# Computernetwerken

## Inhoud



# 1 Algemeen

## 1.1 Definities en begrippen

Een algemeen geaccepteerde definitie van het begrip *computernetwerk*, of kortweg *netwerk*, bestaat niet. Vanaf de jaren ’60, toen onder auspiciën van het Advanced Research Project Agency (ARPA) de eerste schreden werden gezet die leidden tot het ARPAnet en later tot het Internet, zijn er zeer veel technologische en conceptuele ontwikkelingen geweest rond netwerken. Kenmerkende voorwaarden, waarbij de verschijningsvorm en de doelstelling van de eindgebruiker worden benadrukt, zijn echter:

- een stelsel door communicatieverbindingen gekoppelde maar overigens *zelfstandige* informatieverwerkende entiteiten, die
- via deze *verbindingen* en overeenkomstig precieze *voorschriften* interakties kunnen uitvoeren ten behoeve van een of meer specifieke doelen.

Een entiteit is *zelfstandig* omdat het zonder tussenkomst van andere entiteiten een informatieverwerkende taak moet kunnen uitvoeren, en daarom over een zekere mate van intelligentie moet beschikken. Meestal betekent dit dat de entiteit uitgerust is met een computer, vandaar de term computernetwerk. In het vervolg zullen we daarom ook de term computersysteem, of kortweg *systeem*, gebruiken in plaats van entiteit.

Communicatie*verbindingen* zijn nodig omdat de computersystemen geografisch verspreid zijn en de verbindingen gebruiken om afstand te overbruggen. De *topologie* van de verbindingen en de geografische afstand van de systemen spelen een belangrijke rol. Ter aanduiding van de

geografische spreiding van een netwerk onderscheidt men vaak *Local Area Networks* (LAN's), *Metrapolitan Area Networks* (MAN's) en *Wide Area Networks* (WAN's).

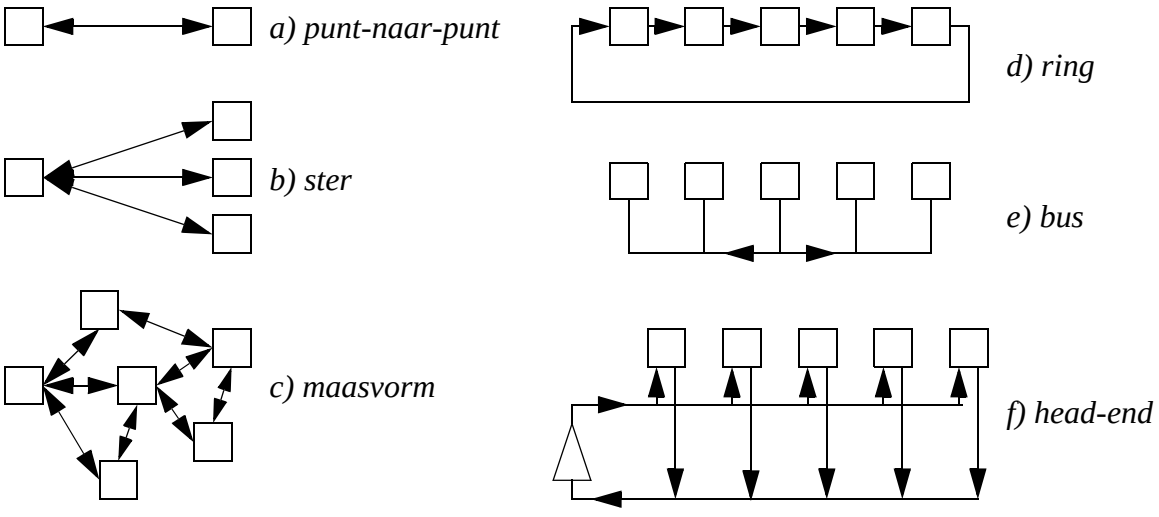
Een verzameling *voorschriften* voor interakties tussen systemen wordt een *protocol* genoemd. Een protocol is in het algemeen zodanig opgesteld dat een maximum aan vrijheid wordt geboden om de voorschriften te implementeren. Aldus gedefinieerde protocollen worden (protocol-) *standaarden* voor *open* systemen genoemd. Ze staan toe dat een netwerk kan worden samengesteld uit systemen die door verschillende fabrikanten geleverd kunnen worden.

Ook de voor de gebruikers relevante resultaten van de interakties tussen systemen worden precies beschreven, zij het op een hoger abstractieniveau. Een eindgebruiker is uiteindelijk niet geïnteresseerd in de interne werking van het netwerk volgens een stelsel van protocollen, maar in de dienst of *service* die door het netwerk wordt geleverd. Een service legt vast wat een gebruiker mag verwachten van een netwerk, en kan dus tevens gebruikt als referentie voor de ontwikkeling van verdere informatieverwerkende taken.

## 1.2 Netwerktopologieën

De *topologie* is de geografische structuur van communicatieverbindingen en systemen die het netwerk vormen. De topologie van een computernetwerk levert belangrijke randvoorwaarden voor de protocollen, het gebruik en het technisch beheer van het netwerk.

Op het eerste gezicht lijkt een ideale verbindingsstructuur er een te zijn waarbij ieder computersysteem met ieder ander computersysteem is verbonden. Communicatieverbindingen zijn echter kostbaar. Bovendien, niet iedereen communiceert met iedereen, en het aantal computersystemen in een computernetwerk kan voortdurend veranderen. Men streeft daarom naar optimalisatie van de verbindingsstructuren, hetgeen aanleiding geeft tot specifieke topologieën zoals punt-naar-punt, ster, maasvorm, ring, bus en head-end (zie Figuur 1.1).




*Figuur: 1.1: Netwerktopologieën*

*Punt-naar-punt (point to point)*

De *punt-naar-punt* verbinding (Figuur 1.1a) is de eenvoudigste topologie. Voor de transmissietechnologie levert deze topologie de minste problemen op, omdat iedere ontvanger slechts op

een enkele zender hoeft te worden afgestemd. Hierdoor kunnen grote afstanden worden overbrugd. De punt-naar-punt verbinding vormt de basis voor andere, meer complexe topologieën, zoals de ster-, maasvorm- en ringtopologie.

*Ster (star)*

In de *ster*topologie (Figuur 1.1b) wordt een centraal systeem via een aparte verbinding gekoppeld met ieder ander systeem. Meestal is er een hiërarchische relatie tussen het centrale systeem, of *master*, en de andere systemen, die dan *slaves* worden genoemd. De stertopologie is kwetsbaar, omdat uitvallen van het centrale systeem het gehele netwerk platlegt. Deze topologie wordt vaak aangetroffen bij oudere systemen met gecentraliseerde intelligentie, maar komt ook voor bij moderne netwerken met toegangscomputers.

*Maasvorm (mesh)*

De *maasvorm*topologie (Figuur 1.1c) ontstaat meestal als de systemen over een groot gebied (bijv. een land of continent) verspreid zijn en de kostenbeheersing van de verbindingen een onregelmatige structuur noodzakelijk maakt. Bij de maasvormtopologie kunnen berichten via verschillende routes de bestemming bereiken, hetgeen aanleiding geeft tot protocollen voor *routering*, *flow control* en *congestion control*.

*Ring*

In de *ring*topologie (Figuur 1.1d) is ieder systeem met twee andere systemen verbonden, nl. een systeem waarvan berichten ontvangen kunnen worden en een systeem waarnaar berichten gestuurd kunnen worden. De berichten volgen dus een vaste route, waardoor de protocollen voor gegevenstransport relatief eenvoudig kunnen zijn. Een bericht blijft in de ring, totdat een systeem het bericht verwijdert (d.w.z., niet verder doorstuurt). Hierdoor zijn *broadcast*toepassingen gemakkelijk te realiseren.

*Bus*

De *bus*topologie (Figuur 1.1e) bestaat uit een niet onderbroken communicatiemedium (bijv. een coaxkabel), waarop verschillende systemen zijn aangesloten. Een systeem kan een bericht op het medium plaatsen, waarna het bericht door ieder systeem gelezen kan worden. De bus heeft hierdoor een inherente broadcasteigenschap, en t.o.v. de ring de voordelen van:

- geringe vertraging, omdat een bericht rechtstreeks en niet via tussenliggende systemen naar zijn bestemming reist;
- grote robuustheid, omdat het uitvallen van een systeem niet de rest van het netwerk uitschakelt.

Daarentegen heeft de bustopologie een complexer *medium access control* protocol nodig, om de toegang tot het gemeenschappelijke medium te controleren en te coördineren.

*Head-end*

De *head-end*topologie (Figuur 1.1f) combineert twee logische (éénrichtings-) bussen, de *up-link* en de *down-link*, die aan één zijde door een versterker, de *head-end*, zijn doorverbonden. Elk systeem is zowel verbonden met de up-link als de down-link. Een systeem kan een bericht op de up-link plaatsen, waarna het bericht zich voortplant naar de head-end, daar wordt versterkt, en op de down-link wordt geplaatst. Daar kan het bericht door ieder systeem worden gelezen. Veel netwerken met een head-endtopologie zijn gebaseerd op de technologie voor kabeltelevisie. Ook satellietnetwerken hebben een head-endtopologie, waarbij de head-end *transponder* wordt genoemd en in de satelliet is geplaatst.

## 1.3 Schakeltechnieken

Het uitwisselen van berichten kan op protocolniveau gerealiseerd worden overeenkomstig de technieken van circuit, packet en message switching.

Bij *circuit switching* wordt een echte (fysieke) verbinding, of *circuit*, opgebouwd tussen twee systemen. Vervolgens kunnen de systemen rechtstreeks, zonder *buffering* in tussengelegen systemen, gegevens uitwisselen. Hierdoor kunnen *throughput* (doorvoer) en *transfer delay* (vertraging) worden gegarandeerd. Circuit switching wordt traditioneel toegepast in publieke telefoonnetwerken. Het belangrijkste nadeel van circuit switching is dat de snelheid van zender en ontvanger precies op elkaar afgestemd moeten zijn. Om deze reden wordt capaciteit in vaste stappen (bijv. van 64 kbit/sec) aan de verbinding toegewezen.

Bij *packet switching* worden gegevens verstuurd in pakketten, of *packets*, met een gegeven maximale of vaste lengte. Een *segmentation and reassembly* protocol zorgt ervoor dat de zender berichten van willekeurige lengte opdeelt in één of meer pakketten en dat de ontvanger de oorspronkelijke berichten weer samenstelt uit de ontvangen pakketten. Pakketten worden afzonderlijk door het netwerk vervoerd en kunnen, in sommige netwerken, verschillende routes volgen. Packet switching staat een variabele throughput toe. Omdat de maximale lengte van de pakketten vaststaat kan de verwerking in (schakel-) systemen relatief efficiënt plaatsvinden. Packet switching wordt gebruikt in de meeste moderne (gegevenstransport-) netwerken, inclusief TCP/IP netwerken en ATM netwerken. In ATM heeft een pakket een korte, vaste lengte, en wordt *cell* genoemd.

Bij *message switching* worden gegevens verstuurd in berichten, of *messages*. Message switching lijkt op packet switching, met het belangrijke verschil dat berichten van variabele lengte zijn. Berichten worden gegenereerd en verwerkt door een applicatie en vertegenwoordigen dus een eenheid van informatie. Net als pakketten worden berichten afzonderlijk door het netwerk vervoerd, en moeten in hun geheel van (bron- of tussen-) systeem naar (tussen- of bestemmings-) systeem worden overgedragen. De nadelen van message switching zijn de grote en variabele buffers die in de systemen beschikbaar moeten zijn en de relatief lange transfer delay. Message switching wordt tegenwoordig alleen nog op applicatieniveau toegepast, bijv. voor *electronic mail*, en wordt dan vaak ook aangeduid met de term *store-and-forwarding*.

## 1.4 Standaardisatie

*Standaardisatie*, of *normering*, speelt een cruciale rol in de ontwikkeling en acceptatie van computernetwerken. Om verschillende systemen te verbinden via een computernetwerk is het nodig dat er gemeenschappelijke protocollen worden gebruikt. Elke gebruiker wil echter zijn protocolimplementaties betrekken van leveranciers en fabrikanten die de beste prijs/prestatie kunnen leveren voor het (specifieke) computerplatform van de gebruiker. Er zijn daarom protocolstandaarden nodig die precies voorschrijven wat nodig is om bepaalde interakties tussen systemen mogelijk te maken, maar geen onnodige beperkingen opleggen aan de fabrikant die de protocollen implementeert.

Behalve protocollen worden om dezelfde reden ook services, (abstracte) *interfaces*, *informatiestructuren* (bijv. voor gebruikersgedefinieerde documenten) en *informatie-objecten* (bijv. voor managementdoeleinden) gestandaardiseerd.

Standaardisatie wordt noodzakelijkerwijs in internationaal verband gedaan, met deelname van fabrikanten, leveranciers, gebruikersgroepen, academia en (soms) overheden. Behalve stan-

daarden voor concrete systemen kunnen standaardisatie-organisaties ook richtlijnen ontwikkelen die systeemstructuren vastleggen. Dergelijke richtlijnen worden *referentiemodellen* genoemd, en hebben als belangrijkste doel om verschillende standaardisatie-activiteiten te coördineren en om de ontwerpcomplexiteit beheersbaar te houden.

De belangrijkste organisaties voor standaardisatie t.b.v. computernetwerken zijn ISO/IEC, ITU, IEEE, IETF, W3C en OMG:

- *ISO* (International Standards Organisation) en *IEC* (International Electrotechnical Commission) zijn samen verantwoordelijk voor de ontwikkeling van protocol- en servicestandaarden voor *OSI* (Open Systems Interconnection). Daarnaast ontwikkelen ze ook een referentiemodel voor *ODP* (Open Distributed Processing).
- *ITU* (International Telecommunication Union) ontwikkelt protocolstandaarden voor publieke netwerken. Op het gebied van OSI en ODP werkt ITU nauw samen met ISO/IEC.
- *IEEE* (Institute of Electrical and Electronics Engineering) ontwikkelt ondermeer standaarden voor *local area* netwerken. Deze standaarden worden zo mogelijk ook als ISO/IEC standaarden gepubliceerd.
- *IETF* (Internet Engineering Task Force) ontwikkelt standaarden voor het *Internet*. Behalve aan protocolstandaarden voor *TCP/IP* netwerken, wordt o.m. ook gewerkt aan standaarden voor Internet applicaties, management en beveiliging. Standaarden worden hier *RFCs* (Request for Comments) genoemd.
- *W3C* (World Wide Web Consortium) ontwikkelt standaarden voor het *World Wide Web* (WWW), momenteel de belangrijkste applicatie van het Internet. Een belangrijk werkgebied van W3C betreft *mark-up languages* voor het definiëren en structureren van webdocumenten.
- *OMG* (Object Management Group) ontwikkelt standaarden voor *CORBA* (Common Object Request Broker Architecture), een infrastructuur voor gedistribueerde applicaties gebaseerd op het object paradigma (*distributed object computing* applicaties). De koppeling van verschillende CORBA platformen is mogelijk via het Internet (TCP/IP).

# 2 Netwerkarchitecturen

Gebruikers stellen steeds grotere eisen aan telematica-toepassingen, computersystemen en communicatiemiddelen, waardoor een groter aantal en een grotere verscheidenheid aan functies ondersteund dienen te worden. Dit geeft aanleiding tot gecompliceerde protocollen. Bovendien stelt de heterogeniteit van computersystemen en communicatiemiddelen stringente eisen aan de implementatie-onafhankelijke definitie (d.w.z. een functionele of abstracte specificatie) van protocollen.

## 2.1 Referentiemodellen en terminologie

Om de complexiteit van protocollen te kunnen beheersen worden de mogelijke interacties tussen computersystemen meestal gestructureerd overeenkomstig een horizontale en verticale structurering. Op deze structuur wordt een aantal abstracte concepten gedefinieerd:

- het service-concept;
- het protocol-concept;
- het interface-concept.

Deze abstracte concepten worden vervolgens overeenkomstig een aantal technische en organisatorische criteria gehanteerd. Een *referentiemodel* is het geheel van de bovengenoemde structuur tezamen met de globale technische criteria op grond waarvan deze structuur is gedefinieerd.

Een referentiemodel levert daarmee een stelsel randvoorwaarden voor de verdere invulling van het model met technische specificaties voor standaard services en protocollen. Door het referentiemodel tezamen met de standaard services en protocollen worden de interacties in een systeem ondubbelzinnig en volledig vastgelegd.

Twee toonaangevende referentiemodellen zijn het *OSI Reference Model* (OSI-RM) en het Internet model (ook wel aangeduid als de *Internet Protocol Suite*, IPS). Een belangrijk verschil in de ontwikkeling van beide referentiemodellen is dat het OSI-RM wordt gekenmerkt door een meer architecturale en theoretische benadering, terwijl het Internet model wordt gekenmerkt door een meer implementatiegerichte en pragmatische benadering. Met name hierdoor is het ontwikkeltraject van de Internet protocollen aanzienlijk korter gebleken en genieten ze een grotere populariteit dan de OSI protocollen.

Gelet op bovenstaande wordt de verdere behandeling van de architectuur van computernetwerken gebaseerd op het OSI-RM. Bovendien zijn OSI concepten ook geldig binnen het Internet model. De behandeling van een aantal veel gebruikte protocollen (in paragraaf 3) wordt daarentegen gestructureerd volgens het Internet model.

### 2.1.1 Verticale structuur

De *verticale structuur* in een referentiemodel ontstaat door in het netwerk substructuren te onderscheiden met gelijke of overeenkomstige interactiepatronen. In het OSI-RM zijn dit de open systemen welke gevormd worden door de computer systemen in het netwerk, zoals hosts, routers, printers en terminals. Hierdoor komt de verticale structuur overeen met de geografische scheiding van de computersystemen.

### 2.1.2 Horizontale structuur

De horizontale structuur ontstaat door binnen ieder verticaal element (open systeem), de functies die de toepassing direct(er) ondersteunen dicht(er) bij de toepassing te plaatsen, en de middelen waarop deze functies steunen daar hiërarchisch onder te plaatsen. Door deze benadering herhaald toe te passen ontstaat een horizontale structuur van *hiërarchisch gerangschikte lagen* van functies. Binnen ieder systeem zijn dezelfde functies op hetzelfde niveau in de hiërarchie, d.w.z. in dezelfde laag, geplaatst.

De functies van de lagen van het OSI-RM zijn ontstaan door ‘decompositie’ van de totale functie van het open systeem. In die zin verschilt dit lagen-model van andere lagen-modellen, waarin lagen eenzelfde functionaliteit op verschillende abstractieniveaus definiëren.

In het OSI-RM wordt iedere laag met een specifiek nummer aangeduid, oplopend van 0 voor de media (het laagst in de hiërarchie) tot 7 voor de applicatiefuncties (het hoogst in de hiërarchie). Lagen worden met de nummers N-1, N, N+1, enz. aangeduid zodra over algemene concepten wordt gesproken. De grens van laag N en N+1 wordt gevormd door de zgn. N-service access points. Een N-service access point modelleert een logische of fysieke plaats waar de functies van de N+1 laag toegang tot de N-service kunnen verkrijgen.

### 2.1.3 Service-concept

Uiteindelijk is de eindgebruiker niet geïnteresseerd in moeilijke protocoloplossingen en ziet hij het totale systeem het liefst als een ondoorzichtige doos, een zgn. *black box*, die bepaalde diensten verleent.

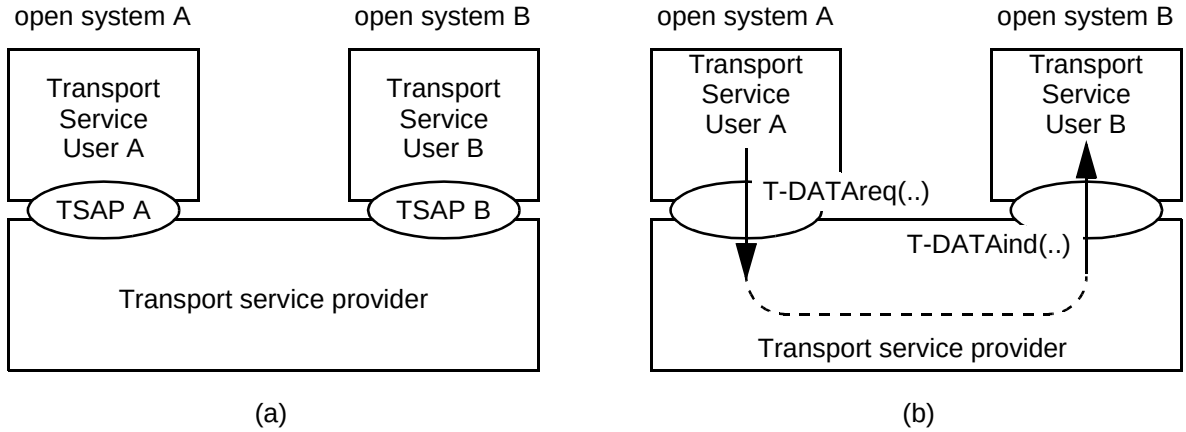
Het *service-concept* komt overeen met dit black box concept. Het definieert welke diensten een gedistribueerd systeem verleent, zonder de interne structuur en distributie van functies van het systeem zichtbaar te maken voor de gebruiker. Het service-concept speelt een vitale rol in de beheersing van de complexiteit van een computernetwerk: het vaststellen van de service vormt het uitgangspunt voor de ontwikkeling van de protocollen die op de service steunen (c.q. deze service gebruiken), zowel als voor de ontwikkeling van de protocollen die deze service moeten implementeren.

Bovendien vervult de service een strategische rol op de lange termijn: nieuwe protocollen die op de service steunen kunnen worden ontwikkeld zonder dat de protocollen die de service implementeren worden aangetast. Hetzelfde geldt ook omgekeerd.

In het OSI-RM wordt het service-concept op ieder niveau in de systeemstructuur gehanteerd. De services vertonen een geneste structuur. In het Internet protocol model daarentegen worden services niet apart gedefinieerd, maar wordt het service concept slechts impliciet gehanteerd.

*N-service*

Een *N-service* definieert het functionele gedrag van alle functies van laag 0 t/m N zoals dat op de grens van laag N en N+1 kan worden geobserveerd. De specificatie van de N-service wordt op de eenvoudigste manier gegeven: in de specificatie kan niet worden herkend hoe de verschillende systemen en verschillende lagen in het functionele gedrag van de N-service bijdragen. D.w.z. de compositie van lagen 0 t/m N wordt beschouwd als een black box. Deze black box wordt een N-service provider genoemd. Figuur 2.1(a) illustreert de transport service provider met twee gebruikers.




*Figuur 2.1: (a) Transport service provider (TSAP = Transport Service Access Point); (b) Executie van twee Transport Service Primitives.*

*Service primitive*

De interactie tussen de service user en de service provider vindt plaats op basis van zgn. service primitives. Een *service primitive* is een elementaire interactie die op een service access point kan plaatsvinden. In de service primitive vindt informatie-uitwisseling plaats waarbij de

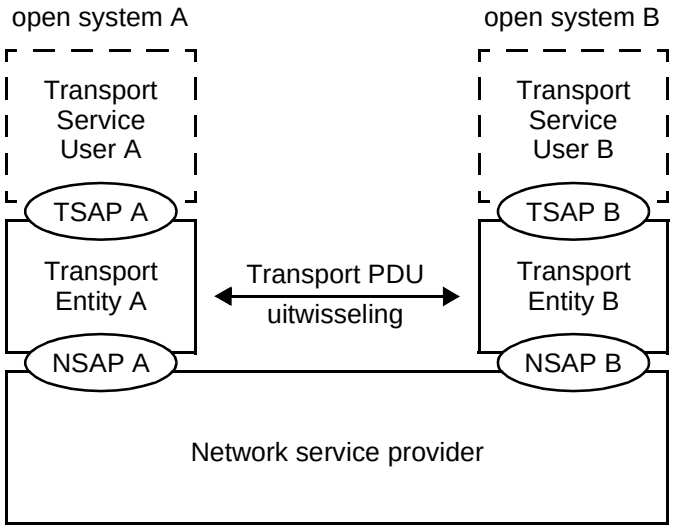
betrokkenen een specifieke verantwoordelijkheid dragen. Een *service-specificatie* definieert de service primitives en de manier waarop deze primitives qua tijdsvolgorde en inhoud van elkaar afhangen. Een service primitive bestaat meestal uit een aantal parameters. Een belangrijke parameter vormt de *N-service data unit* (N-SDU), die dient om data tussen service users uit te wisselen.
Figuur 2.1(b) illustreert de executie van twee transport service primitives: de T-DATAreq(..) SP modelleert het versturen van data door Transport Service User A via de Transport Service Provider en de T-DATAind(..) SP modelleert het afleveren van de data door de Transport Service Provider aan Transport Service User B. Een Service Primitive (SP) wordt gerepresenteerd d.m.v. een pijl, die wijst in de richting van de informatiestroom. De stippellijn die beide pijlen verbindt, in combinatie met de richting van de pijlen, representeert de volgorde waarin de SPs worden uitgevoerd. SP parameters zijn hier voor het gemak weggelaten.

### 2.1.4 Protocol-concept

Een protocol definieert hoe functies van een laag bijdragen aan de service die op de grens met de bovenliggende laag wordt geleverd. Anders dan het service-concept is het protocol-concept bedoeld om de functionele structuur en inhoud van een specifieke laag te definiëren. Het protocol-concept is niet gericht op de gebruiker van een laag, maar op de ontwerper die de functies van de laag moet implementeren, en is dus van belang voor de fabrikant.

*N-protocol*

Een N-protocol definieert de functies van laag N en de manier waarop de interactie van deze functies, via de onderliggende N-1-service, de N-service leveren. Een protocol kan dus niet onafhankelijk gezien worden van de service waarop het steunt en van de service die het moet ondersteunen. Figuur 2.2 geeft een voorbeeld van het *transport protocol* dat steunt op de *network service* om de *transport service* te leveren.




*Figuur 2.2: Transport protocol (NSAP = Network Service Access Point).*

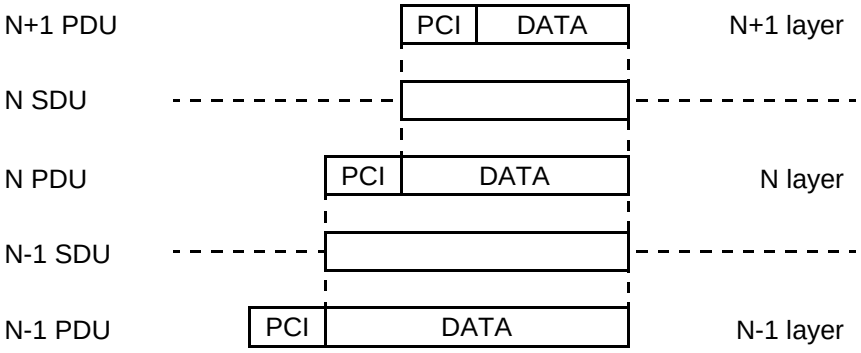
*N-layer subsystem*

De functies die zich binnen een laag en een systeem bevinden worden *N-layer subsystems* genoemd. In Figuur 2.2 zijn twee transport layer subsystems getekend en aangeduid met *transport entities*. Een N-layer subsystem wordt voorts onderverdeeld in één of meerdere zgn. *protocol entities*, afhankelijk of de functies van een laag door één of meerdere protocollen zijn gedefinieerd. In een protocol-specificatie wordt een protocol entity zodanig beschreven dat een

implementeur de maximale vrijheid houdt voor de implementatie van het systeem; hij is echter dusdanig in zijn vrijheid beknot dat de interactie met de andere entiteiten in dezelfde laag (*peer entities*) ondubbelzinnig is gegarandeerd.

*Protocol data unit*

De interactie tussen peer protocol entities geschiedt op basis van zgn. *Protocol Data Units* (PDUs). Deze PDUs worden afgebeeld op de Service Data Units (SDUs) van de onderliggende service. De PDUs worden 'verpakt als data' vervoerd naar de peer entity. Bovendien worden de SDUs weer afgebeeld op de lager liggende PDUs. Deze scheiding is nodig om in iedere laag een andersoortige problematiek te kunnen oplossen. Men noemt dit ook wel 'separation of concerns'. Door deze benadering ontstaat de (zeer schematische) afbeelding van headers in opvolgende PDUs (zie Figuur 2.3). Headers bevatten de (controle) informatie die protocol entiteiten met elkaar uitwisselen om hun activiteiten te coördineren. Headers zijn in Figuur 2.3 als PCI (*Protocol Control Information*) aangegeven.




*Figuur 2.3: Schematische voorstelling van hiërarchisch geordende PDUs (indien geen segmentatie, blocking of concatenatie plaatsvindt, en met uitsluitend data parameters in de SDUs).*

### 2.1.5 Interface-concept

Een *abstract* interface definieert:

- de interactie tussen de service-gebruiker en service provider in termen van service primitives die op één enkel service access point uitgevoerd kunnen worden;
- hoe de service primitives qua tijdvolgorde en inhoud (d.w.z. parameterwaarden) van elkaar afhangen.

Men zou kunnen zeggen dat een abstract interface het service access point volledig definieert. Per laag is er dan ook een ander abstract interface. Een abstract interface is hetzelfde voor een service-specificatie en een protocol-specificatie. Het abstracte interface tussen een N+1-service-gebruiker en een N-service provider is dus hetzelfde als het abstracte interface tussen een N+1-protocol entity en een N-protocol entity.

Binnen een referentiemodel wordt het begrip interface meestal geïnterpreteerd als het zgn. *real interface* en wordt verder niet expliciet gesproken over het abstracte interface, omdat dit impliciet met service-specificaties wordt gegeven. Bij de definitie van een real interface wordt in detail aangegeven hoe de precieze interactie op implementatieniveau tussen de betrokken entiteiten plaatsvindt. Dergelijke specificaties behoren tot de implementatievrijheid van de

fabrikant en horen niet thuis in het referentiemodel en aanverwante standaarden. Verschillende real interfaces kunnen geschikt zijn om een abstract interface te implementeren. Bij fabrikanten bestaan daarom grote weerstanden om tot standaardisatie van (real) interfaces over te gaan. Voorbeelden van real interfaces zijn het socket interface onder UNIX en het winsock interface onder Microsoft Windows.

Real interfaces spelen echter een rol bij conformance testing: de bepaling of de implementatie van een protocol (entity) wel of niet aan de specificatie voldoet. Voordat tot implementatie van een protocol entity kan worden overgegaan moet een keuze worden gemaakt voor de real interfaces waartussen de protocol entity is ingeklemd. De implementaties van deze interfaces worden, afhankelijk van de test-methode, vaak impliciet in de test betrokken.

### 2.1.6 Connection-oriented en connectionless services

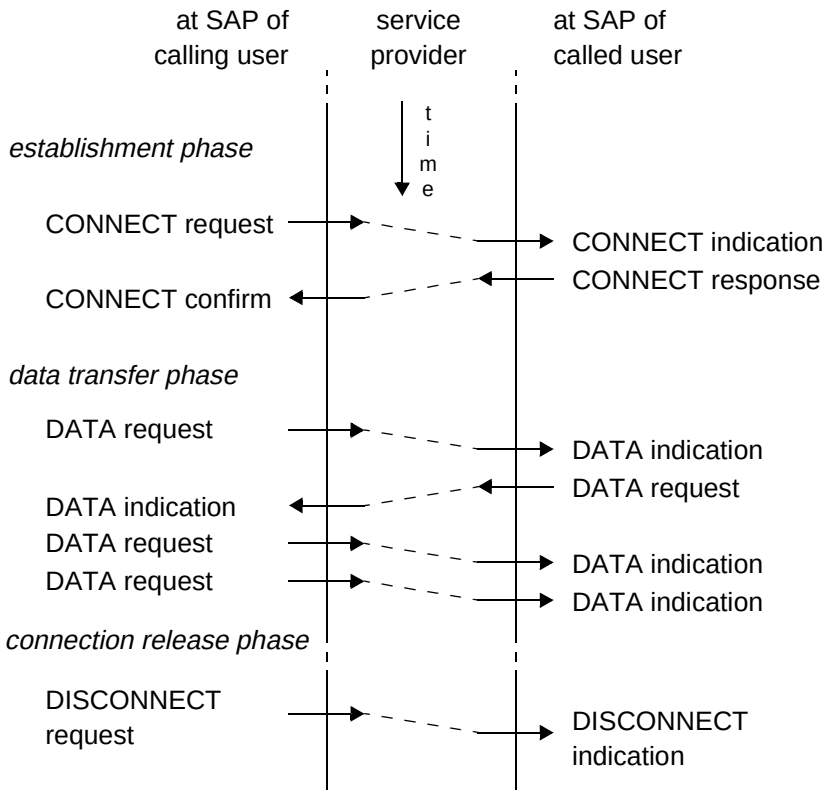
De netwerklaag en de hogere protocollagen in het OSI-RM leveren in het algemeen twee typen services: een *connection-oriented* en een *connectionless* service. Hieronder wordt uitgegaan van de situatie waarin een gebruiker wil communiceren met één andere gebruiker. De gebruiker die het initiatief neemt wordt de *calling user* genoemd en de andere wordt de *called user* genoemd.

*Connection-oriented service*

Bij de *connection-oriented service* zijn beide service-gebruikers en de service provider betrokken in een drie partijen overeenkomst. Deze overeenkomst komt alleen tot stand indien alle partijen het eens zijn omtrent de condities waaronder de gegevens zullen worden uitgewisseld. De connection-oriented service kent een drietal fasen (zie Figuur 2.4):

- de *connection establishment phase* dient voor het opbouwen van de verbinding. Hierin wordt onder andere onderhandeld over de kwaliteit van de verbinding, bijv. *delay* en *throughput*, en worden bepaalde middelen gereserveerd, bijv. bufferruimte en processing capaciteit, die nodig zijn om deze kwaliteit te leveren;
- de *data transfer phase* dient voor de overdracht van gegevens. De volgorde waarin service data units worden ontvangen is gelijk aan de volgorde waarin ze zijn verstuurd;
- de *connection release phase* dient om de verbinding af te breken. Meestal zal een van de gebruikers het initiatief nemen om de verbinding te verbreken. De service provider kan dit

echter ook doen, bijv. in de situatie dat de afgesproken kwaliteit van de verbinding niet langer geleverd kan worden.



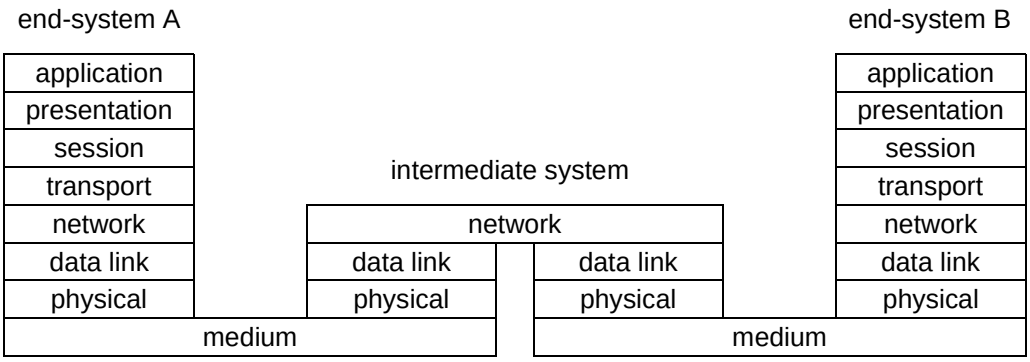

*Figuur 2.4: Voorbeeld van de executie van een connection-oriented service in termen van de service primitiva en hun volgorde relaties*

*Connectionless service*

In de *connectionless service* vindt tussen de drie partijen geen onderhandeling plaats omtrent de condities waaronder de service data units zullen worden uitgewisseld. Iedere uitwisseling van service data units wordt onafhankelijk van overige uitwisselingen beschouwd. De volgorde waarin service data units worden ontvangen hoeft niet gelijk te zijn aan die waarin ze zijn verstuurd. De connectionless service is veel eenvoudiger dan de connection-oriented service en maakt alleen gebruik van data service primitives (zie Figuur 2.4).

## 2.2 OSI model

Figuur 2.5 illustreert de verticale en horizontale structuur van het ISO OSI Reference Model. OSI staat voor Open Systems Interconnection. Met deze term wordt bedoeld: een stelsel van onderling - via netwerken - gekoppelde (computer-)systemen waarvan de services en protocollen zijn gebaseerd op open standaarden. Het staat daarnaast toe dat applicaties (toepassingsactiviteiten) toegankelijk (open) kunnen zijn voor applicaties van andere computersystemen.




*Figuur 2.5: ISO OSI Reference Model*

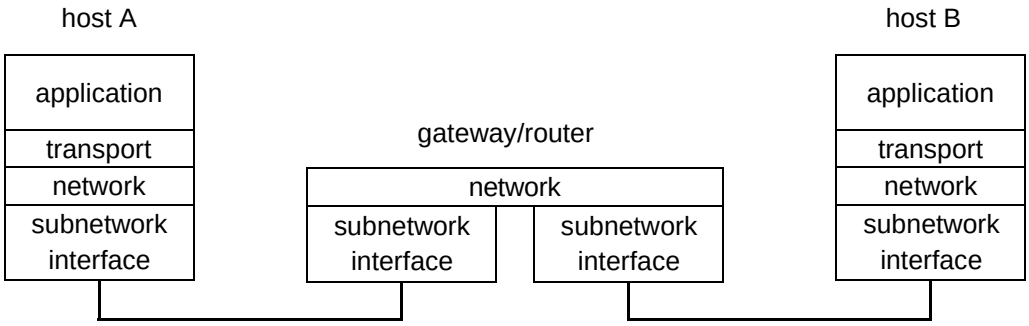
De volgende protocollagen worden in het OSI-RM onderscheiden:

- de *application layer* ondersteunt informatieverwerkende functies van de eindgebruiker direct en zonder tussenkomst van andere lagen. Om deze reden kent deze laag, anders dan bij de onderliggende lagen, geen service access points naar hoger liggende lagen;
- de *presentation layer* verschaft aan application entities middelen voor het overeenkomen van een presentatie-context. Deze context wordt gebruikt om informatie die wordt uitgewisseld eenduidig te kunnen interpreteren, onafhankelijk van de representatie van de informatie. De presentatie-service levert naast deze middelen ook de services die deel uitmaken van de sessie-service;
- de *session layer* verschaft aan application entities de middelen voor een orderlijke en gesynchroniseerde uitwisseling van gegevens. Er worden o.a. mogelijkheden geboden voor het definiëren van synchronisatie-punten (markeringen) in de gegevensstroom, het onderscheiden van verschillende logische eenheden van werk (zgn. activities) en het gebruik van tokens om het gebruik van de verschafte middelen te besturen;
- de *transport layer* heeft tot doel een geoptimalizeerde gegevensoverdracht service aan de session entities te leveren. De functies van de transportlaag optimaliseren de resources van de transportlaag zodanig dat de door de session entities gevraagde service kwaliteit wordt geleverd tegen minimale kosten;
- de *network layer* levert aan transport entities een zogeheten end-to-end gegevensoverdracht service. Dit houdt in dat de netwerklaag zorgt voor het aan elkaar koppelen van data-links en subnetwerken. Op deze manier wordt er een route voor de gegevens mogelijk gemaakt, zodat de gebruikers van de network service rechtstreeks met elkaar kunnen communiceren;
- de *data-link layer* levert een elementaire gegevensoverdracht service aan de network entities voor elk van de subnetwerken waaruit een netwerk is opgebouwd. Voorbeelden van subnetwerken zijn een token ring, FDDI of ISDN netwerk. Belangrijke data-link functies zijn foutdetectie, foutcorrectie en het bewaken van toegang (access) tot het medium, met name voor Local en Metropolitan Area Networks (LANs en MANs);
- de *physical layer* levert een elementaire service die niet veel anders doet dan een bitrij van de ene data-link entity naar een of meerdere andere data-link entities te vervoeren. De functies van de physical layer dienen daarbij om de logische representatie van de informatie (de bits) om te zetten in de fysische signalen zoals die door het medium kunnen worden getransporteerd;
- het *medium* levert een transmissie-service voor de overdracht van signalen. Voorbeelden van media zijn elektrische kabels, de ether en optische fibers. De medium service wordt als geheel geleverd, en niet, zoals bij alle andere (bovenliggende) services wordt gedaan, gestructureerd in protocol entities en een lager liggende service.

## 2.3 Internet model

Figuur 2.6 toont een vaak gehanteerde structuring voor Internet protocollen. De volgende protocollagen worden onderscheiden:

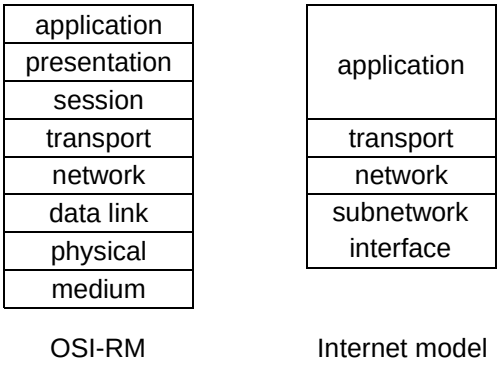
- de *application layer* bestaat uit de applicatieprotocollen die over een TCP/IP netwerk worden aangeboden, zoals DNS, SNMP, FTP, SMTP, NEWS, HTTP, HTML (zie par. 3.4);
- de *transport layer* levert een end-to-end gegevensoverdracht service tussen applicatieprocessen (applicatieprogramma's). Meerdere applicatieprocessen kunnen tegelijkertijd binnen een computersysteem draaien. Deze laag bestaat o.a. uit de protocollen TCP en UDP (zie par. 3.3);
- de *network layer* levert een end-to-end gegevensoverdracht service tussen computersystemen (zgn. hosts). Deze laag bestaat o.a. uit de protocollen IP, ICMP en verscheidene routeringsprotocollen (zie par. 3.2);
- de *subnetwork interface layer* bevat de nodig functies voor het versturen en ontvangen van network layer pakketjes via het lokale subnetwerk waarop een computersysteem is aangesloten. Voorbeelden van protocollen in deze laag zijn ISDN, xDSL, Ethernet, ATM, WDM (zie par. 3.1).




*Figuur 2.6: Internet model*

## 2.4 Vergelijking OSI model en Internet model

Figuur 2.7 illustreert de relatie tussen de functionaliteit van de lagen van het OSI-RM en van het Internet model.




*Figuur 2.7: Functionele relatie tussen de lagen van het OSI-RM en Internet model*

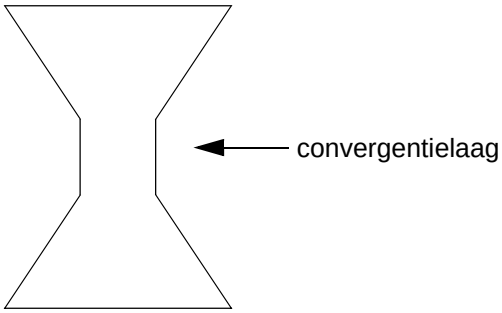
Uit de figuur valt ruwweg het volgende af te leiden: de applicatie laag van het Internet model combineert de functies van de bovenste drie (laag 7, 6 en 5) OSI lagen; de transport en netwerk laag komen ongeveer overeen in beide modellen (in OSI laag 4 en 3); de subnetwerk interface

laag van het Internet model verenigt de functies van de laag 3 en 2 van OSI; en de medium laag (OSI laag 1) komt niet aan bod in het Internet model.

**Zandloper principe**

Een grote diversiteit aan protocollen is gedefinieerd in de application layer van het Internet model om de grote variëteit aan gebruikerswensen te kunnen ondersteunen. Ook de subnetwork interface layer kent diverse protocollen voor de vele subnetwork technologiën die worden gebruikt. Daarentegen worden in de transport en network layers slechts een beperkt aantal protocollen gedefinieerd om end-to-end gegevensoverdracht mogelijk te maken.

Bovenstaande constatering geldt ook voor de overeenkomstige lagen in het OSI-RM. De mate van diversiteit aan protocollen in de verschillende lagen van het OSI and Internet model wordt geïllustreerd aan de hand van een zandloper in Figuur 2.8. De smalle hals wordt wel aangeduid als de convergentielaag, welke overeenkomt met de transport- en netwerklaag in het OSI en Internet model. De protocollen in de convergentielaag zijn stabiel, aangezien de gegevensuitwisseling tussen alle mogelijke applicaties door deze protocollen moeten worden afgehandeld.




*Figuur 2.8: Zandloper principe*

## 2.5 Distributed object computing

De beschikbaarheid van computernetwerken maakt het mogelijk om informatie op een gedistribueerde wijze te verwerken, d.w.z. door de inzet van computers op geografisch verspreide locaties. Een beoogd voordeel van gedistribueerde informatieverwerking (distributed computing) is het gemeenschappelijk gebruik van middelen, zoals verwerkingscapaciteit en informatiediensten. In IT management termen zou dit moeten leiden tot verbeterde integratie van bedrijfsinformatie, verminderde kosten en meer klantgerichte diensten.

Een belangrijke paradigma in de evolutie van gedistribueerde informatieverwerking is client-server gebaseerde informatieverwerking, of kortweg client-server computing. Volgens dit paradigma worden applicaties gesplitst in een relatief eenvoudig client deel en een meer complex server deel. Een client kan een server vragen om een bepaalde dienst te verlenen, gebruik makend van een vraag-antwoord mechanisme. Dit mechanisme wordt ook wel Remote Procedure Call (RPC) genoemd en moet worden ondersteund door een betrouwbare netwerk service. Gemeenschappelijk gebruik van diensten is mogelijk door server interfaces te standaardiseren. Hierdoor kan een server door meerdere clients benaderd worden.

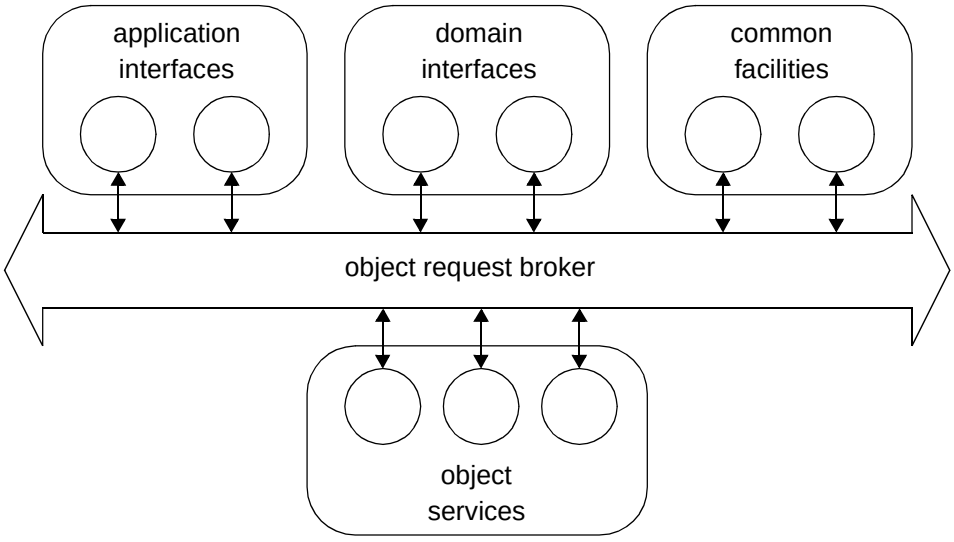
Een volgende stap in de evolutie van client-server computing is de verschuiving van peer-to-peer client-server computing naar volledig gedistribueerde samenwerking tussen componenten, waarin componenten zowel een client als server rol kunnen vervullen.

Op methodologish gebied heeft object-georiënteerd (OO) ontwerpen en programmeren massaal zijn intrede gedaan in de afgelopen 10 jaar. De OO-benadering, ook wel aangeduid met de term object technology (OT), moest een oplossing bieden voor de software crises die was ontstaan doordat de complexiteit van software-ontwikkeling en -onderhoud (zeer) slecht was te controleren. OT draagt bij aan kostenreductie en kwaliteitsverbetering, door ondersteuning te bieden voor hergebruik, uitbreidbaarheid en modulariteit van software.

Distributed computing en object technology zijn goed verenigbaar. Gedistribueerde systemen werden altijd al gemodelleerd in object-achtige termen (componenten en entiteiten met interfaces en services die de interne structuur verbergen), maar OT leverde een algemene methodologie met concrete ondersteuning (i.h.b. programmeertalen). Het gecombineerd gebruik van distributed computing en object technology wordt ook wel aangeduid met de term *distributed object computing* (DOC). De verwachting dat DOC het meest toonaangevende paradigma vormt voor software ontwikkeling in de toekomst, heeft geleid tot de oprichting van de Object Management Groep (OMG). OMG is een consortium bestaande uit fabrikanten van informatiesystemen, gebruikers en universiteiten, die specificaties ontwikkelen voor de ontwikkeling van software. Een belangrijke specificatie ontwikkeld door OMG is het *Object Management Architecture* (OMA) referentiemodel. Dit referentiemodel wordt gezien als een toonaangevende representant van huidige DOC modellen.

### 2.5.1 OMA reference model

Figuur 2.9 illustreert het OMA referentiemodel.




*Figuur 2.9: OMA referentiemodel (interface categorieën)*

Een gedistribueerd systeem wordt gemodelleerd als een verzameling objecten, mogelijk gedistribueerd over geografisch verspreide computersystemen, die samenwerken via een zgn. Object Request Broker (ORB). Een ORB maakt interactie tussen objecten mogelijk, waarbij voor elk object de locatie van andere objecten en de eventueel benodigde communicatiemiddelen transparant is. Een ORB geeft het idee alsof objecten worden geëxecuteerd binnen een virtueel homogeen computersysteem en verbergt de heterogeniteit van de werkelijke computersystemen en computernetwerken waaruit dit virtuele systeem is opgebouwd. Objecten hebben alleen een referentie naar elkaars interfaces nodig om interactie mogelijk te maken. Objecten mogen in een willekeurige programmeertaal worden geïmplementeerd, mits hun interfaces worden gespecifieerd in een standaard taal, de *Interface Definition Language* (IDL), en een standaard afbee-

lding van IDL naar de betreffende programmeertaal is gedefinieerd door OMG. De architectuur van een ORB is gespecifieerd in de *Common Object Request Broker Architecture* (CORBA).

Een object biedt een bepaalde dienst aan via zijn interface. De volgende vier typen interfaces worden onderscheiden, gerangschikt naar algemeenheid:

- *object services*: bieden algemene diensten voor willekeurige objecten, met name functies betreffende de levensloop van objecten en de interactie tussen objecten;
- *common facilities*: bieden algemene diensten voor applicaties binnen vele toepassingsgebieden;
- *domain interfaces*: bieden algemene diensten voor applicaties binnen een bepaald toepassingsgebied;
- *application interfaces*: bieden diensten gerelateerd aan een specifieke applicatie.

# 3 Protocollen

## 3.1 Subnetwerk niveau

De subnetwerk laag heeft tot taak connectiviteit te leveren tussen computers die rechtstreeks met elkaar verbonden zijn via een fysiek medium. Netwerken op subnetwerk niveau onderscheiden zich in twee soorten:

- netwerken waarin de computers rechtsstreeks aangesloten zijn op een fysisch medium dat gemeenschappelijk gebruikt wordt door alle computers samen: l*ocal area netwerken* (LANs), en
- netwerken die zich gedragen als een draad waarover bits verstuurd worden (een bit-pijp, meestal een punt-naar-punt verbinding) en die verzorgd worden door de publieke netwerk operators: *wide area netwerken* (WANs).

### 3.1.1 Local Area Netwerken

Door een LAN wordt een data link service aangeboden. Deze service kan zowel van het connection oriented als ook van het connection-less type zijn.

Het doel van een datalink laag service provider is om data eenheden van eindige lengte, frames genaamd, van zendende service user naar ontvangende service user te transporteren. Hierbij wordt eliminatie van transmissiefouten nagestreefd en wordt er voor gezorgd dat de de frames in dezelfde volgorde doorgegeven worden aan de ontvangende service user als dat de zendende service user ze aangeboden heeft aan de service provider (*sequence integrity*).

De protocolfuncties die door een datalink laag protocol uitgevoerd kunnen worden zijn:

**Framing**

De bits die getransporteerd worden, worden op de datalink laag groepsgewijs samengevoegd tot data eenheden die *frames* genoemd worden. Het gebruik van deze eindige lengte data eenheden maakt het mogelijk de overige protocolfuncties op de datalink laag op relatief eenvoudige wijze uit te voeren.

**Foutafhandeling**

In het geval er transmissiefouten optreden in de fysieke laag service provider staan de ontvangende datalink laag entiteit de volgende opties ter beschikking:

- detecteer de fout in het ontvangen frame;
- laat het daarbij (genereer geen service primitive) en gooi het frame weg;
- genereer een indicatie voor de service user waarin de fout gemeldt wordt;
- corrigeer de fout en genereer een indicatie voor de service user (geef het frame door);
- vraag om een hertransmissie van het frame met de fout of wacht totdat de zendende entiteit het frame nogmaals verstuurd.

**Foutdetectie**

Bij alle service typen probeert het datalink protocol transmissiefouten te detecteren. De detectie is meestal gebaseerd op een *Cyclic Redundancy Check* (CRC). In de zendende protocol entiteit wordt op het informatiewoord een berekening uitgevoerd (verschuiven van de bits gevolgd door de bepaling van de rest na deling door een zogenaamd generator woord) die zich handig laat beschrijven door het informatiewoord op te vatten als een polynoom. Het door deze berekening ontstane codewoord is deelbaar door het generatorwoord. Indien er fouten optreden in de transmissie van het codewoord, ontstaat er bij bepaling van de rest na deling aan de ontvangende kant met grote waarschijnlijkheid een woord ongelijk aan het nul-woord, hetgeen de fout detecteerd.

**Foutcorrectie**

Bij de connectie georienteerde data-link protocollen wordt geprobeerd de gedetecteerde fouten te herstellen. Dit kan gebeuren doordat de ontvangende protocol entiteit wacht totdat de zendende protocol entiteit het bericht opnieuw verstuurd (door het verstrijken van de time-out periode van een uitstaand frame), of door explicitiet om hertransmissie te vragen. Dergelijke protocol mechanismen worden wel aangeduid met *Automatic Repeat reQuest* (ARQ).

In die gevallen waarbij ofwel het fysiek verzenden van bits veel energie kost (bijvoorbeeld bij satelliet verbindingen) ofwel dat variatie in de verzend-duur van frames ongewenst is (bijvoorbeeld bij interactieve real-time toepassingen) wordt gebruik gemaakt van fout corrigerende codes (*forward error correction*, FEC). Bij de connectionless datalink protocollen zulle frames met fouten in het algemeen worden weggegooid. Hierop zal in de hogere protocollagen adequaat gereageerd moeten worden.

**Flow control**

De zendende datalink entiteit kan, bijvoorbeeld bij een groot aanbod van frames door de zendende service user, in staat zijn de ontvangende data link entiteit te overspoelen (bij trage afname van de ontvangende service user). Om dit te voorkomen kan er een *flow control* functie geimplementeerd worden in de datalink laag. Hierbij wordt het mogelijk gemaakt dat de ontvangende entiteit de zendende entiteit afremt, bijvoorbeeld door Start/Stop berichten te verzenden of door gebruik te maken van het sliding window mechanisme (zie paragraaf 3.3).

**Toegangscontrole**

Een van de belangrijkste datalink functies is het bewaken van de toegang tot het gedeelde fysieke medium. Een ruwe classificatie van toegangsmechanismen is gebaseerd op het al dan niet bestaan van een hierarchische relatie tussen de stations op het medium:

- Gecentraliseerde Toegangsmechanismen. Hierbij is sprake van een hierarchische relatie. Er is een station (de master) die de toegang van de andere stations (de slaves) tot het medium

regelt. Dit mechanisme kan berusten op het regelmatig ondervragen (polling) van ieder station of er een bericht moet worden verstuurd of op het zich laten aanmelden van een station dat wil zenden (*arbitration*).
- Gedistribueerde of Symmetrische Toegangsmechanismen. Hier is geen hierarchische relatie tussen de stations aanwezig. Ieder station bevat dezelfde toegangscontrole functie. Wel kan een van de stations een speciale controletaak toegewezen krijgen.

De toegangsmechanismen tot het gedeelde fysische medium zijn sterk afhankelijk van de topologie en geografisch bereik dat een medium heeft. Het Institute of Electrical and Electronic Engineers (IEEE) heeft in haar IEEE802 standaarden een aantal gedistribueerde toegangsmechanismen voor lokale netwerken vastgelegd. Twee hiervan worden hieronder besproken.

*CSMA/CD, Ethernet en Fast Ethernet*

De afkorting CSMA/CD staat voor *Carrier Sense Multiple Access/Collision Detect*. Ieder station dat een bericht wil versturen kijkt eerst of een ander station al aan het zenden is (carrier sense). Is het fysieke medium nog niet bezet, dan begint het station met het versturen van het bericht over het medium. Het mechanisme geeft aanleiding tot het wederzijds verminken (collisions) van berichten wanneer toevalling twee stations gelijktijdig een bericht versturen. Dergelijke collisions zullen door de zendende stations worden gedetecteerd, waarna deze het versturen van de berichten ontbreken. Iedere zender gaat vervolgens een willekeurig lange periode wachten, om het daarna nog eens te proberen. Indien een frame te vaak (meer dan 16 maal) collisions veroorzaakt, zal de zender geen verdere pogingen meer ondernemen dit bericht te versturen.

CSMA/CD is vastgelegd in de IEEE 802.3 standaard; *Ethernet* was een voorloper van deze standaard, waarvan de naam echter ook als losweg-gehanteerde aanduiding voor de 802.3 standaard is blijven voortleven.

In de IEEE802.x standaarden wordt de data link laag opgedeeld in twee sub-lagen:

- De *Logical Link Control* (LLC) *Laag*, waarin ondermeer de Flow control functie, de Fout afhandelings functie en de service primitive afhandelings functie geplaatst zijn.
- De *Medium Access* (MAC) laag, met onder meer de access control functie en de framing functie.

In de IEEE 802.3 standaard wordt een MAC laag beschreven met een (tussen alle stations gezamenlijk) gedeelde link rate van 10 Mb/s. Er zijn meerdere fysieke laag specificaties mogelijk, aangeduid met (bijvoorbeeld) 10Base2, 10Base5, 10BaseT of 10Broad36. Hierin geeft de 10 de link rate in Mb/s, Base duidt het modulatieschema aan (Baseband of Broadband) en het laatste cijfer de maximale segment lengte ([100m]) of de gebruikte kabelsoort (T staat voor Twisted Pair - de welbekende telefoondraad). De meest gebruikte ethernetten zijn op dit moment van het 10BaseT type. Hierin is het gedeelde fysieke medium veelal vervangen door een ethernet switch (een hub), die door middel van twisted pair kabels in een ster topologie verbonden zijn met de aangesloten computers. De afstand die overbrugd wordt tussen hub en stations bedraagt ongeveer 100 m.

In 1992 is de 802.3 werkgroep van de IEEE begonnen aan het moderniseren van de standaard. Het doel hierbij was om een tussen de aangesloten stations gedeelde link rate van 100Mb/s te realiseren. Belangrijke randvoorwaarden aan het ontwerp was het zorg dragen voor backward compatibility: behoudt het protocol mechanisme en de frame formaten, behoudt het extern

observeerbare gedrag (de service) en de interface en zorg ervoor dat de bestaande bekabeling gebruikt kan worden (veelal dus met een afstand van maximaal 100 meter tussen hub en station).

In 1995 heeft dit geleid tot specificatie van 100BaseT standaard (IEEE802.3u). Feitelijk zijn er twee fysieke laag specifikaties gemaakt: de 100Base4T standaard voor gebruik met "spraak kwaliteit" *unshielded twisted pair* (UTP, cat 3) en de 100BaseX voor gebruik met "hoge kwaliteit" unshielded twisted pair (UTP, cat5), *Shielded Twisted Pair* (STP) of glasfiber.

*Token Ring en FDDI*

Bij token-ring zijn de stations door middel van een ringtopologie (zie paragraaf 1.2) onderling verbonden. Stations krijgen om beurten een token toebedeeld. Het token is feitelijk een beurt waarin het station dat het token bezit toegang kan nemen tot de ring. Na het versturen van een bericht, of wanneer het station geen bericht te versturen heeft, geeft het station de beeurt door naar het volgende station in de ring door een special token-boodschap. Aan het token kan een prioriteit worden toegekend; de ontvanger mag het token alleen gebruiken als het een bericht met een gelijke of hogere prioritiet wil versturen. Het token-ring access mechanisme en een fysieke laag specificatie zijn vastgelegd in de IEEE802.5 standaard. De tussen alle stations gedeelde link rate is 4 of 16 Mb/s.

Aan het einde van de jaren tachtig is de Fiber Distributed Data Interface (FDDI) LAN standaaard gespecificeerd door ANSI (American National Standards Institute). Het is nu een internationale standaard ISO9314. In een FDDI LAN zijn de stations door middel van een ring topologie verbonden en kunnen gebruik maken van een tussen alle stations gedeelde link rate van 100 Mb/s. Om de betrouwbaarheid (reliability) van het LAN te vergroten wordt er gebruik gemaakt van twee ringen, waarin de tokens in tegengestelde richting roteren. Indien er een kabelbreuk optreed kan een nieuwe ring gemaakt worden door de twee stations aan weerszijden van de breuk een token te laten teruglussen. De stations zijn verbonden door multi-mode glasfiber, waarbij de ring een totale lengte kan hebben van ongeveer 100 km.

Het is mogelijk om garanties te bieden aan de maximale access delay op het netwerk zodat delay-gevoelig verkeer verzonden kan worden.

FDDI wordt soms gebruikt als LAN backbone om meerdere LANs, verspreid over een grotere afstand dan die welke past bij één gebouw, met elkaar te verbinden. Het gebruik van FDDI heeft niet zo'n hoge vlucht genomen als in de ontwerpfase ervan werd verwacht, mede gezien de prijsstelling en door de opkomst van op ATM gebaseerde lokale en wide area netwerken (zie paragraaf 3.1.2).

### 3.1.2 Wide Area Netwerken

Terwijl local area netwerken de communicatie tussen meerdere computers in een begrensd geografisch gebied mogelijk maken, zijn er situaties waarbij er over grotere afstanden (een stad, een land) tussen één of meer computers gecommuniceerd dient te worden. Men zal dan gebruik maken van de faciliteiten (de services of diensten) die een *publieke netwerk operator* (PNO) biedt. De dienst die zorg draagt voor het transport van gegevens wordt door PNO's wel aangeduid als *dragerdienst* (*bearer service*).

Voor het verbinden van twee computers op grote afstand kan men gebruik maken van het publieke analoge telefoonnetwerk (*Public Switched Telephone Network*, PSTN) en een modulator/demodulator (*modem*). Deze modulator/demodulator zorgt voor de verbindingsopbouw en

voor de omzetting van bits naar analoge signalen die verstuurd kunnen worden over het spraakkanaal.

Indien men meerdere computers, bijvoorbeeld van bedrijfsvestigingen op meerdere locaties, met elkaar wil verbinden kan men zijn toevlucht nemen tot huurlijnen van de PNO's of gebruik maken van de faciliteiten van een publiek datanetwerk (*Public Switched Data Network*, PSDN). PNO's bieden met hun datanetwerken als platform verschillende dragerdiensten voor data communicatie aan zoals (het verouderde) *X.25*, *Frame Relay* (FR) en *Switched Multi-megabyte Data Services* (SMDS).

De PNO's hebben in de loop der tijd de beschikking gekregen over meerdere separate netwerken voor spraak, telex en dataverkeer. In de jaren 80 is er een ontwikkeling in gang gezet om deze netwerken te integreren tot een *Integrated Services Digital Network* (ISDN). Op dit netwerk kan een computer rechtstreeks via een adapterkaart aangesloten worden.

De bitrates die geleverd konden worden door ISDN netwerken werden in het algemeen te laag bevonden voor hoge bit-rate computertoepassingen en voor interactieve video toepassingen. Verder zijn de bit rates in ISDN netwerken erg inflexibel toe te wijzen aan individuele gebruikers en kan een ISDN netwerk niet goed overweg met het sterk wisselende verkeersaanbod dat gegenereerd wordt door computers. Dit heeft er toe geleid dat de PNOs onderzoek en ontwikkeling van Broadband Integrated Services Digital Networks (B-ISDNs) in gang hebben gezet. Deze architectuur is en wordt vastgelegd in standaarden, *recommendations* genaamd, die vastgelegd worden door de *ITU - Telecommunication Standardization Section* (ITU-T).

In de netwerken die gebouwd worden volgens de B-ISDN architectuur wordt een manier van pakket schakelen gebruikt die aangeduidt wordt met *Asynchronous Transfer Mode* (ATM). Hierin is 'transfer mode' ITU-T jargon voor 'pakketschakelmethode' en duidt op een methode waarbij kleine pakketjes van vaste lengte, cellen genaamd, geschakeld worden.

De gebruikers van het publiek net (de telefoontoestellen, terminals of computers) zijn aangesloten op een *toegangsnetwerk* (access netwerk) terwijl de toegangsnetwerken met elkaar verbonden zijn door het *kernnetwerk* (transport netwerk of core netwerk). Voor al deze netwerken worden in het algemeen verschillende technologieen gebruikt en hebben gebruikers die datacommunicatie willen bedrijven verschillende middelen (apparaten) nodig om van de door de telecom operator aangeboden dienst gebruik te maken. Behalve met ISDN en B-ISDN oplossingen is in de afgelopen jarendoor PNOs gepoogd om zowel de bitrates voor de toegangsnetwerken als ook voor de kernnetwerken hoger te maken. Dit heeft geleid tot ontwikkeling van de *Assymmetric Digital Subscriber Line* (ADSL) technologie voor toegangsnetwerken en tot *Wave Division Multiplexing* (WDM) gebaseerde kernnetwerken.

Hieronder worden enkele van bovengenoemde technologieen, netwerken, protocollen en diensten nader besproken.

**Modem standaarden**

In modems wordt gebruik gemaakt van digitale modulatie schemas om de te verzenden bits te vertalen naar analoge signaalvormen die over het fysieke medium (veelal de analoge telefoonlijn) verstuurd kunnen worden. Ook wordt er soms gebruik gemaakt van fout herstellende

mechanismen. De ITU-T heeft in de V-serie recommendations modemstandaarden opgenomen. Enkele daarvan zijn:

- V.32, link rate 9.6 Kb/s;
- V.32bis, link rate 14.4 Kb/s;
- V.34, link rates 28.8 Kb/s en 33.6 Kb/s.

Een nieuwe modemstandaard is V.90, waarbij downstream (van het PSTN naar de gebruiker) 56 Kb/s aangeboden wordt, terwijl er upstream 33.6 Kb/s mogelijk is.

Behalve modems voor analoge telefonie, zijn er in de afgelopen jaren ook modems voor het kabel TV netwerk ontwikkeld. In de IEEE802.14 werkgroep worden standaarden ontwikkeld voor transmissie en multiple access gebruik makend van de kabel infrastructuur.

**Publieke Data Services**

Enkele dragerdiensten voor *dataverkeer* die aangeboden kunnen worden door PNO's zijn:

- *Switched Multimegabit Data Service* (SMDS)

SMDS is geschikt voor interconnectie van LANs in bedrijfsvestigingen op grote afstand van elkaar. SMDS biedt een connection less service. De tussen de gebruikers gedeelde doorvoercapaciteit kan varieren van 2Mb/s tot 155Mb/s. Er wordt gebruik gemaakt van pakketschakeling met een grote maximale pakketgrootte en er bestaat de mogelijkheid van broadcast transmissie en groepsadressering. De doorvoer van verzonden frames is niet gegarandeerd, zodat hogere laag protocollen hiermee rekening moeten houden.

- *X.25*

Een in de jaren 70 door de CCITT ontwikkelde standaard voor *toegang* tot het publieke datanetwerken. De standaard specificeert de interface tussen computers en publieke net en specificeert de laag 1, 2 en 3 protocollen. Er wordt gebruik gemaakt van pakketschakeling waarbij foutafhandeling op een per link basis plaatsvindt. De service is connectie georienteerd waarbij, link rates tot 64 Kb/s aangeboden worden. X.25 is naar huidige maatstaven verouderd.

- *Frame Relay* (FR)

Frame relay is een connectie georienteerd dienst, waarbij een virtuele huurlijn aan de gebruiker aangeboden wordt. Terwijl bij een huurlijn de gebruiker gedurende de hele periode van huur data op de link rate mag verzenden, mag dit bij de virtuele huurlijn niet en moet de hoeveelheid gemiddeld aangeboden verkeer beneden een afgesproken niveau blijven. De doorvoer van paketten is dus maar voor een gedeelte gegarandeerd. Er wordt gebruik gemaakt van pakketschakeling waarbij ten opzichte van X.25 een aantal protocolfuncites zoals foutafhandeling per link ontbreken. De service is beperkt tot het aangeven van begin en einde van een frame en een indicatie van transmissiefouten. De link rates kunnen oplopen van 64Kb/s tot 2 Mb/s en 34Mb/s.

**ISDN Netwerken**

De wens van de PNO's om tot integratie van hun netwerken te komen heeft in de jaren tachtig geleid to de ontwikkeling van Integrated Services Digital Networks (ISDNs). De netwerken zijn gebaseerd op een volledig digitaal netwerk waarin digitale transmissie gebruikt wordt voor bit transport en een apart netwerk voor verbindingsopbouw en -afbraak, het *Common Channel Signalling* (CCS) netwerk.

De interface die standaard aangeboden wordt (*basic access* genaamd) is een full-duplex digitale verbinding bestaande uit twee 64 Kbit/s verbindingen (de B kanalen ) en één 16 Kbit/s voor signalleringsinformatie (het D-kanaal). De B-kanalen worden gebruikt voor de transmissie van spraak of dataverkeer waarbij er bitfouten op het kanaal kunnen optreden. Over het D-kanaal wordt (onder andere) de informatie verzonden voor verbindingsopbouw en afbraak, het signalleringsverkeer. Tegenwoordig spreekt men van *Narrowband ISDN* (N-ISDN).

Hogere link rates zijn echter ook mogelijk, bijvoorbeeld link rates in stapjes van 64 Kbit/s oplopend tot 2 Mb/s, link rates als veelvoud van 2 Mb/s etc. De mogelijk aan te bieden link rates volgen de rates, gestandaardiseerd in de zogenaamde *Synchronous Digital Hierarchy* (SDH) en zijn dus beperkt af te stemmen op de bit rate wensen van de gebruikersapplicaties. Alle link rates liggen vast gedurende de duur van de verbinding en passen zich niet aan aan het verkeersaanbod. Verder is de apparatuur die gebruikt wordt om in SDH netwerken verbindingen met verschillende link rates bij elkaar te voegen en uit elkaar te halen (de zogenaamde *add-drop multiplexers*) erg kostbaar. Deze eigenschappen zijn er mede de oorzaak van dat er gezocht is naar meer flexibiliteit in het allokeren van bitrates voor gebruikersverbindingen.

De huidige populariteit van ISDN aansluitingen wordt vooral veroorzaakt doordat de link rate voor computertoepassingen net iets hoger is dan de link rate die door analoge modems aangeboden wordt en (waarschijnlijk meer belangrijk) doordat er gelijktijdig telefonie en Internet Access mogelijk is. De huidige trend is dat er gezocht wordt naar breedbandige toegangs netwerken waarmee de ISDN basic access capability overtroffen gaat worden.

**B-ISDN en ATM**

De door de ITU-T in gang gezette ontwikkeling van de Broadband Integrated Digital Networks heeft eind jaren ’80 geleid heeft tot het vastleggen van de B-ISDN architectuur in de I series Recommendations. Een belangrijke keuze die daarbij gemaakt moest worden is welke vorm van pakketschakelen men zou gaan gebruiken voor toekomstige publieke op B-ISDN gebaseerde netwerken. Deze keuze heeft geleid tot het gebruik van 53 byte grote pakketten, cellen genaamd, als eenheid van schakelen en multiplexen in het netwerk. Het is deze manier van pakketschakelen en multiplexen die de *Asynchronuous Transfer Mode* (ATM) netwerken hun naam geven.

Begin jaren ’90 ontstond er interesse vanuit de data-communicatiewereld om niet-publieke op ATM gebaseerde netwerktechnologie aan te bieden. Op dat moment leek het mogelijk om op vrij korte termijn een op ATM gebaseerd *LAN* met link rates van 155Mb/s te implementeren. Hiertoe is door een aantal datacom fabrikanten het *ATM Forum* opgericht dat tot doel heeft om ATM LANs te standaardiseren en producten hiervoor te ontwikkelen. Deze netwerken zijn LANs in de zin van dat ze veelal privé-eigendom van een organisatie zijn en dat ze in de TCP/IP context gebruikt worden op subnetwerk niveau. In principe echter zijn het volledig pakket geschakelde netwerken met een wereldwijde interconnectie potentie.

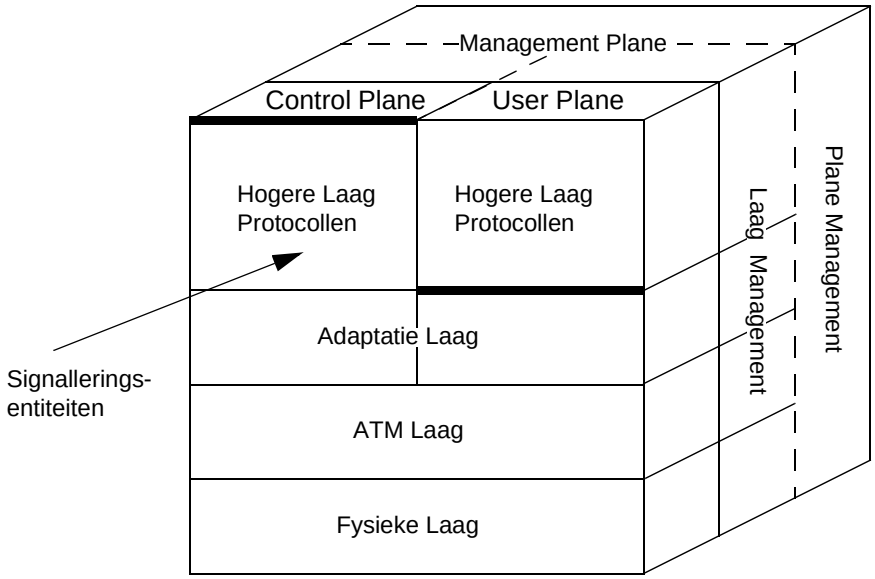
Het standaardisatieproces van op ATM gebaseerde netwerken is in de jaren negentig het resultaat geweest van samenwerking tussen de ITU-T en het ATM Forum, zodat niet alleen de I series recommendations van belang zijn maar ook de ATM Forum *implementation agreements* zoals ze genoemd worden.

De service, aangeboden door een op ATM gebaseerd netwerk, is connectie georienteerd. In het onderstaande zullen we deze service aanduiden met *ATM Bearer Service*.

Met als uitgangspunt het B-ISDN protocol referentie model wordt deze service geconstrueerd, waardoor het duidelijk zal worden wat de service, protocollen en interfaces van deze netwerken zijn. Vervolgens wordt een voorbeeld van het gedrag van de ATM bearer service provider getoond.

*B-ISDN Protocol Referentie Model*

In Figuur 3.1 is het B-ISDN *Protocol Referentie Model* (PRM) weergegeven.




*Figuur 3.1: B-ISDN Protocol Reference Model*

Het model is te gebruiken om protocol functionaliteit te structureren. Het bestaat uit drie zogenaamde planes en vier protocol lagen. De planes representeren de protocolfunctionaliteit die men aantreft in de drie netwerken die door PNO's gebruikt worden om hun services te bieden:

- een *user netwerk*, gebruikt voor data en spraakverkeer van de gebruikers van het publieke netwerk,
- een *controle netwerk* voor transport van verbindingsopbouw, -modificatie en verbindingsafbraak informatie (de *signalleringsberichten*), en
- een *management netwerk*, gebruikt om het publiek net operationeel te houden.

De vier protocol lagen zijn:

- *Fysieke Laag*

De fysieke laag netwerken, bijvoorbeeld Synchronous Digital Hierarchy (SDH) of Synchronous Optical Network (SONET) netwerken, leveren een connectie georienteerde datalink laag service in de vorm van punt naar punt verbindingen tussen ATM schakelaars (met een vaste link rate). De fysical laag in het B-ISDN PRM ligt dus *niet* op het niveau van de fysieke laag van het OSI model!

Er zijn veel fysieke laag interfaces in de loop der tijd ontstaan, veelal aangeduid met lastig te onthouden acroniemen. Enkele fysieke laag interfaces zijn in Tabel 3.1 opgenomen.

| Interface | Link Rate [Mb/s] | Medium |
|---|---|---|
| DS1 (T1, primary rate USA | 1.544 | Coax |
| E1, primary rate Europa | 2.048 | Coax |
| SDH STM1 (SONET STS3c) | 155.52 | Single Mode Fibre |
| SDH STM4c (SONET STS12c) | 622.08 | Single Mode Fibre |

*Tabel 3.1: Enkele fysieke laag interfaces*

• *ATM Laag*

Deze protocollaag implementeert een connectie georienteerde pakket schakel functie (*forwarding* of *switching* functie). De ATM cellen worden in een ATM schakelaar naar een output poort gedirigeerd, afhankelijk van de inhoud van forwarding tabellen in de schakelaar en de header informatie van de cellen. De forwarding tabellen worden ofwel door het control netwerk (via het signalleringssysteem) ofwel door het management netwerk van de juiste informatie voorzien. De verbindingsopbouw vindt dus *niet* plaats door functionaliteit in de ATM laag zelf. In de ATM laag zelf vind ook geen routering plaats, deze wordt verzorgd door functionaliteit in de control plane (ofwel door functionaliteit in de management plane).

De connectie terminologie in ATM netwerken is nogal verwarrend. Voor end-to-end connecties (van ATM bearer SAP naar ATM bearer SAP) hanteert men het begrip *Virtual Channel Connection* (VCC). Deze kunnen zowel door het signaleringssyteem opgezet zijn (*Switched Virtual Circuit* (SVC)) als ook door het management systeem (*Permanent Virtual Circuit* (PVC)). Daarnaast is er ook nog sprake van *Virtual Path Connections* (VPCs) die te beschouwen zijn als een bundel van Virtual Channels.

Pas nadat de forwarding tabellen gevuld zijn bestaat de VCC en kunnen er ATM cellen verzonden worden. De service die de ATM laag dan aanbiedt is bestaat uit het gedrag in de data fase van een connectie georienteerde service provider. De cellen komen in dezelfde volgorde aan als dat ze verzonden zijn, er kunnen cellen verloren gaan, fout afgeleverd zijn of gecorrumpeerd zijn.

• *ATM Adaptatie Laag* (AAL)

In de user plane vinden we AAL entiteiten alleen in de computers aangesloten op het netwerk. Het doel van deze laag is om in de eindsystemen extra functionaliteit aan te bieden die noodzakelijk is voor specifieke applicaties. Zo is er AAL1 voor het emuleren van circuits voor het verzenden van spraakverkeer. Deze adaptatielaag zorgt voor een tijdsrelatie tussen zendende en ontvangende computer. AAL3/4 en AAL5 worden gebruikt voor het verzenden van computerdata. De functionaliteit van deze laatste twee lagen is het best te beschrijven als een *segmentatie en reassembly* functie. Grote paketten worden door de zendende adaptatielaag entiteit in cellen opgedeeld en aan de ontvangende kant weer samengevoegd tot grote paketten. In de control plane treffen we adaptatielaag entiteiten *in* de schakelaars. Deze entiteiten implementeren de S*ignalling Adaptation Layer* (SAAL) waardoor een betrouwbare service voor het verzenden van signalleringsberichten over signallerings VCCs mogelijk wordt.

• *Hogere protocollagen*

In de user plane laat het B-ISDN PRM ruimte over voor applicaties en applicatieprotocollen die gebruik maken van de connectie die door de bearer service provider geboden wordt (bijvoorbeeld gedigitaliseerde spraak, audio of video). In de control plane moet men hier zich de signalleringsentiteiten in de schakelaars voorstellen die zorg dragen voor de verbindingsopbouw, -modificatie en -afbraak (feitelijk dus het vullen, veranderen en wissen van de forwarding tabellen in de schakelaars).

*Voorbeeldgedrag van de ATM Bearer Service Provider*

De ATM bearer service provider kan nu als volgt begrepen worden. Allereerst wordt er een verbinding op ATM niveau opgezet tussen twee ATM SAPs (een user VCC). Dit gebeurt door samenwerkend signalleringss entiteiten die via speciale signallerings VCCs signalleringsberichten uitwisselen. De route van de user VCC door het netwerk wordt door het signalleringssysteem bepaald en de forwarding tabellen van de ATM switches langs de gekozen route worden gevuld.

Door middel van een SetUp signallerings primitieve moet de gebruiker onder andere specificeren:

- met welke gebruiker(s) er gecommuniceerd moet worden.

Gebruikers worden geidentificeerd door 20 byte lange ATM adressen, te vergelijken met telefoonnummers.

- welke *Quality of Service* (QoS) parameters voor de connectie van belang zijn.

Enkele voorbeelden van parameters: end to end vertraging (*delay*), variabiliteit in delay (*Cell Delay Variation* (CDV)) en cell loss (*Cell Loss Ratio* (CLR)).
Niet voor alle applicaties is bijvoorbeeld delay variatie een relevante parameter. Voor dataverkeer is delay variatie niet relevant, voor spraakverkeer wel; voor videoverkeer is cell loss toe te staan, voor dataverkeer niet. De keuze van de relevante QoS parameters bepaalt de zogenaamde *ATM Transfer Capability* (in ITU-T terminologie) of *ATM Service Categorie* (volgens de ATM Forum terminologie). In Tabel 3.2 zijn enkele voorbeelden gegeven.

- wat de *verkeersparameters* van de connectie zijn.

Dese worden vastgelegd in de zogenaamde *Source Traffic Descriptor*. Een source traffic descriptor kan onder andere bestaan uit de *Peak Cell Rate* (PCR); een bovengrens aan de gemiddelde cell rate, genaamd *Sustainable Cell Rate* (SCR) en een *Maximum Burst Size* (MBS), de maximale tijdsduur waarin cellen met de peak cell rate verstuurd mogen worden. Afhankelijk van de gekozen transfer capability of transfer mode dienen meer of minder verkeersparameters door de gebruiker gespecificeerd te worden. Zie tabel 3.2.

- de *Quality of Service class* waartoe de connectie moet behoren.

Per service categorie kan een netwerk operator of ATM-LAN leverancier besluiten meerdere klassen van QoS parameter*waarden* aan te bieden. In een netwerk (of voor een dragerdienst) zijn bijvoorbeeld voor de CBR Service Category twee QoS Klassen gedefinieerd: QoS Class 1 met een Cell Loss Ratio van $10^{-10}$ voor de applicaties die een betrouwbare connectie nodig hebben en QoS Class 2 met een Cell Loss ratio van $10^{-6}$ voor applicaties die fouttoleranter zijn.

- de AAL adaptatielaag waarvan de gebruiker gebruik wenst te maken.

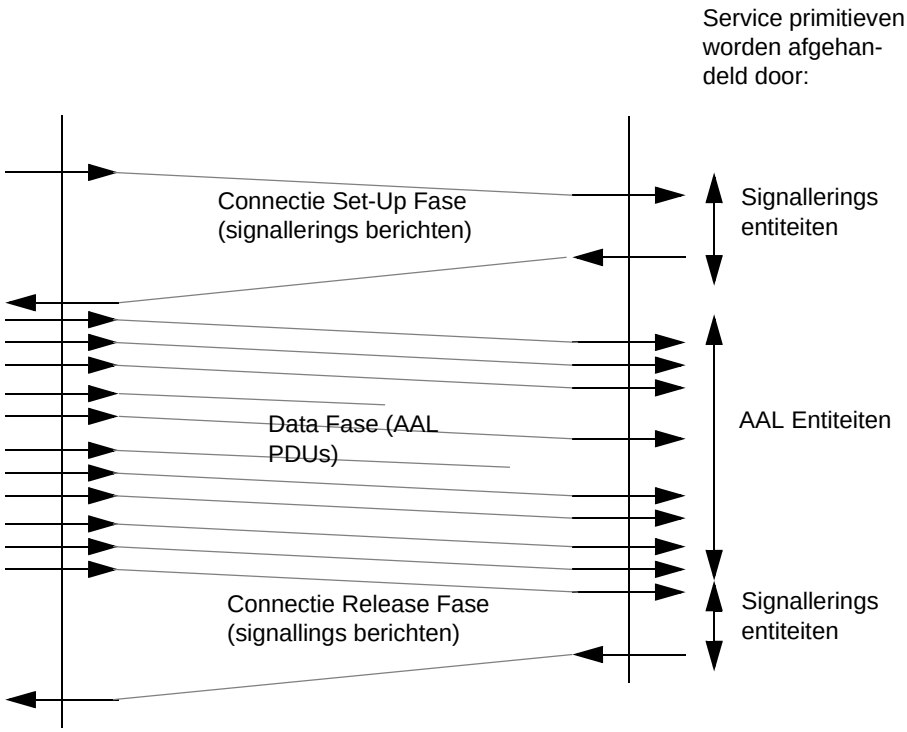
Behalve de al genoemde adaptatielagen (AAL1, AAL3/4 en AAL5) is het ook mogelijk voor de gebruiker om een AAL zonder functionalitiet te kiezen, de lege AAL of AAL0. De gebruiker heeft dan de mogelijkheid tot het versturen van ATM cellen.

| ATM Forum: ATM Service Category | ITU-T: ATM Transfer Capability | Source Traffic Descriptor | Relevante QoS Parameters | | |
|---|---|---|---|---|---|
| | | | CLR | Delay | CDV |
| Constant Bit Rate (CBR) | Deterministic Bit rate (DBR) | PCR | Ja | Ja | Ja |
| Real-Time Variable Bit rate (rt-VBR) | real-time Statistical Bit rate (rt-SBR) | PCR, SCR, MBS | Ja | Ja | Ja |
| Non Real-Time Variable Bit rate (nrt-VBR) | Non real-time Statistical Bit rate (nrt-SBR) | PCR, SCR, MBS | Ja | Nee | Nee |
| Available Bit Rate (ABR) | Available Bit Rate (ABR) | PCR en andere parameters | Ja | Nee | Nee |
| Unspecified Bit Rate (UBR) | Unspecified Bit Rate (UBR) | PCR | Nee | Nee | Nee |

*Tabel 3.2: Enkele ATM Service Categorieen en ATM Transfer Capabilities*

Nadat de verbinding, logisch gezien. op de ATM laag tot stand gebracht is worden in de zendende en ontvangende computer de juiste adaptatielagen gekozen en is de bearer service provider klaar om AAL paketten te transporteren.

In Figuur 3.1 is in het B-ISDN PRM de service boundary aangegeven die een gebruiker van de connectie georienteerde ATM Bearer Service gebruikt. De signalering service primitieven worden afgehandeld door de signaleringsentiteiten in de control plane en de data transport service primitieven door adaptatielaag entiteiten in de user plane (zie Figuur 3.2).




*Figuur 3.2: Voorbeeldgedrag van de ATM Bearer Service Provider*

De dienst die een PNO aan kan bieden, indien een op B-ISDN gebaseerde infrastructuur gebruikt word, wordt aangeduid met *Cell Relay*. Cell relay is een voorbeeld van een *universele dragerdienst*: de gebruikers kunnen hun spraak, audio, video en dataverkeer in de vorm van ATM *cellen* aanbieden aan het publiek net. Aan de gebruikers kan een volledige connectie georienteerde *ATM Bearer Service* aangeboden worden, ofwel in de vorm van switched virtual circuits of in de vorm van permanent virtual circuits. De link rates kunnen varieren van 2 Mb/s tot 34 Mb/s en zelfs 155Mb/s en 622Mb/s behoren tot de mogelijkheden.

De ATM bearer service provider kan ook gebruikt worden om frame relay of SMDS aan gebruikers aan te bieden (bijvoorbeeld om er voor te zorgen dat een gebruiker bestaande applicaties en interfaces kan behouden: ‘legacy reasons’). Ook is het mogelijk om IP verkeer te versturen, gebruik makend van de ATM bearer service provider.

Het is op dit moment op z’n minst onduidelijk en op z’n best niet waarschijnlijk dat *cell relay* op grote schaal aangeboden gaat worden door PNO’s. Er is ten tijde van schrijven een ontwikkeling gaande waarbij PNO’s een *IP bearer service* aanbieden voor dataverkeer en ondertussen de ontwikkelingen voor verbeteringen aan het IP protocol op gebied van spraak, audio en video nauwlettend volgen en proberen te sturen. Dit leidt tot ontwikkelingen om de bestaande SDH en SONET netwerken geschikt te maken voor IP verkeer.

**ADSL**

*Asymmetric Digital Subscriber Line* (ADSL) is een hoge snelheid transmissietechnologie ontwikkeld voor een bestaand telefoon access netwerk. Dit telefoon access netwerk kan zowel een analoog netwerk zijn (Plain Old Telephone Sytem, POTS) als ook een ISDN netwerk. De link rate voor dataverkeer downstream (van het netwerk naar de gebruiker) varieert van 1.5Mb/s tot 9Mb/s en upstream van 16Kb/s tot 640Kb/s over één paar koperdraden, afhankelijk van de

kwaliteit en lengte van de verbinding. ADSL biedt een bidirectionele dataservice naast de bestaande telefoonservice.

Een van de manieren waarop bij ADSL het aanbieden van meerdere kanalen mogelijk over een POTS access netwerk wordt gemaakt is door Frequentie Multiplexing. Hierbij worden, naast de frequentieband van het bestaande telefoonkanaal (ongeveer 4 KHz breed) een frequentieband voor upstream verkeer en een frequentieband voor downstream verkeer gebruikt. In de afgelopen jaren zijn veldexperimenten gedaan met ADSL. Indien ADSL commercieel aangeboden gaat worden zullen PNO's via hoge link rates toegang tot IP diensten kunnen bieden.

**WDM**

*Wave Division Multiplexing* (WDM) netwerken worden gezien als netwerken die een oplossing kunnen bieden voor het onderling verbinden van toegangsnetwerken met een hoge bit rate. Immers, de link rates in het kernnetwerk moet vele malen hoger kunnen zijn als die van de toegangsnetwerken.

Wave Division Multiplexing is een multiplexing methode waarbij meerdere optische signalen (van meerdere gebruikers) met verschillende golflengte gezamenlijk over één enkele glasvezel geplaatst worden, om zo de capaciteit van de fiber beter te gebruiken en afzonderlijke communicatiekanalen tussen gebruikers te realiseren. De capaciteit van een glasvezel ligt in de orde grootte van Terabits/s; in een veel gebruikt golflengtegebied is een bandbreedte beschikbaar van 25000 GHz. Link rates voor afzonderlijke kanalen kunnen in de orde van Gigabits liggen (transmissiesystemen van 100 Gb/s werden in 1996 door leveranciers aangekondigd).

Daarnaast is het mogelijk om WDM schakelaars te bouwen waarmee een optisch signaal met een bepaalde golflengte op een bepaalde link geschakeld kan worden naar een optisch signaal met een andere golflengte op een andere link. Deze schakelaars maken netwerken mogelijk van volledige optische verbindingen tussen access netwerken. Een netwerk management systeem kan dit kern netwerk configureren.

Op dit moment zijn producten voor op WDM gebaseerde netwerken met link rates varierend van 2 Gb/s tot 8 Gb/s in de handel. Er zijn al PNO's die WDM voor hun kernnetwerk gebruiken. Of PNO's ooit WDM gebaseerde dragerdiensten gaan aanbieden is vooralsnog onduidelijk.

## 3.2 Netwerk niveau

De netwerklaag heeft tot taak 'end-to-end connectiviteit' te leveren aan haar gebruikers, de transport entiteiten. Deze end-to-end connectiviteit wordt gerealiseerd door verschillende type subnetwerken met behulp van routers zodanig met elkaar te verbinden, dat een rechtreeks communicatiepad ontstaat tussen eindsystemen. Om te kunnen communiceren tussen ieder willekeurig paar eindgebruikers, moeten alle routers en alle eindsystemen in het netwerk in principe hetzelfde netwerkprotocol ondersteunen.

Begin jaren negentig waren er meerdere netwerkprotocollen die met elkaar streden om de gunst van de gebruiker: de ITU werkte samen met ISO-IEC aan *X.25*, ISO-IEC werkte daarnaast aan het *ConnectionLess Network Protocol* (CLNP), Novell had het *Internet Packet eXchange* (IPX) protocol, dat weer was afgeleid van het oude *Internet Datagram Protocol* (IDP) van Xerox, de IETF had sinds 1981 het *Internet Protocol* (IP) etc. Halverwege de jaren negentig werd duidelijk dat van al deze protocollen IP de winnaar zou worden. Momenteel zijn er vele tientallen miljoenen systemen via IP met elkaar verbonden, en word IP zelfs gebruikt voor het leveren van telefoondiensten.

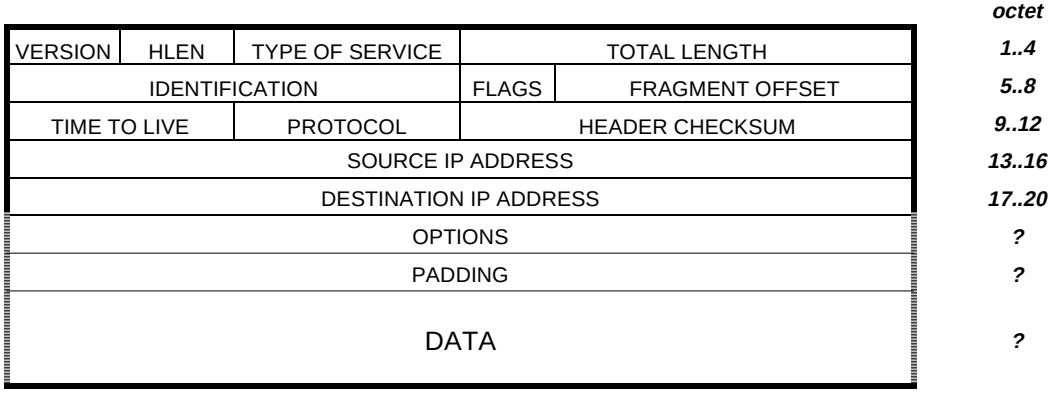
IP is een zogeheten ‘connectionless’ protocol, dus een protocol waarbij niet eerst een verbinding wordt opgebouwd voordat gebruikersdata wordt verstuurd. Het voordeel van een connectionless netwerkprotocol is dat routers geen status informatie omtrent verbindingen hoeven te bewaren en daardoor relatief eenvoudig zijn te bouwen. Een nadeel van een connectionless netwerkprotocol is dat ieder PDU het complete bron- en bestemmingsadres moet bevatten, waardoor de PDU header relatief groot wordt. Een tweede nadeel is dat er geen netwerk capaciteit kan worden gereserveerd, zodat het niet zeker is of er wel voldoende capaciteit is om het PDU tijdig op de bestemming af te leveren. Deze nadelen wegen echter niet op tegen het voordeel van een grotere eenvoud.

**IP-PDU structuur**

De structuur van een IP-PDU, ook wel datagram genoemd, is vastgelegd in RFC 791 en weergegeven in Figuur 3.3. De header begint met een veld voor het versienummer (de huidige versie van IP is 4), en heeft daarna een veld waarin de header-lengte wordt aangegeven. Deze is minimaal 20 octetten (bytes), maar kan groter worden als de header ‘options’ bevat voor b.v. ‘route-recording’ of ‘source-routing’. Met behulp van het ‘Type of Service’ (TOS) veld kan worden aangegeven of het pakket moet worden verstuurd over een pad met minimale vertraging, maximale capaciteit, maximale betrouwbaarheid etc. Alhoewel dit veld tot voor kort weinig werd gebruikt, is de verwachting dat het in de toekomst een grotere rol gaat spelen. Voor het aangeven van de totale PDU lengte zijn 2 octetten gereserveerd, zodat een IP PDU nooit groter kan worden dan 65535 octetten. In de praktijk worden echter bijna nooit pakketten gestuurd groter dan 8192 bytes; veel toepassingen beperken zich zelfs tot 512 octetten. Het ‘identification’ veld bevat een uniek volgnummer, en wordt evenals de twee volgende octetten gebruikt ten behoeve van ‘fragmentatie’ en ‘reassembly’. Het ‘Time to Live’ veld is nodig om PDUs te verwijderen als ze, ten gevolge van inconsistente routeringstabellen, in het netwerk blijven circuleren en de bestemming niet bereiken. Het veld wordt door de bron op één of andere waarde gezet (meestal 32 of 64), en vervolgens door iedere router die wordt gepasseerd verlaagd. Als de waarde nul is bereikt, wordt het PDU weggegooid. Het PDU bevat ook een veld dat aangeeft welk protocol de data heeft gegenereerd en aan welk protocol de ontvangende IP entiteit de data dus moet doorgeven. In veel gevallen zal dit een transport protocol zijn, maar het is ook mogelijk dat een routeringsprotocol of een ander controle protocol de data heeft gegenereerd. Om fouten te kunnen detecteren bevat de header ook een checksum veld; merk op dat deze checksum alleen wordt berekend over de header, en niet over de data.

**Adressering**

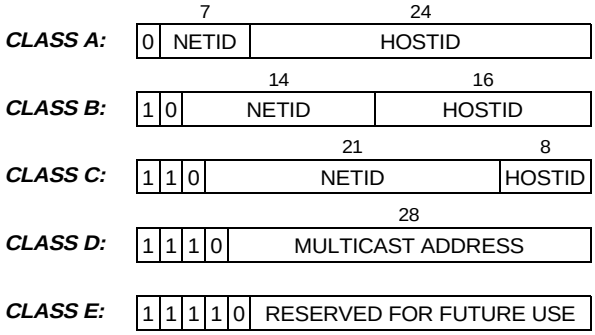
IP is oorspronkelijk ontworpen om netwerken van verschillende gebruikers door middel van één backbone met elkaar te verbinden. Om systemen wereldwijd uniek te kunnen identificeren,




*Figuur 3.3: Structuur van het IP PDU*

heeft ieder systeem (of preciezer gezegd: ieder netwerk-interface) een eigen 32 bits adres. Dit adres bestaat uit twee delen: een NETID die het gebruikersnet wereldwijd uniek identificeert, en een HOSTID die het systeem binnen het gebruikersnet uniek identificeert. NETIDs worden uitgegeven door de Internet Corporation for Assigned Names and Numbers (ICANN), de opvolger van de Internet Assigned Numbers Autority (IANA). HOSTIDs worden beheerd door de locale netwerkbeheerder.

Omdat er grote en kleine gebruikersnetwerken zijn, zijn er verschillende adresformaten (zie Figuur 3.4). Het class A adres is bedoeld voor grote gebruikersnetwerken, met meer dan 65536 gebruikers. Omdat in dit geval het NETID een lengte heeft van 7 bits, kunnen er wereldwijd niet meer dan 128 van dergelijke netwerken bestaan. Class B adressen zijn bedoeld voor middelgrootte gebruikersnetwerken, en class C voor netwerken met niet meer dan 256 gebruikers. Class D is tenslotte bedoeld voor multicast toepassingen.



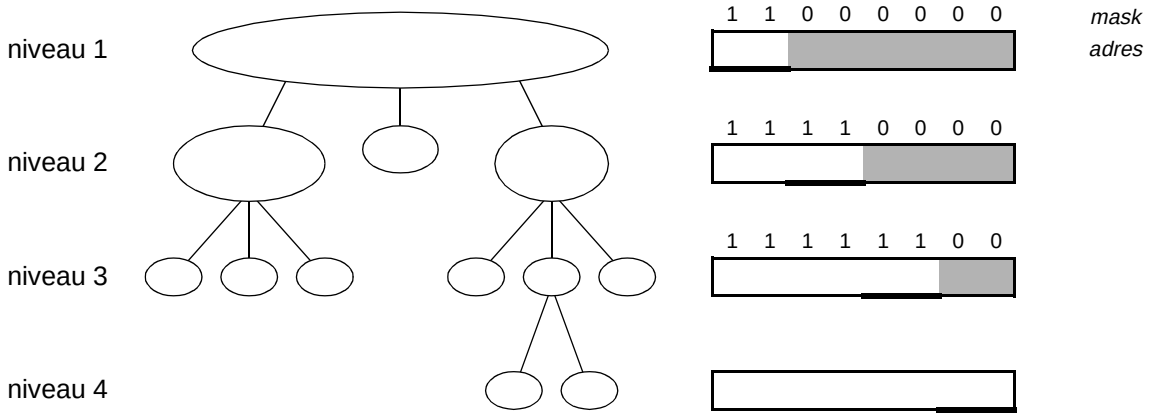

*Figuur 3.4: structuur van IP adressen*

Omdat IP PDUs een source adres veld bevatten, moet een systeem voordat het gaat communiceren zijn eigen netwerkadres kennen. In principe kan de netwerkbeheerder dit adres in de configuratie fase aan het systeem kenbaar maken, maar deze oplossing heeft als nadeel dat de beheerder ieder systeem fysiek moet benaderen en ieder systeem permanent geheugen moet bezitten om het adres op te slaan. Er is daarom een speciaal protocol ontwikkeld dat door systemen in een LAN omgeving kan worden gebruikt om tijdens de start-up fase het eigen IP adres op te vragen. Dit protocol heet het *Reverse Address Resolution Protocol* (RARP), en maakt gebruik van een LAN-broadcast bericht. Dit bericht wordt ontvangen door ieder systeem dat op het LAN is aangesloten, maar wordt alleen door de RARP-server beantwoord. Deze server gebruikt het 48 bit LAN adres, dat wereldwijd uniek is omdat het door de fabrikant in iedere LAN chip wordt ingebakken, om het IP adres te bepalen. In plaats van RARP wordt ook wel het *BOOTstrap Protocol* (BOOTP) gebruikt; dit protocol stuurt niet alleen het IP adres terug, maar het hele Operating System.

De IP adresstructuur stamt uit het eind van de jaren zeventig toen de wereld nog bestond uit mainframes en mini-computers en de PC nog moest worden uitgevonden. In die tijd dacht nog niemand aan wereldwijde netwerken met daaraan gekoppeld miljoenen apparaten. Achteraf gezien is het dan ook niet verwonderlijk dat beter een andere adresstructuur gekozen had kunnen worden. Deze andere structuur zou een groter aantal class B netwerken moeten ondersteunen; het tekort aan class B adressen is zelfs één van de belangrijkste drijfveren voor de ontwikkeling van een nieuwe versie van IP (IPv6).

Omdat er veel meer systemen op het Internet werden aangesloten dan verwacht, ontstond tevens het probleem dat de routeringstabellen te groot werden en het zoeken in die tabellen te veel tijd ging kosten. De tabellen konden worden verkleind als de oorspronkelijke architectuur, waarin werd aangenomen dat het Internet bestaat uit een verzameling van gebruikersnetwerken die

door middel van één backbone zijn gekoppeld, werd vervangen door een architectuur waarin een hierarchie van gebruikersnetwerken is toegestaan. Dankzij een dergelijke hierarchie hoeven gebruikersnetwerken niet allemaal meer direct met de backbone te worden verbonden, zodat de routeringstabellen in de backbone kleiner worden, en hoeven eindsystemen niet allemaal op hetzelfde gebruikersnetwerk te worden aangesloten, zodat de routeringstabellen in de gebruikernetwerken eveneens kleiner worden. Het nadeel is natuurlijk wel dat tussen de niveaus extra routers worden geplaatst, waardoor een deel van de tijdwinst bij het zoeken weer verloren gaat.



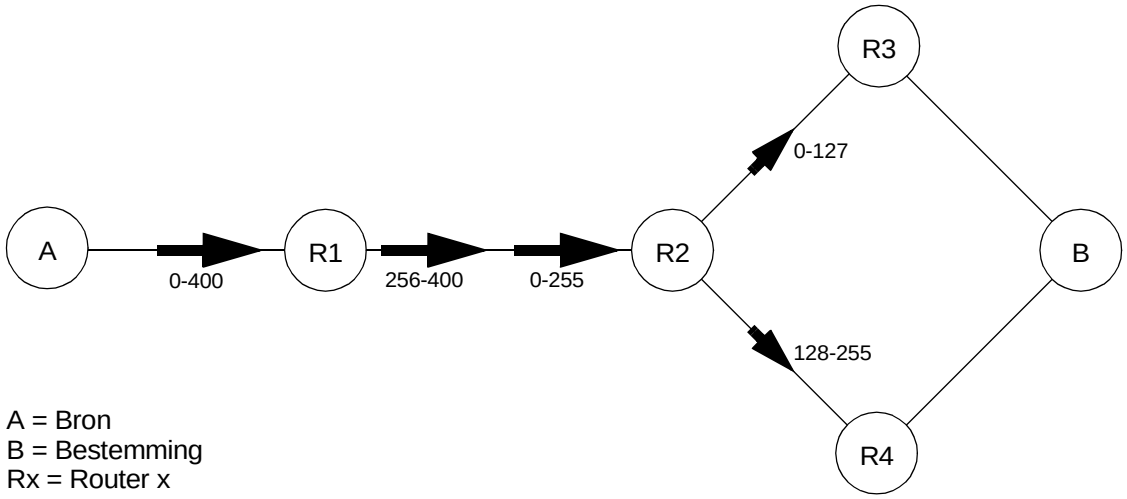

*Figuur 3.5: Subnet masks*

De techniek om een hierarchie van gebruikersnetwerken binnen de bestaande adresstructuur te realiseren werd eind jaren tachtig geïntroduceerd en maakt gebruik van zogeheten *'subnet masks'*. Deze techniek wordt geïllustreerd met behulp van het voorbeeld van Figuur 3.5. In dit voorbeeld worden op niveau 1 niet meer alle bits, maar alleen de eerste twee bits van het adres geïnterpreteerd. De netwerkbeheerder heeft hiertoe alle routers op niveau 1 voorzien van een subnet mask waarvan de eerste twee bits zijn gezet. Deze mask wordt door een router over het bestemmingsadres gelegd; het effect hiervan is dat alleen die bits van het bestemmingsadres worden geinterpreteerd waarvoor de mask de waarde 1 heeft. In het voorbeeld van Figuur 3.5 worden de routers op niveau 2 voorzien van een subnet mask waarvan de eerste vier bits zijn gezet. Het effect hiervan is dat de routers op dit niveau alleen kijken naar de eerste vier bits van het adres; omdat de eerste twee adres bits reeds op niveau 1 zijn geïnterpreteerd, wordt er in de praktijk vooral gekeken naar de bits drie en vier. Dit mechanisme herhaalt zich op de onderliggende niveaus.

**Fragmentatie en reassembly**

Het IP protocol kan gebruikt worden om verschillende soorten subnetwerken met elkaar te verbinden. Ieder type subnetwerk heeft echter zijn eigen beperkingen voor wat betreft de maximale PDU lengte. Ethernet heeft bijvoorbeeld als beperking 1500 octetten, Token ring en FDDI (afhankelijk van de maximum token rotation timer) ruim 8000 octetten, Token bus beperkt de PDU lengte tot 5000 octetten etc. Om al deze soorten subnetwerken te kunnen gebruiken, hadden de IP ontwerpers er voor kunnen kiezen de maximale IP PDU lengte te beperken tot een waarde die door ieder type subnetwerk wordt ondersteund. Omdat er echter ook subnetwerken zijn die 128 octetten als bovengrens stellen, zou de maximale IP PDU lengte klein zijn uitgevallen en zou er een ongunstige verhouding ontstaan tussen de header (van 20 octetten) en nuttige gebruikersdata. In plaats van te kiezen voor een kleine maximale PDU lengte, hebben de ontwerpers van IP er voor gekozen dat routers, indien nodig, het IP PDU in kleine fragmenten mogen opdelen. Dit proces heet fragmentatie.

IP fragmenten worden net zo behandeld als niet gefragmenteerde IP PDUs. Het is dus mogelijk dat fragmenten die bij hetzelfde bericht behoren verschillende routes door het netwerk nemen. Het weer samenvoegen van de fragmenten om het oorspronkelijke bericht te reconstrueren (reassembly) kan dan ook niet door een router in het netwerk plaatsvinden, maar alleen door de bestemming.



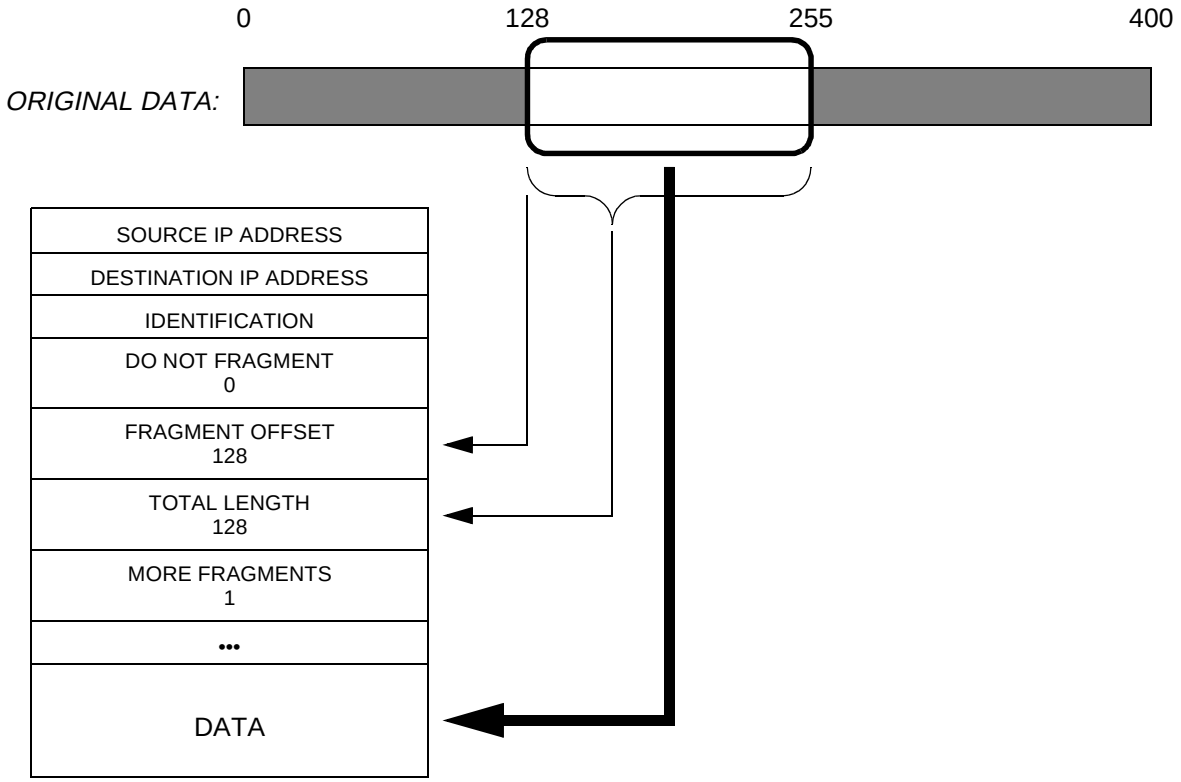

*Figuur 3.6: Fragmentatie*

Figuur 3.6 geeft een voorbeeld van fragmentatie. De bron stuurt een IP bericht met een lengte van 401 octetten. Dit bericht wordt door router 1 gesplitst in twee fragmenten; het eerste fragment bevat de eerste 256 octetten en het tweede fragment de overige 145 octetten. Het eerste fragment wordt door router 2 wederom gesplitst; de eerste 128 octetten worden verstuurd via router 3 en de overige 128 octetten via router 4. Om de figuur niet nodeloos ingewikkeld te maken is het verder versturen van de fragmenten niet getekend.

Figuur 3.7 laat aan de hand van het fragment dat van router 2 naar router 4 wordt gestuurd zien welke velden in het IP PDU bij fragmentatie een rol spelen. In eerste instantie zijn dit natuurlijk het source en destination IP adres. Om bij reassembly te kunnen bepalen tot welk bericht een fragment behoort, geeft de bron ieder bericht een uniek identificatienummer. Bij fragmentatie houden de diverse fragmenten het oorspronkelijke identificatienummer. Door de *'do not fragment'* flag te zetten, kan de bron verhinderen dat routers het bericht fragmenteren; er bestaat dan wel het gevaar dat het bericht moet worden weggegooid omdat het te groot is om voor het volgende subnetwerk. De '*fragment offset'* bepaalt de positie van het eerste octet van het fragment in het oorspronkelijke bericht; in het voorbeeld is dit positie 128. De betekenis van het *'fragment length'* veld spreekt voor zich; de '*more fragments'* flag geeft tenslotte aan of dit het laatste fragment van het oorspronkelijke bericht is (de waarde is dan 0), of niet (de waarde is dan 1).

**Foutafhandeling**

Voor het melden van fouten heeft IP een eigen protocol: het *Internet Control Message Protocol* (ICMP). ICMP berichten worden verstuurd in het data veld van het IP PDU. ICMP berichten worden vaak gegenereerd door routers en eindsystemen nadat een fout in het IP protocol is




*Figuur 3.7: IP PDU en fragmentatie*

gedetecteerd, het is echter ook mogelijk dat ICMP berichten worden verstuurd als een bepaalde vraag moet worden beantwoord. Figuur 3.8 geeft een overzicht van mogelijke ICMP berichten.

| ICMP bericht | Foutmelding | Vraag |
|---|---|---|
| Destination unreachable | X | |
| Source quench | X | |
| Redirect | X | |
| Time exceeded | X | |
| IP parameter problem | X | |
| Echo request / reply | | X |
| Timestamp request / reply | | X |
| Address mask request / reply | | X |

*Figuur 3.8: ICMP berichten*

Het *'destination unreachable'* bericht heeft een speciaal veld om de precieze foutoorzaak aan te geven. De meest voorkomende oorzaken zijn: netwerk onbekend, netwerk onbereikbaar, bestemming onbekend, bestemming onbereikbaar, protocol onbereikbaar, poort onbereikbaar, fragmentatie vereist maar de *'do not fragment'* flag is gezet. Het *'source quench'* bericht biedt de mogelijkheid tot een beperkte vorm van congestie control en kan door een router naar een bron worden verstuurd als die bron meer berichten aanbiedt dan de router kan verwerken. Omdat dit congestie control bericht voor extra netwerkbelasting zorgt, en dus voor een verhoging van de congestie, en de meeste implementaties het bericht negeren, wordt het in de praktijk niet vaak meer toegepast. Het *'redirect'* bericht wordt door een router verstuurd als het een IP bericht ontvangt en het weet dat dit IP bericht beter via een andere router verstuurd had kunnen worden.

Het *'time exceeded'* ICMP bericht wordt verstuurd als het *'time to live'* (TTL) veld (zie ) de waarde nul bereikt. Alhoewel deze fout oorspronkelijk is bedoeld voor het detecteren van inconsistente routeringstabellen, wordt het versturen van deze foutmelding in de praktijk vaak uitgelokt door een te lage waarde voor het TTL veld te kiezen. Een bekend programma dat hiervan gebruik maakt is *traceroute*. Dit programme verstuurd eerst een IP bericht met een TTL waarde gelijk aan 1. De eerste router op het pad verlaagt deze TTL waarde, ziet dat de waarde 0 wordt, en stuurt het *'time exceeded'* ICMP bericht. Omdat ieder ICMP bericht het IP adres bevat van de zender van dit bericht, weet het traceroute programma het adres van de eerste router op het pad naar de bestemming. Vervolgens stuurt het traceroute programme een IP bericht met een TTL waarde gelijk aan 2. De tweede router op het pad zal nu de ICMP foutmelding sturen en het traceroute programma weet nu het adres van de tweede router op het pad. Dit proces herhaalt zich, totdat de adressen van alle routers op het pad door het traceroute programma zijn ontdekt.

Een ander bekend programma dat gebruik maakt van ICMP, is *ping*. Dit programma genereert één of meerdere ICMP *'echo request'* berichten, en meet de tijd tot de bijbehorende ICMP *'echo reply'* berichten worden ontvangen. Omdat met *ping* op een eenvoudige wijze de bereikbaarheid van bestemmingen kan worden getest, kan het als meest gebruikt IP beheerprogramma worden beschouwd.

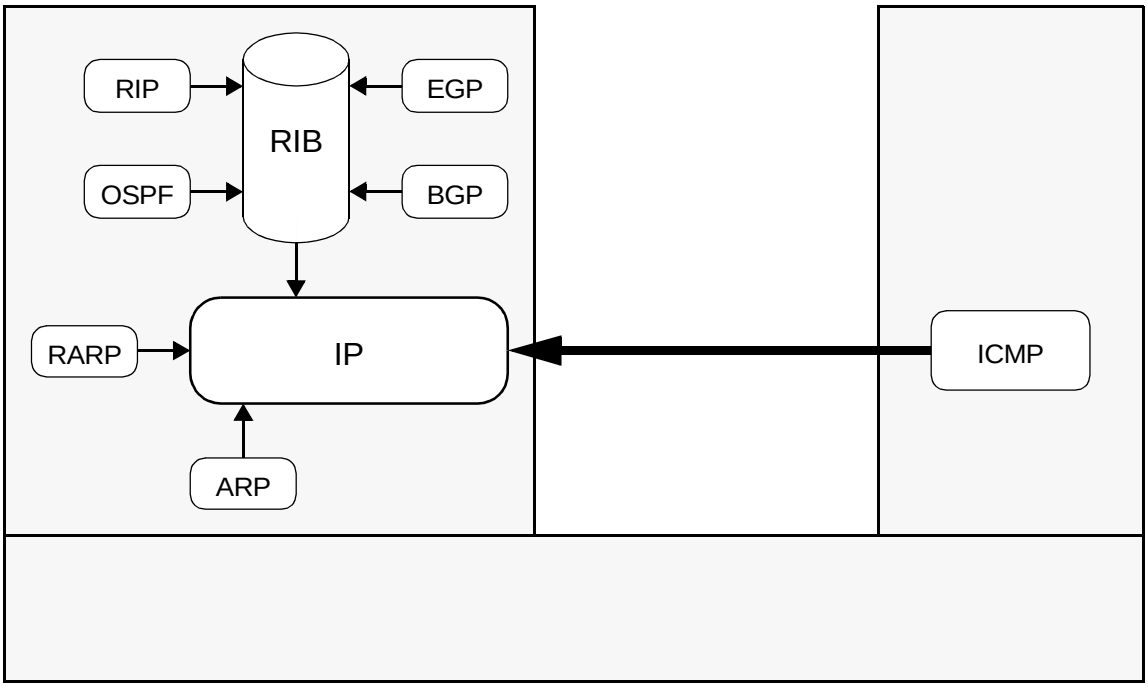
De *'timestamp request'* wordt verstuurd indien een systeem wil weten hoe laat het is. De ontvanger stuurt als reactie een *'timestamp reply'*, met daarin de tijd en een nauwkeurigheid van milliseconde. De datum wordt niet teruggestuurd.
De *'address mask request'* wordt verstuurd indien een systeem wil weten welke *'subnet mask'* wordt gebruikt (). Het wordt vaak gebruikt in combinatie met het RARP protocol.

**Routering**

Omdat IP volgens het 'connectionless' principe werkt, moet een router voor ieder binnenkomend PDU bepalen in welke richting de PDU moet worden doorgestuurd. De router haalt hiertoe het IP bestemmingsadres uit het IP-PDU, en gebruikt dit adres om de zogeheten *Routing Information Base* (RIB) te raadplegen. Het resultaat van de RIB query is het IP adres van het volgende systeem op het pad naar de bestemming. Indien de router en het volgende systeem met behulp van een LAN zijn verbonden, moet ook het LAN adres van dit volgende systeem worden bepaald. Hiertoe kan het *Address Resolution Protocol* (ARP) worden gebruikt, dat net als het eerder besproken RARP protocol, een broadcast bericht verstuurt over het LAN. Alhoewel deze broadcast door ieder systeem dat op het LAN is aangesloten wordt ontvangen, zal alleen het systeem reageren dat zijn eigen IP adres in het ARP bericht herkent. Het ARP bericht dat als antwoord wordt teruggestuurd bevat het LAN adres dat bij het gevraagde IP adres behoort; de router heeft dus nu alle gegevens om het IP-PDU door te sturen. Om toekomstige netwerkbelasting ten gevolge van ARP te verminderen, zal de router de zo even gevonden relatie tussen IP en LAN adres in de zogeheten ARP cache opslaan.

Om de RIB initieel te vullen en vervolgens dynamisch aan te passen aan veranderingen in de netwerk topologie, zijn een aantal *routeringsprotocollen* gedefinieerd. Deze protocollen kunnen in twee categorieen worden ingedeeld: interior gateway protocollen en exterior gateway protocollen; voor de eerste categorie wordt ook wel de naam intra-domain protocol gebruikt, en voor de tweede inter-domain protocol. Een interior gateway protocol wordt gebruikt binnen het domein van een enkele operator; een exterior gateway protocol wordt gebruikt om routeringsinformatie tussen operators uit te wisselen. Voorbeelden van interior gateway protocollen zijn het oude *Routing Information Protocol* (RIP) en het nieuwe *Open Shortest Path First* (OSPF)

protocol; voorbeelden van exterior gateway protocollen zijn het oude *Exterior Gateway Protocol* (EGP) en het nieuwe *Border Gateway Protocol* (BGP). Figuur 3.9 toont de relatie tussen de tot nu toe genoemde netwerklaag protocollen.




*Figuur 3.9: Relatie tussen netwerklaagprotocollen*

Een belangrijk technisch verschil tussen de diverse routeringsprotocollen is dat OSPF van het zogeheten *link-state* type is, en de overige van het *distance vector* type. Bij link state protocollen kent iedere router de totale netwerktopologie en is het detecteren en voorkomen van routeringsfouten relatief eenvoudig. Een nadeel van link state protocollen is dat er veel informatie in de RIB moet worden opgeslagen en dat het berekenen van routes relatief veel processing power kost. Deze problemen treden niet op bij distance vector protocollen, omdat routers hierbij slechts een deel van de netwerktopologie hoeven te weten. Deze protocollen hebben echter weer andere problemen, zoals trage aanpassing na veranderingen in de netwerktopologie en het gevaar dat door een al dan niet bewuste fout netwerkverkeer de verkeerde kant kan worden opgestuurd waarna het verloren gaat (black holes).

Om de afmeting van routeringstabellen te beperken, wordt steeds vaker gebruik gemaakt van *Classless Interdomain Routing* (CIDR). Bij deze techniek kunnen, door gebruik te maken van de eerder genoemde subnet masks, meerdere regels in een routeringstabel worden samengevoegd.

**Multicast**

Naast *unicast*, waarbij een IP-PDU naar een enkel systeem wordt gestuurd, en *broadcast*, waarbij een IP-PDU naar alle systemen wordt gestuurd, ondersteund IP ook de mogelijkheid om een PDU naar een groep van systemen te sturen. Deze mogelijkheid heeft *multicast*, en is vooral handig voor toepassingen zoals ‘multi-party video conferencing’. Om met deze toepassing te kunnen experimenteren, is er binnen het Internet zelfs een specifieke infrastructuur gecreeerd onder de naam MBone. Toepassingen die van multicast gebruik maken, moeten er voor zorgen dat er weinig besturingsinformatie terug hoeft te worden gestuurd naar de multicast bron, en zijn daarom in de meeste gevallen gebaseerd op het connectionless transport protocol UDP.

Multicast kan vooral in een LAN omgeving efficient worden geïmplementeerd. Om in een dergelijke omgeving de leden van een multicast groep dynamisch te kunnen wijzigen, is er een apart protocol gedefinieerd: het *Internet Group Management Protocol* (IGMP). Bij dit protocol adverteert een router die met de buitenwereld is verbonden met een bepaalde regelmaat alle bekende multicast groepen. Als er binnen de LAN omgeving systemen zijn die lid willen worden van een bepaalde multicast groep, stellen ze de router hiervan op de hoogte. Als de router vervolgens van de buitenwereld IP-PDUs ontvangt met de bewuste multicast groep als adres, dan zal de router deze IP-PDUs over het LAN verder versturen. Mocht er na verloop van tijd vanuit het LAN geen enkel systeem meer interesse in de bewuste multicast groep hebben gemeld, dan stopt de router met het doorsturen van de aan deze groep geadresseerde IP-PDUs.

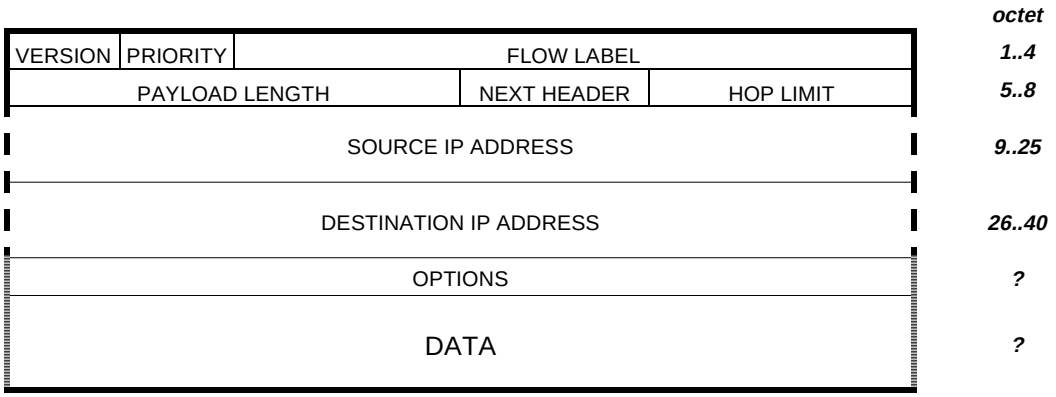
**QoS ondersteuning**

Binnen het oorspronkelijke Internet werd ieder PDU op dezelfde manier behandeld, ongeacht of het PDU afkomstig was van een real-time toepassingen of van een toepassingen waarbij geen eisen werden gesteld aan de Quality of Service (QoS). Om toch onderscheid te kunnen maken tussen bijvoorbeeld video conferencing en email, heeft men het *Resource Reservation Protocol* (RSVP) ontwikkeld. Met behulp van RSVP kunnen eindgebruikers capaciteit op het pad tussen bron en bestemming reserveren; het is hierdoor mogelijk kwaliteitsgaranties te geven voordat een bepaalde toepassing wordt gestart.

Om RSVP goed te kunnen ondersteunen, moeten routers veel informatie opslaan. Omdat het bijhouden van deze informatie problematisch blijkt te zijn, wordt er sinds 1997 gewerkt aan een alternatieve techniek die de naam *Differentiated Services* (DiffServ) heeft gekregen. Deze techniek maakt gebruik van het Type of Service veld in het IP-PDU (), en vereist geen extra informatie in routers.

**IPv6**

Om ook in de toekomst verdere groei van het Internet mogelijk te maken is een nieuwe versie van het IP protocol ontwikkeld: IP versie 6. De verbeteringen die in deze versie zijn doorgevoerd hebben vooral betrekking op de lengte van het IP adres en QoS ondersteuning door middel van flows. Omdat de IPv6 PDU (zie Figuur 3.10) minder velden bevat dan de IPv4 PDU (), kan een IPv6 router PDUs sneller doorsturen.




*Figuur 3.10: structuur van het IPv6 PDU*

IPv6 adressen hebben een lengte van 128 bits, en zijn dus vier keer groter dan de oorspronkelijke IPv4 adressen. In plaats van een starre adres structuur gebaseerd op een vijftal classes (A t/m E - Figuur 3.4), is het met IPv6 mogelijk adressen op een aantal manieren te structureren: provider based global, link local, site local, embedded IPv4 en loopback.

Een interessante vernieuwing is het zogeheten *anycast* adres. Dit adres kan worden gebruikt voor het identificeren van een groep van systemen; een router die een PDU met een anycast adres ontvangt, zal het PDU doorsturen naar het systeem dat binnen deze groep het best bereikt kan worden. Anycast adressen zijn vooral nuttig voor bedrijven met meerdere identieke web servers, die de belasting willen verdelen over de verschillende servers.

Het IPv6 flow label kan gebruikt worden om het pad waarvoor met behulp van RSVP capaciteit is gereserveerd, te identificeren. Alle PDUs met dezelfde bronadres - flow label combinatie worden op dezelfde manier door een router behandeld.

## 3.3 Transport niveau

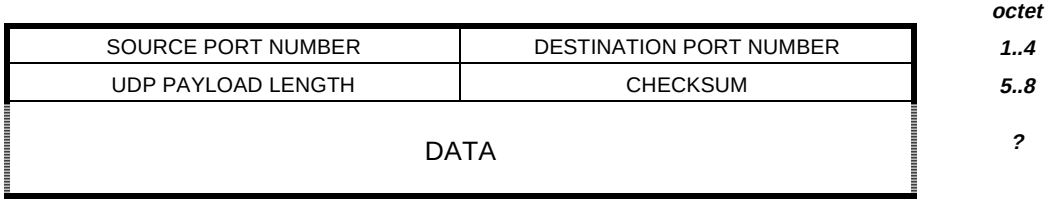
Er zijn twee Internet transport protocollen: het connectionless *User Datagram Protocol* (UDP) en het connection-oriented *Transmission Control Protocol* (TCP). Om te kunnen identificeren voor welk applicatielaag protocol de data van het transport protocol is bestemd, maken zowel UDP als TCP gebruik van zogenaamde *well-known port-numbers* (zie Tabel 3.3).

| Port-number | applicatielaag protocol |
|---|---|
| 20 | FTP (data) |
| 21 | FTP (control) |
| 23 | Telnet |
| 25 | SMTP |
| 53 | DNS |
| 80 | HTTP |
| 161 | SNMP (Get / Set) |
| 162 | SNMP (Trap) |

*Tabel 3.3: Voorbeelden van well-known port-numbers*

**UDP**

Het *User Datagram Protocol* (UDP) is een eenvoudig protocol. De belangrijkste functie van UDP is het identificeren van het applicatielaag protocol waartoe de UDP data behoort; om deze functie te kunnen verrichten heeft het UDP-PDU (zie Figuur 3.11) een veld om het bron port-number en een veld om het bestemmings port-number aan te geven. Naast een veld om de lengte van de UDP data aan te geven, heeft UDP nog een checksum veld, waarmee fouten kunnen worden gedetecteerd in de UDP header en data. Vaak worden transmissiefouten reeds op datalink niveau gedetecteerd, en wordt het checksum veld op transport niveau alleen gebruikt voor het detecteren van IP reassembly fouten. Nadat een fout is gevonden, wordt het bijbehorende PDU weggegooid. UDP heeft geen flow control of hertransmissie functies; de overdracht via UDP is derhalve onbetrouwbaar.




*Figuur 3.11: UDP-PDU structuur*

**TCP**

Het *Transmission Control Protocol* (TCP) is een betrouwbaar, en daardoor complex protocol.

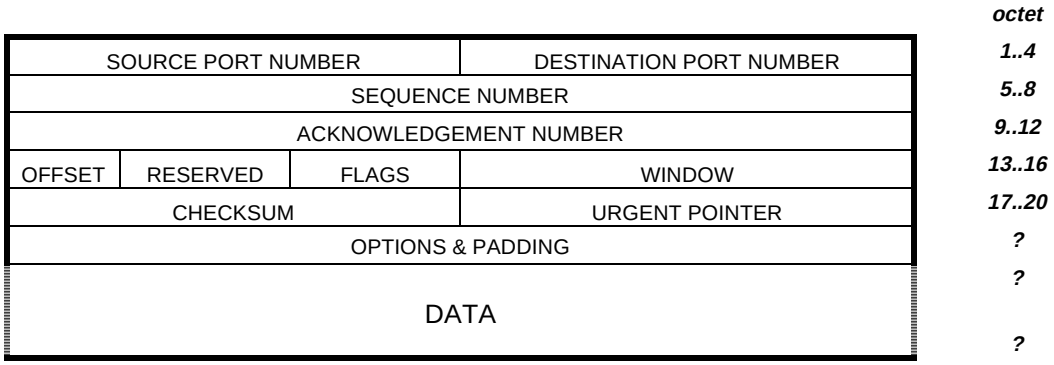
In tegenstelling tot de meeste andere transport protocollen, is TCP *stroom-georiënteerd*. De betekenis hiervan is dat de gebruiker van TCP een stroom van data octetten aanlevert, in plaats van complete pakketten. De TCP entiteit slaat deze stroom van octetten op in een buffer, en bepaalt in principe zelf wanneer de data in de buffers wordt verstuurd. Door het zetten van een speciale *push flag*, kan de gebruiker echter ook aangeven dat de data onmiddellijk verstuurd moet worden. Deze optie is vooral nuttig bij interactieve applicaties, zoals *telnet*.

TCP onderscheid drie fases: verbindingopbouw, data-uitwisseling en verbindingbeëindigen. Verbindingopbouw kan op twee manieren:

- De TCP gebruiker geeft aan dat hij bereid is data te ontvangen. De bijbehorende primitieve heet *passive open* en wordt vooral gebruikt door servers. Indien gewenst kan de gebruiker aangeven dat de bereidheid alleen bestaat voor data van een specifieke bron.
- De TCP gebruiker geeft aan dat hij van plan is data te versturen naar een specifieke bestemming. De bijbehorende primitieve heet *active open* en wordt vooral gebruikt aan de client zijde van de verbinding.

Nadat de verbinding is opgebouwd, kan met behulp van de *send* primitieve data worden verstuurd. Net als bij UDP, bevatten TCP-PDUs (zie Figuur 3.12) een *checksum* veld waarmee fouten kunnen worden gedetecteerd. Omdat TCP is voorzien van een hertransmissie functie, kunnen PDUs die niet (correct) zijn aangekomen opnieuw worden verstuurd. De hertransmissie functie maakt gebruik van timers en een tweetal velden in het TCP-PDU: het *sequence number* veld en het *acknowledgement number* veld. Samen met het *window* veld worden deze velden eveneens gebruikt voor TCP's flow control functie. Met behulp van deze functie kan de ontvanger aangeven hoeveel data hij bereid is te ontvangen.

De verbinding wordt gewoonlijk afgebroken door middel van een *close* primitieve. Deze primitieve zorgt er voor dat alle nog in de buffers aanwezige data alsnog wordt verstuurd. Indien er wat is misgegaan, kan de verbinding ook worden afgebroken met behulp van de *abort* primitieve; eventueel in de buffers aanwezige data wordt dan niet meer verstuurd maar weggegooid.




*Figuur 3.12: structuur van het TCP PDU*

## 3.4 Applicatie niveau

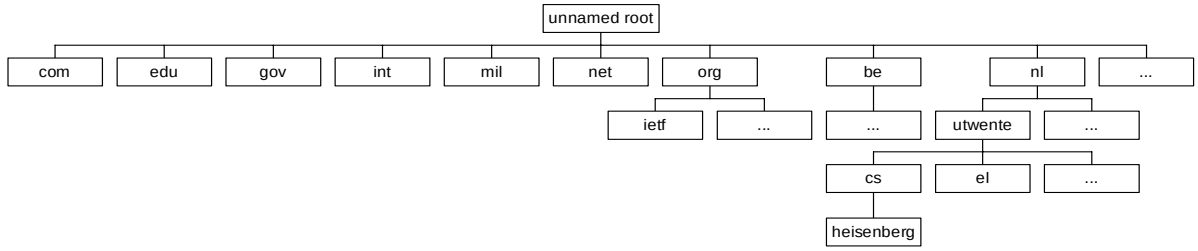
Zoals in hoofdstuk 2 reeds is aangegeven bestaan er zeer vele applicatieprotocollen die over TCP/IP netwerken worden aangeboden, ieder met zijn eigen kenmerkende functionaliteit die aan de gebruikersbehoeften tegemoet komt. Juist door deze veelheid van applicaties en applicatieprotocollen is het onmogelijk deze allen hier te bespreken. In plaats hiervan zijn er een aantal geselecteerd en deze worden in de volgende paragrafen besproken, hierbij zal ingegaan worden op principes die worden gebruikt. Voor het verkrijgen van meer gedetailleerde informatie, de specificaties van de applicatieprotocollen zijn vastgelegd in RFCs.

### 3.4.1 Domain Name System (DNS)

Om via het Internet communicatie tussen twee hosts te bewerkstelligen moet van zowel de zendende als de ontvangende host het IP-adres bekend zijn, immers uiteindelijk wordt de informatie in IP-pakketten verpakt en ieder IP pakket moet zowel het IP-adres van de zender als van de ontvanger bevatten. Echter, vanuit het perspectief van een gebruiker zijn IP adressen uitermate onhandig: ze zijn moeilijk te onthouden, er kunnen makkelijk foutieve adressen worden opgegeven en, wellicht het allerbelangrijkste, een IP adres bevat weinig tot geen intuitief begrijpelijke informatie. Deze problemen worden opgelost door het Domain Name System (DNS).

DNS is vastgelegd in twee RFC's: een RFC specificeert de concepten en faciliteiten die door DNS worden geboden, de andere legt details vast over de specificatie en implementatie. Met behulp van DNS kunnen domain-namen omgezet worden in IP-adressen en omgekeerd. Internet applicaties, zoals TELNET, FTP en HTTP maken hiervan gebruik zodat een gebruiker slechts de domain-naam hoeft in te voeren teneinde de communicatie met de betreffende host te bewerkstelligen.

Domain-namen zijn hierarchisch opgebouwd, ze hebben dus een bepaalde (boom-)struktuur. In Figuur 3.13 is dit geïllustreerd. De wortel van de boom is de 'unnamed root'. Op het eerste hierarchische niveau bestaan een zevental generieke domeinen ('com' voor commerciele organisaties, 'edu' voor onderwijs instellingen, 'gov' voor andere gouvermentele organisaties in de U.S.A, 'int' voor internationale organisaties, 'mil' voor de militaire organisatie in de U.S.A, 'net' voor netwerken en 'org' voor overige organisaties), en een groot aantal land domeinen waaronder: 'nl' en 'be' voor respectievelijk Nederland en Belgie. Verder opsplitsing van de domain-namen gebeurt op opeenvolgende hierarchische niveaux, bijvoorbeeld 'utwente' voor de Universiteit Twente en 'cs' voor de Faculteit Informatica en 'el' voor de Faculteit Elektrotechniek. Zo is er bijvoorbeeld een computer binnen de Faculteit Informatica met de naam 'heisenberg', de domain-naam van deze host is: heisenberg.cs.utwente.nl. Het bijbehorende IP-adres is 130.89.10.98.




*Figuur 3.13: Hierarchische structuur van DNS*

Om dit naam-adres transformaties uit te kunnen voeren is een gegevensbank nodig waarin deze gegevens zijn opgeslagen. In het geval van het DNS is dit een gedistribueerde gegevensbank: de gegevens zijn, verspreid over vele zogenaamde DNS servers, beschikbaar. Wanneer aan een DNS server om het IP-adres van en een bepaalde domain-naam wordt gevraagd en deze server heeft die informatie niet, dan wordt dit verzoek op basis van die domain-naam doorgestuurd naar een andere DNS server.

Het DNS maakt hoofdzakelijk gebruik van de UDP transport service, in een aantal specifieke situaties wordt de TCP transport service gebruikt. De transformatie van hostnaam naar IP-adres (of omgekeerd) wordt gerealiseerd met een DNS-request naar een DNS-server te versturen, de doorop verkregen DNS-response bevat het resultaat van de transformatie. Het DNS kent slecht

1 struktuur van berichten, zowel de DNS-request als de DNS-response kunnen in deze ene berichtenstruktuur worden gevat.

### 3.4.2 Simple Network Management Protocol (SNMP)

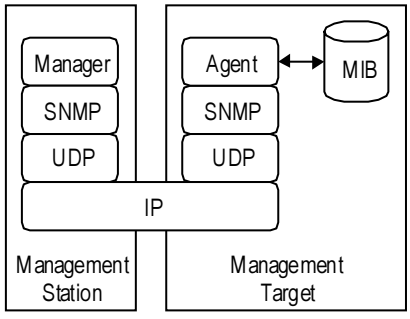
Naarmate een netwerk groter wordt, wordt ook de behoefte aan netwerk management groter teneinde het netwerk te kunnen monitoren en, waar en wanneer nodig, netwerk parameters aan te kunnen passen om zodoende een (sub-)optimaal functioneren van het netwerk te bewerkstelligen.

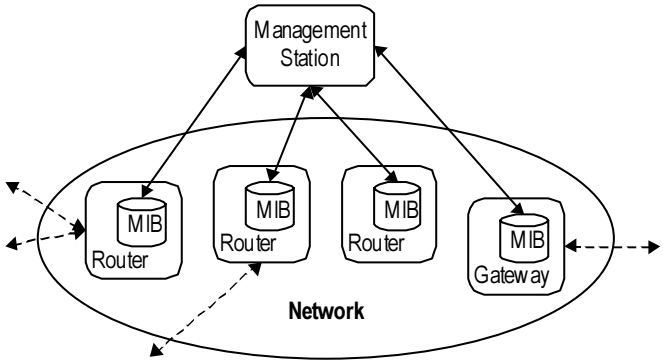
Binnen diverse organisaties is en wordt aan netwerk management standaarden gewerkt. Voor Internet management is door de IETF een management raamwerk ontwikkeld. Dit raamwerk is gebaseerd op het Manager-Agent model, hierin is de agent de te managen entiteit (bijv. router, bridge, firewall), en de manager is een entiteit (doorgaans een workstation) waar management informatie verzameld en verwerkt wordt en vanaf waar agents bestuurd kunnen worden (zie Figuur 3.14).

Het SNMP raamwerk bestaat uit drie hoofddelen:

- Management Information Base (MIB): dit is een een gegevensbank die informatie over een netwerk element bevat (bijvoorbeeld: status en type van poorten, verkeer over iedere poort), deze informatie is opgeslagen in het te managen element. Gegevens die kunnen worden opgeslagen in een MIB zijn variabelen (aangeduid met object identifiers, OID's) en tabellen.
- SNMP voor de uitwisseling vanmanagement informatie tussen de agent en de manager.
- Structuur van Management Informatie (SMI): Dit is een document waarin is vastgelegd waaraan een MIB moet voldoen en hoe deze gespecifeerd moet worden. SMI zelf is dus meer een meta-document.

Door het aantal implementaties dat voorhanden is, is SNMP de de facto standaard voor netwerk management. In 1990 verscheen SNMPv1; inmiddels is SNMPv3 in een zeer ver gevorderd stadium en zal in 1999 als RFC verschijnen.




Op het eerste gezicht lijkt het voor de hand te liggen dat SNMP gebruik maakt van het TCP transport protocol, immers management informatie is belangrijk en kan dus beter niet verloren gaan. Echter, management is nodig wanneer een aantal onderdelen niet meer geheel naar behoren functioneren of wanneer er erg veel verkeer op het netwerk is, onder zulke omstan-

digheden is het beter om UDP te gebruiken: statistisch gezien is de kans dan namelijk het grootst dat de berichten toch nog getransporteerd worden.

Het SNMPv1 protocol kent vijf verschillende typen berichten die kunnen worden uitgewisseld:

- get-request: dit is een bericht van manager naar agent om de waarde van een variable op te vragen.
- get-next-request: alle objecten in de MIB zijn geordend, met behulp van dit bericht wordt om de waarde van de volgende variable gevraagd.
- set-request: dit is een bericht van manager naar agent om de waarde van een variable te veranderen.
- get-response: dit is een bericht van agent naar manager in reactie op een get-request, get-next-request of een set-request.
- trap: dit is een bericht van de agent naar de manager n.a.v. een bijzondere of onverwachte gebeurtenis die zich in de agent heeft voorgedaan.

### 3.4.3 TELNET

TELNET is een van eerste Internet applicaties, het dateert reeds uit 1969. TELNET is een acronym van 'Telecommunications Network Protocol'. Met behulp van TELNET kan een remote login worden gedaan, d.w.z. een gebruiker die is ingelogd op een host kan met behulp van TELNET inloggen op een andere op het Internet aangesloten host ongeacht het fabrikaat van beide hosts en het besturingssyteem waar deze hosts gebruik van maken. Daarmee heeft de gebruiker dus toegang gekregen tot het arsenaal van bestanden en commando's die op de remote host van toepassing zijn.

Omdat er verschillende soorten besturingssystemen zijn (bijvoorbeeld Unix en Windows) moeten er speciale maatregelen getroffen worden om het op afstand werken op een host mogelijk te maken. Het concept waar TELNET gebruik van maakt is die van de Network Virtual Terminal (NVT), dit is een imaginaire terminal en zowel de local als de remote host beelden hun terminal specifieke commando's en informatie af op de NVT. De communicatie tussen de local en remote terminal gaat via de voor de TELNET-sessie geldende NVT.

De TELNET applicatie maakt gebruik van een enkele TCP connectie. Zowel de signalleringsberichten (commando's) als data worden via deze TCP verbinding getransporteerd, we spreken hier van 'in-band-signaling'. Om commando's en data van elkaar te kunnen scheiden wordt ieder commando byte voorafgegaan door het 'Interpret As Command' byte (IAS).

### 3.4.4 File Transfer Protocol (FTP)

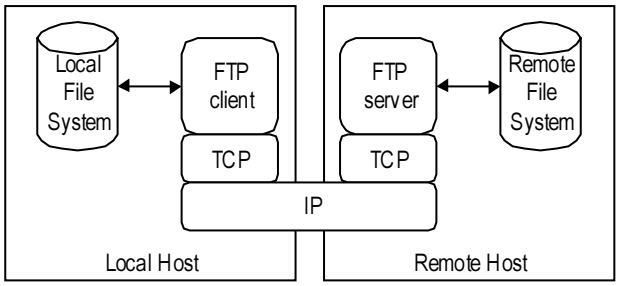
Het File Transport Protocol (FTP) maakt het mogelijk om bestanden tussen een local en remote host worden uit te wissellen. Het protocol is zodanig ontworpen dat het protocol werkt onafhankelijk van het soort host, het besturingssysteem, de filestruktuur en gebruikt karakter-set. Om dit te bereiken wordt net als bij TELNET gebruik gemaakt van een Network Virtual Terminal.

Om FTP te kunnen gebruiken, moet de gebruiker een account op de remote host hebben of de remote host moet een anonymous FTP toestaan. In dit laatste geval kan doorgaans ingelogd worden op de remote host met usename 'anonymous' en een email-adres als password.

FTP maakt gebruik van de TCP service, wanneer een FTP sessie wordt gestart wordt er via een TCP verbinding een control verbinding opgezet. Deze verbinding wordt gebruikt voor alle

commando's en de respons hierop tussen local en remote host (zie Figuur 3.15). Voorts zijn er drie situaties waarvoor een tweede FTP verbinding wordt opgezet:

- versturen van een bestand van local naar remote host,
- versturen van een bestand van remote naar local host, en
- versturen van de directory inhoud van remote naar local host.




*Figuur 3.15: FTP model*

Aangezien het voor de besturing van een FTP sessie noodzakelijk is om een garantie te hebben dat beide hosts hetzelfde 'beeld' van de toestand van sessie hebben moet de communicatie tussen beide betrouwbaar zijn, het is daarom voor de handliggend dat de control verbinding gebruik maakt van een TCP verbinding. Bestanden die worden uitgewisseld kunnen van willekeurige omvang zijn. Het zal vaak zo zijn dat een bestand in meerdere paketten moet worden gesplits om te kunnen worden verstuurd. Wanneer dit transport onbetrouwbaar gebeurt (bijvoorbeeld door gebruik te maken van UDP), dan kan het gebeuren dat in het ontvangen bestand hiaten zitten. In z'n algemeenheid zal dit ongewenst zijn, derhalve is voor bestandstransfer TCP de logische keuze.

### 3.4.5 E-mail protocollen

Een van de populairste toepassingen van het Internet is e-mail (electronic mail). Een e-mail bericht bestaat uit drie componenten, namelijk:

- *Envelope*: Dit zijn de e-mail adressen van de verzender en de ontvanger van het bericht, ze worden gebruikt om het bericht te verzenden.
- *Header*: Dit zijn velden in een e-mail bericht die toegevoegde informatie over het e-mail bericht bevatten, zoals: tijdstip waarop het verstuurd is, tijdstip waarop het aangekomen is, onderwerp van het bericht.
- *Body*: dit is de inhoud van het e-mail bericht.

E-mail berichten bestaan volledig uit ASCII karakters. Om de hieraan inherente beperkingen weg te nemen kan met behulp van MIME (Multipurpose Internet Mail Extensions) de body van een bericht zodanig worden ingericht dat deze gecodeerde niet-ASCII bestanden bevat.

Alhoewel populair en eenvoudig in gebruik, is e-mail een complexe dienst. Ook worden diverse applicatieprotocollen gebruikt voor het realiseren van de end-to-end dienst.

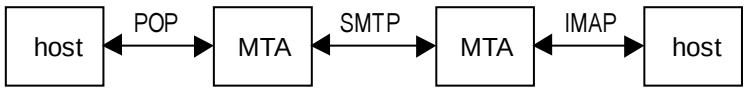
Ieder gebruiker van de e-mail dienst heeft een postbus (mailbox). Deze postbus is (doorgaans) niet de host van de gebruiker, maar een ander systeem waar meerdere gebruikers hun postbus hebben: dit systeem verzorgt de daadwerkelijke e-mail uitwisseling en wordt Message Transfer Agent (MTA) genoemd. Wanneer een gebruiker een e-mail bericht heeft samengesteld en deze

gaat versturen dan wordt het bericht eerst van de host naar de MTA getransporteerd, vervolgends wordt het bericht doorgestuurd naar de MTA van de ontvanger en het e-mail bericht wordt in de juiste postbus opgeslagen. De ontvanger kan vervolgens het e-mail bericht ophalen uit de postbus en overbrengen naar de eigen host. De end-to-end verzending van e-mail berichten bestaat dus feitelijk uit drie stappen.

Er zijn een aantal redenen om de e-mail service op deze wijze te organiseren:

- de computer van de gebruiker hoeft niet altijd aan te staan om toch altijd e-mail berichten te kunnen ontvangen. Het spreekt hierbij voor zich dat in dit scenario de MTAs 24 uur per dag operationeel zijn.
- juist omdat zo'n populaire en veel gebruikte Internet dienst is, kan door deze architectuur special purpose computer ingeze worden (voor de MTAs) waarbij de hosts van gebruikers ontlast worden voor andere taken.

Voor de communicatie tussen de host van de gebruiker en de MTA kan o.a. het POP of het IMAP protocol worden gebruikt, het SMTP protocol wordt gebruikt voor de communicatie tussen MTAs (zie Figuur 3.16).

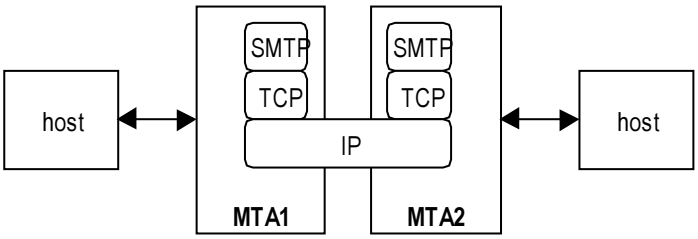



*Figuur 3.16: Illustratie van 'end-to-end' e-mail service*

**Simple Mail Transfer Protocol (SMTP)**

Het Simple Mail Transfer Protocol (SMTP) is gespecificeerd in RFC 821, het voorziet in betrouwbare communicatie tussen twee MTAs onder gebruikmaking van een TCP verbinding. Bij normaal (correct) verlopende e-mail uitwisseling worden de volgende vijf commando's achtereenvolgens uitgewisseld:

- HELO: identificatie de zendende MTA.
- MAIL: identificatie van de verzender van het e-mail bericht.
- RCPT: identificatie van de ontvanger van het e-mail bericht.
- DATA: verzenden van de inhoud van het e-mail bericht.
- QUIT: beëindigen de e-mail uitwisseling tussen de twee MTAs.




*Figuur 3.17: Positionering van het SMTP protocol*

### Post Office Protocol (POP)

Zoals in voorgaande paragraaf is uigelegd, is het SMTP protocol vooral bedoeld voor de communicatie tussen MTAs. Wanneer een gebruiker zijn electronisch mail wil lezen, kan deze gebruik maken van het Post Office Protocol (POP).

De ontwikkeling van POP is reeds begonnen in 1985, inmiddels is er versie 3. De functionaliteit die POP biedt is:

- Ophalen van e-mail berichten en verwijderen van de berichten op de MTA.
- Ophalen van e-mail berichten en de berichten op de MTA laten staan.
- Vragen of er nieuwe e-mail berichten zijn aangekomen.

Deze beperkte functionaliteit is in de praktijk bijzonder krachtig en nuttig. Vaak vanaf de eigen werkplek de eerste functie gebruikt worden, zodoende wordt er voor gezorgd dat de geheugengebruik op de MTA binnen de perken blijft. Wanneer een gebruiker (tijdelijk) vanaf een andere werkplek de e-mail berichten wil lezen, zal meestal van de tweede functie gebruik worden gemaakt.

Omdat de functionaliteit beperkt is en ook het protocol redelijk eenvoudig is, worden er geen hoge eisen gesteld aan de computer die door de gebruiker gebruikt wordt. Om redenen die gelijksoortig zijn aan die voor SMTP, maakt POP ook gebruik van de transport service TCP. Wanneer de TCP verbinding is opgezet wordt er eerst een autorisatie-fase afgehandeld (doorgaans d.m.v, gebruikersnaam en password), immers, alleen de geidentificeerde persoon mag toegang hebben tot zijn of haar e-mail berichten.

### Internet Message Access Protocol (IMAP)

Een alternatief voor POP is het Internet Message Access Protocol (IMAP). De meest recente standaard van IMAP is versie 4 (revisie 1). Net als POP wordt IMAP gebruikt voor de communicatie tussen de MTA en de host van de gebruiker. Het grote verschil de twee is dat IMAP meer functionaliteit biedt dan POP. Met IMAP is het, in tegenstelling tot POP, o.a. mogelijk om:

- alleen de headers van de e-mail berichten op te halen,
- alleen de bodies van specifieke e-mail berichten op te halen,
- alleen e-mail berichten op te halen die aan een bepaald selectiecriterium voldoen,
- e-mail berichten op de MTA te markeren (bijv. met ’deleted’, ’answered’ en ’unseen’).

Het model dat ten grondslag ligt aan het IMAP protocol is een client-server model met uitgebreide mogelijkheden voor een dialoog tussen gebruikers-host en MTA en online verwerking van e-mail berichten, terwijl POP gebasseerd is op bulk transport van (alle) e-mail berichten van MTA naar gebruikers-host in combinatie met offline verwerking.

## 3.4.6 Newsgroups, News en Usenet

Een ’Newsgroup’ is, vanuit het perspectief van de gebruiker, een virtueel prikbord waar discussies over een specifiek onderwerp kunnen worden gevoerd. Het Internet kent inmiddels enkele tienduizenden newsgroups.

Berichten die naar een newsgroup worden gestuurd en gelezen kunnen worden heten ’news messages’. De struktuur waaraan news messages moeten voldoen vertoont bijzonder veel overeenkomsten met e-mail berichten: aan news berichten worden een aantal extra eisen gesteld.

In tegenstelling tot de e-mail service, waarbij een gebruiker een e-mail bericht naar een ander gebruiker verstuurd, worden news berichten niet naar de e-mail server van een andere gebruiker gestuurd maar naar een news server. Voor het lezen van news berichten binnen een bepaalde newsgroup moeten deze opgehaald worden van een news server.

Wereldwijd zijn er talloze news servers die van elkaars bestaan afweten en nieuw binnengekomen news berichten aan elkaar doorgeven, ze vormen een virtueel netwerk binnen het Internet. Dit netwerk heet USENET.

De newsgroups zijn hierarchisch geordend, de naam van een newsgroup is gebaseerd op deze hierarchische ordening. Op het hoogste niveau van de hierarchy bestaan onder meer de volgende clusters:

- comp - over computer gerelateerde onderwerpen
- rec - bevat newgroups over recreatie gerelateerde onderwerpen
- sci - newgroups over wetenschappelijke onderwerpen
- soc - over maatschappelijke onderwerpen

Deze vier clusters zitten in de zogenaamde mainstream hierarchiën, d.w.z. ze maken onderdeel uit van het ‘officiele’ USENET en nieuwe news berichten worden dus doorgegeven aan alle news servers. De wijze waarop nieuwe newsgroups kunnen worden gevormd is aan regels gebonden om wildgroei te voorkomen. Voorts kent iedere newsgroup doorgaans een voorzitter om totale chaos binnen een nieuwsgroup te voorkomen.

Naast de ‘mainstream’ hierarchiën bestaan er ook alternatieve hierarchiën. Deze zijn volledig vrij voor wat betreft het oprichten van een nieuwe newsgroup. Formeel maken ze echter geen deel uit van USENET en berichten naar deze newsgroup worden dan ook niet doorgegeven naar andere news servers.

### 3.4.7 HyperText Transfer Protocol (HTTP)

Het HyperText Transfer Protocol (HTTP) is het meest gebruikte protocol op het World Wide Web. Met dit protocol kunnen zogenaamde hypertext en hypermedia bestanden worden verzonden: een hypertext bestand is een tekst bestand met daarin opgenomen verwijzingen naar andere bestanden, hypermedia bestanden zijn bestanden die andersoortige informatie bevatten zoals: beelden, audio en video. Met HTTP kunnen deze bestanden over het Internet worden verstuurd.

Bestanden waar naar verwezen wordt kunnen zich op willekeurige plaatsen op het Internet bevinden, en daarmee is technisch gezien de globalisering van informatie een feit. Essentieel voor een gebruiker is dat het ophalen van informatie snel gebeurt, met dit doel voor ogen is HTTP ontworpen. Het protocol maakt gebruik van TCP verbindingen. Een enkele transactie verloopt als volgt:

- eerst wordt een verbinding opgezet tussen de host en de (gewenste) server;
- vervolgens wordt een request verstuurd van de host naar de server;
- daarna volgt een response van de server naar de host;
- de transactie wordt afgesloten door het verbreken van de verbinding.

Wanneer een bestand wordt opgehaald, dan is het mogelijk dat er meerdere transacties nodig zijn.

### 3.4.8 Inter-ORB protocollen

CORBA wordt algemeen beschouwd als een van de belangrijkste open standaarden voor *middleware*, waarbij middleware gezien wordt als een softwarelaag tussen technologie-afhankelijke computer- en netwerkplatformen enerzijds en de eindgebruikerapplicaties anderzijds. Middleware heeft als taak de heterogeniteit van de onderliggende platformen af te schermen van de applicatie-ontwikkelaar en gebruiker, zodat applicaties sneller en gemakkelijker gebouwd en gebruikt kunnen worden. Hiertoe wordt elk platform uitgerust met een *Object Request Broker* (ORB) die verzoeken (*requests*) van een client aan een server 'object' en de eventuele antwoorden (*responses*) van het server object naar de client transporteert. Deze ORB service is onafhankelijk van de onderliggende platformen en van de lokatie, implementie en toestand van het server object.

Om het samenwerken van ORBs van verschillende fabrikanten te garanderen is een standaard inter-ORB protocol nodig. CORBA definieert twee niveaus van inter-ORB protocollen: het *General Inter-ORB Protocol* (*GIOP*) en (o.a.) het *Internet Inter-ORB Protocol* (*IIOP*). GIOP heeft betrekking op de algemeen geldende verzameling berichtenformaten en -representaties voor ORB-samenwerking over willekeurige verbindingsgeoriënteerde transportlaag services. IIOP specificeert de uitwisseling van GIOP-berichten over een TCP/IP-netwerk en maakt dus gebruik van de TCP transportlaag service (zie par. 3.3).

Naast IIOP zijn er nog andere mogelijkheden om GIOP-berichten uit te wisselen. Zo zijn er *Environment Specific Inter-ORB Protocol*s (*ESIOP*s) geïdentificeerd die samenwerking in omgevingen met een bestaande (andere) infrastructuur voor gedistribueerde applicaties, zoals het Distributed Computing Environment (DCE), mogelijk moet maken. Daarnaast kunnen in de toekomst inter-ORB protocollen voor andere netwerken dan TCP/IP ontwikkeld worden, bijvoorbeeld een inter-ORB protocol dat direct op een ATM netwerk wordt geplaatst.

### 3.4.9 HTML en XML

HTML (*Hyper Text Markup Language*) is momenteel de algemeen gebruikte 'taal' voor publicatie op het Web. Omdat met HTML 'documenten' worden geschreven die door iedere Web browser kunnen worden weergegeven, kunnen we HTML ook als een protocol (-taal) beschouwen. HTML-documenten zijn in deze visie PDUs.

HTML is gebaseerd op een vereenvoudigde deelverzameling van de *Standard Generalized Markup Language* (SGML). Een HTML-document bevat, naast de eigenlijke inhoud die door de eindgebruiker wordt bepaald, *markup* om de logische structuur van de inhoud te beschrijven. De markup bestaat uit zogenaamde *tags*, waarmee documentelementen worden afgebakend. Zo zijn er ondermeer tags en corresponderende documentelementen voor titels, paragrafen, lijsten en tabellen. Een belangrijke eigenschap van HTML is de mogelijkheid om in HTML-documenten verwijzingen te definiëren naar andere (delen van) documenten en naar afbeeldingen, foto's, pictogrammen etc. De verwijzing naar een ander document wordt door de browser zichtbaar gemaakt als een *hypertext link*: een op een speciale manier weergegeven stuk tekst, bijvoorbeeld een onderstreept woord. Door de pijl van de muis op de link te plaatsen en te klikken wordt het betreffende document opgehaald en door de browser getoond. Verwijzingen naar afbeeldingen e.d. worden gebruikt als flexibel alternatief voor het direct incorporeren van de vaak omvangrijke representaties ervan in documenten. De browser laad de afbeeldingen automatisch en toont de afbeeldingen op de juiste (met de markup beschreven) plaatsen in het document.

In principe kan er een onderscheid gemaakt worden tussen de abstracte representatie van een document, de weergave van het document op een of ander medium (bijv. een computerscherm), en de definitie van de structurerings- en verwerkingsregels m.b.t. het document (o.m. de verzameling gedefinieerde tags en hun betekenis). Een HTML-document definieert de abstracte representatie. De weergave (visuele effecten, maar ook de controle over alternatieve uitvoer zoals spraak of Braille) kan worden gedefinieerd door een *style sheet*, die in het HTML-document gerefereerd moet worden. De structurerings- en verwerkingsregels tenslotte kunnen gedefinieerd worden in een *document type definition* (DTD), die dan ook in het HTML-document gerefereerd moeten worden. De huidige versie van HTML is HTML/4.0. Deze versie kan verwijzen naar een style sheet die geschreven is in de HTML-specifieke *Cascading Style Sheet* taal en naar een HTML-DTD. De relatief late beschikbaarheid van een officiële HTML-DTD verklaart de vele (strict genomen) foutieve HTML-documenten die momenteel op het Web te vinden zijn.

XML (*eXtensible Markup Language*) is een gestroomlijnde versie van SGML, en oorspronkelijk bedacht als de opvolger van HTML. XML kan gebruikt worden om HTML uit te breiden, maar kan ook voor verschillende andere doeleinden gebruikt worden (bijv. om synchronisatie tussen mediacomponenten te beschrijven, of om Web interfaces te definiëren). XML maakt een stricter onderscheid tussen weergave, abstracte representatie en applicatie-specifieke regels dan HTML. Bovendien is XML uitbreidbaar, in tegenstelling tot HTML, in de zin dat het gebruikergedefinieerde tags toestaat en dus willekeurige types documentelementen. Er zijn dus vele verschillende DTDs mogelijke voor XML, die elk toegesneden zijn op een specifiek toepassingsgebied. Eén zo'n toepassingsgebied, waarvoor momenteel documentelement types worden ontwikkeld, is *electronic commerce / electronic data interchange (EDI)*.

# Literatuur


L. L. Peterson, B. Davie. *Computer Networks: A Systems Approach*. Morgan Kaufman, 1996.

A. Pope. *The CORBA Reference Guide - Understanding the Common Object Request Broker Architecture*. Addison-Wesley, 1998.

W. Stallings. *SNMP, SNMPv2 and RMON: Practical Network Management*. Addison-Wesley, 1996.

W.R. Stevens. *TCP/IP Illustrated, Volume 1: The Protocols*. Addison-Wesley, 1994.

W.R. Stevens. *TCP/IP Illustrated, Volume 3: TCP for Transactions, HTTP, NNTP and the UNIX Domain Protocols*. Addison-Wesley, 1996.

A. S. Tanenbaum. *Computer Networks*. Prentice-Hall, 1996


Meer informatie over standaarden is te vinden op de Web pages van de respectivelijke standaardisatie-organisaties:

ATM Forum, http://www.atmforum.com/
IEEE. URL: http://www.ieee.org/
IETF, http://www.ietf.org/

ISO, http://www.iso.org/
ITU, http://www.itu.int/publications/
OMG, http://www.omg.org/
W3C, http://www.w3.org/