SKALIERBARKEIT IM INTERNET

Problemstellung:

Wie kann die IP-Technologie in weltweiten Netzen eingesetzt werden?

Lernziel:

- Die Teilnehmer sollen die Skalierungsaspekte bei IP (Adressierung und Routing) erklären können.

Inhalt:

- Subnetting
- Classless Routing (CIDR)
- Interdomain Routing (BGP)
- Routing Areas (OSPF)

Skalierbarkeit im Internet

Abb. INT–1 zeigt die Struktur des Internets von 1990. Dort sind diverse sog. *Autonome Systeme* (AS) miteinander verbunden. Autonome Systeme sind dabei sowohl die angeschlossenen Universitäten (wie z.B. Stanford und Berkeley), als auch lokale Netzanbieter (Provider) wie z.B. BARRNET oder Westnet, aber auch das NSFNET Backbone, das alle Teile miteinander verbindet.

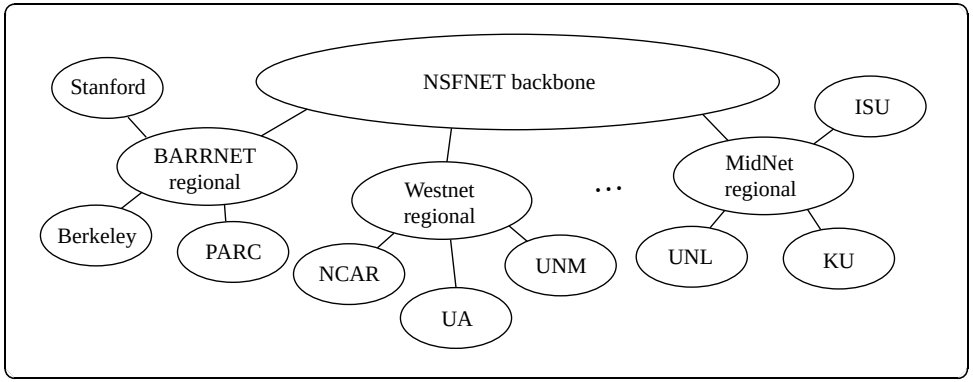

Abb. INT–1 Die Baumstruktur des Internets (1990)

Heute gibt es in etwa 50.000 autonome Systeme im Internet. Welche Skalierbarkeitsprobleme die mit sich bringt wird klar, wenn man sich die beiden folgenden Fakten vor Augen hält:

- Netzwerknummern werden an autonome Systeme vergeben.
- Routing erfolgt auf der Basis der Netzwerknummern.

Somit gibt es zwei zentrale Skalierbarkeitsprobleme:

1. Ausnutzung des Adreßraums
2. Handhabbarkeit der Routing-Tabellen

Diese Probleme sollen im Folgenden behandelt werden. Subnetting beschäftigt sich mit der Ausnutzung des Adreßraums. Classless Routing mit beiden Problemen, wobei Interdomain Routing sich mehr um das Routing-Problem dreht. IPv6 schließlich wird als Ausblick in zukünftige Entwicklungen kurz beleuchtet.

Subnetting

Subnetting verbessert die Ausnutzung des IP-Adreßraums. Die IP-Adressierung sieht im wesentlichen die folgenden drei Netzwerk-Klassen vor:

- **Klasse A**: 126 Netze zu jeweils 16 Millionen Rechnern
- **Klasse B**: 16.382 Netze zu jeweils 64K Rechnern
- **Klasse C**: 2 Millionen Netze zu jeweils 254 Rechnern

IP-Forwarding basiert auf der Annahme, daß Rechner im Netzwerk mit der gleichen Nummer sich direkt untereinander erreichen können. Somit ist es mehr als fraglich, inwieweit Klasse A Netzwerke sinnvoll sind. Aber auch bei Klasse B wird es zumindest sehr aufwendig, bis zu 64K Rechner in einem Netzwerk direkt zu verbinden.

Andererseits ist es ein erstrebenswertes Ziel, einem autonomen System genau eine Netzwerknummer zuzuweisen. Dadurch wird die Menge der Routing-Information (von außerhalb des AS) minimiert. Außerdem erlaubt dies dem AS, im Rahmen seines IP-Netzwerks, Adressen (z.B. an neue Rechner) zu vergeben, ohne mit Stellen außerhalb des eigenen Bereichs kommunizieren zu müssen. Deshalb wird üblicherweise angestrebt, einem AS ein Netzwerk einer Klasse zuzuweisen, das unter Berücksichtigung mittelfristigen Wachstums den Adreßbedarf des AS abdecken kann. Da Organisationen sehr schnell mehr als 254 Rechner haben können, sind Netze der Klasse B am interessantesten. Allerdings sind diese Netzwerke auch mittlerweile aufgebraucht.

Schließlich sollte noch bedacht werden, daß die starre Aufteilung des IP-Adreßraums in die drei Klassen zu potentiell sehr schlechter Ausnutzung des Adreßraums führt. Ein AS mit mehr als 254 Rechnern belegt automatisch 64K Adressen; ein AS mit zwei Rechnern belegt mindestens 254 Adressen. Eine flexiblere Zuweisung von Adreßkontingenten wird durch Classless Routing ermöglicht (siehe folgenden Abschnitt).

Hier soll zunächst die als Subnetting bekannte Technik vorgestellt werden, die es erlaubt, Netze der Klasse B (typischerweise) in mehrere Teile (Subnetze) aufzuspalten. Diese Technik erlaubt es dann beispielsweise einer Universität, ein Klasse B Netzwerk über seinen Campus zu verteilen und dabei mehrere LANs zu betreiben, die ein gemeinsames Campus-Netzwerk bilden.

Abb. INT–2 zeigt, wie die Adressierung mit Subnetzen funktioniert. Die Abbildung geht vom häufigsten Fall aus, dem Subnetting in Klasse B Netzen. Die Technik funktioniert aber auch für die anderen Klassen. Eine Klasse B Adresse besteht aus 16 Bit Netzwerknummer und 16 Bit Hostadresse. Für das Subnetting wird zusätzlich eine Subnetzmaske eingeführt. Alle Bits der Subnetzmaske, die auf "1" gesetzt sind, zeigen an welche Bits zur Subnetzdresse gehören. Alle "0" Bits bezeichnen den Rechner-spezifischen Teil der Adresse. Im Beispiel der Abbildung sind die ersten 24 Bits der Adresse zum Subnetz gehörend, während die letzten 8 Bits die Rechner (und Router) in den Subnetzen bezeichnen. Es hat sich als Konvention etabliert, daß jeweils die vordersten Bits fortlaufend zur Bestimmung von Subnetzen verwendet werden. Prinzipiell können auch Lücken in der Folge verwendet werden. Dies verkompliziert aber die Überschaubarkeit der Adressierung und wird nicht empfohlen. Im Beispiel der Abbildung wird somit ein Klasse B Netzwerk in 256 Teilnetze zu jeweils 254 Adressen unterteilt. Dies ist die am weitesten verbreitete Unterteilung.

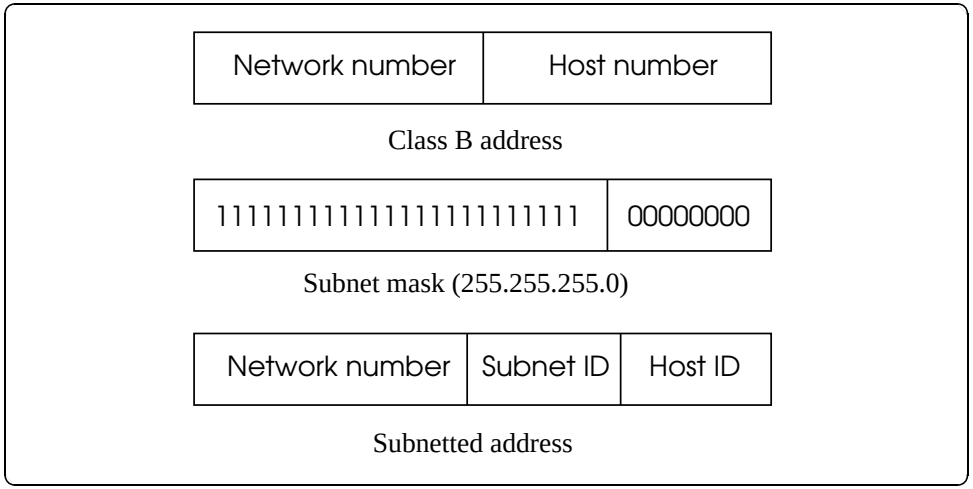

Abb. INT–2 Adressierung mit Subnetzen

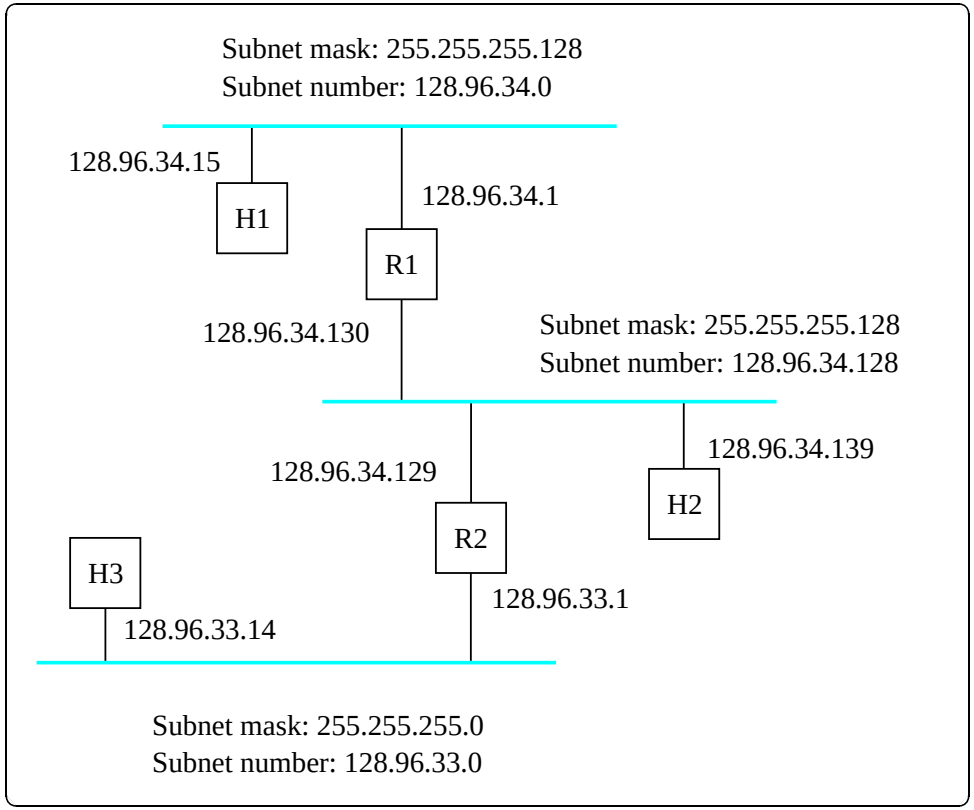

Abb. INT–3 Ein Beispiel für Subnetting

Abb. INT–3 zeigt ein (etwas ungewöhnliches) Beispiel für Subnetting. Hier werden 24 bzw. 25 Bits für die Subnetze verwendet. Dadurch wird deutlich, daß Subnetzmasken flexibel eingesetzt werden können, um ggf. Subnetze intern weiter aufzuteilen. Jeder Rechner bestimmt Subnetznummern durch eine bitweise UND-Verknüpfung zwischen der (eigenen oder fremden) Host-Adresse und der Subnetzmaske.

IP-Forwarding mit Subnetting

Durch die Einbeziehung der Subnetzmasken erweitert sich das IP-Forwading wie folgt. Für jeden Eintrag der Forwarding-Tabelle muß durch bitweises UND zwischen der Subnetzmaske und der Zieladresse festgestellt werden, oder die Zieladresse in diesem Subnetz liegt. Einer der Einträge der Forwarding-Tabelle ist ein *Default*-Eintrag.

```
D = destination IP address
for each forward table entry: <SubnetNumber, SubnetMask, NextHop>
    D1 = SubnetMask & D
    if D1 == SubnetNumber
      if NextHop is an interface
        deliver datagram directly to destination
      else
        deliver datagram to NextHop (a router)
```

Skalierbarkeit durch Subnetting

Der Nutzen der Subnetting-Technik kann wie folgt zusammengefaßt werden. Subnetting verbessert die Ausnutzung des Adreßraums, weil mehrere (logisch in Zusammenhang stehende) physikalische Netzwerke innerhalb eines Klasse B (oder Klasse C) Netzes zusammengefaßt werden können. Weiterhin wird die Routing Information außerhalb des in Subnetze aufgeteilten Netzes minimiert, weil für alle Subnetze gemeinsam nur ein einziger Routing-Eintrag existieren muß.

Präsenzübung: *LAN-Konfiguration*

- Das UNIX Kommando ifconfig wurde zum Anzeigen der Konfiguration von Netzwerk-Interface Parametern verwendet.

```
aretha:~ $ ifconfig lis0
lis0: flags=8e1<UP,NOTRAILERS,RUNNING,NOARP,SIMPLEX>
     inet 141.99.92.8 netmask ffffff00  ipmtu 4352

aretha:~ $ ifconfig lo0
lo0: flags=c89<UP,LOOPBACK,NOARP,MULTICAST,SIMPLEX>
     inet 127.0.0.1 netmask ff000000  ipmtu 1536

aretha:~ $ ifconfig tu0
tu0: flags=c63<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST,SIMPLEX>
     inet 141.99.130.50 netmask ffffff00 broadcast 141.99.130.255
     ipmtu 1500

paradox:~ $ ifconfig fta0
fta0: flags=8c63<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST,SIMPLEX>
     inet 141.99.216.12 netmask ffffff00 broadcast 141.99.216.255
     ipmtu 4352

paradox:~ $ ifconfig ln0
ln0: flags=c63<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST,SIMPLEX>
     inet 141.99.130.208 netmask ffffff00 broadcast 141.99.130.255
     ipmtu 1500

paradox:~ $ ifconfig lo0
lo0: flags=c89<UP,LOOPBACK,NOARP,MULTICAST,SIMPLEX>
     inet 127.0.0.1 netmask ff000000  ipmtu 1536
```

- Welche Information z.B. zur Adressierung und zur Netzwerk-Hardware entnehmen Sie diesen Angaben?

Traceroute

- Traceroute ist ein UNIX-Utility zum Debugging von Netzwerk-Verbindungen.
- Traceroute verfolgt Routing-Pfade vom Sender zum Empfänger.
- Brauchbare Alternative zur Record-Route Option von IP:
  - Die RR-Option wurde nicht von allen Routern implementiert.
  - Die RR-Option (von PING eingesetzt) benötigt auf dem Empänger-Rechner einen lauffähigen Dämon-Prozeß.
  - Der IP-Header hat nur Platz für bis zu neun Adressen.
    - ◇ Das reicht im heutigen Internet bei weitem nicht aus.
- Traceroute benötigt lediglich ICMP und beim Empänger-Rechner ein UDP–Modul.
- Das TTL-Feld im IP-Header wird auf jedem Router–Knoten dekrementiert, über den das Paket läuft. Wenn ein Router ein Paket länger als eine Sekunde puffert, dann wird TTL um die Anzahl Sekunden reduziert.
- Ein Router, der ein IP-Paket mit TTL $\in \{0, 1\}$ erhält, darf dieses nicht weiterleiten.
  - Stattdessen sendet er ein ICMP Paket "time exceeded" an den Sender zurück.
    - ◇ Darin gibt der betreffende Router seine Adresse an.
  - Ein Host hingegen darf ein Paket mit TTL $\in \{0, 1\}$ an eine Applikation zustellen.
- Traceroute beginnt nun, Pakete an die Zieladresse mit TTL = 1 abzusenden. Danach mit TTL = 2, TTL = 3 etc.
  - Dadurch sendet jeder Router auf dem Weg zum Empfänger je ein ICMP "time exceeded" mit seiner Adresse.
- Das Erreichen des Zielhosts muß anderweitig erkannt werden.
  - Dies geschieht, indem das gesendete IP-Paket ein UDP-Paket mit einer "unwahrscheinlichen" Port-Nummer (> 30.000) ist.
  - Wenn es (wie zu erwarten) auf der Empfänger–Maschine keinen Prozeß gibt, der auf diesem Port empfängt, erzeugt der Empfänger nun ein ICMP Paket "port unreachable".
- Traceroute bekommt so nun auf jeden Fall eine Fehlermeldung und kann damit den (beliebig langen) Weg zum Empfänger verfolgen.
- LAN-Beispiel:

(Zeigt den Einsatz von Subnetting an der Uni Siegen.)

```
paragon:~ $ traceroute www.uni-siegen.de
traceroute to sinfo.hrz.Uni-Siegen.DE (141.99.128.4), 30 hops max, 40 byte packets
 1  atm-gw.informatik.uni-siegen.de (141.99.92.254)  9 ms  8 ms  8 ms
 2  sinfo.hrz.uni-siegen.de (141.99.128.4)  3 ms  3 ms  4 ms
```

- WAN-Beispiel:

```
paragon:~ $ traceroute www.carleton.ca
traceroute to rideau.ccs.carleton.ca (134.117.1.17), 30 hops max, 40 byte packets
 1  atm-gw.informatik.uni-siegen.de (141.99.92.254)  8 ms  8 ms  8 ms
 2  SI-B-WiN-Router.hrz.uni-siegen.de (141.99.128.249)  3 ms  2 ms  2 ms
 3  Uni-Siegen1.WiN-IP.DFN.DE (188.1.6.85)  3 ms  4 ms  2 ms
 4  ZR-Koeln1.WiN-IP.DFN.DE (188.1.160.25)  9 ms  6 ms  5 ms
 5  ZR-Frankfurt1.WiN-IP.DFN.DE (188.1.144.34)  10 ms  8 ms  8 ms
 6  IR-Perryman1.WiN-IP.DFN.DE (188.1.144.78)  96 ms  96 ms  97 ms
 7  bordercore3-hssi0-0.Washington.mci.net (166.48.41.249)  98 ms  97 ms  100 ms
 8  * * core2.NorthRoyalton.mci.net (204.70.4.45)  137 ms
 9  core2-hssi-3.Chicago.mci.net (204.70.1.254)  194 ms  287 ms  385 ms
10  borderx2-fddi-1.Chicago.mci.net (204.70.185.68)  150 ms  150 ms  145 ms
11  canet.Chicago.mci.net (204.70.185.122)  157 ms  162 ms  153 ms
12  psp.on.canet.ca (205.207.238.141)  160 ms  162 ms  166 ms
13  exterior.onet.on.ca (192.68.55.102)  164 ms *  167 ms
14  ott-rt1-exterior-if.onet.on.ca (130.185.4.10)  173 ms  165 ms  171 ms
15  ottawa1-ott-rt1-if.onet.on.ca (130.185.23.1)  170 ms *  173 ms
16  * carleton-ottawa1-if.onet.on.ca (130.185.16.18)  826 ms  517 ms
17  onet-gate.carleton.ca (134.117.18.1)  495 ms  881 ms  717 ms
18  rideau.ccs.carleton.ca (134.117.1.17)  633 ms  639 ms  549 ms
```


Classless Routing (CIDR)

Bis heute sind allein die Netzwerknummern der Klasse B aufgebraucht. Adressen der Klasse C sind jedoch noch reichlich(?) vorhanden. Generell könnte man also mehrere Netze der Klasse C an ein autonomes System ausgeben. Ein Vorteil dieses Verfahrens wäre die flexible Anpassung der Adreßraumvergabe an die Größe des AS (in Einheiten von 254 Rechnern). Allerdings wird dadurch das Routing erheblich verkompliziert, weil ein AS nun viele Routing-Einträge (von außerhalb) haben muß. Weiterhin verkomplizieren sich auch die Routing-Tabellen für Verkehr innerhalb des AS.

Das Problem läßt sich lösen, wenn das Routing nicht starr an den Grenzen der Adreß-Klassen ausgerichtet wird. Dann kann ein AS beispielsweise mehrere fortlaufende Klasse-C Adressen erhalten und diesen Block wie eine einzige Netzwerk-Adresse verwenden. Genau dies geschieht beim *Classless Interdomain Routing (CIDR)*.

Beispiel für einen Block von Klasse C Adressen: Die Netze 192.4.16 bis 192.4.31 werden an ein AS vergeben. Die ersten 20 Bits der Adressen sind dabei identisch:
1100.0000 0000.0100 0001 Dies ergibt einen Block von 4K Adressen.

Als Schreibweise hat sich die Angabe des Präfix gefolgt von der Anzahl signifikanter Bits eingebürgert, also in unserem Beispiel 192.4.16/20.

Für die praktische Einsetzbarkeit von CIDR müssen die verwendeten Routing-Protokolle damit umgehen können. Ein Beispiel dafür ist BGP, das vorwiegend zwischen AS im Internet eingesetzt wird. Die CIDR-Adressierung ist auch (in Analogie zum Subnetting) als *Supernetting* bekannt.

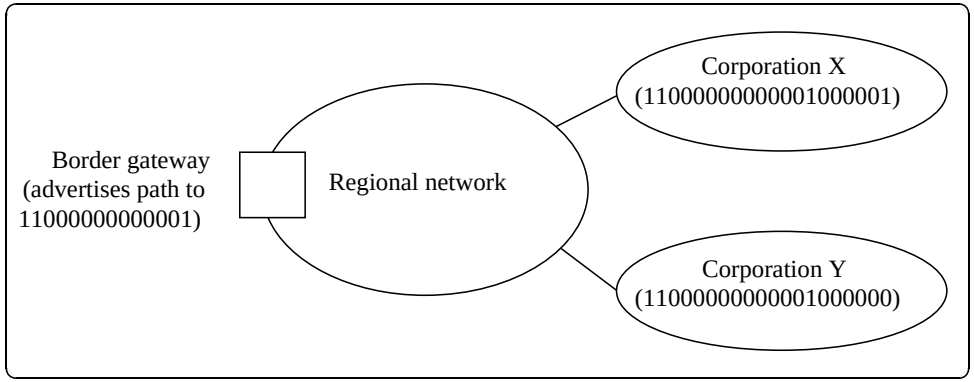

Abb. INT–4 Zusammenfassung von Routen mit CIDR)

Abb. INT–4 zeigt eine weitere Einsatzmöglichkeit von CIDR. Große Adreßblöcke werden an Netzwerk-Provider vergeben, die dann über einen einzigen Routing-Eintrag von außen erreichbar sind. Innerhalb des Provider-Netzwerks kann dann ein solcher Adreßblock weiter aufgespalten werden.

IP-Forwarding mit CIDR

Der Einsatz von CIDR verändert natürlich auch die Forwarding-Mechanismen. Durch die Adreß-Präfixe variabler Länge ergeben sich auch überlappende Präfixe. Beispielsweise könnte eine Forwarding-Tabelle die beiden folgenden Einträge enthalten:

- (171.69/16, R1)
- (171.69.10/24, R2)

Ein ankommendes Packet für 171.69.10.5 könnte theoretisch von beiden Einträgen behandelt werden. Offensichtlich ist aber der Eintrag für R2 gemeint. Somit muß mit CIDR der jeweils längste passende Eintrag verwendet werden. Ein anderes Packet, bestimmt für 171.69.20.5, passt nur zum Eintrag für Router R1, der somit der längste passende Eintrag für dieses Packet ist. Durch die Existenz von Einträgen verschiedener Länge muß die Forwarding-Tabelle prinzipiell als Forwarding-Baum organisiert werden, um die mehr oder weniger spezifischen Einträge korrekt zu verarbeiten.

Interdomain Routing (BGP)

Abb. INT–5 zeigt ein Netzwerk mit zwei autonomen Systemen. Charakteristisch ist die Anordnung der Router. R2 und R4 sind sogenannte *Border Gateways*, die an der Grenze der autonomen Systeme Daten-Verkehr in das jeweils andere AS weiterleiten. Die Router innerhalb der AS werden *Interior Gateways* genannt. (Vergleiche Kapitel *Routing*!) Verfahren für das Routing innerhalb eines AS wurden in jenem Kapitel behandelt. Da ein AS auch als (administrative) *Domain* bezeichnet wird, werden Verfahren für das Routing zwischen mehreren AS als *Interdomain Routing* Verfahren bezeichnet.

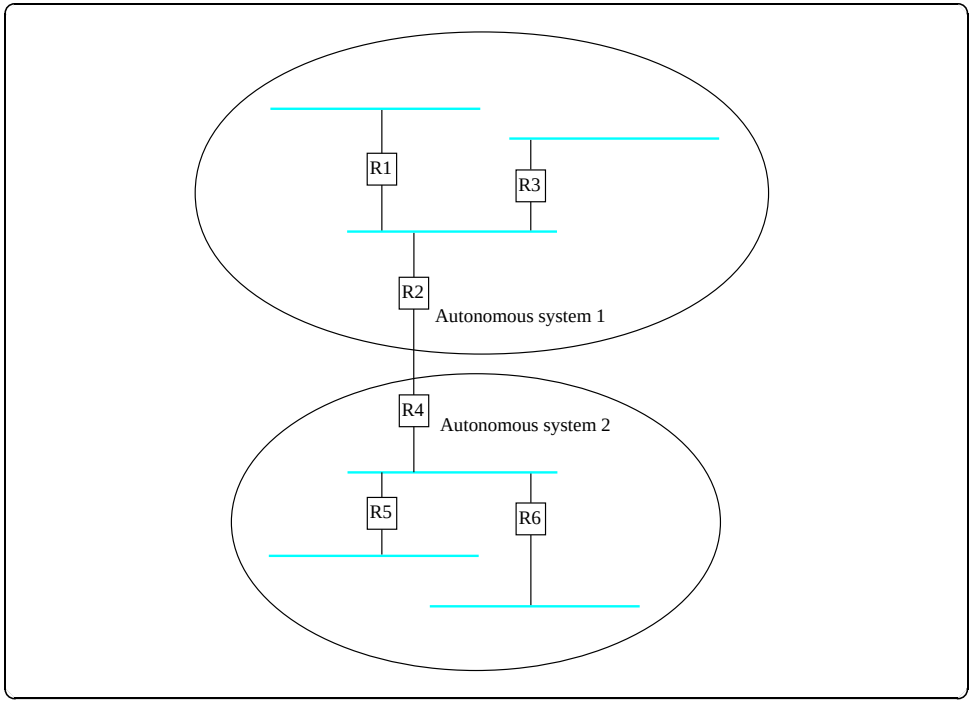

Abb. INT–5 Ein Netzwerk mit zwei autonomen Systemen (AS)

Klassifikation Autonomer Systeme

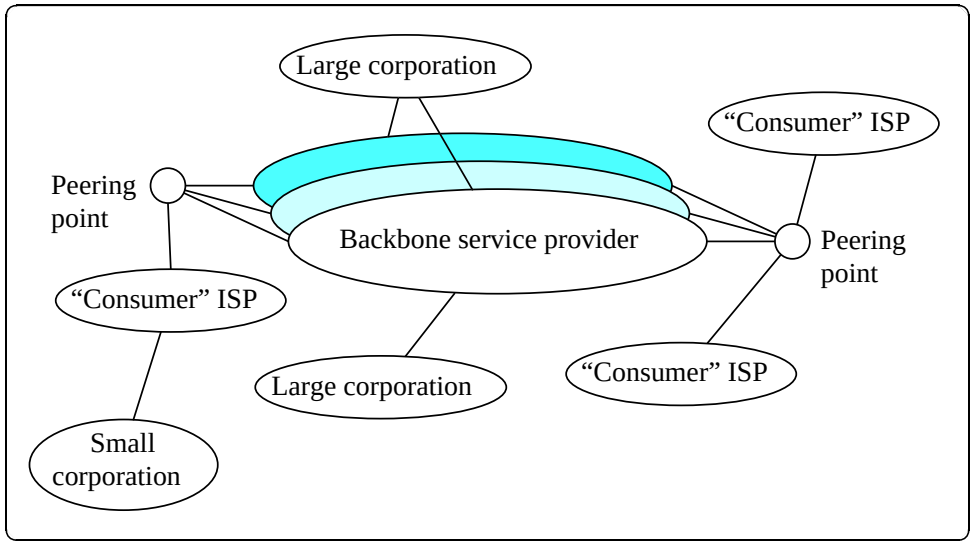

Abb. INT–6 Die Multi-Backbone Struktur des heutigen Internet

Abb. INT–6 zeigt die abstrakte Struktur des heutigen Internets. Daraus läßt sich die folgende Klassifikation für autonome Systeme ableiten. Generell unterscheidet man zwischen *lokalem Verkehr* und *Transit-Verkehr*. Bei lokalem Verkehr liegt ein Endpunkt einer Kommunikation im jeweiligen AS. Transit-Verkehr durchquert ein AS.

- **Stub AS**: hat einen einzigen Anschluß an ein anderes AS. Ein Stub AS hat ausschließlich lokalen Verkehr.
- **Multihomed AS**: hat mehrere Anschlüsse an andere AS, verweigert aber Transit-Verkehr.
- **Transit AS**: hat mehrere Anschlüsse an andere AS und leitet Transit-Verkehr durch (neben möglichem lokalem Verkehr).

Im Beispiel der Abb. INT–6 ist die “Small Corporation” ein Stub AS, die “Large Corporation” am oberen Bildrand ist ein Multihomed AS, während der “Consumer ISP” links sowie die “Backbone Service Provider” Transit AS darstellen.

Das wichtigste Ziel von *Interior Gateway Protocols* (wie z.B. RIP und OSPF) ist das Auffinden optimaler (kurzer) Pfade innerhalb einer Routing Domain (eines autonomen Systems). Bei *Border Gateway Protocols* ist das Ziel etwas anders. Zunächst einmal ist die Routing-Domain wesentlich größer. Heute gibt es in etwa 50.000 Einträge. Dies ist bedingt durch CIDR – ohne CIDR wären es noch wesentlich mehr. Weiterhin sind die autonomen Systeme nun einmal autonom: die lokal verwendeten Routing-Verfahren werden vollständig lokal im AS verwaltet. Es steht jedem AS frei, das lokale Routing je nach eigenen Interessen zu organisieren. Als Konsequenz gibt es keine einheitliche Metrik. Numerische Bewertungen einzelner Links sind deshalb für die Verfahren anderer AS bedeutungslos, da hierüber keine Information vorliegt. Schließlich ist es auch fraglich inwieweit ein AS den Routen-Bewertungen anderer AS vertrauen könnte; die Bewertungen könnten eventuell aus kommerziellen Interessen oder aus anderen Gründen manipuliert oder zumindest unzuverlässig sein. Weiterhin werden Routen “politisch” bestimmt. Ein multihomed AS wäre zwar rein technisch in der Lage, Transit-Verkehr durchzuleiten, verweigert dies allerdings, da dies nicht seine Aufgabe ist. (Beispielsweise könnte ibm.com sehr wohl weltweit Transit-Verkehr weiterleiten, würde dadurch aber seine eigenen Netze belasten.) Abgesehen davon werden Routen durch Geschäftspolitik und Strategien bestimmt. Dies hängt davon ab, welche AS mit welchen anderen AS beispielsweise Abkommen zur Weiterleitung unterhalten. (Dies ist schließlich die Geschäftsgrundlage der Service-Provider.)

Das Interdomain Routing muß von beliebig verbundenen Autonomen Systemen ausgehen; die Verbindungen stellen einen zwar zusammenhängenden, aber unstrukturierter Graphen gegebenenfalls mit Zyklen dar. Wichtigstes Ziel des Interdomain Routings ist somit die Erreichbarkeit aller autonomen Systeme auf zyklenfreien Pfaden. Gute Pfade (als Approximation an optimale Pfade) sind dabei von sekundärer Bedeutung.

Border Gateway Protocol (BGP)

BGP (in der aktuellen Version BGP-4) wird heute im Internet zum Interdomain Routing eingesetzt. Jedes teilnehmende AS ernennt dabei ein oder mehrere sogenannte *Speaker*, die mit den Speakern der anderen AS Routing-Information austauschen. Neben den Speakern haben die AS dann noch ein oder mehrere Gateways (Router), die ein- und ausgehenden Verkehr weiterleiten. Speaker und Gateways können auch in einem Gerät vereinigt sein; zur Lastbalancierung wird dies aber üblicherweise aufgeteilt.

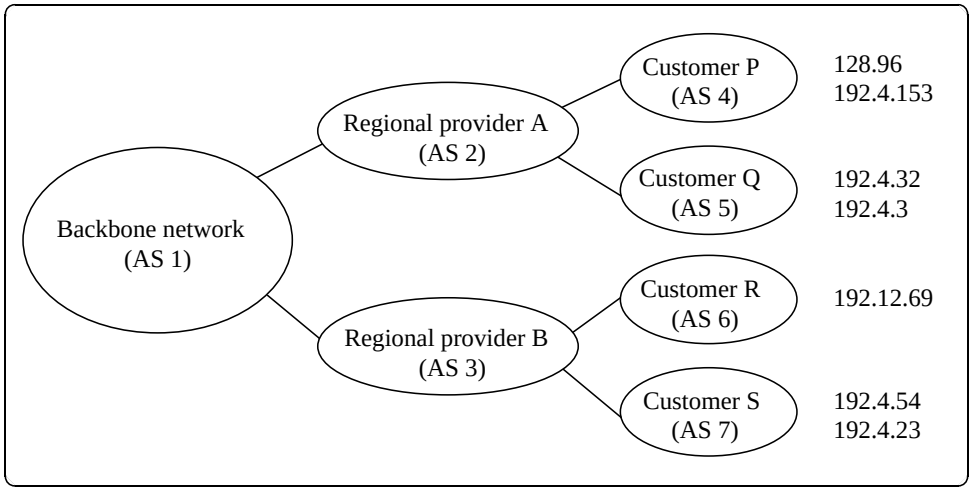

Abb. INT–7 Beispiel für ein Netzwerk mit BGP

Abb. INT–7 zeigt ein Beispiel für BGP. Die autonomen Systeme AS2 und AS3 annoncieren jeweils, daß die rechts angegebenen Netzwerknummern direkt über sie erreichbar sind. Der Backbone, AS1, annonciert die Erreichbarkeit über den Pfad (AS1, AS2) bzw. (AS1, AS3). Bei BGP werden jeweils komplette Pfade vom AS bis zum (Endpunkt) Netzwerk weitergeleitet. Dieses Verfahren vermeidet zirkuläre Pfade, die in einem Multibackbone-Netzwerk sonst sehr schnell entstehen könnten. Weiterhin werden bei BGP die Routing-Entscheidungen von "politischen" der AS-Betreiber abhängig gemacht. Dabei können Routen über bestimmte andere AS (Netzwerk-Betreiber oder z.B. Staaten) ausgeschlossen werden.

Bei BGP werden die Autonomen Systeme über weltweit eindeutige numerische Adressen identifiziert. Genau wie IP-Netznummern werden die AS-Identifikationen von einer zentralen Stelle zugeteilt. Die globale Eindeutigkeit der AS-Identifikation wird zum Funktionieren des Verfahrens benötigt. Im Format der BGP-Nachrichten sind AS-Identifikationen als 16 bit-Zahlen festgeschrieben. Dadurch sind ca. 65.000 Autonome Systeme unterscheidbar. Vergleicht man dies mit den derzeit 50.000 bestehenden Systemen, so scheint die Obergrenze in naher Zukunft erreicht zu werden. Allerdings müssen

nur nicht-Stub AS eindeutig identifiziert werden, um Zyklen in Pfaden zu vermeiden. Da die weitaus größte Anzahl bestehender AS jedoch Stub-AS sind, können diese uneindeutig bezeichnet werden, so daß die AS-Nummerierung in BGP in absehbarer Zeit nicht Gefahr läuft an ihre Grenzen zu stoßen.

BGP-4 ist direkt auf die Verwendung von CIDR abgestimmt; alle Netzwerke werden in der CIDR Präfix-Notation mit variabler Länge identifiziert. Zur Weitermeldung nicht mehr bestehender (oder defekter) Routen verwendet BGP auch negative Information; die sogenannten "withdrawn routes" werden zusammen mit den positiven Routing-Informationen weitergeleitet, um die weitere Verwendung der nicht mehr bestehenden Pfade möglichst schnell zu unterbinden.

Skalierbarkeit mit BGP

- Das Interdomain-Routing basiert auf der Anzahl der BGP-Knoten, die der Anzahl der Autonomen Systeme entspricht.
- Die Optimierung der Routen zwischen den AS kann sich auf das Finden von Routen zwischen Border Gateways beschränken.
- Das Intradomain-Routing ist in seiner Komplexität auf die Anzahl der Netzwerke innerhalb eines AS beschränkt.

Integration von Intradomain und Interdomain Routing

Zum Interdomain Routing wird üblicherweise BGP verwendet. Innerhalb der autonomen Systeme (Domains) werden jeweils lokale Routing-Verfahren eingesetzt, die nicht zwischen das AS abgestimmt werden. Dadurch kommt den Border Gateways die Rolle zu, an beiden Routing-Systemen teilzunehmen. Analog zu Abb. INT–7 gibt es dabei drei Varianten, nach denen dies geschehen kann.

- **Customer Netzwerk**: ein Customer Netzwerk ist ein Stub AS oder Multihomed AS. Sein(e) Border Gateway Router dienen dabei als Default Router für ein- und ausgehenden Verkehr. Die Router innerhalb des AS senden deshalb ausgehenden Verkehr einfach an ihren Default Router. Der oder die Default Router kommunizieren dann mit den Border Gateways der AS, an denen sie angeschlossen sind.
- **Provider Netzwerk**: Ein (lokales) Provider Netzwerk liegt in einem Baum-artigen Bereich des Netzes, der noch nicht zum Backbone selbst gehört. Dadurch wird die Struktur noch relativ einfach gehalten; es gibt noch eine Default-Route für unbekannte Ziele. Deshalb kann innerhalb eines solchen Provider AS auch noch ein Verfahren wie OSPF angewendet werden, wobei die Border Gateways Routen zu den Customer Netzwerken annoncieren.
- **Backbone Netzwerk**: Ein Backbone Netzwerk kennt keine Default Route mehr. Für jedes Netzwerk muß hier eine Route bekannt sein. Deshalb muß hier mit wesentlich mehr (z.B. 50.000) Einträgen gearbeitet werden. Dafür wird ein als *Interior BGP* (IBGP) bekanntes, zweistufiges Verfahren angewendet. Dabei liefert BGP mit den anderen AS die Abbildung von Netzwerk-Adressen auf die Border-Router des AS. Innerhalb des AS (zwischen allen lokalen Routern und den eigenen Border Routern) werden die kürzesten Pfade untereinander bestimmt (z.B. mit OSPF). Ein Border-Router, der ein Packet von außen empfängt, sucht dann zuerst den zugehörigen Ausgangs-Border-Router, und benutzt dorthin die jeweils optimale Route innerhalb des eigenen AS.

Routing Areas (OSPF)

BGP ist das im Internet verwendete Interdomain Routing Verfahren, das im Hinblick auf Skalierbarkeit für große Systeme entworfen wurde. Für Intradomain Routing (innerhalb eines Autonomen Systems, "AS") wurden im Kapitel "Routing" die zwei Verfahren RIP und OSPF vorgestellt. Beide Verfahren sind jedoch nur in AS von begrenzter Größe einsetzbar. RIP kann nur Pfade mit maximaler Länge 15 behandeln (wegen des Count-to-Infinity Problems). OSPF benötigt Flooding als Kommunikationsmechanismus und den relativ zeitaufwändigen Shortest-Path Algorithmus und wird dadurch in seiner Skalierbarkeit limitiert.

Allerdings verfügt OSPF noch über eine bislang nicht erwähnte Hierarchieebene, die sogenannten Routing Areas. Abb. INT–8 zeigt ein Autonomes System, das in mehrere solcher Areas unterteilt ist.

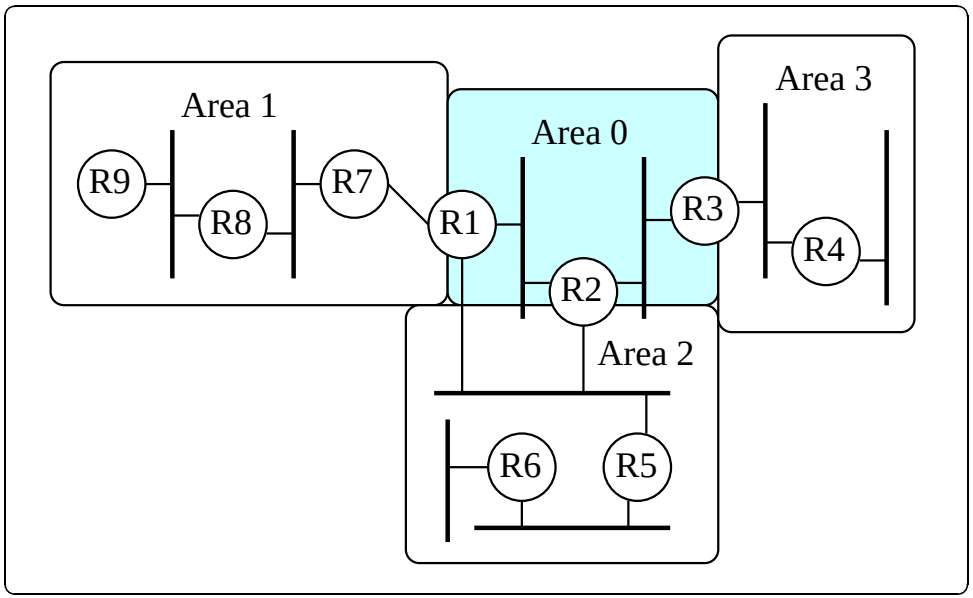

Abb. INT–8 Ein Autonomes System unterteilt in Routing Areas

Eine Area hat dabei einen besonderen Status. Dies ist Area 0, die auch Backbone Area genannt wird. Die Areas werden untereinander über die "Area Border Router" (ABR) verbunden. (nicht zu verwechseln mit den Border Gateways von BGP) Ein ABR ist somit Bestandteil sowohl der Backbone Area als auch von einer oder mehreren anderen Areas. Innerhalb einer Area findet das Routing wie bereits bekannt über OSPF statt. Allerdings werden die Link-State Packete eines nicht-ABR Routers nicht über die Area-Grenze hinaus weitergeleitet. Dadurch reduziert sich der Aufwand für Flooding und Routen-Berechnung und macht beides besser skalierbar. Allerdings müssen die über die nicht-ABR Router erreichbaren Pfade auch außerhalb der jeweiligen Area bekannt gemacht werden. Zu diesem Zweck sendet jeder ABR Information über solche Pfade weiter und gibt dabei an, daß die entsprechenden Netze über den ABR selbst erreichbar seien. Die Metriken werden dabei entsprechend angepasst, so daß die Kosten der Pfade korrekt weiter gegeben werden.

Nachrichten von einer Area zur anderen werden immer durch die Backbone Area geleitet. Dies ist eine Konsequenz des Designs des soweit beschriebenen Routing-Verfahrens. Wenn eine Area mit mehr als einem ABR am Backbone abgeschlossen ist, so werden für verschiedene Ziel-Areas ggf. auch die entsprechend günstigeren Pfade verwendet. (Wie beschrieben leiten die ABR die Metriken entsprechend weiter.) Im Beispiel von Abb. INT–8 ist Area 2 über R1 und R2 an der Area 0 angeschlossen. Für Ziele in Area 1 wird dabei R1 verwendet. Ziele in Area 3 hingegen werden über R2 geroutet.

Allerdings werden mögliche direkte Verbindungen zwischen nicht-Backbone Areas nicht verwendet. Dadurch sind die verwendeten Pfade gegebenenfalls nicht optimal. Dies ist der Preis, der für die bessere Skalierbarkeit zu zahlen ist. Der Tradeoff zwischen perfekter Optimalität und Skalierbarkeit eines Verfahrens ist ein häufig wiederkehrendes Problem in Rechnernetzen. (Zum Beispiel wird bei BGP die Optimalität des Routings noch weit mehr zugunsten der Skalierbarkeit vernachlässigt.)