeinsame Reservierung mit einem expliziten Geltungsbereich. Die SE-Methode erzeugt eine einzige Reservierung, in der Datenströme von allen auf dem Pfad liegenden Sendern zusammenkommen. Wie bei der FF-Reservierung werden die Sender (und damit der Geltungsbereich) explizit angegeben, indem der Empfänger die Reservierung vornimmt.

41.5.4 Folgen der Reservierungsmethoden

Sowohl bei der WF- als auch der SE-Methode handelt es sich um gemeinsame Reservierungen, die für Multicast-Anwendungen geeignet sind, in denen anwendungsspezifische Einschränkungen es unwahrscheinlich machen, daß mehrere Datenquellen gleichzeitig übertragen. Ein Beispiel hierfür ist das Audio-Conferencing, bei dem eine begrenzte Anzahl von Leuten gleichzeitig spricht. Jeder Empfänger kann einen WF- oder SE-Reservation-Request zweimal für einen Audio-Kanal absetzen (um ein Übersprechen zu ermöglichen). Die FF-Methode erzeugt unabhängige Reservierungen für die Datenströme verschiedener Sender. Die FF-Methode ist besser für Videosignale geeignet. Leider ist es nicht möglich, gemeinsame mit getrennten Reservierungen zusammenzufassen.

41.6 RSVP – Soft-State-Implementierung

Im Zusammenhang mit einem RSVP bezieht sich ein *Soft-State* (etwa: weicher Zustand) auf einen Zustand in Routern und Endknoten, der sich durch bestimmte RSVP-Nachrichten aktualisieren läßt. Die Soft-State-Eigenschaft ermöglicht es einem Netzwerk, dynamische Änderungen von Gruppenzugehörig-

keiten zu unterstützen und an Änderungen beim Routing anzupassen. Allgemein gesprochen wird der Soft-State von einem auf RSVP basierendem Netzwerk eingesetzt, um dem Netzwerk Änderungen des Zustands zu ermöglichen, ohne Endpunkte zu konsultieren. Dies steht im Gegensatz zu einer Switched-Circuit-Architektur, in der ein Endpunkt einen Aufruf tätigt und im Falle eines Fehlschlags einen neuen Aufruf tätigt.

Die Protokollmechanismen von RSVP stellen ein allgemeines Mittel für die Erstellung und Pflege eines verteilten Reservierungszustands über ein Netz von Multicast- und Unicast-Zustellungspfaden bereit.

Für die Wartung eines Reservierungszustands verfolgt RSVP einen Soft-State in Router- und Host-Knoten. Der Soft-State wird durch Pfad- und Reservation-Request-Nachrichten erzeugt und regelmäßig aufgefrischt. Der Zustand wird gelöscht, wenn vor Ablauf eines Zeitintervalls bis zum Aufräumen keine passenden Refresh-Nachrichten ankommen. Der Soft-State kann außerdem als Folge einer expliziten Abbau-Nachricht (*Teardown*) gelöscht werden. RSVP überprüft den Soft-State regelmäßig, um Refresh-Nachrichten für Pfad- und Reservation-Request-Nachrichten zu erstellen und an nachfolgende Hops weiterzuleiten.

Wenn sich eine Route ändert, initialisiert die nächste Pfadnachricht den Zustand des Pfads auf der neuen Route. Spätere Reservation-Requests richten einen Reservierungszustand ein. Der Soft-State im jetzt ungenutzten Segment läuft aus. (Die RSVP-Spezifikation fordert die Aufnahme von neuen Reservierungen durch das Netzwerk zwei Sekunden nach einer Änderung der Topologie.)

Wenn es zu Zustandsänderungen kommt, verbreitet RSVP diese Änderungen ohne Verzögerung vom einen zum anderen Ende eines RSVP-Netzwerks. Wenn sich der empfangene vom gespeicherten Zustand unterschiedet, wird der gespeicherte Zustand aktualisiert. Wenn als Resultat Änderungen an den zu generierenden Refresh-Nachrichten auftreten, werden sofort Refresh-Nachrichten generiert und weitergeleitet.

41.7 RSVP – Modell des Ablaufs

Unter RSVP sind Ressourcen für einfache Datenströme (d.h. unidirektionale Datenströme) reserviert. Jeder Sender ist logisch vom Empfänger getrennt, aber jede Anwendung kann als Sender und Empfänger agieren. Empfänger sind für die Requests nach Reservierungen von Ressourcen zuständig. Im Bild 41.3 ist diese allgemeine Betriebsumgebung dargestellt, während der nachfolgende Abschnitt eine Übersicht über die genaue Abfolge von Ereignissen gibt.

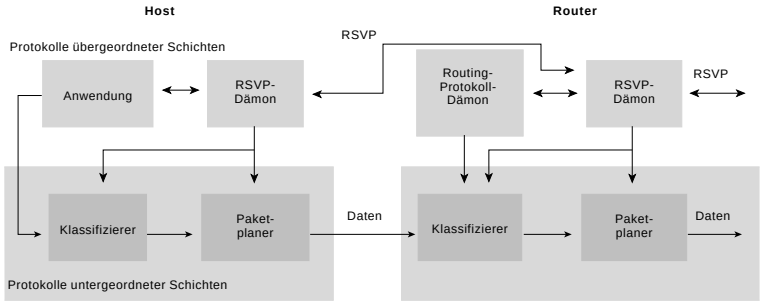


Bild 41.3: Die Betriebsumgebung von RSVP reserviert Ressourcen für unidirektionale Datenströme

41.7.1 Allgemeiner Protokollablauf von RSVP

Der Vorgang der Ressourcenreservierung unter RSVP beginnt, sobald ein RSVP-Dämon auf lokale Routing-Protokolle zurückgreift, um Routen zu erhalten. Ein Host verschickt IGMP-Nachrichten, um eine Multicast-Gruppe zusammenzustellen, und RSVP-Nachrichten, um Ressourcen entlang des Pfads bzw. der Pfade für die Zustellung von dieser Gruppe zu reservieren. Jeder Router, der sich an der Ressourcenreservierung beteiligen kann, übergibt ankommende Datenpakete an einen Paketklassifizierer und reiht sie anschließend nach Bedarf in einen Paketplaner (*packet scheduler*) ein. Der RSVP-Paketklassifizierer legt die Route und die Klasse der Dienstqualität für jedes Paket fest. Der RSVP-Planer weist auf dem von jeder Schnittstelle verwendeten Medium der Verbindungsschicht Ressourcen für die Übertragung zu. Wenn das Medium der Verbindungsschicht die Dienstqualitäten selber verwalten kann, ist der Paketplaner für die Aushandlung mit der Verbindungsschicht zuständig, um die von RSVP geforderte Dienstqualität zu erhalten.

Der Planer selbst teilt auf einem Medium mit passiver Dienstqualität, wie beispielsweise einer gepachteten Leitung, Kapazitäten für die Paketübertragung zu und kann weiterhin andere Systemressourcen wie beispielsweise CPU-Zeiten oder Zwischenspeicher zuweisen. Ein Request nach einer Dienstqualität, der typischerweise von einer Anwendung auf einem Empfänger-Host stammt, wird an die lokale RSVP-Implementierung als ein RSVP-Dämon übergeben.

Anschließend wird RSVP verwendet, um den Request an alle Knoten (Router und Hosts) entlang des/der umgekehrten Datenpfade(s) an die Datenquelle(n) zu übergeben. In jedem Knoten wendet das RSVP-Programm eine als Zugangskontrolle (*admission control*) bezeichnete lokale Entscheidungsprozedur an, um zu bestimmen, ob er die angeforderte Dienstqualität bieten kann. Wenn die Zugangskontrolle erfolgreich absolviert wird, setzt das RSVP-Programm die Parameter des Paketklassifizierers und -planers entsprechend der gewünschten Dienstqualität. Wenn die Zugangskontrolle an einem beliebigen Knoten fehlschlägt, liefert das RSVP-Programm eine Fehlernachricht an die Anwendung zurück, die den Request ausgelöst hat.

41.7.2 RSVP – Tunneln

Es ist unmöglich, RSVP oder ein anderes neues Protokoll zu einem Zeitpunkt im gesamten Internet einzusetzen. Tatsächlich kann es passieren, daß RSVP nie überall eingesetzt wird. RSVP muß daher auch dann für einen korrekten Protokollbetrieb sorgen, wenn zwei RSVP-fähige Router über eine beliebige Ansammlung von Nicht-RSVP-Routern verbunden sind. Eine dazwischenliegende Nicht-RSVP-Ansammlung kann keine Ressourcenreservierungen vornehmen, so daß für einen Dienst keine Garantien gegeben werden können. Wenn eine derartige Ansammlung allerdings über eine ausreichende Kapazität verfügt, kann sie akzeptable und hilfreiche Echtzeitdienste bereitstellen.

Um Verbindungen von RSVP-Netzwerken durch Nicht-RSVP-Netzwerke hindurch zu unterstützen, greift RSVP auf das Tunneln zurück, das bei Nicht-RSVP-Ansammlungen automatisch auftritt. Für das Tunneln ist es erforderlich, daß RSVP- und Nicht-RSVP-Router Pfadnachrichten an die Ziel-

adresse weiterleiten, indem sie eine lokale Routing-Tabelle verwenden. Wenn eine Pfadnachricht eine Nicht-RSVP-Ansammlung durchquert, tragen die Kopien der Nachricht die IP-Adresse des letzten RSVP-fähigen Router. Request-Nachrichten für die Reservierung werden an den nächsten RSVP-fähigen Router weitergeleitet.

Zwei Argumente wurden gegen die Implementierung des Tunnelns in einer RSVP-Umgebung vorgebracht. Erstens wird RSVP eher vereinzelt als allgemein eingesetzt werden. Zweitens kann das Tunneln effektiver gestaltet werden, indem eine Andrangskontrolle (*congestion control*) in Situationen implementiert wird, von denen bekannt ist, daß sie in hohem Maße gefordert sind.

Ein sporadischer oder vereinzelter Einsatz bedeutet, daß einige Teile des Netzwerks RSVP vor anderen aktiv implementiert haben. Wenn RSVP von einem Ende zum anderen erforderlich ist, gibt es keine Vorzüge ohne einen nahezu universellen Einsatz, der aber unwahrscheinlich ist, solange ein frühzeitiger Einsatz keine wesentlichen Vorteile aufzeigt.

Eine Lösung: Weighted Fair-Queuing

Das Vorhandensein einer Technik zum Erzwingen einer effektiven Ressourcenreservierung (wie beispielsweise das Weighted Fair-Queuing Schema von Cisco) an einer als Engpaß wirkenden Stelle kann sich positiv auswirken. Das Tunneln stellt nur dann eine Gefahr dar, wenn der Engpaß innerhalb einer Nicht-RSVP-Domain liegt und sich nicht vermeiden läßt. Im Bild 41.4 ist eine RSVP-Umgebung mit einem Tunnel zwischen Netzwerken dargestellt, die auf RSVP basieren.

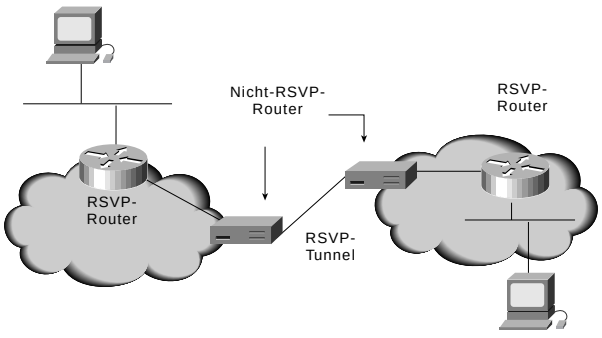


Bild 41.4: Eine RSVP-Umgebung kann einen Tunnel zwischen Netzwerken aufweisen, die auf RSVP basieren

41.8 RSVP – Nachrichten

RSVP unterstützt vier grundlegende Nachrichtentypen: *Reservation-Request-Nachrichten, Pfadnachrichten, Fehler-* und *Acknowledgment-Nachrichten* sowie *Abbaunachrichten*. Diese werden in den folgenden Abschnitten kurz beschrieben.

41.8.1 Reservation-Request-Nachrichten

Eine Reservation-Request-Nachricht wird von jedem empfangenden Host an die Sender geschickt. Diese Nachricht folgt der von den Datenpaketen verwendeten Route in umgekehrter Richtung bis zu den sendenden Hosts. Ein Reservation-Request muß den sendenden Hosts zugestellt werden, damit sie für den ersten Hop geeignete Parameter zur Kontrolle des Datenverkehrs setzen können. RSVP verschickt keine positiven Acknowledgment-Nachrichten (Bestätigung).

41.8.2 Pfadnachrichten

Eine Pfadnachricht (*Path*) wird von jedem Sender auf denjenigen Unicast- oder Multicast-Routen verschickt, die von einem oder mehreren Routing-Protokollen bereitgestellt werden. Eine Pfad-Nachricht wird zum Speichern des Pfadzustands in jedem Knoten verwendet. Der Pfadzustand wird verwendet, um Reservation-Requests in der umgekehrten Richtung weiterzuleiten.

41.8.3 Fehler- und Acknowledgment-Nachrichten

Es gibt drei Arten von Fehler- und Acknowledgment-Nachrichten: *Pfadfehlernachrichten, Reservation-Request-Fehlernachrichten* und *Reservation-Request-Acknowledgment-Nachrichten*.

Pfadfehlernachrichten resultieren aus Pfadnachrichten und bewegen sich auf Sender zu. Pfadfehlernachrichten werden unter Rückgriff auf den Pfadzustand Hop für Hop weitergeleitet. Bei jedem Hop stellt die IP-Zieladresse die Unicast-Adresse des vorangegangenen Hop dar.

Reservation-Request-Fehlernachrichten resultieren aus Reservation-Request-Nachrichten und bewegen sich auf den Emp-

fänger zu. Reservation-Request-Fehlernachrichten werden unter Rückgriff auf den Reservierungszustand Hop für Hop weitergeleitet. Bei jedem Hop stellt die IP-Zieladresse die Unicast-Adresse des nächsten Hop-Knoten dar. In Fehlernachrichten können folgende Informationen enthalten sein:

- Admission failure (Zugang nicht geglückt)
- Bandwidth unavailable (Bandbreite ist nicht verfügbar)
- Service not supported (Dienst wird nicht unterstützt)
- Bad flow specification (Unzulässige Datenstrom-Spezifikation)
- Ambiguous path (Pfad nicht eindeutig)

Reservation-Request-Acknowledgment-Nachrichten werden beim Auftreten eines Reservation-Confirmation-Objekts in einer Reservation-Request-Nachricht versendet. Diese Acknowledgment-Nachricht enthält eine Kopie der Reservierungsbestätigung (*reservation confirmation*). Eine Acknowledgment-Nachricht wird an die Unicast-Adresse eines empfangenden Host geschickt, und die Adresse wird dem Reservation-Confirmation-Objekt entnommen. Eine Reservation-Request-Acknowledgment-Nachricht wird Hop für Hop an den Empfänger weitergeleitet (um den Mechanismus zum Überprüfen der Hop-für-Hop-Vollständigkeit unterzubringen).

41.8.4 Abbaunachrichten

RSVP-Abbaunachrichten (*Teardown*) entfernen den Pfad- und Reservierungszustand, ohne die Zeitspanne bis zum Aufräumen abzuwarten. Abbaunachrichten können von einer Anwendung auf einem Endsystem (Sender oder Empfänger) oder einem Router als Resultat des Ablaufens eines Zustands veranlaßt werden. RSVP unterstützt zwei Arten von Abbau-Nachrichten: *Pfadabbaunachrichten* und *Reservation-Request-Abbaunachrichten*. Pfadabbaunachrichten löschen den Pfadzustand (wodurch der Reservierungszustand gelöscht wird), wandern zu allen dem Ausgangspunkt nachfolgenden Empfängern und werden wie Pfadnachrichten weitergeleitet. Reservation-Request-Abbaunachrichten löschen den Reservierungszustand, wandern zu allen vor dem Ausgangspunkt

liegenden passenden Sendern und werden wie entsprechende Reservation-Request-Nachrichten weitergeleitet.

41.9 RSVP – Paketformat

Im Bild 41.5 ist das Paketformat für RSVP dargestellt. In den nachfolgenden Zusammenfassungen wird eine Übersicht über die in der Abbildung dargestellten Header- und Objektfelder gegeben.

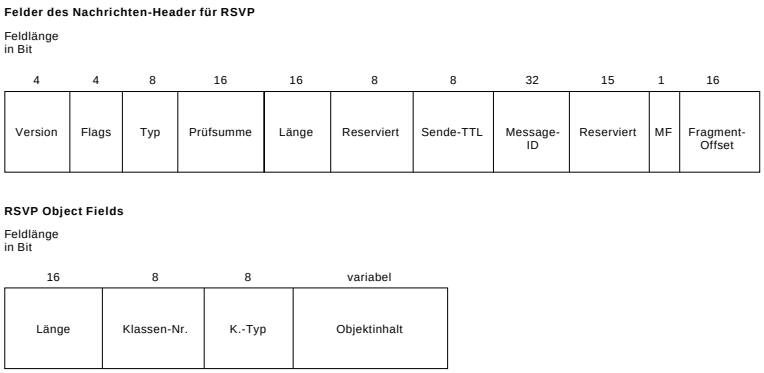


Bild 41.5: Ein Paketformat für RSVP besteht aus Nachrichten-Headern und Objektfeldern

41.9.1 Felder des Nachrichten-Headers für RSVP

Der Nachrichten-Header für RSVP umfaßt folgende Felder:

- *Version* – 4-Bit; gibt die Versionsnummer des Protokolls an (derzeit Version 1).
- *Flags* – 4-Bit; derzeit keine Flags definiert.
- *Typ* – 8-Bit; sechs mögliche (ganzzahlige) Werte, wie in der Tabelle 41.1 dargestellt.

Tabelle 41.1: Werte des Felds für den Nachrichtentyp

Wert	**Nachrichtentyp**
1	Pfad
2	Reservation-Request
3	Pfadfehler
4	Reservation-Request-Fehler
5	Pfadabbau
6	Reservation-Abbau
7	Reservation-Request-Acknowledgment

– *Prüfsumme* – 16-Bit; eine normale TCP/UDP-Prüfsumme für den Inhalt der RSVP-Nachricht, wobei das Feld *Checksum* durch Null ersetzt ist.

– *Länge* – 16-Bit; die Länge dieses RSVP-Pakets in Byte, einschließlich des allgemeinen Header und der nachfolgenden Objekte variabler Länge. Falls das Flag *More Fragments* (MF) gesetzt ist oder das Feld *Fragment Offset* ungleich Null ist, handelt es sich hierbei um die Länge des aktuellen Fragments einer umfangreicheren Nachricht.

– *Sende-TTL* – 8-Bit; gibt den Wert für die IP-Lebensdauer (*Time-To-Live* = TTL) an, mit der die Nachricht übertragen wurde.

– *Nachrichten-ID* – 32-Bit; stellt einen Bezeichner bereit, den alle Fragmente einer Nachricht von einem gegebenen nächsten bzw. vorhergegangenen RSVP-Hop gemein haben.

– *More Fragments (MF)* – Flag; niederwertigstes Bit eines 1-Byte-Word, dessen andere sieben höherwertigen Bits reserviert sind. WF wird für alle Fragmente einer Nachricht außer dem letzten aktiviert.

– *Fragment Offset* – 24-Bit; gibt den Byte-Offset des Fragments in der Nachricht an.

41.9.2 Objektfelder für RSVP

Die Objektfelder für RSVP setzen sich folgendermaßen zusammen:

– *Länge* – 16-Bit; enthält die Gesamtlänge des Objekts in Byte (muß immer mindestens 4 und ein Vielfaches von 4 sein).

– *Klassen-Nr.* – Gibt die Objektklasse an. Jede Objektklasse trägt eine Bezeichnung. Eine RSVP-Implementierung muß die in der Tabelle 41.2 aufgeführten Klassen erkennen.

Das hochwertigste Bit von *Klassen-Nr.* legt fest, welche Handlung ein Knoten ausführen soll, wenn er die *Klassen-Nr.* eines Objekts nicht erkennt.

– *K-Typ* – Objekttyp, eindeutig innerhalb von *Klassen-Nr.* Die maximale Länge des Objektinhalts beträgt 65528 Byte.

Die Felder *Class-Num* und *C-Type* (zusammen mit dem Flag-Bit) können gemeinsam als eine 16-Bit-Zahl verwendet werden, um einen eindeutigen Typ für jedes Objekt zu definieren.

– *Objektinhalt* – Die Felder *Länge*, *Klassen-Nr.* und *K-Typ* geben die Form des Objektinhalts an. In der Tabelle 41.2 finden Sie Definitionen der Objektklassen, die als Objektinhalt vorkommen können.

Tabelle 41.2: Objektklassen für RSVP

Objektklasse	Beschreibung
Null	Enthält die *Klassen-Nr.* Null und ihr *K-Typ* wird ignoriert. Ihre Länge muß mindestens 4 sein, kann aber auch ein Vielfaches von 4 sein. Ein Null-Objekt kann überall in einer Abfolge von Objekten auftreten, und der Inhalt wird vom Empfänger ignoriert.
Session	Enthält die IP-Zieladresse und möglicherweise eine allgemeine Zielschnittstelle, um eine bestimmte Sitzung für die weiteren nachfolgenden Objekte zu definieren (in jeder RSVP-Nachricht erforderlich).
RSVP Hop	Trägt die IP-Adresse des RSVP-fähigen Knotens, der diese Nachricht verschickt hat.
Time Values	Enthält, sofern vorhanden, Werte für die Zeitspanne bis zum Aktualisieren und die Lebensdauer (TTL) des Zustands, mit denen die Vorgabewerte überschrieben werden.
Style	Definiert die Reservierungsmethode und methodenspezifischen Informationen, bei denen es sich nicht um eines der Objekte *Flow Specification* oder *Filter Specification* handelt (in einer Reservation-Request-Nachricht enthalten).
Flow Specification	Definiert eine gewünschte Dienstqualität (in einer Reservation-Request-Nachricht enthalten).
Filter Specification	Definiert eine Untermenge von Datenpaketen für eine Sitzung, welche die gewünschte Dienstqualität erhalten sollen (von einem *Flow Specification*-Objekt innerhalb einer Reservation-Request-Nachricht angegeben).

Tabelle 41.2: Objektklassen für RSVP (Fortsetzung)

Objektklasse	Beschreibung
Sender Template	Enthält die IP-Adresse eines Senders und eventuell einige zusätzliche Informationen zum Demultiplexen, um einen Sender zu identifizieren (in einer Pfadnachricht enthalten).
Sender TSPEC	Definiert die Eigenschaften für den Verkehr des Datenstroms eines Senders (in einer Pfadnachricht enthalten).
Adspec	Enthält Bekanntmachungsdaten (*advertising data*) in einer Pfadnachricht.
Error Specification	Gibt einen Fehler an (in einer Pfad-Fehler- oder Reservation-Request-Fehlernachricht enthalten).
Policy Data	Enthält Informationen, die es einem lokalen Sicherheitsmodul ermöglichen, zu entscheiden, ob eine verbundene Reservierung verwaltungsgemäß zulässig ist (in einer Pfad- oder Reservation-Request-Nachricht enthalten).
Integrity	Enthält kryptographische Daten, um den Ursprungsknoten zu authentifizieren und möglicherweise den Inhalt dieser Reservation-Request-Nachricht zu überprüfen.
Scope	Eine explizite Angabe des Geltungsbereichs für das Weiterleiten einer Reservation-Request-Nachricht.
Reservation Confirmation	Enthält die IP-Adresse eines Empfängers, der eine Bestätigung angefordert hat. Kann entweder bei einem Reservation-Request oder einem Reservation-Request-Acknowledgement auftreten.

KAPITEL 42

Routing Information Protocol (RIP)

42.1 Hintergrund

Beim Routing Information Protocol (RIP) handelt es sich um ein Protokoll nach dem Distance-Vector-Verfahren, das als Metrik auf den Hop-Count zurückgreift. RIP ist im globalen Internet weit verbreitet für den Einsatz beim Routing des Datenverkehrs und stellt ein *Interior-Gateway-Protokoll* (IGP) dar, das heißt, es führt das Routing innerhalb eines einzelnen autonomen Systems durch. Exterior-Gateway-Protokolle, wie beispielsweise das Border Gateway Protocol (BGP), sind für das Routing zwischen verschiedenen autonomen Systemen zuständig. Ursprünglich erschien RIP als das Xerox-Protokoll GWINFO. Eine spätere, als *routed* (»Route De« gesprochen) bezeichnete Version, wurde 1982 mit dem Berkeley Standard Distribution (BSD) Unix ausgeliefert. RIP selber entwickelte sich zu einem Internet-Routing-Protokoll, und andere Protokollsammlungen greifen auf modifizierte Fassungen von RIP zurück. Das AppleTalk Routing Table Maintenance Protocol (RTMP) und das Banyan VINES Routing Table Protocol (RTP) basieren beispielsweise beide auf der RIP-Version des Internet Protocol (IP). Die letzten Erweiterungen zu RIP finden sich in der RIP-2-Spezifikation, die mehr Informationen in RIP-Paketen zuläßt und einen einfachen Mechanismus für die Authentifizierung bereitstellt.

IP-RIP wird in zwei Dokumenten formal definiert: Request For Comments (RFC) 1058 und 1723. Die RFC 1058 (1988) beschreibt die erste Implementierung von RIP, wohingegen die

RFC 1723 (1994) die RFC 1058 aktualisiert. Die RFC 1058 ermöglicht es RIP-Nachrichten, mehr Informationen und Sicherheitsmerkmale aufzunehmen.

In diesem Kapitel werden die grundlegenden, mit RIP verbundenen Fähigkeiten und Eigenschaften beschrieben. Zu den behandelten Themen gehören der Vorgang des Routing-Update, die Routing-Metriken von RIP, die Stabilität des Routing sowie die Routing-Timer.

42.2 Routing-Updates

RIP verschickt Routing-Update-Nachrichten in regelmäßigen Abständen und wenn die Netzwerk-Topologie sich ändert. Sobald ein Router ein Routing-Update empfängt, das Änderungen für einen Eintrag enthält, aktualisiert er die neue Route in seiner Routing-Tabelle. Der Wert der Metrik für den Pfad wird um eins erhöht, und auf den Sender wird als nächster Hop verwiesen. RIP-Router halten nur die beste Route (die Route mit dem niedrigsten Metrikwert) zu einem Ziel vor. Nach der Aktualisierung der Routing-Tabelle beginnt der Router sofort damit, Routing-Updates zu verschicken, um andere Netzwerk-Router über die Änderung zu informieren. Diese Updates werden unabhängig von den regelmäßig geplanten Updates der RIP-Router verschickt.

42.3 RIP – Routing-Metrik

RIP verwendet für die Entfernungsmessung zwischen dem Quell- und einem Zielnetzwerk eine einzige Routing-Metrik (Hop-Count). Jedem Hop auf einem Pfad von der Quelle zum Ziel wird ein Hop-Count zugewiesen, der üblicherweise 1 ist. Sobald ein Router ein Routing-Update empfängt, das einen neuen oder geänderten Eintrag für das Zielnetzwerk enthält, erhöht der Router den im Update angegebenen Metrikwert um eins und trägt das Netzwerk in die Routing-Tabelle ein. Die IP-Adresse des Senders wird als nächster Hop verwendet.

RIP schützt vor sich endlos fortsetzenden Routing-Schleifen, indem ein Grenzwert für die Anzahl der Hops auf einem Pfad von der Quelle zu einem Ziel implementiert wird. Die maximale Anzahl von Hops auf einem Pfad beträgt 15. Wenn ein

Router ein Routing-Update empfängt, das einen neuen oder geänderten Eintrag enthält und wenn das Erhöhen des Metrikwerts um eins dafür sorgen würde, daß die Metrik unendlich (also 16) beträgt, dann wird das Zielnetzwerk als unerreichbar angesehen.

42.4 RIP – Stabilitätsmerkmale

Für eine schnelle Anpassung an Änderungen in der Netzwerk-Topologie bringt RIP eine Reihe von Stabilitätsmerkmalen mit, die es mit vielen Routing-Protokollen gemein hat. Beispielsweise implementiert RIP die *Split-Horizon-* und *Hold-Down*-Mechanismen, um das Weiterleiten falscher Routing-Informationen zu vermeiden. Weiterhin bewahrt der Grenzwert für den Hop-Count in RIP vor sich endlos fortsetzenden Routing-Schleifen.

42.5 RIP – Timer

RIP verwendet zur Ausführungsregulierung eine Vielzahl von Timern. Hierzu zählen ein *Routing-Update-Timer*, ein *Route-Timeout* und ein *Route-Flush-Timer*. Der Routing-Update-Timer bemißt das Zeitintervall zwischen den regelmäßigen Routing-Updates. Üblicherweise ist er auf 30 Sekunden gestellt, wobei jedesmal eine geringe, zufällig bestimmte Anzahl von Sekunden hinzugezählt wird, um Kollisionen zu vermeiden. Jeder Eintrag in der Routing-Tabelle verfügt über einen mit ihm verbundenen Timer für das Route-Timeout. Wenn die Zeit des Route-Timeout abgelaufen ist, wird die Route als ungültig markiert, aber noch so lange in der Tabelle aufbewahrt, bis der Route-Flush-Timer abläuft.

42.6 Paketformate

Die folgenden Abschnitte konzentrieren sich auf die Paketformate von IP RIP und IP RIP 2, die im Bild 42.1 und Bild 42.2 dargestellt sind. Jeder Abbildung folgt eine kurze Beschreibung der dargestellten Felder.

42.6.1 Paketformat von RIP

Im Bild 42.1 ist das IP-RIP-Paketformat dargestellt.

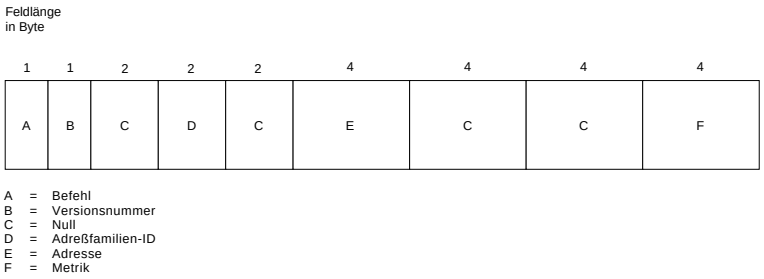


Bild 42.1: Ein IP-RIP-Paket setzt sich aus neun Feldern zusammen

Die Felder des im Bild 42.1 dargestellten Paketformats für IP RIP werden im folgenden kurz beschrieben:

- *Befehl* – Zeigt an, ob es sich bei dem Paket um ein Request (Anforderung) oder um ein Response (Antwort) handelt. Ein Request fordert dazu auf, daß der Router seine ganze Routing-Tabelle oder einen Teil davon senden soll. Bei einem Response kann es sich um ein unaufgefordertes regelmäßiges Routing-Update oder um eine Reaktion auf einen Request handeln. Responses enthalten Einträge für Routing-Tabellen. Für den Transport umfangreicher Routing-Tabellen werden mehrere RIP-Pakete verwendet.
- *Versionnummer*– Gibt die verwendete RIP-Version an. Dieses Feld kann auf verschiedene, möglicherweise inkompatible Versionen hinweisen.
- *Null* – Nicht verwendet.
- *Adreßfamilien-ID (AFI)* – Gibt die verwendete Adreßfamilie an. RIP wurde entworfen, um Routing-Informationen für mehrere unterschiedliche Protokolle zu übertragen. Jeder Eintrag verfügt über eine ID für die Adreßfamilie, der die Art der angegebenen Adresse kennzeichnet. Die AFI für IP ist 2.
- *Adresse* – Gibt die IP-Adresse des Eintrags an.
- *Metrik* – Zeigt an, wie viele Hops (Router) des Internetzwerks auf dem Weg zum Ziel durchquert wurden. Dieser Wert liegt für eine gültige Route zwischen 1 und 15, oder 16 für eine unerreichbare Route.

Die Felder *AFI*, *Adresse* und *Metrik* dürfen bis zu 25 Mal in einem einzigen IP-RIP-Paket vorkommen. (Bis zu 25 Ziele können in einem einzigen RIP-Paket aufgeführt werden.)

42.6.2 Paketformat von RIP 2

Die RIP-2-Spezifikation (in der RFC 1723 beschrieben) ermöglicht weitere Informationen in RIP-Paketen und bietet einen einfachen Mechanismus für die Authentifizierung. Im Bild 42.2 ist das Paketformat von IP RIP 2 dargestellt.

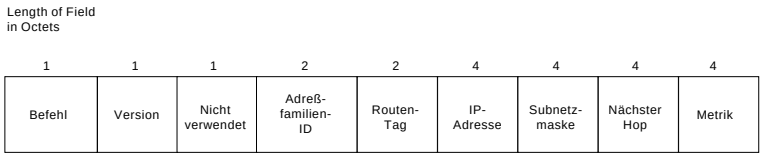


Bild 42.2: Ein IP-RIP-2-Paket setzt sich aus ähnlichen Feldern zusammen wie ein IP-RIP-Paket

Die Felder des im Bild 42.2 dargestellten Paketformats für IP RIP 2 werden im folgenden kurz beschrieben:

- *Befehl* – Zeigt an, ob es sich bei dem Paket um ein Request (Anforderung) oder um ein Response (Antwort) handelt. Ein Request fordert dazu auf, daß der Router seine ganze Routing-Tabelle oder einen Teil davon senden soll. Bei einem Response kann es sich um ein unaufgefordertes regelmäßiges Routing-Update oder um eine Reaktion auf einen Request handeln. Responses enthalten Einträge für Routing-Tabellen. Für den Transport umfangreicher Routing-Tabellen werden mehrere RIP-Pakete verwendet.
- *Version* – Gibt die verwendete RIP-Version an. Bei der Implementierung eines beliebigen RIP-2-Felds in einem Paket für RIP oder dem Einsatz der Authentifizierung wird dieses Feld auf 2 gesetzt.
- *Nicht verwendet* – Wert ist auf Null gesetzt.
- *Adreßfamilien-ID (AFI)* – Gibt die verwendete Adreßfamilie an. RIP wurde entworfen, um Routing-Informationen für mehrere unterschiedliche Protokolle zu übertragen. Jeder Eintrag verfügt über eine ID für die Adreßfamilie, der die Art der angegebenen Adresse kennzeichnet. Die AFI für IP ist 2. Falls die AFI für den ersten Eintrag der Nachricht

0xFFFF ist, enthält der Rest des Eintrags Authentifizierungsinformationen. Derzeit stellt ein einfaches Kennwort den einzigen Authentifizierungstyp dar.

- *Routen-Tag* – Stellt eine Methode zum Unterscheiden von internen Routen (von RIP erfahren) und externen Routen (von anderen Protokollen erfahren) bereit.
- *IP-Adresse* – Gibt die IP-Adresse des Eintrags an.
- *Subnetzmaske* – Enthält die Subnetzmaske für den Eintrag. Falls dieses Feld auf Null gesetzt ist, wurde für diesen Eintrag keine Subnetzmaske angegeben.
- *Nächster Hop* – Gibt die IP-Adresse des nächsten Hop an, an den Pakete für den Eintrag weitergeleitet werden sollen.
- *Metrik* – Zeigt an, wie viele Hops (Router) des Internetzwerks auf dem Weg zum Ziel durchquert wurden. Dieser Wert liegt für eine zulässige Route zwischen 1 und 15, oder 16 für eine unerreichbare Route.

Die Felder *AFI*, *Adresse* und *Metrik* dürfen bis zu 25 Mal in einem einzigen IP-RIP-Paket vorkommen. Somit können bis zu 25 Einträge für Routing-Tabellen in einem einzigen RIP-Paket aufgeführt werden. Falls die *AFI* eine authentifizierte Nachricht kennzeichnet, können nur 24 Einträge für Routing-Tabellen angegeben werden.

KAPITEL 43

Simple Multicast Routing Protocol (SMRP)

43.1 Hintergrund

Beim Simple Multicast Routing Protocol (SMRP) handelt es sich um ein Protokoll für die Transportschicht, das für das Weiterleiten von Multimedia-Datenströmen über AppleTalk-Netzwerke entwickelt wurde. Es unterstützt das QuickTime Conferencing (QTC) von Apple. SMRP sorgt für eine verbindungslose und möglichst gute Zustellung von Multicast-Datagrammen und stützt sich für Dienste auf die zugrundeliegenden Protokolle für die Netzwerkschicht. Insbesondere erleichtert SMRP die Übertragung von Daten von einer einzigen Quelle zu mehreren Zielen. Dieses Kapitel konzentriert sich auf die funktionalen Elemente und die Protokollausführung von SMRP. Im Bild 43.1 ist eine verallgemeinerte SMRP-Umgebung dargestellt.

Bei der Schaffung von SMRP übernahm Apple einige Strategien und Konzepte von anderen Protokollen und Techniken. Dadurch erhielten viele Begriffe in Apples SMRP-Umgebung eine eigene Bedeutung. In der Tabelle 43.1 sind die SMRP-spezifischen Begriffe und deren Definitionen zusammengefaßt. In diesem Kapitel wird auf diese Begriffe zurückgegriffen.

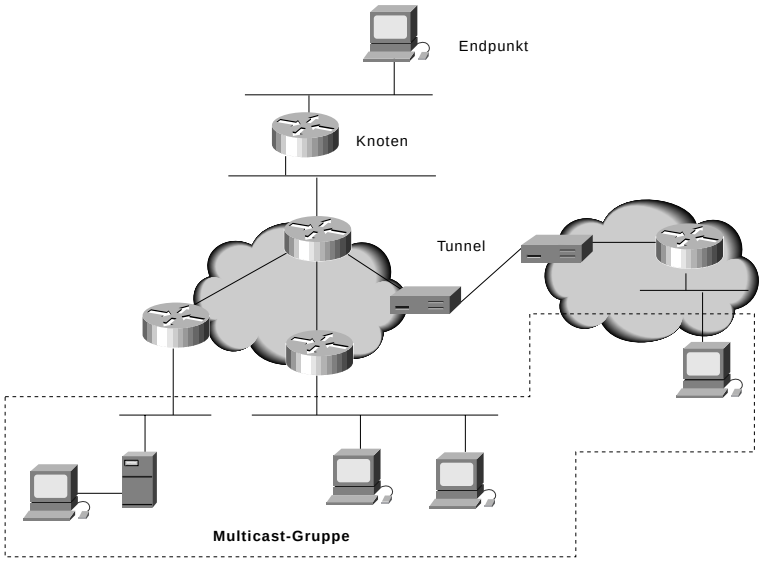


Bild 43.1: Eine allgemeine SMRP-Umgebung erstreckt sich von einer Multicast-Gruppe bis zu einem Endpunkt

43.2 SMRP – Multicast-Transportdienste

SMRP wurde entworfen, um Routern und Rechnern an Endpunkten den Austausch von Multicast-Paketen über Protokolle der Netzwerkschicht zu ermöglichen. SMRP kann die Zuordnungen von Multicast-Adressen verwalten und macht es einer einzelnen Quelle möglich, an eine eindeutige Multicast-Gruppenadresse gerichtete Daten zu verschicken. Empfänger schließen sich dieser Gruppe an, falls sie am Empfang von Daten für diese Gruppe interessiert sind. Zur Unterstützung dieser Funktionen greift SMRP auf einige Dienste zurück. Die folgenden Beschreibungen konzentrieren sich auf die wesentlichen Vorgänge und Techniken, welche die SMRP-Dienste ermöglichen. Zu diesen zählen: die Adreßverwaltung, das Multicast Transaction Protocol (MTP), die Knotenverwaltung, die Verwaltung von Multicast-Routen, das Weiterleiten von Daten sowie die Topologieverwaltung.

Begriff	Definition
Direkt benachbarter Endpunkt	In bezug auf einen Knoten bzw. Endpunkt: ein Endpunkt im gleichen lokalen Netzwerk bzw. ein durch einen Tunnel angebundener Knoten.
Direkt benachbarter Knoten	In bezug auf einen Knoten oder Endpunkt: ein Knoten im gleichen lokalen Netzwerk.
Tochter-Endpunkt	Ein direkt benachbarter Endpunkt, an den ein Knoten Multicast-Daten schickt.
Tochterknoten	In bezug auf einen Mitgliederknoten einer Gruppe: ein weiter vom Erzeuger-Endpunkt entfernter Nachbarknoten.
Tochteranschluß	In bezug auf eine Gruppe: ein Anschluß, der die Schnittstelle zu einem oder mehreren Tochterknoten darstellt.
Erzeuger-Endpunkt	Der Endpunkt, der die Erzeugung der Gruppe gefordert hat, und die Quelle der Daten, die an die Gruppe weitergeleitet werden.
Erzeugerknoten	Der Primärknoten, der die Gruppe erzeugt hat.
Designierter Knoten	Ein SMRP-Router, der zum Primär- oder Sekundärknoten ernannt worden ist.
Zielbaum	Der in einem lokalen Netzwerk wurzelnde umfassende Baum mit auf das lokale Netzwerk ausgerichteten Pfaden.
Verteilungsbaum	Die aktive Untermenge eines Quellbaums, die für das Weiterleiten von Multicast-Daten für eine Gruppe verwendet wird.
Endpunkt	Eine nicht weiterleitende Quelle oder ein nicht weiterleitendes Ziel von Multicast-Paketen.
Gruppe	Eine Menge von empfangenden Endpunkten oder eine Multicast-Adresse.
Zusammenschluß	Begriff für den Vorgang, Mitglied einer Gruppe zu werden.
Anschließender Pfad	Ein Pfad im Zielbaum für ein lokales Netzwerk, der zum Erreichen eines Erzeugerknotens verwendet wird und der mittels des Distance-Vector-Algorithmus von SMRP erzeugt wurde.
Verlassen	Begriff für den Vorgang, die Mitgliedschaft in einer Gruppe aufzugeben.
Lokales Netz	Eine Datenverbindung mit gemeinsamen Zugriff und das zugehörige Protokoll für die Netzwerkschicht. Ein LAN kann mehr als ein lokales Netz unterstützen.
Mitglieder-Endpunkt	Ein Endpunkt, der Mitglied einer Gruppe ist.

Tabelle 43.1: SMRP-spezifische Begriffe und deren Definition

Tabelle 43.1: SMRP-spezifische Begriffe und deren Definition (Fortsetzung)

Begriff	Definition
Mitgliederknoten	Ein Knoten, der sich im Verteilungsbaum einer Gruppe befindet.
Nachbarknoten	In bezug auf einen Mitgliederknoten einer Gruppe: ein direkt benachbarter Knoten, der sich im Verteilungsbaum für die Gruppe befindet.
Knoten	Ein SMRP implementierender Router.
Vateranschluß	In bezug auf eine Gruppe: der Anschluß, der die Schnittstelle zum Vaterknoten darstellt.
Vaterknoten	In bezug auf einen Mitgliederknoten einer Gruppe: der näher am Erzeuger-Endpunkt befindliche Nachbarknoten.
Anschluß	Eine Schnittstelle eines SMRP-Routers zu einem lokalen Netzwerk oder einem Tunnel.
Urheberanschluß	Die Adresse des Knotens, der für die Behandlung von Gruppenanfragen verantwortlich ist.
Primärknoten	Der in einem Lokalen Netzwerk für das Erstellen von Gruppen zuständige Knoten.
Umgekehrter Pfad	Die Umkehrung eines anschließenden Pfads, ein Pfad im Quellbaum für ein lokales Netz, der zum Weiterleiten von Multicast-Daten verwendet wird.
Sekundärknoten	Derjenige Knoten, der zur Übernahme der Aufgaben eines verschwindenden Primärknotens bereit ist.
Quellbaum	Der in einem lokalen Netzwerk wurzelnde umfassende Baum mit vom lokalen Netzwerk weisenden Pfaden.
Umfassender Baum	Eine verbundene Menge von Pfaden, die lokale Netzwerke zwischen allen Knoten eines Internetzwerks mit nur einem Pfad zwischen jeweils zwei Knoten verwendet.
Tunnel	Eine Punkt-zu-Punkt-Verbindung zwischen Knoten nicht direkt benachbarter Netzwerke durch Router, die kein SMRP implementieren.

43.2.1 SMRP – Verwaltung von Multicast-Adressen

Die Adressierung von SMRP basiert auf dem lokalen Netzwerk eines Erzeuger-Endpunkts. Eine SMRP-Adresse besteht aus zwei Teilen: einer Netzwerk-Nummer mit einer Länge von 3 Byte und einer Socketnummer mit einer Länge von 1 Byte.

Jedes lokale Netzwerk wird mit einem Bereich eindeutiger Netzwerk-Nummern konfiguriert.

Bei der Zuordnung von Netzwerk-Nummern müssen die Netzwerk-Nummern lokalen Netzen für SMRP zugewiesen werden und im gesamten Internetzwerk eindeutig sein. Jedem lokalen Netz kann ein beliebiger fortlaufender Bereich von 3-Byte Netzwerk-Nummern zugewiesen werden. Die Anzahl der für ein lokales Netz verfügbaren Multicast-Gruppen entspricht dem 254-fachen der Anzahl zugewiesener Netzwerk-Nummern. Netzwerk-Nummern können konfiguriert oder aufgrund der Netzwerk-Nummer zugrundeliegender Netzwerk-Protokolle zugeordnet werden. Bereiche eindeutiger Netzwerk-Nummern können für unterstützte Netzwerk-Protokolle reserviert werden.

Im Falle der Zuordnung von Multicast-Adressen müssen Multicast-Adressen von SMRP auf Multicast-Adressen der Netzwerk-Schicht abgebildet werden, denen wiederum Multicast-Adressen der Verbindungsschicht zugeordnet sind. Für jede Art von Netzwerk-Schicht muß für SMRP ein Block mit Multicast-Adressen vorgehalten werden. Günstigstenfalls erfolgt eine direkte Zuordnung der Adressen. Meistens ist eine direkte Zuordnung aber nicht möglich, und mehrere Multicast-Adressen von SMRP werden auf eine einzige Multicast-Adresse der Netzwerk-Schicht abgebildet.

Die Art und Weise, in der Multicast-Adressen auf Adressen der Netzwerk-Schicht abgebildet werden, hängt von der Netzwerk-Schicht ab. Wenn Multicast-Adressen der SMRP-Transportschicht sich nicht direkt auf Multicast-Adressen der Netzwerk-Schicht abbilden lassen, ist eine Filterung der Multicast-Adressen von SMRP erforderlich. Wenn Multicast-Adressen der Netzwerk-Schicht sich nicht direkt auf Multicast-Adressen der Verbindungsschicht abbilden lassen, wird von der Netzwerk-Schicht erwartet, daß sie nicht unterstützte Multicast-Adressen ausfiltert.

Es gibt folgende Voreinstellungen für Multicast-Adressen der Netzwerk-Schicht: *AllEndpoints, AllNodes* und *AllEntities*. AllEndpoints-Nachrichten, die an diese Multicast-Adresse geschickt werden, werden an alle Endpunkte in einem Netzwerk übertragen. AllNodes-Nachrichten, die an diese Multicast-Adresse geschickt werden, werden an alle SMRP-Routingkno-

ten in einem Netzwerk übertragen. Und AllEntities-Nachrichten, die an diese Multicast-Adresse geschickt werden, werden an alle Endpunkte und an alle SMRP-Routingknoten in einem Netzwerk übertragen.

43.2.2 SMRP – Multicast-Transaktionsprotokoll (MTP)

SMRP bezieht ein Multicast-Transaktionsprotokoll (MTP) mit ein, das drei Transaktionstypen vorsieht: Knoten, Endpunkt und gleichzeitig Knoten/Endpunkt. Die Kommunikation zwischen direkt benachbarten Knoten und zwischen Knoten und Endpunkten erfolgt mittels Request/Response-Transaktionen.

Responses erfolgen immer als Unicast. MTP sorgt beim Auftreten von Netzwerkfehlern für eine erneute Übertragung von Requests und/oder Responses. Nur Hello- und Designated-Node-Request-Pakete werden als Multicast-Nachrichten verschickt, alle übrigen als Unicasts. Requests von Endpunkten zu Knoten werden als Multicasts verschickt, während Requests von Knoten zu Endpunkten entweder als Unicasts oder als Multicasts verschickt werden.

Der grundlegende MTP-Entwurf, wie er auf SMRP-Routern implementiert ist, setzt zwei Warteschlangen für alle Transaktionen ein: eine Request-Warteschlange und eine Response-Warteschlange. Die Einträge der Request-Warteschlange werden gelöscht, sobald der Router das erhaltene Response bearbeitet hat. Das Response wird unter Zuhilfenahme einer im Eintrag angegebenen Rücksendeadresse (*callback*) bearbeitet, sofern es zu einem Request paßt.

Nach der Bearbeitung des Response wird das Request gelöscht. Wenn das Request unbeantwortet bleibt, wird ein intern generiertes Reject-Response an die Rücksendeadresse geschickt. Requests können in Abhängigkeit vom Kontext an eine Unicast-Adresse oder an die Multicast-Adressen AllNodes oder AllEndpoints geschickt werden. Solange sie nicht explizit umadressiert werden, werden Requests an die Multicast-Adresse AllNodes geschickt.

Die Einträge der Response-Warteschlange werden bei Empfang eines Request-Pakets erzeugt. Auf den Eintrag wird während der gesamten Bearbeitung des Request verwiesen, und der bearbeitete Eintrag bleibt so lange in der Warteschlange,

bis er abläuft und aus der Warteschlange gelöscht wird. Falls ein vervielfältigtes Request empfangen wird, wird es ignoriert, wenn der SMRP-Router das ursprüngliche Request noch bearbeitet oder wenn ein vervielfältigtes Response nach Beendigung der Bearbeitung generiert wird. Für einige empfangene Requests ist es erforderlich, daß ein SMRP-Routing-Knoten weitere Requests generiert. In diesem Fall wird das bzw. werden die ursprüngliche(n) Request(s) von der Bearbeitungsprozedur für die Rücksendung bearbeitet, die zum Request-Eintrag des Routing-Knotens gehört.

43.2.3 SMRP – Knotenverwaltung

SMRP baut für die Transporterlaubnis von Multicast-Datagrammen auf einige Beziehungen zwischen Knoten auf, zu denen *designierte Knoten*, *direkt benachbarte Knoten* und *Tunnelknoten* gehören.

Bei designierten Knoten handelt es sich um SMRP-Router, die als Primär- oder Sekundärknoten angegeben sind. Ein designierter Primärknoten ist für die Zuteilung von Gruppenadressen zuständig. Ein Primärknoten ist für jedes lokale Netzwerk mit SMRP-Knoten erforderlich. Ein designierter Sekundärknoten ist erforderlich, falls ein lokales Netzwerk über mehr als einen Knoten verfügt. Der Sekundärknoten wird für die Pflege einer Kopie der Gruppenerzeugungstabelle (*Group Creation Table*) verwendet und wird zum Primärknoten, falls der Primärknoten für ein Netzwerk ausfällt.

Der wesentliche Vorgang beim Bestimmen des Primär- und Sekundärknotens beginnt beim Hochfahren, wenn ein Knoten zuerst versucht, zum designierten Sekundärknoten in jedem lokalen Netz zu werden. Bei Erfolg versucht der Knoten anschließend, zum designierten Primärknoten zu werden. Transaktionen werden entweder von einem Primärknoten-Request oder einem Sekundärknoten-Request eingeleitet. Das Fehlen einer Antwort auf den Request zeigt an, daß die Aushandlung erfolgreich war, wohingegen eine positive Antwort anzeigt, daß die Aushandlung fehlschlug. Wenn zwei Knoten gleichzeitig versuchen, zum designierten Primärknoten oder zum designierten Sekundärknoten zu werden, wird der Knoten mit der niedrigeren Unicast-Adresse für die Netzwerk-Schicht zum designierten Knoten. Ein Primärknoten verschickt anschlie-

ßend Add-Group-Entry-Pakete und Remove-Group-Entry-Pakete an den Sekundärknoten für ein lokales Netzwerk, um für eine identische Gruppenerzeugungstabelle zu sorgen.

In bezug auf einen bestimmten Knoten oder Endpunkt existiert ein direkt benachbarter Knoten im gleichen lokalen Netzwerk. Die Knoten verschicken über jeden Anschluß regelmäßig Hello-Pakete. Wenn von einem direkt benachbarten Knoten innerhalb einer bestimmten Zeitspanne kein Hello-Paket empfangen wird, wird der Status für den direkten Nachbarn im Knoten auf »Außer Betrieb« gesetzt, und die zugehörigen Routen werden als unerreichbar markiert. Notify-Pakete werden an jeden direkt benachbarten Knoten geschickt, sobald ein Anschlußzustand des Knotens in einen anderen Betriebszustand übergeht. Jeder Knoten führt für jeden direkt benachbarten Knoten einen Eintrag in der Knotentabelle. Der Tabelleneintrag wird erstellt, wenn zum ersten Mal ein Paket vom direkt benachbarten Knoten empfangen wird. Tabelleneinträge enthalten den Zeitpunkt des letzten Hello-Pakets und dessen Zustand.

Bei Tunnelknoten handelt es sich um Punkt-zu-Punkt-Verbindungen zwischen Knoten in nicht direkt benachbarten Netzwerken durch Router, die kein SMRP implementieren. Es gibt zwei verschiedene Arten von Tunnelknoten: Tunnel zwischen Knoten und Tunnel zwischen einem Knoten und einem Endpunkt.

Tunnelknoten werden hinsichtlich der Verwendung von Hello-Paketen und Notify-Paketen von jedem Knoten auf dieselbe Weise als Einträge in der Tabelle für direkt benachbarte Knoten geführt wie andere benachbarte Knoten. Entsprechend ermöglicht es SMRP Tunnelknoten, sich auf die gleiche Weise wie jeder andere direkt benachbarte Knoten Gruppen anzuschließen oder diese zu verlassen.

Cisco unterstützt keine Tunnelknoten. SMRP kann jedoch in die Lage versetzt werden, Tunnel in der Netzwerk-Schicht zwischen nicht direkt benachbarten Knoten zu betreiben.

43.2.4 SMRP – Multicast-Routen

SMRP stützt sich für die Ermittlung von Routen für Multicast-Verkehr auf ein Weiterleitungsschema, das auf einem umfassenden Baum basiert. Dieses Verfahren der Routenermittlung beruht auf dem Einsatz eines Distance-Vector-Algorithmus. Ein Knoten verschickt beim Hochfahren und bei Routenänderungen Distance-Vector-Request-Pakete an direkt benachbarte Knoten. Die durch den Vektor angegebene Distanz entspricht der zum Erreichen eines bestimmten Bereichs von Netzwerk-Nummern benötigten Anzahl von Hops. Knoten enthalten einen Vektor für jeden Eintrag in der Tabelle für Netzwerk-Routen und verschicken so viele Pakete, wie für den Versand aller Vektoren erforderlich sind. Bei Routenänderungen verschickt jeder Knoten Distance-Vector-Request-Pakete an jeden direkt benachbarten Knoten.

Wenn eine Route an einem Anschluß empfangen wird, muß die Adresse des Urheberanschlusses für die Route für alle Anschlüsse gesetzt werden. Da die Gruppenadresse an die Netzwerk-Adresse gebunden ist, wird die Adresse des Urheberanschlusses auch dann verwendet, wenn ein Knoten ein Request für bestimmte Gruppen bearbeiten muß. Wenn es sich bei der Adresse des Urheberanschlusses um die Knotenadresse selber handelt, ist der Knoten für das Request zuständig. Bei Knoten mit demselben Pfad wird der für ein Request zuständige Knoten darüber bestimmt, welcher der Knoten über die höchste Netzwerk-Adresse verfügt.

Wenn ein Knoten ein Distance-Vector-Request mit Einträgen für unbekannte lokale Netzwerke empfängt, werden in der Netzwerk-Routentabelle für den Knoten Netzwerk-Bereiche für damit verbundene lokale Netzwerke hinzugefügt, wobei die empfangene Distanz um eins erhöht wird. Der direkt benachbarte Knoten, der das Distance-Vector-Paket geschickt hat, wird daraufhin zum Vaterknoten für das lokale Netzwerk. Ein Tabelleneintrag wird aktualisiert, wenn ein Distance-Vector-Paket für bekannte lokale Netzwerke empfangen wird und die Distanz plus eins niedriger ausfällt als der Eintrag in der Netzwerk-Routentabelle. Ein Tie-Break gelangt zum Einsatz, wenn ein Distance-Vector-Paket von einem direkt benachbarten Knoten mit derselben Distanz zu einem lokalen Netzwerk empfangen wird. Der Tie-Break wird als der direkt

benachbarte Knoten mit einer höheren Unicast-Adresse der Netzwerk-Schicht festgelegt. Dieser Knoten wird als der Vaterknoten für das lokale Netzwerk ausgewiesen.

43.2.5 SMRP – Verwaltung von Multicast-Gruppen

Die Teilnahme an einer Multicast-Gruppe wird unter SMRP durch ein Verfahren geregelt, das Verhandlungen zwischen Endpunkten und Knoten miteinbezieht. Ein Endpunkt versucht sich einer Gruppe anzuschließen, indem er einen Knoten in einem lokalen Netzwerk kontaktiert. Ein beliebig angesprochener Knoten ist für das Anschließen des Verteilungsbaums für die Gruppe zuständig, indem er Pfade zu einem vorhandenen Verteilungsbaum aktiviert. Knoten verlassen einen Verteilungsbaum für eine Gruppe, indem sie Pfade deaktivieren, sobald keine Mitglieder-Endpunkte für die Gruppe auf diesen Pfaden mehr vorhanden sind. Für die Verwaltung von SMRP-Gruppen sind vier grundlegende Vorgänge erforderlich: das Erzeugen (*create*), Anschließen (*join*), Verlassen (*leave*) und Löschen (*delete*).

Ein Endpunkt schickt ein Create-Group-Request an den designierten Primärknoten, sobald er mit dem Versenden von Daten an eine Gruppe beginnen will. Der Primärknoten weist daraufhin eine unbenutzte Gruppenadresse zu und erstellt einen Eintrag in der Gruppenerzeugungstabelle. Schließlich liefert der Primärknoten die Gruppenadresse an den Erzeuger-Endpunkt und schickt ein Add-Group-Request an den Sekundärknoten, sofern dieser vorhanden ist.

Endpunkte verschicken Requests, um den Anschluß an eine Multicast-Gruppe einzuleiten. Der Vaterknoten für eine Gruppe in einem lokalen Netzwerk reagiert auf Join-Group-Request-Pakete von Endpunkten. (Ein Knoten stellt fest, ob er der Vaterknoten ist, indem er die Netzwerk-Nummer in der Gruppenadresse untersucht.) Wenn der Vaterknoten für eine Gruppe ein Join-Group-Request erhält und der Knoten noch kein Mitglied der Gruppe ist, leitet der Knoten den Join-Group-Request an den Erzeugerknoten der Gruppe, weiter. Schließlich erreicht das Join-Group-Request-Paket einen Mitgliederknoten oder den Erzeugerknoten der Gruppe und ein Join-Group-Confirm-Paket wird über den umgekehrten Pfad zurückgeschickt. Der Mitglieder- oder Erzeugerknoten

fügt einen Tochteranschluß in der Gruppenweiterleitungstabelle hinzu, falls das Join-Group-Request an diesem Anschluß empfangen wurde. Sobald Daten auf dem umgekehrten Pfad ankommen, werden sie an alle Tochteranschlüsse weitergeleitet. Wenn der Erzeugerknoten das erste Join-Request für eine Gruppe erhält, leitet er das Request an den Erzeuger-Endpunkt weiter, damit dieser mit dem Verschicken von Daten beginnen kann.

Zum Verlassen einer Multicast-Gruppe verschicken Endpunkte Leave-Group-Request-Pakete für ihr lokales Netz. Der Vaterknoten der Gruppe in einem lokalen Netzwerk liefert ein Leave-Group-Confirm-Paket an den Endpunkt zurück und verschickt ein Group-Member-Request-Paket für den Tochteranschluß. Falls der Vaterknoten kein Group-Member-Confirm-Paket für den Tochteranschluß von einem Mitgliederknoten oder Endpunkt erhält, entfernt der Vaterknoten diesen Anschluß aus dem Eintrag. Falls keine Tochteranschlüsse mehr im Eintrag des Vaterknotens vorhanden sind, setzt er den Status des Eintrags auf »Verlassen« und schickt ein Leave-Group-Request-Paket im Verteilungsbaum an seinen Vaterknoten. Jeder betroffene Vaterknoten entfernt bei Erhalt des Leave-Group-Confirm-Pakets den Eintrag aus seiner Gruppenweiterleitungstabelle.

Der Endpunkt verschickt ein Delete-Group-Request, sobald er mit dem Senden von Daten an die Gruppe aufhören will. Nur der designierte Primärknoten antwortet auf dieses Request.

43.2.6 Weiterleiten von Multicast-Datagrammen

Das Weiterleiten von Daten unter SMRP betrifft Knoten, die Multicast-Datagramme über aktive Pfade des Quellbaums für eine bestimmte Gruppe weiterleiten. Ein aktiver Pfad verfügt über Mitglieder-Endpunkte für die Gruppe, oder es handelt sich um einen Pfad, der als Durchgangspfad benötigt wird, um andere aktive Pfade zu erreichen. Die Teilmenge der aktiven Pfade für den Quellbaum stellt den Verteilungsbaum für die Gruppe dar. Zum Weiterleiten von Daten unter SMRP gehören eine Reihe von Verhandlungen zwischen Endpunkten und Knoten. Im allgemeinen empfangen Knoten Multicast-Datagramme, wenn Endpunkte Daten an eine Gruppe schicken. Der Erzeuger-Endpunkt kann Datenpakete mit einer

Multicast-Adresse für die Netzwerk-Schicht an sein lokales Netzwerk schicken, sobald er ein Join-Request vom Erzeugerknoten empfangen hat. Vaterknoten im lokalen Netzwerk erhalten diesen Multicast und leiten das Paket an alle Tochteranschlüsse in der Weiterleitungstabelle der Gruppe weiter. Ein Knoten verschickt ein Paket nur dann als Multicast für ein lokales Netzwerk, wenn es sich bei ihm um den Vaterknoten für die Gruppe dieses lokalen Netzwerks handelt und die Daten auf dem Vateranschluß für die Gruppe empfangen wurden. Knoten leiten Daten außerdem an direkt benachbarte Tunnelknoten weiter, die Mitglieder der Gruppe sind. Im Falle eines SMRP-Tunnels werden Multicast-Datagramme in einem Unicast-Paket der Netzwerk-Schicht gekapselt.

43.2.7 Umgang mit Topologieänderungen unter SMRP

SMRP-Einheiten legen Topologiekarten an, um Änderungen von Pfaden oder Mitgliedschaften in einer SMRP-Umgebung zu verwalten. SMRP sind einige typische Topologieänderungen bekannt und definiert spezifische Techniken für deren Behandlung.

Verschwundene Mitglieder-Endpunkte

Um verschwundene Mitglieder-Endpunkte zu entdecken, schicken Knoten regelmäßig ein Group-Member-Request-Paket an jeden aktiven Tochteranschluß. Jeder Mitgliederknoten und Endpunkt liefert ein Group-Member-Confirmation-Paket an den Vaterknoten zurück. Wenn der Vaterknoten keine Group-Member-Confirmation-Pakete erhält, schickt der Knoten ein Leave-Group-Request-Paket an seinen Vaterknoten und löscht anschließend den Gruppeneintrag.

Verlassene Gruppen

Um verlassene Gruppen aufzuspüren, schicken Erzeugerknoten regelmäßig ein Group-Creator-Request-Paket an den Erzeuger-Endpunkt. Wenn der Erzeugerknoten nach einigen Versuchen keine Group-Creator-Confirm-Pakete empfängt, wird die Gruppe gelöscht. Netzwerk-Routentabellen werden auf dem neuesten Stand gehalten, indem Knoten bei Routenänderungen Distance-Vector-Pakete an ihre direkt benachbarten Knoten schicken. Damit wird es Knoten ermöglicht, das Rou-

ting von Multicast-Gruppen aufgrund von Änderungen in der Topologie zu ändern.

43.3 SMRP – Beispiel für eine Transaktion

Zu einer typischen Transaktionssitzung unter SMRP gehört ein die Gruppe erzeugender Macintosh-Arbeitsplatz, weitere sich der Gruppe anschließende Macintosh-Arbeitsplätze sowie Daten, die an die Gruppenmitglieder verschickt werden.

In einer typischen SMRP-Transaktionssitzung verschickt ein Macintosch (nennen wir dieses System Erzeuger-Mac) zuerst ein Create-Group-Request an alle Knoten in einem bestimmten Netzwerk. Der primäre Router für das lokale Netzwerk weist eine nicht verwendete Gruppenadresse zu und liefert diese Adresse an den Erzeuger-Mac zurück. Ein Macintosh in einem entfernten Netzwerk (nennen wir ihn Mitglieder-Mac) entdeckt den Erzeuger-Mac über das *Name Binding Protocol* (NBP).

Der Erzeuger-Mac antwortet dann mittels eines NBP-Response mit der Gruppenadresse. Der Mitglieder-Mac schickt ein Join-Group-Request an alle Knoten. Ein entfernter Router (nennen wir ihn Router M) mit einer gültigen Route zu der Gruppe und einem richtigen Urheberanschluß schickt ein Join-Group-Request an den primären Router.

Der primäre Router empfängt schließlich das Join-Group-Request und schickt es an den Erzeuger-Mac. Er fügt der Gruppe in der Weiterleitungstabelle außerdem den ankommenden Anschluß hinzu. Der Erzeuger-Mac bestätigt das Join-Group-Request und schickt Daten an die Gruppe. Der primäre Router erhält die Daten und leitet sie an die Tochteranschlüsse der Gruppe weiter.

Schließlich gelangen die Daten zu Router M, der die Gruppe in der Weiterleitungstabelle nachschlägt und die Multicast-Daten weiterleitet. Der Mitglieder-Mac empfängt daraufhin für die Gruppe bestimmte Daten.

43.4 SMRP – Paketformat

Im Bild 43.2 ist das allgemeine Paketformat für SMRP dargestellt.

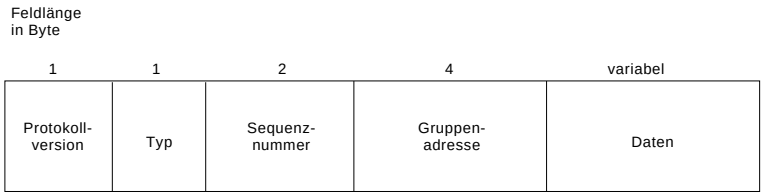


Bild 43.2: Ein allgemeines SMRP-Paket setzt sich aus fünf Feldern zusammen

Die Felder des im Bild 43.2 dargestellten Paketformats für SMRP werden im folgenden kurz beschrieben:

- *Protokoll-Version* – Zeigt die Version von SMRP an.
- *Typ* – Besteht aus zwei Teilfeldern. Die ersten 2 Bit modifizieren den von den hinteren 6 Bit angegebenen Pakettyp, um feststellen zu können, ob es sich bei einem Paket um ein Transaktionspaket handelt und gegebenenfalls, um welche Art von Transaktion es sich handelt.
- *Sequenznummer* – Ordnet in Transaktionen Response- und Request-Pakete einander zu, um doppelte Requests und Responses zu vermeiden. Alle Pakettypen sind Transaktionspakete und tragen eine Sequenznummer ungleich Null (mit Ausnahme von Multicast-Datenpaketen und Hello-Paketen, deren Sequenznummern auf Null gesetzt sind).
- *Gruppenadresse* – Dient als designierter Primärknoten und weist allen Multicast-Quellen im lokalen Netzwerk Gruppenadressen zu. Einem lokalen Netzwerk können mehrere Netzwerk-Nummern zugewiesen werden, die aber in einem fortlaufenden Bereich liegen müssen. Knoten müssen Netzwerk-Nummern derart konfigurieren, daß sie für jedes lokale Netzwerk und jeden Primärknoten eindeutig sind, um Konflikte mit Multicast-Adressen zu verhindern. Wenn ein Primärknoten eine neue Gruppenadresse zuweist, weist er der Netzwerk-Nummer willkürlich eine unbenutzte Gruppenadresse zu.

– *Daten* – Variiert in Abhängigkeit vom SMRP-Pakettyp. In Tabelle 43.2 sind die Eigenschaften der auf den Pakettypen basierenden Daten zusammengefaßt.

Pakettyp	**Übertragene Daten**	**Größe**
Multicast-Data	Daten	Variabel, abhängig von der Datagrammgröße der Netzwerk-Schicht
Hello	Anschlußzustand	2 Byte
Notify	Anschlußzustand	1 Byte
Designated Node	Keine	0 Byte
Distance-Vector	Multicast-Vektor	8 Byte
Create Group	Keine	0 Byte
Delete Group	Keine	0 Byte
Join Group	Keine	0 Byte
Add Group Entry	Unicast-Adresse einer Netzwerk-Schicht	Variabel, abhängig vom Adreßformat der Netzwerk-Schicht
Remove Group Entry	Keine	0 Byte
Leave Group	Keine	0 Byte
Creator Request	Keine	0 Byte
Member Request	Keine	0 Byte
Reject	Fehlerkennzeichnung	Short Integer im Bereich von –7700 bis –7710; abhängig vom Fehler

Tabelle 43.2: Auf Pakettypen basierende Eigenschaften von Daten

Kapitel 44: IBM Netzwerkverwaltung
Kapitel 45: Remote Monitoring (RMON)
Kapitel 46: Simple Network Management Protocol (SNMP)

TEIL 7

Netzwerk-Verwaltung

Der Teil 7, »Netzwerk-Verwaltung«, umreißt die Funktionen, die von verschiedenen, weit verbreiteten Umgebungen zur Netzwerk-Verwaltung bereitgestellt werden. In jeweils eigenen Kapiteln wird auf folgende Themen eingegangen:

IBM-Netzwerk-Verwaltung – Behandelt die Grundlagen der IBM-Netzwerk-Verwaltung für SNA und APPN.

Remote Monitoring (RMON) – Behandelt die Eigenschaften und Funktionen dieser standardisierten Überwachungsspezifikation.

Simple Network Management Protocol (SNMP) – Behandelt SNMP, das weit verbreitete Internet-Protokoll zur Netzwerk-Verwaltung.

KAPITEL 44

IBM-Netzwerk-Verwaltung

44.1 Hintergrund

Die IBM-Netzwerk-Verwaltung bezieht sich auf jede Architektur, die für die Verwaltung von Netzwerken unter der IBM Systems Network Architecture (SNA) oder dem Advanced Peer-to-Peer Networking (APPN) zum Einsatz kommt. Die IBM-Netzwerk-Verwaltung ist Teil der IBM Open-Network Architecture (ONA) und wird hauptsächlich unter Einsatz von Verwaltungsplattformen wie beispielsweise NetView durchgeführt. Sie unterteilt sich in fünf Funktionen, die den im Open-System-Interconnection-Modell (OSI) angegebenen Funktionen zur Netzwerk-Verwaltung ähneln. In diesem Kapitel erfolgen kurze Beschreibungen der funktionalen Bereiche bei der IBM-Netzwerk-Verwaltung, der Netzwerk-Verwaltungsarchitektur ONA und von Verwaltungsplattformen. Im Bild 44.1 ist ein einfaches verwaltetes IBM-Netzwerk dargestellt.

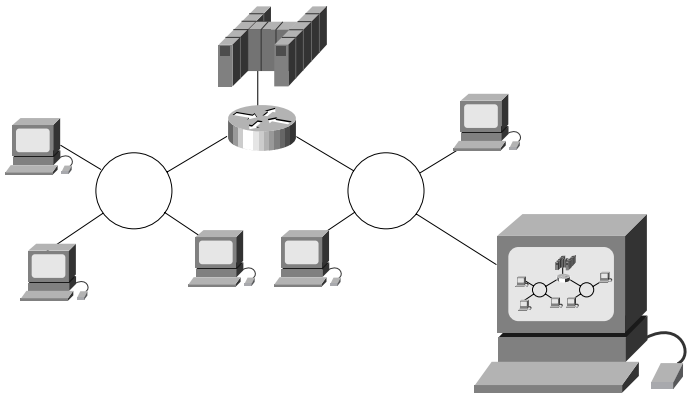


Bild 44.1: Die IBM Netzwerk-Verwaltung geht mit SNA- oder APPN-Netzwerken um

44.2 IBM – Funktionale Bereiche der Netzwerk-Verwaltung

IBM unterteilt die Netzwerk-Verwaltung in folgende fünf benutzerorientierte Aufgabenbereiche: *Konfigurationsverwaltung*, *Performance- und Accounting-Verwaltung*, *Problemverwaltung*, *Betriebsverwaltung* und *Änderungsverwaltung*.

44.2.1 IBM – Konfigurationsverwaltung

Die Konfigurationsverwaltung (*Configuration Management*) von IBM überwacht Informationen, welche die physikalischen und logischen Eigenschaften von Netzwerk-Ressourcen sowie die Beziehungen dieser Ressourcen untereinander beschreiben. Ein zentrales Verwaltungssystem speichert Daten in einer Datenbank zur Konfigurationsverwaltung und kann Informationen wie beispielsweise Versionsnummern der Systemsoftware oder des Microcode, Seriennummern von Hard- oder Software, physikalische Aufenthaltsorte von Geräten sowie Namen, Adressen und Telefonnummern für Kontakte enthalten. Wie zu erwarten entspricht die Konfigurationsverwaltung von IBM nahezu der OSI-Konfigurationsverwaltung.

Die Möglichkeiten der Konfigurationsverwaltung helfen bei der Pflege einer Bestandsliste für die Netzwerk-Ressourcen und bei der Sicherstellung, daß sich Änderungen an der Netzwerk-Konfiguration in der Datenbank für die Netzwerk-Verwaltung wiederfinden. Die Konfigurationsverwaltung stellt weiterhin Informationen bereit, die von den Systemen zur Pro-

blemverwaltung und zur Änderungsverwaltung aufgegriffen werden. Systeme zur Problemverwaltung verwenden diese Informationen, um Versionsunterschiede zu vergleichen und um die Eigenschaften von Netzwerk-Ressourcen ausfindig zu machen, zu identifizieren und zu überprüfen. Systeme zur Änderungsverwaltung verwenden diese Informationen, um die Auswirkungen von Änderungen zu analysieren und um Änderungen in Zeiten minimaler Netzwerk-Belastung einzuplanen.

44.2.2 IBM – Performance- und Accounting-Verwaltung

Die Performance- und Accounting-Verwaltung von IBM stellt Informationen über die Leistungsdaten der Netzwerk-Ressourcen bereit. Zu den Aufgaben der Performance- und Accounting-Verwaltung gehören die Überwachung der Antwortzeiten von Systemen, das Messen der Verfügbarkeit von Ressourcen, das Messen des Einsatzes von Ressourcen sowie das Optimieren, Verfolgen und Steuern der Netzwerk-Performance. Die dabei gesammelten Informationen sind hilfreich, um festzustellen, ob Ziele für die Netzwerk-Performance erreicht wurden und ob aufgrund der Leistung Verfahren zur Ermittlung von Schwachstellen eingeleitet werden sollten. Die Performance- und Accounting-Verwaltung von IBM erfüllt Aufgaben, die denen der OSI-Performance-Verwaltung und der OSI-Accounting-Verwaltung ähneln.

44.2.3 IBM – Problemverwaltung

Die Problemverwaltung (*Problem Management*) von IBM ähnelt der OSI-Fehlerverwaltung darin, daß sie Fehlerbedingungen behandelt, die für einen Benutzer zum Verlust der vollen Funktionalität einer Netzwerk-Ressource führen. Die Problemverwaltung erfolgt in fünf Schritten: Problemermittlung, Problemdiagnose, Problemumgehung und -wiederherstellung, Problemlösung sowie Problemverfolgung und -kontrolle. Die *Problemermittlung* besteht im Entdecken eines Problems und dem Beenden der notwendigen Schritte (beispielsweise das Eingrenzen des Problems auf ein bestimmtes Teilsystem), um mit der Problemdiagnose beginnen zu können. Die *Problemdiagnose* besteht darin, die genaue Ursache des Problems und die zu dessen Lösung benötigte Handlung zu ermitteln. Die

Problemumgehung und -wiederherstellung besteht aus Versuchen, das Problem entweder teilweise oder vollständig zu umgehen. Sie sorgt lediglich für eine vorübergehende Lösung und vertraut auf eine Problemlösung, die das Problem dauerhaft löst. Die *Problemlösung* besteht aus Unternehmungen, das Problem zu beseitigen. Sie beginnt üblicherweise nach der Beendigung der Problemdiagnose und umfaßt häufig korrigierende Handlungen wie beispielsweise das Ersetzen von fehlerhafter Hard- oder Software. Die *Problemverfolgung und -kontrolle* schließlich besteht aus der Verfolgung des Problems, bis eine endgültige Lösung gefunden ist. Wichtige, das Problem beschreibende Informationen werden in einer Problemdatenbank gespeichert.

44.2.4 IBM – Betriebsverwaltung

Die Betriebsverwaltung (*Operations Management*) von IBM besteht aus der Verwaltung verteilter Netzwerk-Ressourcen von einer zentralen Stelle aus, wobei zwei Arten von Funktionen eingesetzt werden: betriebsverwaltende Dienste und allgemeine Betriebsdienste. Betriebsverwaltende Dienste besitzen die Fähigkeit, entfernte Ressourcen mittels folgender Funktionen zentral zu kontrollieren: Ressourcenaktivierung und -deaktivierung, Befehlsstreichung und Zeiteinstellung. Betriebsverwaltende Dienste können automatisch als Reaktion auf bestimmte Meldungen zu Systemproblemen eingeleitet werden.

Allgemeine Betriebsdienste ermöglichen die Verwaltung von nicht explizit durch andere Verwaltungsbereiche angesprochenen Ressourcen, wobei eine besondere Kommunikation durch neue, leistungsfähigere Anwendungen verwendet wird. Die allgemeinen Betriebsdienste stellen zwei wichtige Dienste zur Verfügung: den Ausführungsbefehl und den Dienst zur Ressourcenverwaltung. Der Ausführungsbefehl stellt ein standardisiertes Mittel zum Ausführen entfernter Befehle dar. Der Dienst zur Ressourcenverwaltung sorgt für eine Möglichkeit, Informationen in einer vom Kontext unabhängigen Weise zu transportieren.

44.2.5 IBM – Änderungsverwaltung

Die Änderungsverwaltung verfolgt Netzwerk-Änderungen und pflegt Änderungsdateien auf entfernten Knoten. Netzwerk-

Änderungen treten hauptsächlich aus zwei Gründen auf: sich ändernde Anwenderbedürfnisse und das Umgehen von Problemen. Zu den sich ändernden Anwenderbedürfnissen gehören die Aktualisierung von Hard- und Software, neue Anwendungen und Dienste sowie weitere Faktoren, die ständig die Bedürfnisse von Netzwerk-Benutzern verändern. Das Umgehen von Problemen ist notwendig, um unerwartete Änderungen handhaben zu können, die aus dem Ausfall von Hardware, Software oder anderen Netzwerk-Komponenten resultieren. Die Änderungsverwaltung versucht, Probleme zu minimieren, indem sie ein geordnetes Vorgehen bei Netzwerk-Änderungen fördert und Änderungsdateien verwaltet, die Netzwerk-Änderungen protokollieren.

44.3 IBM-Architekturen zur Netzwerk-Verwaltung

Zwei der bekanntesten Architekturen zur Netzwerk-Verwaltung von IBM sind die *Open Network Architecture* (ONA) und *SystemView*.

44.3.1 Open Network Architecture (ONA)

Bei der Open Network Architecture (ONA) handelt es sich um eine allgemeine Architektur zur Netzwerk-Verwaltung, die vier wesentliche Verwaltungseinheiten definiert: Brennpunkt, Sammlungspunkt, Eintrittspunkt und Dienstpunkt.

Der Brennpunkt (*Focal Point*) ist eine Verwaltungseinheit, die den Betrieb einer zentralisierten Netzwerk-Verwaltung unterstützt. Er reagiert auf Alarmmeldungen von Endrechnern, pflegt Verwaltungsdatenbanken und stellt eine Benutzerschnittstelle für den Netzwerk-Verwalter bereit. Es gibt drei Arten von Brennpunkten: primäre, sekundäre und verschachtelte. Der primäre Brennpunkt führt sämtliche Aufgaben eines Brennpunkts durch. Der sekundäre Brennpunkt dient als Reserve für primäre Brennpunkte und wird eingesetzt, falls ein primärer Brennpunkt ausfällt. Der verschachtelte Brennpunkt sorgt in umfangreichen Netzwerken für die Unterstützung einer verteilten Verwaltung. Verschachtelte Brennpunkte sind für die Weiterleitung wesentlicher Informationen an umfassendere Brennpunkte zuständig.

Sammelpunkte (*Collection Points*) tragen Informationen von selbstenthaltenen SNA-Teilnetzwerken an Brennpunkte weiter. Üblicherweise werden sie eingesetzt, um Daten von IBM-Peer-to-Peer-Netzwerken in die ONA-Hierarchie weiterzuleiten.

Bei einem Eintrittspunkt (*Entry Point*) handelt es sich um ein SNA-Gerät, das ONA für sich selber und andere Geräte implementieren kann. Die meisten standardmäßigen SNA-Geräte sind als Eintrittspunkte geeignet.

Bei einem Dienstpunkt (*Service Point*) handelt es sich um ein System, das Nicht-SNA-Geräten den Zugriff auf ONA ermöglicht und das im wesentlichen ein Gateway zu ONA darstellt. Dienstpunkte sind dazu in der Lage, Brennpunkte mit Verwaltungsinformationen über Nicht-SNA-Geräte zu versorgen, Befehle von Brennpunkten zu empfangen, Befehle in ein für Nicht-SNA-Geräte verständliches Format zu übersetzen sowie Befehle zur Ausführung an Nicht-SNA-Geräte weiterzuleiten.

Im Bild 44.2 sind die Beziehungen der verschiedenen Verwaltungseinheiten von ONA zueinander dargestellt.

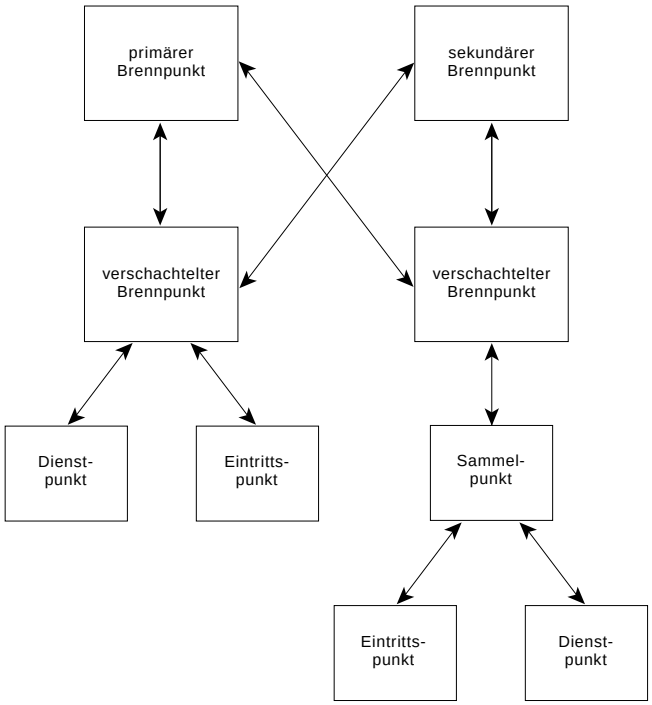


Bild 44.2: Die vier Arten von Brennpunkten sind innerhalb der ONA-Umgebung miteinander verbunden

44.3.2 SystemView

SystemView stellt eine Entwurfsvorlage für das Erstellen von verwaltenden Anwendungen dar, die in der Lage sind, Informationssysteme für viele Anbieter zu verwalten. SystemView beschreibt, wie heterogene Netzwerke verwaltende Anwendungen mit anderen Verwaltungssystemen umgehen. Es handelt sich hierbei um die offizielle Systemverwaltungsstrategie der IBM Systems Application Architecture (SAA).

44.4 IBM – Plattformen zur Netzwerk-Verwaltung

Die Netzwerk-Verwaltung von IBM ist auf mehreren Plattformen implementiert. Zu diesen gehören: *NetView, LAN Network Manager* (LAN) und *Simple Network Management Protocol* (SNMP).

44.4.1 NetView

Bei NetView handelt es sich um eine umfassende IBM-Enterprise-Netzwerk-Verwaltungsplattform, die zentralisierte SNA Netzwerk-Verwaltungsdienste zur Verfügung stellt. Sie wird auf IBM Mainframes eingesetzt und ist ein Bestandteil der ONA. NetView setzt sich aus der *Einrichtung zur Befehlskontrolle, Hardware-Monitor, Session-Monitor, Hilfefunktion, Status-Monitor, Performance-Monitor* und *Distribution-Manager* zusammen. Die Einrichtung zur Befehlskontrolle bietet die Möglichkeit zur Netzwerk-Kontrolle, indem grundlegende Befehle für den Verwalter und den Dateizugriff an Anwendungen, Controller, Betriebssysteme und NetView/PC (eine Schnittstelle zwischen NetView und Nicht-SNA-Geräten) aufgerufen werden, die auf der Virtual Telecommunications Access Method (VTAM) aufbauen. Der Hardware-Monitor überwacht das Netzwerk und warnt den Netzwerk-Verwalter bei Auftreten eines Hardware-Fehlers automatisch. Der Session-Monitor fungiert als VTAM-Performance-Monitor und sorgt für die Ermittlung von Software-Problemen und für die Konfigurationsverwaltung. Die Hilfefunktion dient NetView-Benutzern als Hilfe und besteht aus einer Möglichkeit zum Blättern, einem Help-Desk und einer Bibliothek üblicherweise auftretender Betriebssituationen eines Netzwerks. Der

Status-Monitor faßt die Statusinformationen des Netzwerks zusammen und stellt sie dar. Der Performance-Monitor überwacht die Leistungsdaten der Frontend-Prozessoren (FEDs), des Network Control Program (NCP) und weiterer zugehöriger Ressourcen. Der Distribution-Manager plant und verfolgt die Verteilung von Daten, Software und Microcode in SNA-Umgebungen.

44.4.2 LAN Network Manager (LNM)

Beim LAN Network Manager (LNM) handelt es sich um eine IBM-Anwendung zur Netzwerk-Verwaltung, die Token-Ring-LANs von einer zentralen Stelle aus steuert. LNM ist ein Produkt, das auf der OS/2 Extended Edition basiert und mit IBM NetView (das auf solche LNM-Aktivitäten wie beispielsweise Alarmmeldungen achtet) sowie anderer Verwaltungssoftware von IBM zusammenarbeitet.

44.4.3 Simple Network Management Protocol (SNMP)

Die Netzwerk-Verwaltung von IBM kann mittels SNMP implementiert werden. In Kapitel 46, »Simple Network Management Protocol (SNMP)«, finden Sie nähere Informationen zur Implementierung von SNMP.

KAPITEL 45

Remote Monitoring (RMON)

45.1 Hintergrund

Beim *Remote Monitoring* (RMON) handelt es sich um eine allgemeine Überwachungsspezifikation, die es verschiedenartigen Netzwerk-Monitoren und Konsolensystemen ermöglicht, Daten der Netzwerk-Überwachung auszutauschen. RMON versieht Netzwerk-Administratoren mit größeren Freiheiten bei der Auswahl von Meßsonden und Konsolen für die Netzwerk-Überwachungen mit zu ihren besonderen Netzwerk-Bedürfnissen passenden Eigenschaften. Dieses Kapitel liefert einen kurzen Überblick über die RMON-Spezifikation, wobei die RMON-Gruppen im Mittelpunkt stehen.

Die RMON-Spezifikation definiert eine Reihe von statistischen Werten und Funktionen, die RMON-gemäße Konsolen-Manager und Netzwerk-Meßsonden untereinander austauschen können. Damit versorgt RMON Netzwerk-Administratoren mit umfassenden Informationen für die Fehlerdiagnose, Planung und Leistungsoptimierung von Netzwerken.

RMON wurde von der Anwenderschaft mit Unterstützung durch die Internet Engineering Task Force (IETF) definiert und wurde 1992 in Form der RFC 1271 (für Ethernet) als Standard vorgeschlagen. Daraufhin wurde RMON 1995 in Form der RFC 1757 zum Standard im Entwurfsstadium erklärt, wodurch die RFC 1272 letztendlich überflüssig wurde.

Im Bild 45.1 ist eine RMON-Meßsonde dargestellt, die dazu in der Lage ist, ein Ethernet-Segment zu überwachen und statistische Werte an eine RMON-gemäße Konsole zu übertragen.

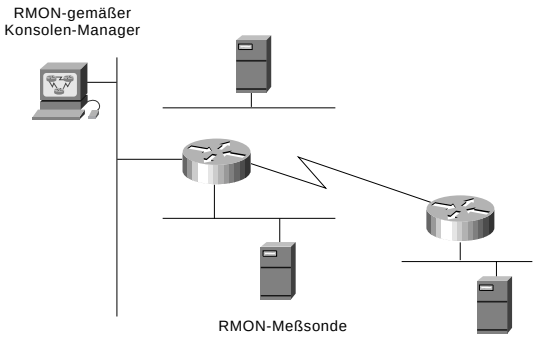


Bild 45.1: Eine RMON-Meßsonde kann statistische Werte an eine RMON-Konsole schicken

45.2 RMON-Gruppen

RMON überbringt Informationen in neun aus Überwachungselementen bestehenden RMON-Gruppen, von denen jede für bestimmte Arten von Daten sorgt, um allgemeine Bedingungen der Netzwerk-Überwachung zu erfüllen. Jede der Gruppen ist optional, so daß Anbieter nicht alle Gruppen in der *Management Information Base* (MIB) zu unterstützen brauchen. In der Tabelle 45.1 sind die neun in der RFC 1757 für Ethernet RMON MIB angegebenen Überwachungsgruppen zusammengestellt.

Tabelle 45.1: Überwachungsgruppen von RMON

RMON-Gruppe	**Funktion**	**Elemente**
Statistics	Enthält statistische Werte, die von der Meßsonde für jede überwachte Schnittstelle dieses Geräts gemessen werden.	Gelöschte Pakete, gesendete Pakete, gesendete Bytes (Oktett), Broadcast-Pakete, Multicast-Pakete, CRC-Fehler, Runts, Giants, Fragmente, Jabbers, Konflikte sowie Zähler für Pakete, die 64–128, 128–256, 256–512, 512–1024 und 1024–1518 Byte umfassen.

RMON-Gruppe	Funktion	Elemente
History	Zeichnet regelmäßige statistische Messungen in einem Netzwerk auf und speichert sie für die spätere Verwendung.	Meßzeitraum, Anzahl der Messungen, gemessene(s) Element(e).
Alarm	Entnimmt Variablen der Meßsonde regelmäßig statistische Meßwerte und vergleicht sie mit zuvor konfigurierten Schwellenwerten. Falls die überwachte Variable einen Schwellenwert übersteigt, wird ein Event generiert.	Umfaßt die Alarm-Tabelle und erfordert die Implementierung der Event-Gruppe. Alarmtyp, Intervall, Anfangsschwelle, Endschwelle.
Host	Enthält mit jedem im Netzwerk vorkommenden Host verbundene statistische Werte.	Host-Adresse, empfangene und übertragene Pakete und Bytes sowie Broadcast-, Multicast- und Fehler-Pakete.
HostTopN	Bereitet Tabellen vor, die diejenigen Hosts beschreiben, die eine nach einem ihrer statistischen Werte sortierte Auflistung anführen. Bei den statistischen Werten handelt es sich um Messungen eines ihrer grundlegenden Erfassungswerte über einen vom verwaltenden Rechner vorgegebenen Zeitraum. Somit handelt es sich um ratenorientierte statistische Werte	Statistische Werte, Host(s), Anfangs- und Endpunkt einer Messung, Ratengrundlage, Dauer.
Matrix	Speichert statistische Werte für Konversationen zwischen Paaren zweier Adressen. Sobald das Gerät eine neue Konversation feststellt, erzeugt es einen neuen Eintrag in seiner Tabelle.	Adressenpaare für Quelle und Ziel sowie Pakete, Bytes und Fehler für jedes Paar.

Tabelle 45.1: Überwachungsgruppen von RMON

Tabelle 45.1: Überwachungsgruppen von RMON

RMON-Gruppe	Funktion	Elemente
Filters	Ermöglicht das Überprüfen von Paketen gemäß einer Filtergleichung. Diese passenden Pakete bilden einen Datenstrom, der erfaßt werden kann oder Events erzeugen kann.	Bit-Filtertyp (maskiert oder nicht maskiert), Filterausdruck (Bit-Ebene), Bedingungsausdruck (und, oder, nicht) in bezug auf andere Filter.
Packet Capture	Ermöglicht das Erfassen von Paketen, nachdem sie über einen Kanal geflossen sind.	Größe des Zwischenspeichers für erfaßte Pakete, Füllstand (Alarm), Anzahl erfaßter Pakete.
Events	Steuert die Erzeugung und Mitteilung von Ereignissen dieses Geräts.	Ereignistyp, Beschreibung, Zeitpunkt der letzten Ereignisversendung.

KAPITEL 46

Simple Network Management Protocol (SNMP)

46.1 Hintergrund

Beim *Simple Network Management Protocol* (SNMP) handelt es sich um ein Protokoll für die Anwendungsschicht, das den Austausch von Verwaltungsinformationen zwischen Netzwerk-Geräten erleichtert. Es ist ein Bestandteil der Protokollsammlung Transmission Control Protocol/Internet Protocol (TCP/IP). SNMP gestattet es Netzwerk-Administratoren, die Netzwerk-Performance zu verwalten, Netzwerk-Probleme aufzuspüren und zu lösen und das Wachstum des Netzwerks zu planen.

Es gibt zwei Versionen von SNMP: SNMP Version 1 (SNMPv1) und SNMP Version 2 (SNMPv2). Beide Versionen verfügen über einige gemeinsame Eigenschaften, SNMPv2 bietet allerdings auch einige Erweiterungen wie beispielsweise zusätzliche Protokolloperationen. Die Standardisierung einer weiteren Version von SNMP – SNMP Version 3 (SNMPv3) – ist anhängig. Dieses Kapitel beschreibt die Protokolloperationen von SNMPv1 und SNMPv2. Im Bild 46.1 ist ein einfaches mit SNMP verwaltetes Netzwerk dargestellt.

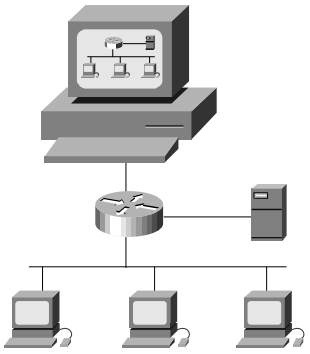


Bild 46.1: SNMP ermöglicht den Austausch von Netzwerk-Informationen zwischen Geräten

46.2 SNMP – Grundlegende Komponenten

Ein mit SNMP verwaltetes Netzwerk setzt sich aus drei wesentlichen Komponenten zusammen: verwaltete Geräte, Agenten und Netzwerk-Verwaltungssysteme (*Network Management Systems* = NMSs).

Bei einem verwalteten Gerät handelt es sich um einen Netzwerk-Knoten, der über einen SNMP-Agenten verfügt und der sich in einem verwalteten Netzwerk befindet. Verwaltete Geräte sammeln und speichern Verwaltungsinformationen und stellen diese Informationen mittels SNMP den NMSs zur Verfügung. Bei verwalteten Geräten, die manchmal auch als Netzwerk-Elemente bezeichnet werden, kann es sich um Router und Zugriffsserver, Switches und Bridges, Hubs, Hostcomputer oder Drucker handeln.

Bei einem Agenten handelt es sich um ein netzwerkverwaltendes Software-Modul, das sich auf einem verwalteten Gerät befindet. Ein Agent verfügt über lokales Wissen von Verwaltungsinformationen und übersetzt diese Informationen in eine Form, die zu SNMP kompatibel ist.

Ein NMS führt Anwendungen aus, die verwaltete Geräte überwachen und steuern. NMSs stellen den Großteil der für die Netzwerk-Verwaltung erforderlichen Bearbeitungs- und Speicherressourcen bereit. In jedem verwalteten Netzwerk müssen ein oder mehrere NMSs vorhanden sein.

Im Bild 46.2 sind die Beziehungen zwischen diesen drei Komponenten dargestellt.

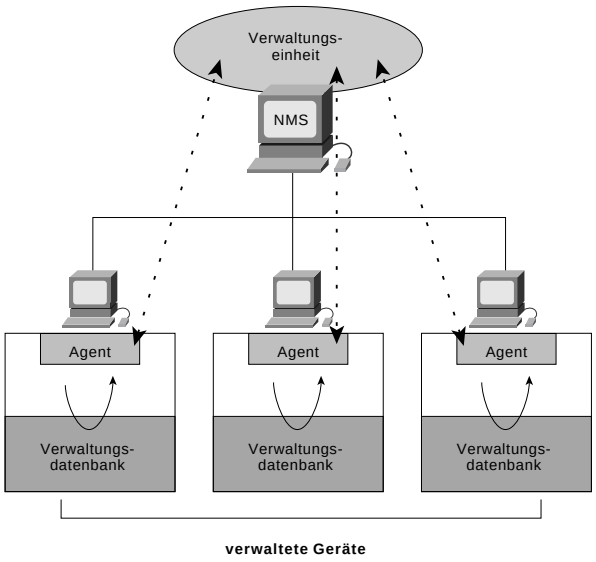


Bild 46.2: Ein per SNMP verwaltetes Netzwerk setzt sich aus verwalteten Geräten, Agenten und NMSs zusammen

46.3 SNMP – Grundlegende Befehle

Verwaltete Geräte werden mittels folgender vier grundlegenden Befehle von SNMP überwacht und gesteuert: *Read*, *Write*, *Trap* und *Traversal-Operationen*.

Der Befehl *Read* wird von einem NMS verwendet, um verwaltete Geräte zu überwachen. Das NMS untersucht verschiedene Variablen, die von verwalteten Geräten vorgehalten werden.

Der Befehl *Write* wird von einem NMS verwendet, um verwaltete Geräte zu steuern. Das NMS ändert die Werte von in verwalteten Geräten gespeicherten Variablen.

Der Befehl *Trap* wird von verwalteten Geräten verwendet, um dem NMS asynchron Ereignisse mitzuteilen. Beim Auftreten bestimmter Arten von Ereignissen schickt ein verwaltetes Gerät einen Trap an das NMS.

Traversal-Operationen werden vom NMS verwendet, um zu ermitteln, welche Variablen ein verwaltetes Gerät unterstützt und um der Reihe nach Informationen in Variablentabellen wie beispielsweise einer Routing-Tabelle zu sammeln.

46.4 SNMP – Management Information Base (MIB)

Eine Management Information Base (MIB) ist eine Sammlung von hierarchisch organisierten Informationen. Auf MIBs wird mittels eines Protokolls zur Netzwerk-Verwaltung wie beispielsweise SNMP zugegriffen. Sie setzen sich aus verwalteten Objekten zusammen und werden durch Objekt-IDs identifiziert.

Ein verwaltetes Objekt (manchmal auch als MIB-Objekt, Objekt oder MIB bezeichnet) stellt eine von einer beliebigen Anzahl von Eigenschaften eines verwalteten Geräts dar. Verwaltete Objekte bestehen aus einer oder mehreren Objektinstanzen, bei denen es sich im wesentlichen um Variablen handelt.

Es gibt zwei Arten verwalteter Objekte: skalare und tabellarische. Skalare Objekte definieren eine einzige Objektinstanz. Tabellarische Objekte definieren mehrere zusammengehörige Objektinstanzen, die in MIB-Tabellen zusammen gruppiert sind.

Ein Beispiel für ein verwaltetes Objekt ist *atInput*, bei dem es sich um ein skalares Objekt handelt, das als einzige Objektinstanz den ganzzahligen Wert enthält, der die Gesamtzahl der an einer Router-Schnittstelle eingegebenen AppleTalk-Pakete angibt.

Eine Objekt-ID weist ein verwaltetes Objekt in der MIB-Hierarchie eindeutig aus. Die MIB-Hierarchie kann als ein Baum mit namenloser Wurzel betrachtet werden, dessen Ebenen von unterschiedlichen Organisationen zugewiesen werden. Im Bild 46.3 ist der MIB-Baum dargestellt.

Die Objekt-IDs der obersten MIB-Ebene gehören zu unterschiedlichen Standardisierungsorganisationen, wohingegen die Objekt-IDs der untergeordneten Ebenen von angeschlossenen Organisationen belegt werden.

Anbieter können eigene Zweige definieren, die verwaltete Objekte für ihre eigenen Produkte enthalten. MIBs, die nicht standardisiert wurden, befinden sich üblicherweise im Zweig *experimental*.

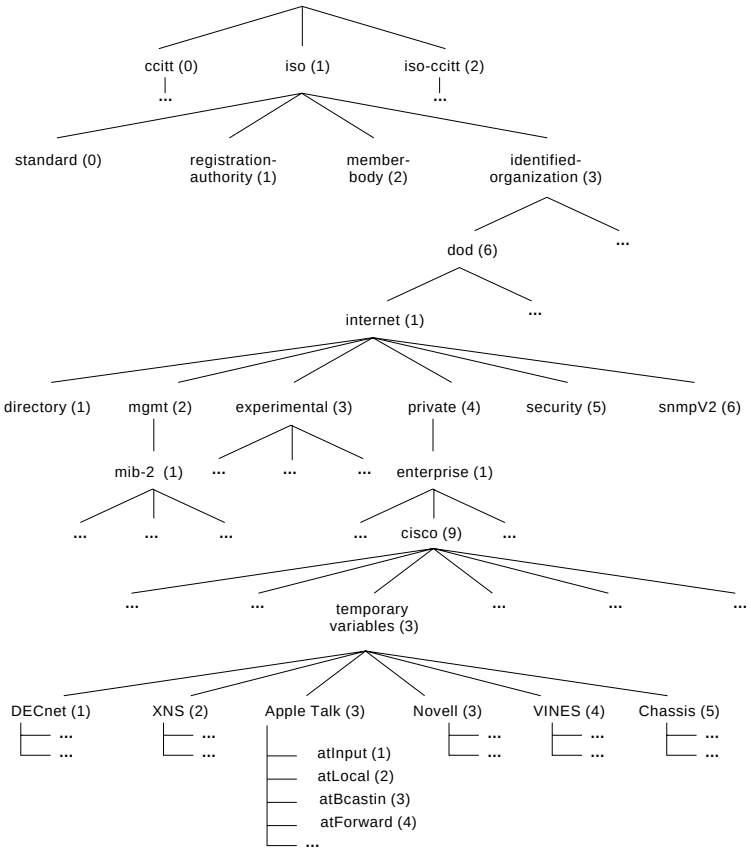


Bild 46.3: Der MIB-Baum stellt die verschiedenen Hierarchien dar, die von unterschiedlichen Organisationen zugewiesen wurden

Das verwaltete Objekt *atInput* kann entweder durch die Objektbezeichnung *iso.identified-organization.dod.internet. private. enterprise.cisco.temporary variables.AppleTalk.atInput* oder durch den entsprechenden Objektdeskriptor *1.2.6.1.4.1.9.3.3.1* eindeutig identifiziert werden.

46.5 SNMP und Datendarstellung

SNMP muß Inkompatibilitäten zwischen verwalteten Geräten erklären und diese ausgleichen. Verschiedene Computer verwenden verschiedene Techniken zur Datendarstellung, wodurch es zu Zugeständnissen für die Fähigkeit von SNMP zum Informationsaustausch zwischen verwalteten Geräten kommen kann. SNMP setzt eine Teilmenge der Abstract Syntax Notation One (ASN.1) ein, um sich auf die Kommunikation zwischen unterschiedlichen Systemen einzustellen.

46.6 SNMP Version 1 (SNMPv1)

SNMP Version 1 (SNMPv1) ist die erste Implementierung des Protokolls SNMP. Es ist im Request For Comments (RFC) 1157 beschrieben und funktioniert innerhalb der Spezifikationen für die Structure of Management Information (SMI). SNMPv1 arbeitet mit Protokollen wie beispielsweise dem User Datagram Protocol (UDP), Internet Protocol (IP), OSI Conectionless Network Service (CLNS), AppleTalk Datagram Delivery Protocol (DDP) und Novell Internet Paket Exchange (IPX). SNMPv1 wird häufig eingesetzt und stellt das Standardprotokoll für die Netzwerk-Verwaltung im Internet dar.

46.6.1 SNMPv1 und Structure of Management Information (SMI)

Die Structure of Management Information (SMI) definiert die Regeln für das Beschreiben von Verwaltungsinformationen mittels der Abstract Syntax Notation One (ASN.1). Die SMI von SNMPv1 ist in dem RFC 1155 definiert. Sie sorgt für drei Schlüsselspezifikationen: ASN.1-Datentypen, SMI-spezifische Datentypen und MIB-Tabellen von SNMP.

SNMPv1 und ASN.1-Datentypen

Die SMI von SNMPv1 gibt an, daß mit allen verwalteten Objekten eine bestimmte Teilmenge von ASN.1-Datentypen verbunden ist. Drei ASN.1-Datentypen werden benötigt: *Name*, *Syntax* und *Encoding*. Der Name dient als Objekt-ID. Die Syntax definiert den Datentyp des Objekts (beispielsweise Ganzzahl oder Zeichenkette). Die SMI setzt eine Teilmenge für ASN.1-Syntaxdefinitionen ein. Die Encoding-Daten beschreiben, auf welche Weise mit einem verwalteten Objekt verbundene Informationen für die Übertragung über das Netzwerk als eine Reihe von Dateneinträgen formatiert werden.

SNMPv1 und SMI-spezifische Datentypen

Die SMI von SNMPv1 beschreibt den Einsatz einiger SMI-spezifischer Datentypen, die in zwei Kategorien eingeteilt werden: *einfache Datentypen* und *anwendungsweite Datentypen*.

Es werden drei einfache Datentypen in der SMI von SNMPv1 definiert, die alle eindeutige Werte darstellen: *Integers*, *Octet Strings* und *Object-IDs*. Bei den Integern handelt es sich um vorzeichenbehaftete Ganzzahlen im Bereich von –2.147.483.648 bis 2.147.483.647. Bei den Octet Strings handelt es sich um sortierte Abfolgen von null bis 65.535 Oktetten. Object-IDs stammen aus der Menge sämtlicher entsprechend den ASN.1-Regeln erteilter Objekt-IDs.

In der SMI von SNMPv1 gibt es sieben anwendungsweite Datentypen: *Network Addresses*, *Counter*, *Gauges*, *Time Ticks*, *Opaques*, *Integers* und *Unsigned Integers*. Netzwerk-Adressen stehen für eine Adresse einer bestimmten Protokollfamilie. SNMPv1 unterstützt nur 32-Bit-IP-Adressen. Counter sind positive Ganzzahlen, die erhöht werden, bis sie einen Maximalwert erreichen und dann wieder bei Null beginnen. In SNMPv1 sind 32 Bit für die Zählergröße vorgesehen. Gauges sind positive Ganzzahlen, die erhöht oder verringert werden können, aber den erreichten Maximalwert beibehalten. Ein Time Tick steht für die Hundertstelsekunden nach irgendeinem Ereignis. Ein Opaque steht für eine beliebige Kodierung, die für die Übergabe beliebiger Informationszeichenketten verwendet wird, die nicht der von der SMI benutzten genauen Datentypisierung entsprechen. Ein Integer stellt Informationen mit vorzeichenbehafteten ganzzahligen Werten dar. Dieser Datentyp definiert den Datentyp Integer neu, der unter ASN.1 über eine beliebige, unter SMI aber über eine begrenzte Genauigkeit verfügt. Ein Unsigned Integer stellt Informationen mit vorzeichenlosen ganzzahligen Werten dar und ist nützlich, wenn Werte immer positiv ausfallen. Dieser Datentyp definiert den Datentyp Integer neu, der unter ASN.1 über eine beliebige, unter SMI aber über eine begrenzte Genauigkeit verfügt.

SNMP – MIB-Tabellen

Die SMI von SNMPv1 definiert hochgradig strukturierte Tabellen, die für die Gruppierung der Instanzen eines tabellarischen Objekts (also ein mehrere Variablen enthaltendes Objekt) eingesetzt werden. Tabellen setzen sich aus keiner oder mehreren Zeilen zusammen, die in einer Weise indiziert sind, die es SNMP gestattet, eine ganze Zeile mit einem einzigen Get-, GetNext- oder Set-Befehl zu ermitteln oder zu ändern.

46.6.2 SNMPv1 – Protokolloperationen

Bei SNMP handelt es sich um ein einfaches Request-Response-Protokoll. Das Netzwerk-Managementsystem (NMS) löst ein Request aus, und verwaltete Geräte liefern Responses zurück. Solches Verhalten wird durch den Einsatz einer der folgenden vier Protokolloperationen realisiert: *Get*, *GetNext*, *Set* und *Trap*. Die Get-Operation wird vom NMS verwendet, um den Wert eines oder mehrerer Objektinstanzen von einem Agenten zu erhalten. Kann der auf die Get-Operation reagierende Agent nicht für alle Objektinstanzen einer Auflistung Werte liefern, so stellt er gar keine Werte zur Verfügung. Die GetNext-Operation wird vom NMS verwendet, um den Wert von der nächsten Objektinstanz in einer Tabelle oder Auflistung eines Agenten zu erhalten. Die Set-Operation wird vom NMS verwendet, um die Werte von Objektinstanzen in einem Agenten zu setzen. Die Trap-Operation wird von Agenten verwendet, um dem NMS asynchron wichtige Ereignisse mitzuteilen.

46.7 SNMP Version 2 (SNMPv2)

SNMP Version 2 (SNMPv2) ist eine Weiterentwicklung der ersten Version, SNMPv1. Ursprünglich wurde SNMPv2 im Jahre 1993 als eine Reihe von Vorschlägen für Internet-Standards veröffentlicht; derzeit handelt es sich dabei um einen Standard in der Entwurfsphase. Wie bereits SNMPv1, funktioniert SNMPv2 innerhalb der Spezifikationen für die Structure of Management Information (SMI). Theoretisch bringt SNMPv2 einige Verbesserungen gegenüber SNMPv1 mit sich, zu denen beispielsweise zusätzliche Protokolloperationen gehören.

46.7.1 SNMPv2 und Structure of Management Information (SMI)

Die Structure of Management Information (SMI) definiert die Regeln für das Beschreiben von Verwaltungsinformationen mittels der Abstract Syntax Notation One (ASN.1).

Die SMI von SNMPv2 ist in der RFC 1902 definiert und bringt einige Ergänzungen und Erweiterungen für SMI-spezifische Datentypen mit; unter anderem für *Bit Strings*, *Network*

Addresses und *Counter*. Bit Strings sind nur in SNMPv2 definiert und umfassen keine oder mehrere bezeichnete Bits, die einen Wert angeben. Netzwerk-Adressen stehen für eine Adresse einer bestimmten Protokollfamilie. SNMPv1 unterstützt nur 32-Bit-IP-Adressen, während SNMPv2 auch andere Arten von Adressen unterstützt. Counter sind positive Ganzzahlen, die erhöht werden, bis sie einen Maximalwert erreichen, und dann wieder bei Null beginnen. In SNMPv1 sind 32 Bit für die Zählergröße vorgesehen. In SNMPv2 sind Zähler mit 32 und 64 Bit definiert.

SMI-Informationsmodule

Weiterhin beschreibt die SMI von SNMPv2 Informationsmodule, die eine Gruppe zusammengehöriger Definitionen angeben. Es gibt drei Arten von SMI-Informationsmodulen: *MIB-Module*, *Compliance-Statements* und *Capability-Statements*. MIB-Module enthalten Definitionen von sich aufeinander beziehenden verwalteten Objekten. Compliance-Statements stellen eine systematische Methode zum Beschreiben einer Gruppe von verwalteten Objekten dar, die implementiert sein müssen, um einen Standard zu erfüllen. Capability-Statements werden eingesetzt, um den genauen Grad an Unterstützung anzuzeigen, den ein Agent hinsichtlich einer MIB-Gruppe fordert. Ein NMS kann sein Verhalten gegenüber Agenten entsprechend der mit jedem Agenten verbundenen Compliance-Statements anpassen.

SNMPv2 – Protokolloperationen

Die in SNMPv1 verwendeten Operationen *Get*, *GetNext* und *Set* stimmen mit den in SNMPv2 verwendeten überein. Allerdings ergänzt und erweitert SNMPv2 einige Protokolloperationen. Beispielsweise erfüllt die SNMPv2-Operation *Trap* die gleiche Funktion wie in SNMPv1. Sie greift allerdings auf ein anderes Nachrichtenformat zurück und soll das Trap von SNMPv1 ersetzen.

SNMPv2 definiert weiterhin zwei neue Protokolloperationen: *GetBulk* und *Inform*. Die Operation GetBulk wird vom NMS verwendet, um umfangreiche Datenblöcke wie beispielsweise mehrere Zeilen einer Tabelle rationell zu erhalten. GetBulk füllt eine Response-Nachricht mit so vielen der angeforderten

Daten wie möglich. Die Operation Inform gestattet es einem NMS, Trap-Informationen an ein anderes NMS zu senden und eine Antwort zu erhalten. Wenn der auf GetBulk-Operationen reagierende Agent unter SNMPv2 nicht für alle Variablen einer Auflistung Werte liefern kann, stellt er Teilergebnisse zur Verfügung.

46.8 SNMP – Verwaltung

Bei SNMP handelt es sich um ein Protokoll für die verteilte Verwaltung. Ein System kann entweder ausschließlich als NMS oder Agent fungieren oder aber beide Funktionen übernehmen. Wenn eine System sowohl als NMS als auch als Agent fungiert, kann ein anderes NMS verlangen, daß das System verwaltete Geräte anfordert und eine Zusammenfassung der angeeigneten Informationen zur Verfügung stellt oder daß es lokal gespeicherte Verwaltungsinformationen mitteilt.

46.9 SNMP – Sicherheit

SNMP mangelt es an jeglicher Fähigkeit zur Authentifizierung, was zu einer Anfälligkeit für eine Vielzahl von Sicherheitsbedrohungen führt. Zu diesen zählen die *Täuschung*, die *Veränderung von Informationen*, die *Veränderung der Abfolge und des Timings von Nachrichten* sowie die *Preisgabe von Informationen*. Eine Täuschung liegt vor, wenn eine unautorisierte Einheit mit der mutmaßlichen Identität einer autorisierten Verwaltungseinheit Verwaltungsoperationen durchzuführen versucht. Die Veränderung von Informationen bezieht sich auf eine unautorisierte Einheit, die eine von einer autorisierten Einheit generierte Nachricht zu verändern versucht, so daß die Nachricht zu einer nicht autorisierten Verwaltungsoperation für das Accounting oder die Konfiguration führt. Zu einer Veränderung der Abfolge und des Timings von Nachrichten kommt es, wenn eine unautorisierte Einheit eine von einer autorisierten Einheit generierte Nachricht erneut abruft, verzögert oder kopiert und später wiedergibt. Zu einer Preisgabe von Informationen kommt es, wenn eine unautorisierte Einheit in verwalteten Objekten gespeicherte Werte herauszieht oder durch die Überwachung des Datenaustauschs zwischen Managern und Agenten von meldepflichtigen Ereignissen er-

fährt. Da SNMP keine Authentifizierung implementiert, realisieren viele Anbieter die Set-Operation nicht und reduzieren SNMP damit auf ein reines Überwachungsmedium.

46.10 SNMP – Zusammenarbeit

In der gegenwärtigen Spezifikation ist SNMPv1 in zwei Schlüsselbereichen nicht kompatibel zu SNMPv2: Nachrichtenformate und Protokolloperationen. SNMPv2-Nachrichten verwenden im Vergleich zu SNMPv1 andere Formate für den Header und die Protokolldateneinheiten (*Protocol Data Units* = PDUs). Außerdem verwendet SNMPv2 zwei Protokolloperationen, die in SNMPv1 nicht beschrieben sind. Darüber hinaus definiert die RFC 1908 zwei mögliche Strategien zur Koexistenz von SNMPv1/v2: Proxy-Agenten und »zweisprachige« Netzwerk-Verwaltungssysteme.

46.10.1 Proxy-Agenten

Ein SNMPv2-Agent kann folgendermaßen als Proxy-Agent für verwaltete Geräte von SNMPv1 agieren:

- Ein NMS von SNMPv2 ruft einen für einen SNMPv1-Agenten gedachten Befehl auf.
- Das NMS schickt die SNMP-Nachricht an den SNMPv2-Proxy-Agenten.
- Der Proxy-Agent leitet Get-, GetNext- und Set-Nachrichten unverändert an den SNMPv1-Agenten weiter.
- GetBulk-Nachrichten werden vom Proxy-Agenten in GetNext-Nachrichten umgewandelt und dann an den SNMPv1-Agenten weitergeleitet.
- Der Proxy-Agent bildet Trap-Nachrichten unter SNMPv1 auf Trap-Nachrichten unter SNMPv2 ab und leitet diese an das NMS weiter.

46.10.2 »Zweisprachige« Netzwerk-Verwaltungssysteme

»Zweisprachige« Netzwerk-Verwaltungssysteme (NMS) unter SNMPv2 unterstützen sowohl SNMPv1 als auch SNMPv2. Um diese doppelte Verwaltungsumgebung zu unterstützen,

muß eine Verwaltungsanwendung im »zweisprachigen« NMS eine Verbindung zu einem Agenten aufbauen. Daraufhin untersucht das NMS in einer lokalen Datenbank gespeicherte Informationen, um festzustellen, ob der Agent SNMPv1 oder SNMPv2 unterstützt. Entsprechend der Information in der Datenbank kommuniziert das NMS unter Verwendung der geeigneten Version von SNMP mit dem Agenten.

46.11 SNMP-Referenz: SNMPv1 Nachrichtenformate

Nachrichten unter SNMPv1 enthalten zwei Teile: einen Nachrichten-Header und eine Protocol Data Unit (PDU). Im Bild 46.4 ist das grundlegende Format einer Nachricht unter SNMPv1 dargestellt.

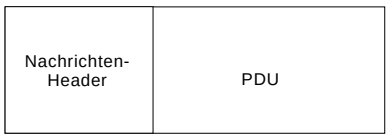


Bild 46.4: Eine Nachricht unter SNMPv1 setzt sich aus einem Header und einer PDU zusammen

46.11.1 SNMPv1 – Nachrichten-Header

Nachrichten-Header unter SNMPv1 enthalten zwei Felder: *Versionsnummer* und *Gemeinschaftsname*. Im folgenden werden diese Felder kurz beschrieben:

- *Versionsnummer* – Gibt die Version des verwendeten SNMP an.
- *Gemeinschaftsname* – Definiert eine Zugriffsumgebung für eine Gruppe von NMSs. Von NMSs innerhalb der Gemeinschaft sagt man, daß sie sich in derselben Verwaltungsdomain befinden. Gemeinschaftsnamen dienen als eine dürftige Form der Authentifizierung, da Geräte, die den richtigen Gemeinschaftsnamen nicht kennen, von SNMP-Operationen ausgeschlossen sind.

46.11.2 SNMPv1 – Protokolldateneinheit (PDU)

Protokolldateneinheiten (*Protocol Data Units* = PDUs) unter SNMPv1 enthalten einen bestimmten Befehl (Get, Set usw.)

und Operanden, welche die in der Transaktion einbezogenen Objektinstanzen angeben. Die PDU-Felder sind unter SNMPv1 wie von ASN.1 vorgeschrieben von variabler Länge. Im Bild 46.5 sind die Felder für die Transaktionen Get, GetNext, Response und Set PDUs unter SNMPv1 dargestellt.

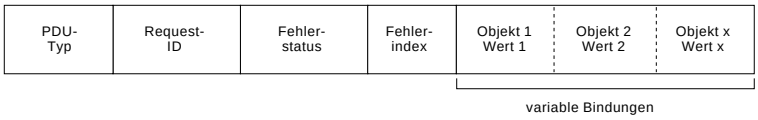


Bild 46.5: Get, GetNext, Response und Set PDUs enthalten unter SNMPv1 dieselben Felder

Im folgenden werden die im Bild 46.5 dargestellten Felder kurz beschrieben:

- *PDU-Typ* – Gibt den Typ der übertragenen PDU an.
- *Request-ID* – Verbindet SNMP-Requests mit Responses.
- *Fehlerstatus* – Gibt einen von mehreren Fehlern und Fehlertypen an. Nur die Response-Operation setzt dieses Feld. Andere Operationen belegen dieses Feld mit dem Wert 0.
- *Fehlerindex* – Verbindet einen Fehler mit einer bestimmten Objektinstanz. Nur die Response-Operation setzt dieses Feld. Andere Operationen belegen dieses Feld mit dem Wert 0.
- *Variable Bindungen* – Dient als Datenfeld der PDU unter SNMPv1. Jede Variablenbindung verknüpft eine bestimmte Objektinstanz mit dessen aktuellem Wert (mit Ausnahme von Get- und GetNext-Requests, für die der Wert ignoriert wird).

46.11.3 Format von Trap PDU

Im Bild 46.6 sind die Felder von Trap PDU unter SNMPv1 dargestellt.

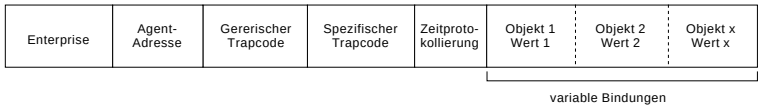


Bild 46.6: Trap PDU setzt sich unter SNMPv1 aus acht Feldern zusammen

Im folgenden werden die im Bild 46.6 dargestellten Felder kurz beschrieben:

- *Enterprise* – Gibt die Art des verwalteten Objekts an, das den Trap generiert hat.
- *Agent-Adresse* – Stellt die Adresse des verwalteten Objekts bereit, das den Trap generiert hat.
- *Generischer Traptyp* – Gibt einen von mehreren generischen Traptypen an.
- *Spezifischer Trapcode* – Gibt einen von mehreren spezifischen Trapcodes an.
- *Zeitprotokollierung* – Stellt die seit der letzten Initialisierung des Netzwerks bis zum Erzeugen des Traps verstrichene Zeit bereit.
- *Variable Bindungen* – Dient als Datenfeld der Trap PDU unter SNMPv1. Jede Variablenbindung verknüpft eine bestimmte Objektinstanz mit dessen aktuellem Wert.

46.12 SNMP-Referenz: SNMPv2 Nachrichtenformate

Nachrichten unter SNMPv1 setzen sich aus einem Header und einer PDU zusammen. Im Bild 46.7 ist das grundlegende Format einer Nachricht unter SNMPv2 dargestellt.

Bild 46.7: Nachrichten unter SNMPv2 setzen sich ebenfalls aus einem Header und einer PDU zusammen

46.12.1 SNMPv2 – Nachrichten-Header

Nachrichten-Header unter SNMPv2 enthalten zwei Felder: *Versionsnummer* und *Gemeinschaftsname*. Im folgenden werden diese Felder kurz beschrieben:

- *Versionsnummer* – Gibt die Version des verwendeten SNMP an.

– *Gemeinschaftsname* – Definiert eine Zugriffsumgebung für eine Gruppe von NMSs. Von NMSs innerhalb der Gemeinschaft sagt man, daß sie sich in derselben Verwaltungsdomain befinden. Gemeinschaftsnamen dienen als eine dürftige Form der Authentifizierung, da Geräte, die den richtigen Gemeinschaftsnamen nicht kennen, von SNMP-Operationen ausgeschlossen sind.

46.12.2 SNMPv2 – Protokolldateneinheit (PDU)

SNMPv2 beschreibt zwei PDU-Formate, die von den SNMP-Protokolloperationen abhängen. Die Felder von PDU unter SNMPv2 verfügen wie von der ASN.1 vorgeschrieben über eine variable Länge.

Im Bild 46.8 sind die Felder von Get, GetNext, Inform, Response, Set und Trap PDUs unter SNMPv2 dargestellt.

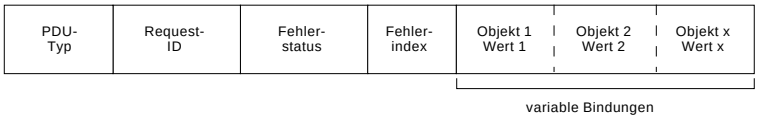


Bild 46.8: Get, GetNext, Inform, Response, Set und Trap PDUs enthalten unter SNMPv2 dieselben Felder

Im folgenden werden die im Bild 46.8 dargestellten Felder kurz beschrieben:

– *PDU-Typ* – Gibt den Typ der übertragenen PDU an (Get, GetNext, Inform, Response, Set oder Trap).

– *Request-ID* – Verbindet SNMP-Requests mit Responses.

– *Fehlerstatus* – Gibt einen von mehreren Fehlern und Fehlertypen an. Nur die Response-Operation setzt dieses Feld. Andere Operationen belegen dieses Feld mit dem Wert 0.

– *Fehlerindex* – Verbindet einen Fehler mit einer bestimmten Objektinstanz. Nur die Response-Operation setzt dieses Feld. Andere Operationen belegen dieses Feld mit dem Wert 0.

– *Variable Bindungen* – Dient als Datenfeld der PDU unter SNMPv2. Jede Variablenbindung verknüpft eine bestimmte Objektinstanz mit dessen aktuellem Wert (mit Ausnahme von Get- und GetNext-Requests, für die der Wert ignoriert wird).

Format von GetBulk PDU

Im Bild 46.9 sind die Felder von GetBulk PDU unter SNMPv2 dargestellt.

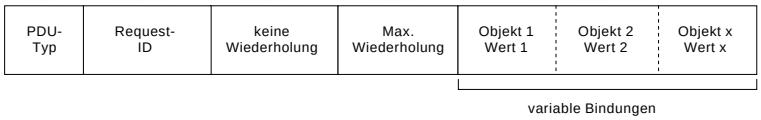


Bild 46.9: GetBulk PDU setzt sich unter SNMPv2 aus sieben Feldern zusammen

Im folgenden werden die im Bild 46.9 dargestellten Felder kurz beschrieben:

- *PDU-Typ* – Weist die PDU als eine GetBulk-Operation aus.
- *Request-ID* – Verbindet SNMP-Requests mit Responses.
- *Keine Wiederholung* – Gibt die Anzahl von Objektinstanzen im Feld *Variable Bindungen* an, die von Beginn des Request an nur einmal erhalten werden sollen. Dieses Feld wird verwendet, wenn es sich bei einigen der Instanzen um skalare Objekte mit nur einer Variable handelt.
- *Max. Wiederholung* – Definiert, wie oft andere als die durch das Feld *Keine Wiederholung* angegebenen Variablen maximal erhalten werden sollen.
- *Variable Bindungen* – Dient als Datenfeld der PDU unter SNMPv2. Jede Variablenbindung verknüpft eine bestimmte Objektinstanz mit dessen aktuellem Wert (mit Ausnahme von Get- und GetNext-Requests, für die der Wert ignoriert wird).