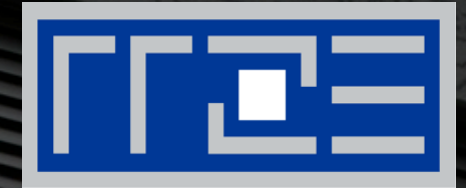


REGIONALES RECHENZENTRUM ERLANGEN [RRZE]



Routing im Intra- und Internet

Helmut Wünsch
RRZE-Kolloquium
Vorlesung „Grundzüge der Datenkommunikation“
27. Januar 2016

Themen

- Routing – allgemein
 - Statisches Routing
 - Dynamisches Routing
 - Routing im Internet
- Routingprotokolle
 - distance vector (Bsp. RIP)
 - link state (Bsp. OSPF)
 - path vector (Bsp. BGP)

Routing

Was ist Routing?

- Alle Rechner in einem lokalen Netz (LAN) können untereinander direkt (auf Layer2) kommunizieren
- Ein lokales Netz ist (wie der Name schon sagt) lokal begrenzt
- Wollen Rechner eines LANs mit Rechnern eines anderen LANs kommunizieren, braucht es sog. *Router*
- Ein Router ist i.d.R. ein dediziertes Gerät,
 - › welches mehrere Schnittstellen besitzt,
 - › an welche wiederum jeweils LANs oder auch andere, (z.T. weit entfernte) Router angeschlossen sind,
 - › zwischen denen der Router Datenpakete (nur Layer3, d.h. IP) umsetzt

Beispiele von Routern



Quelle: wikipedia.org

Typischer Router für den Heimbereich:
Vermittlung zwischen zwei Netzen

- LAN / WLAN (Heimnetz)
- WAN (Internet)



Quelle: wikipedia.org

Hochleistungsrouter im Providerumfeld:
I.d.R. viele Schnittstellen, Vermittlung
zwischen Tausenden von Netzbereichen

Routing

- Warum Routing?
 - Logische (effiziente) Trennung von Subnetzen
 - Skalierung: Sehr Dynamisches Wachstum von Netzwerken möglich
 - Sicherheit (Möglichkeit der Zugriffskontrolle)
 - Stabilität (Forwarding vs. Flooding)
 - Erhöhung von Redundanz und Performance

Grobe Funktionsweise Routing im LAN mit IPv4/IPv6

1. Fall (kein Router nötig)

IP-Adressen von Quell- und Zielrechner sitzen im gleichen LAN

Bsp.: alle Rechner im Netz `192.168.1.* /24`

- Quellrechner kann dann das IP-Paket direkt an MAC-Adresse des Zielrechners im LAN schicken
- Ermittlung der MAC-Adresse des Zielrechners:
 - › Bei IPv4: per ARP-Request
 - › Bei IPv6: per NDP-Request
- Woher weiß Quellrechner, dass Zieladresse im gleichen LAN angesiedelt ist?
- Durch die Netzmaske!
 - › definiert die „Größe“ eines IP-Netzes
 - › Sollte tunlichst auf jedem Rechner im selben LAN gleich konfiguriert sein!

/slash	# Hosts	Netmask	Wildcard
/30	4	255.255.255.252	0.0.0.3
/29	8	255.255.255.248	0.0.0.7
/28	16	255.255.255.240	0.0.0.15
/27	32	255.255.255.224	0.0.0.31
/26	64	255.255.255.192	0.0.0.63
/25	128	255.255.255.128	0.0.0.127
/24	256	255.255.255.0	0.0.0.255
/23	512	255.255.254.0	0.0.1.255
/22	1,024	255.255.252.0	0.0.3.255
/21	2,048	255.255.248.0	0.0.7.255
/20	4,096	255.255.240.0	0.0.15.255
/19	8,192	255.255.224.0	0.0.32.255
/18	16,384	255.255.192.0	0.0.63.255
/17	32,768	255.255.128.0	0.0.127.255
/16	65,536	255.255.0.0	0.0.255.255
/15	131,072	255.254.0.0	0.1.255.255
/14	262,144	255.252.0.0	0.3.255.255
/13	524,288	255.248.0.0	0.7.255.255
/12	1,048,576	255.240.0.0	0.15.255.255
/11	2,097,152	255.224.0.0	0.31.255.255
/10	4,194,304	255.192.0.0	0.63.255.255
/9	8,388,608	255.128.0.0	0.127.255.255
/8	16,777,216	255.0.0.0	0.255.255.255

Quelle: c128.com

Grobe Funktionsweise Routing im LAN mit IPv4/IPv6

2. Fall (mit Router)

IP-Adressen von Quell- und Zielrechner in unterschiedlichen LANs

- D.h. Quellrechner hat Anhand seiner Netzmaske ermittelt, dass Zielpartner nicht im selben LAN sitzt
 - › Bsp: Kommunikation 192.168.0.10 nach 192.168.178.30 (jew. /24)
- Quellrechner schickt das IP-Paket an MAC-Adresse des Routers
- Dazu muss dem Quellrechner der Router im LAN bekannt sein!
(Genauso wie Netzmaske essentieller Teil der Netzwerkkonfiguration)
- Router leitet IP-Paket weiter:
 - › An MAC-Adresse des Empfängers im Ziel-LAN, sofern er dieses direkt erreichen kann
 - › An anderen Router

Status: **Connected**
Ethernet is currently active and has the IP address 192.168.1.10.

Configure IPv4:

IP Address:

Subnet Mask:

Router:

DNS Server:

Search Domains:

Grobe Funktionsweise Routing im LAN mit IPv4/IPv6

- Woher weiß der Router, wo welche Zielnetze liegen?
 - › Jeder Router führt eine Art „Adressdatenbank“, die sog. *Routing-Tabelle*.
- Routing-Tabelle enthält Infos zu
 - › direkt am Router erreichbaren LANs („directly connected“),
 - › zu entfernten, über andere angeschlossene Router erreichbare Netze („next hop“ routing)
- Aufbau einer Routingtabelle: i.d.R. mind. Zielnetz, Zielinterface, Metrik
- Größe der Routingtabelle kann zwischen zwei Netzen (DSL-Router) bis hin zu mehrere hunderttausend Netzen (Internet BGP Router) umfassen

Ziel	Intf.	Metrik
192.168.0.0/24	Interface A	1
192.168.0.0/24	Interface B	5
192.168.178.0/24	Interface C	1
10.0.0.0/16	Interface D	1

Exemplarische Routingtabelle

Routing

Wie wird die Routingtabelle aufgebaut?

- **Statisches Routing:**
Manuelle Konfiguration der Routing-Tabelle auf Router
- **Dynamisches Routing:**
Router unterhalten sich untereinander und bauen Routingtabelle selbstständig über Routingprotokolle auf

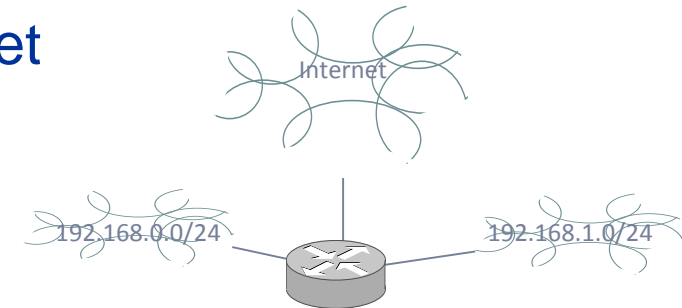
Statisches Routing (I)

■ Einfachstes Bsp: DSL-Router zuhause

- z.B. „Fritzbox“
- Routet i.d.R. nur zwischen zwei Netzen: Lokales Heimnetz und Internet
- Triviale Routingtabelle:
 - › Heimnetzwerk (z.B. 192.168.178.0/24) → LAN-Port 1-4
 - › Internet („default route“) → LAN-Port 5

■ Anderes Bsp: Firewall-Router für kleines Firmen-Intranet

- Z.B. jeweils eine Netzwerkschnittstelle für
 - › Bürorechner 192.168.0.0/24
 - › Servernetz 192.168.1.0/24
 - › Internet



- zugehörige statische Routingtabelle:

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
192.168.0.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
0.0.0.0	0.0.0.0	0.0.0.0	U	0	0	0	ppp0

Dynamisches Routing: Sinn und Zweck

Durch dynamisches Routing sollen Router...

- ... Routinginformationen selbständig untereinander austauschen
- ... selbständig die Netztopologie „lernen“
- ... somit selbständig für jedes Paket den jeweils besten Weg zum Ziel ermitteln
- ... selbständig auf Veränderungen in der Topologie reagieren
- ... gut wie möglich Fehler vermeiden (z.B. Schleifentopologien)

Statisches vs. Dynamisches Routing

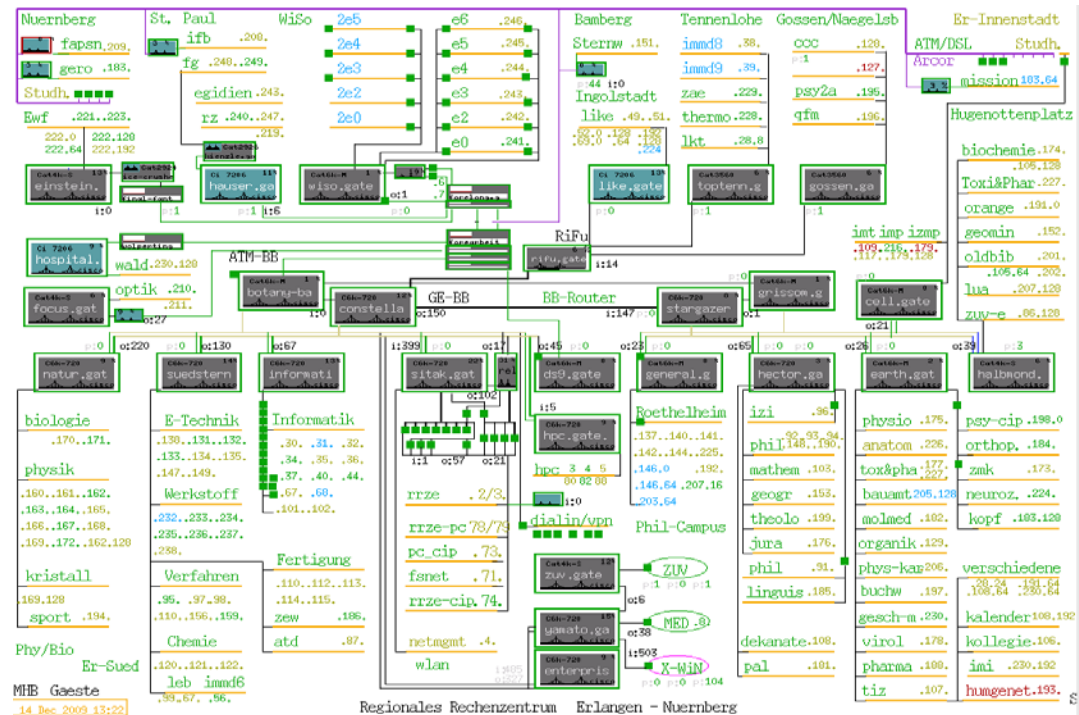
Gegenüberstellung

- Dynamisches Routing kann durch falsche Informationen gestört werden
- Dynamisches Routing erzeugt Netzlast
- Statisches Routing nur bei einfachen Netztopologien handhabbar
- Keine Backup-Pfade bei statischem Routing
- Statisches Routing mit mehr als einem Router ist arbeitsintensiv bei Änderungen und fehleranfällig

Bsp. Dynamisches Routing (I)

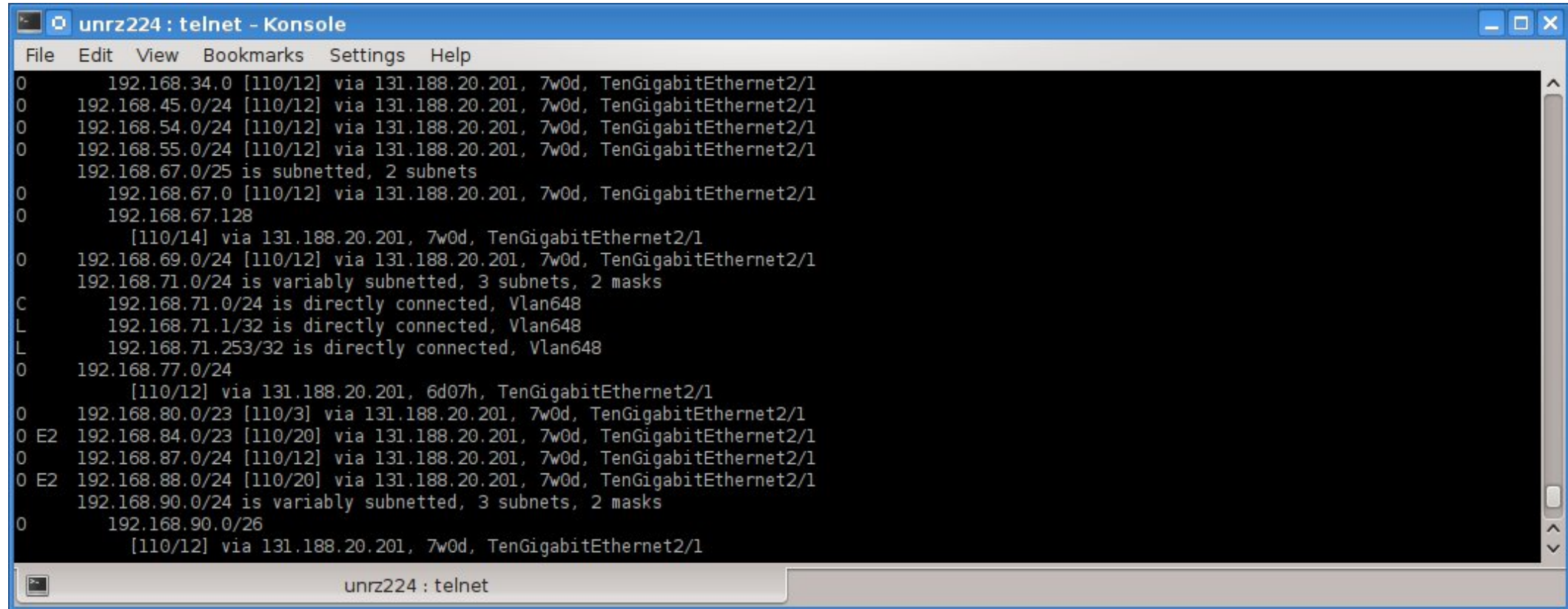
FAU Backbonenetz

- Universitärer Backbone aufgebaut mit ca. 30 Backbone-Routern, verteilt auf mehrere Städte
- Statisches Routing auf jeden Router indiskutabel
- Änderung des Netzes an beliebiger Stelle lässt Backbone automatisch reorganisieren
- verschiedene Subnetze in den Routingtabellen!



Bsp. Dynamisches Routing (II)

Uni-Backbone: Auszug Routingtabelle



```
unrz224 : telnet - Konsole
File Edit View Bookmarks Settings Help
0      192.168.34.0 [110/12] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0      192.168.45.0/24 [110/12] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0      192.168.54.0/24 [110/12] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0      192.168.55.0/24 [110/12] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0      192.168.67.0/25 is subnetted, 2 subnets
0      192.168.67.0 [110/12] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0      192.168.67.128
0      [110/14] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0      192.168.69.0/24 [110/12] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0      192.168.71.0/24 is variably subnetted, 3 subnets, 2 masks
C      192.168.71.0/24 is directly connected, Vlan648
L      192.168.71.1/32 is directly connected, Vlan648
L      192.168.71.253/32 is directly connected, Vlan648
0      192.168.77.0/24
0      [110/12] via 131.188.20.201, 6d07h, TenGigabitEthernet2/1
0      192.168.80.0/23 [110/3] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0 E2   192.168.84.0/23 [110/20] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0      192.168.87.0/24 [110/12] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0 E2   192.168.88.0/24 [110/20] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
0      192.168.90.0/24 is variably subnetted, 3 subnets, 2 masks
0      192.168.90.0/26
0      [110/12] via 131.188.20.201, 7w0d, TenGigabitEthernet2/1
```

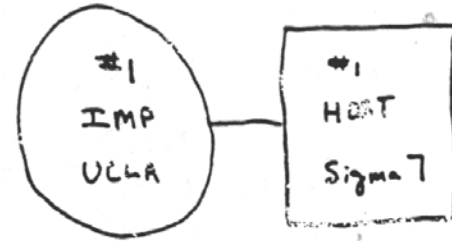
Bsp. Dynamisches Routing (III)

Das Internet

- 1969 als „ARPANET“ entstanden
- Durch die ARPA als Forschungsprojekt initiiert
- Gründungsmythos: Kommunikationsnetz, robust gegen „nuklearen Zerstörung“
- Tatsächlich: Projekt, um einzelne Uni- und Forschungsnetze im Land dezentral und effizient (über Telefonleitungen) zusammenzuschalten
- Grundprinzip geblieben bis heute: Internet nach wie vor aufgebaut aus einzelnen, unabhängig voneinander verwalteten Netzwerken von Provider/Uni/Regierung,...
- Einzelnetzwerke im Internet auch bezeichnet als „AS“ (Autonome Systeme)

Historik

Internet „Arpanet“ im Frühstadium



**September 1969:
1 Knoten**

Uni Kalifornien

THE ARPA NETWORK

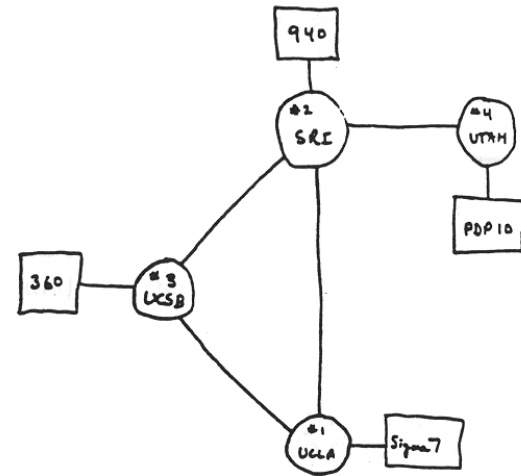
SEPT. 1969

1 NODE

FIGURE 6.1 Drawing of September 1969
(Courtesy of Alex McKenzie)

Historik

Internet „Arpanet“ im Frühstadium (2)



THE ARPA NETWORK

DEC 1969

4 NODES

FIGURE 6.2 Drawing of 4 Node Network
(Courtesy of Alex McKenzie)

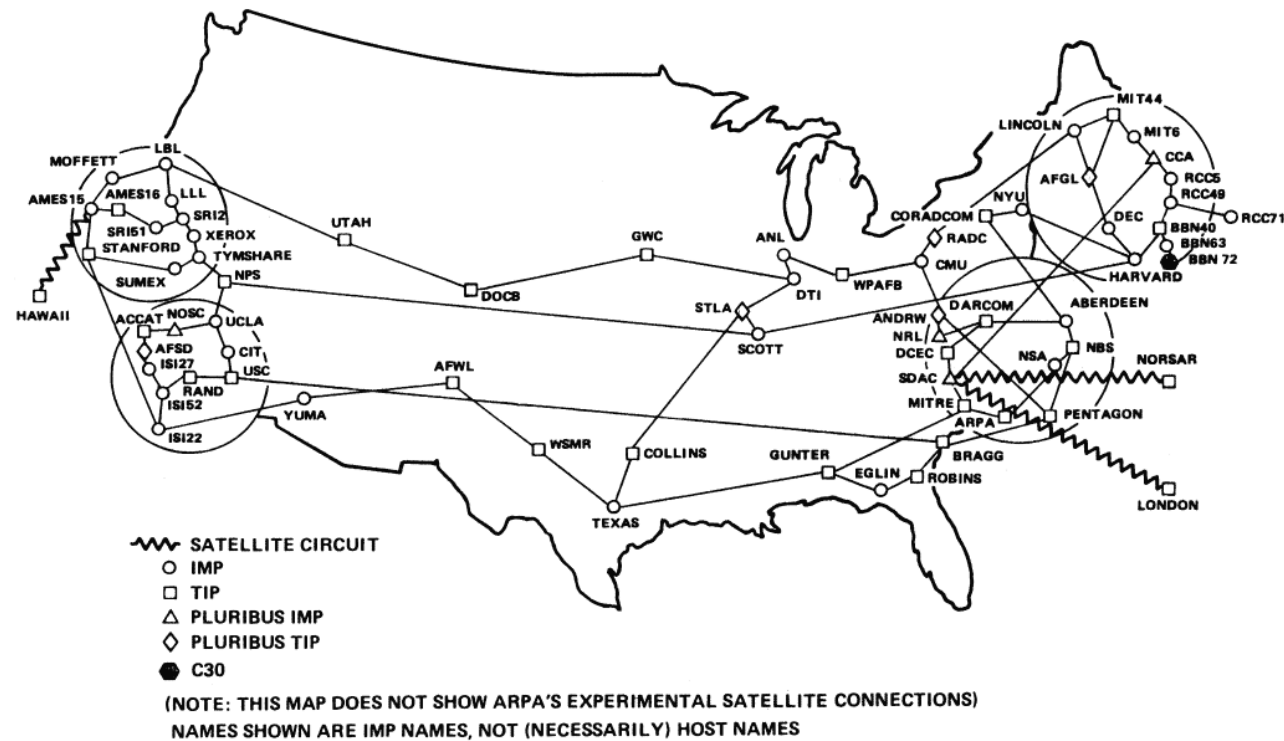
Dezember 1969:
4 Knotenpunkte

- Kalifornien
- Utah
- Stanford
- Santa Barbara

Historik

Internet „Arpanet“ im Frühstadium (3)

ARPANET GEOGRAPHIC MAP, OCTOBER 1980



Historik

Weiterer Werdegang:

- Abspaltung des „Milnet“ aus dem Arpnet
- Arpnet -> NSFNet (Abschaltung Arpnet 1989)
- NSFNet -> Internet (90er)

Routing im Internet

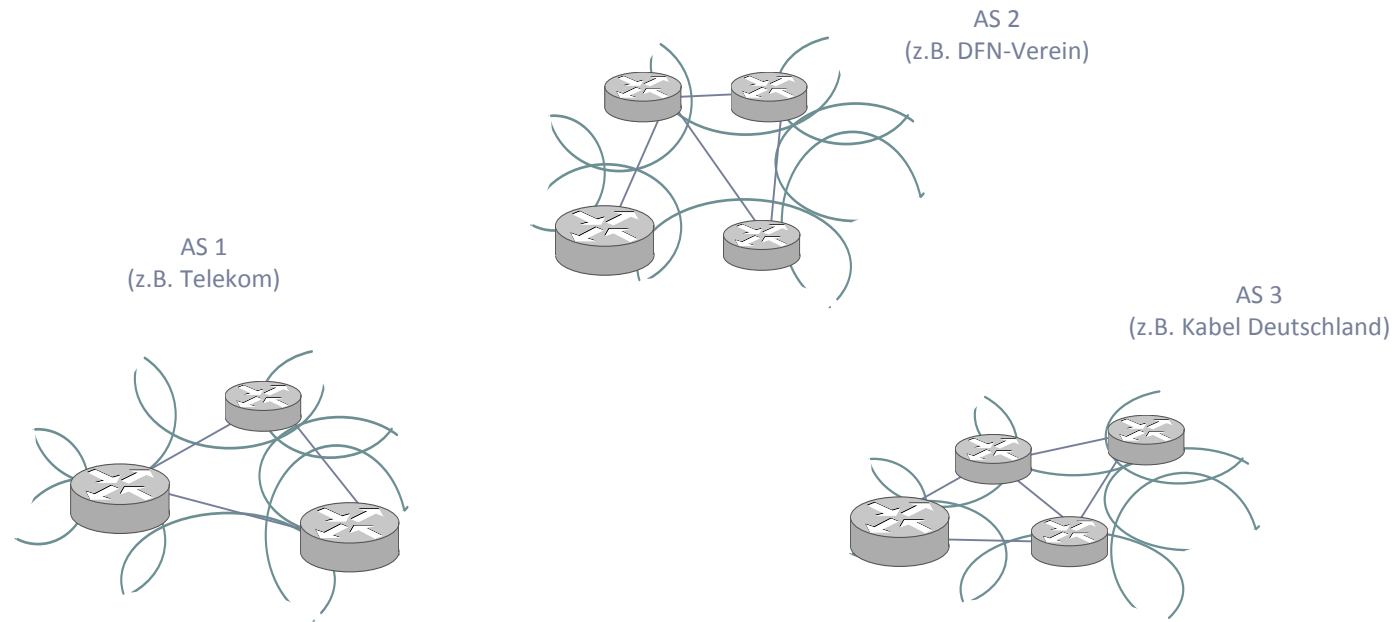
Heutiges Internet

- Bis heute besteht das Internet aus zigtausend Einzelnetzwerken, jeweils unter Kontrolle ihres Betreibers (Telcos, Firmen, Unis, Behörden,...)
- Diese Einzelnetzwerke werden auch bezeichnet als „AS“ (Autonome Systeme)
- AS sind somit die „Einzelbausteine“ des Internets

Vom AS zum Internet (I)

„Autonomes System“:

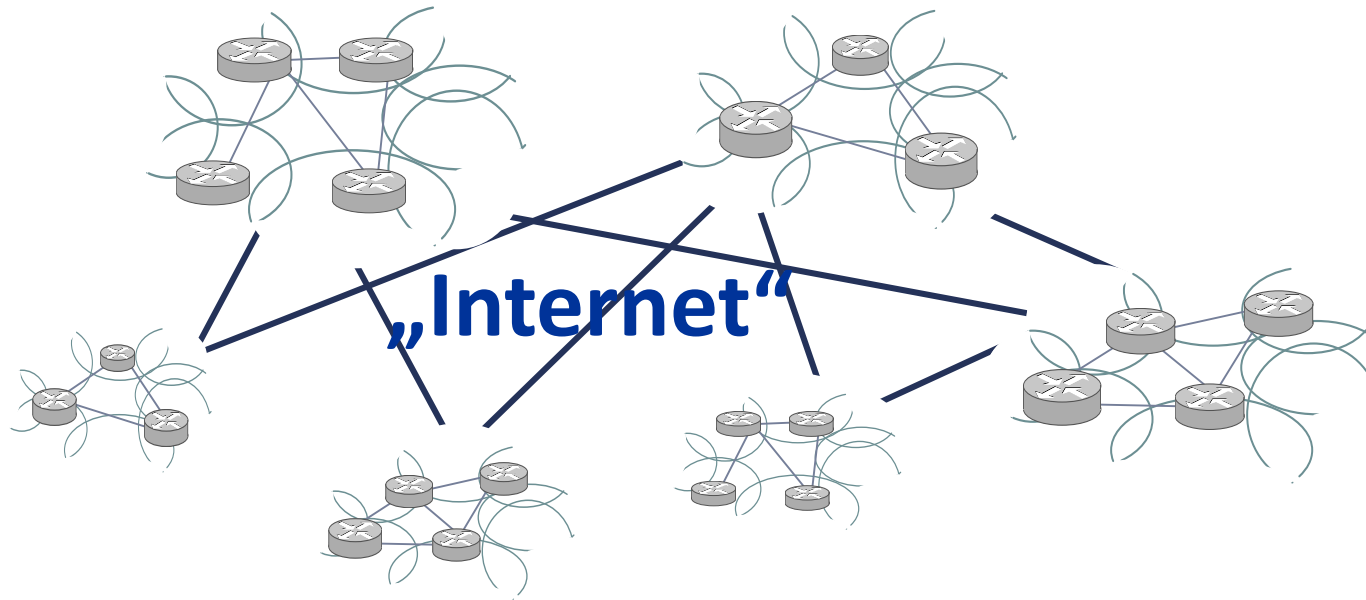
- IP-Netzwerk unter organisatorisch eigener Verwaltung
- I.d.R. leistungsfähige Netze Privater oder Öffentlicher Betreiber (Firmen, Unis, Telcos, Behörden,...)



Vom AS zum Internet (II)

Vernetzungen von mehreren AS

- Wenn sich nun AS untereinander vernetzen wollen...
- ...schalten sie in eigener Verantwortung Verbindungen zwischen sich...
(„Peering“ bzw. „Transit“ Verhandlungen)
- ...und bilden dadurch zusammen ein immer größer werdendes Netz der Netze, das **“Inter-Net”**



Vom AS zum Internet (IV)

Nach Einbindung eines AS in das Internet:

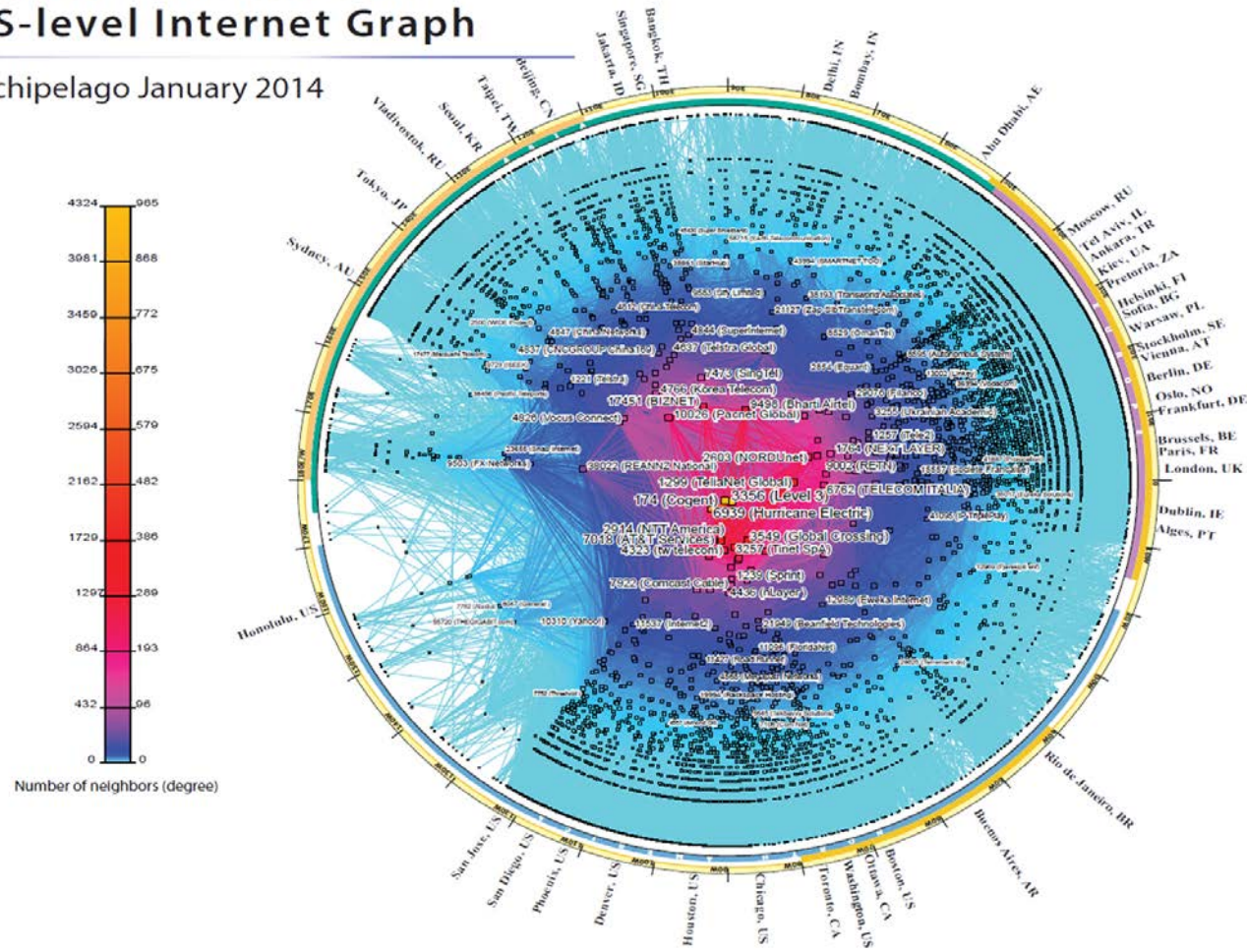
Jeder Border-Router synchronisiert sich mit seinen Partnern per BGP (Border Gateway Protokoll):

- Jedes AS beheimatet einerseits nur einen Bruchteil aller IPv4/v6-Netze
(Bsp. AS-680 des DFN: „Heimat“ der meisten Uni-Netze in Deutschland)
- Aber: Jeder Border-Router eines jeden AS kennt die Netze inkl. Routen aller anderen AS
- → Extrem große Routingtabellen auf allen Border-Routern:
Derzeit (2015): > 450.000 IPv4-Routen müssen auf jedem Border-Router eines AS vorgehalten werden

Internet heute: Vermaschung per IPv4

CAIDA's IPv4 AS Core AS-level Internet Graph

Archipelago January 2014



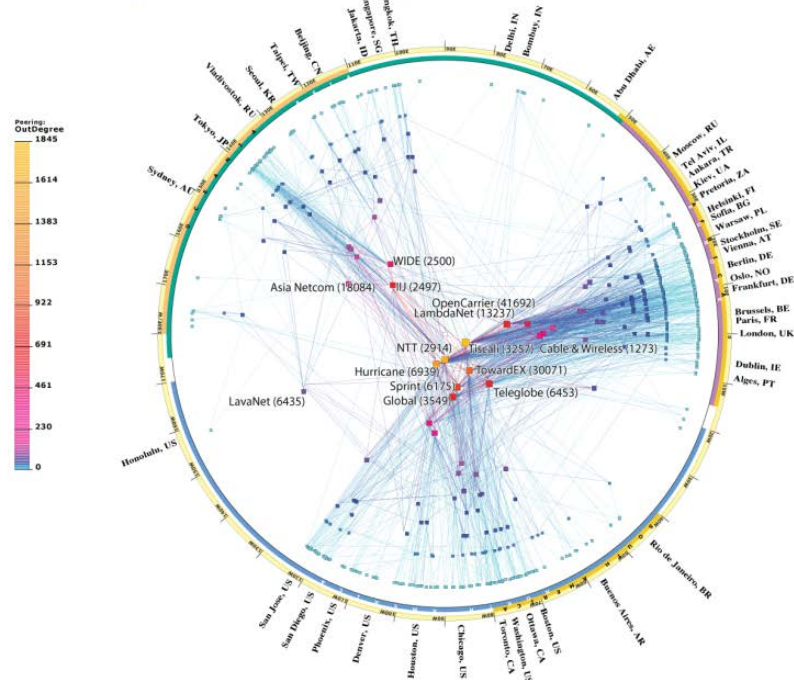
Copyright © 2014 UC Regents. All rights reserved.

Quelle: www.caida.org

„Erwachsenwerden“ von IPv6 2008 vs. 2014

CAIDA's IPv6 AS Core AS-level INTERNET GRAPH

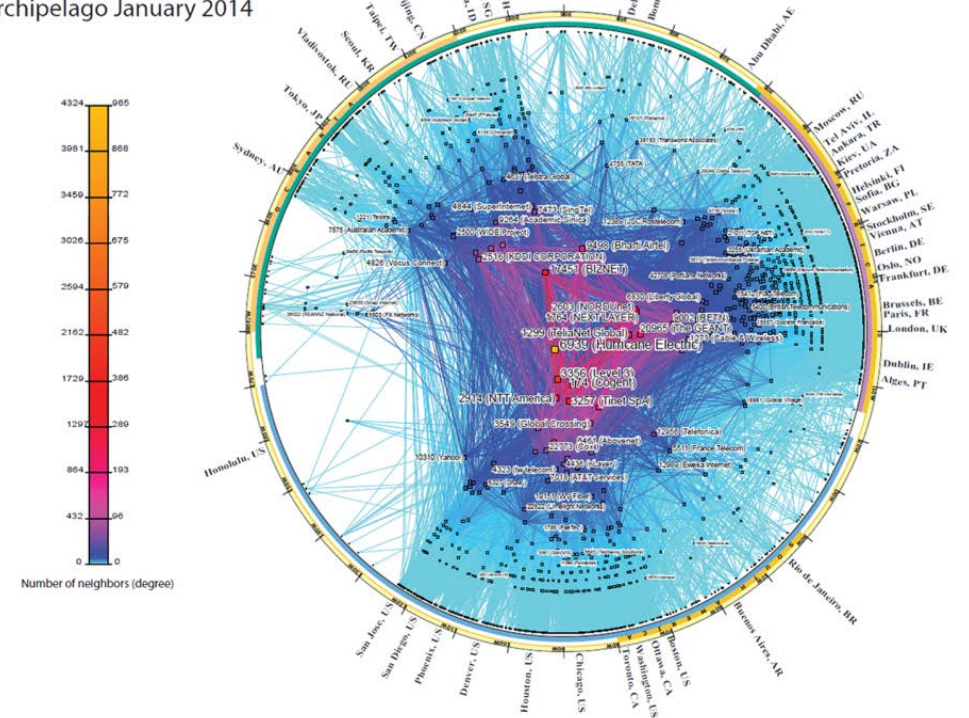
Community Collected January 2008



copyright © 2008 UC Regents. all rights reserved.

CAIDA's IPv6 AS Core AS-level Internet Graph

Archipelago January 2014



Copyright © 2014 UC Regents. All rights reserved.

Quelle: www.caida.org

Vom AS zum Internet (V)

Autonome Systeme als „anarchische“ Strukturen



The screenshot shows the heise Security website interface. The top navigation bar includes 'News', 'Hintergrund', and 'Erste Hilfe'. Below this, there are several news snippets. One snippet is titled 'Fehlkonf' and mentions 'Ein offenbar einigen Teile BGP-Router Einträgen) in'. Another snippet is titled 'Chinesischer Provid Internets' and mentions 'Der kleinere chinesische Inter einen Konfigurationsfehler ein IP-Netzen zuständig erklärt. Über Router, für welche Netze (auto anderen Netze sie erreichen können.'.

News-Meldung vom 15.11.2014 06:57

« Vorige | Nächste »

91. Treffen der IETF: Das Kapern von BGP-Routen verhindern

vorlesen / MP3-Download



Immer wieder wird Internet-Verkehr unbemerkt über seltsame Wege zum eigentlichen Ziel umgeleitet. Ob es sich um Abhör-Aktionen handelt oder nur um Pannen, ist oft unklar. Nun könnten Netzbetreiber ein Mittel dagegen in die Hand bekommen.

sam verantwortlich

Leinkrieg Ursache für YouTube-Ausfall

« Vorige | Nächste »

ir YouTube-Ausfall

Teile des

etwork Coordination Centre (RIPE) die der Ausfall des Online-ursacht wurde. Laut Daniel NCC, sowie Tiziana Refice und Luca atte die Pakistan Telecom schlicht pressen annonciert. Bereits eine ein (nach CIDR-Terminologie) /24-n zahlreichen Routern eingetragen.

Dynamische Routingprotokolle

Dynamische Routingprotokolle

Logische Unterscheidung

- Internal Gateway Protocol (IGP)

- Für dynamisches Routing innerhalb eines AS (z.B. Universitätscampus)

- › Z.B. RIP, OSPF, IGRP,...

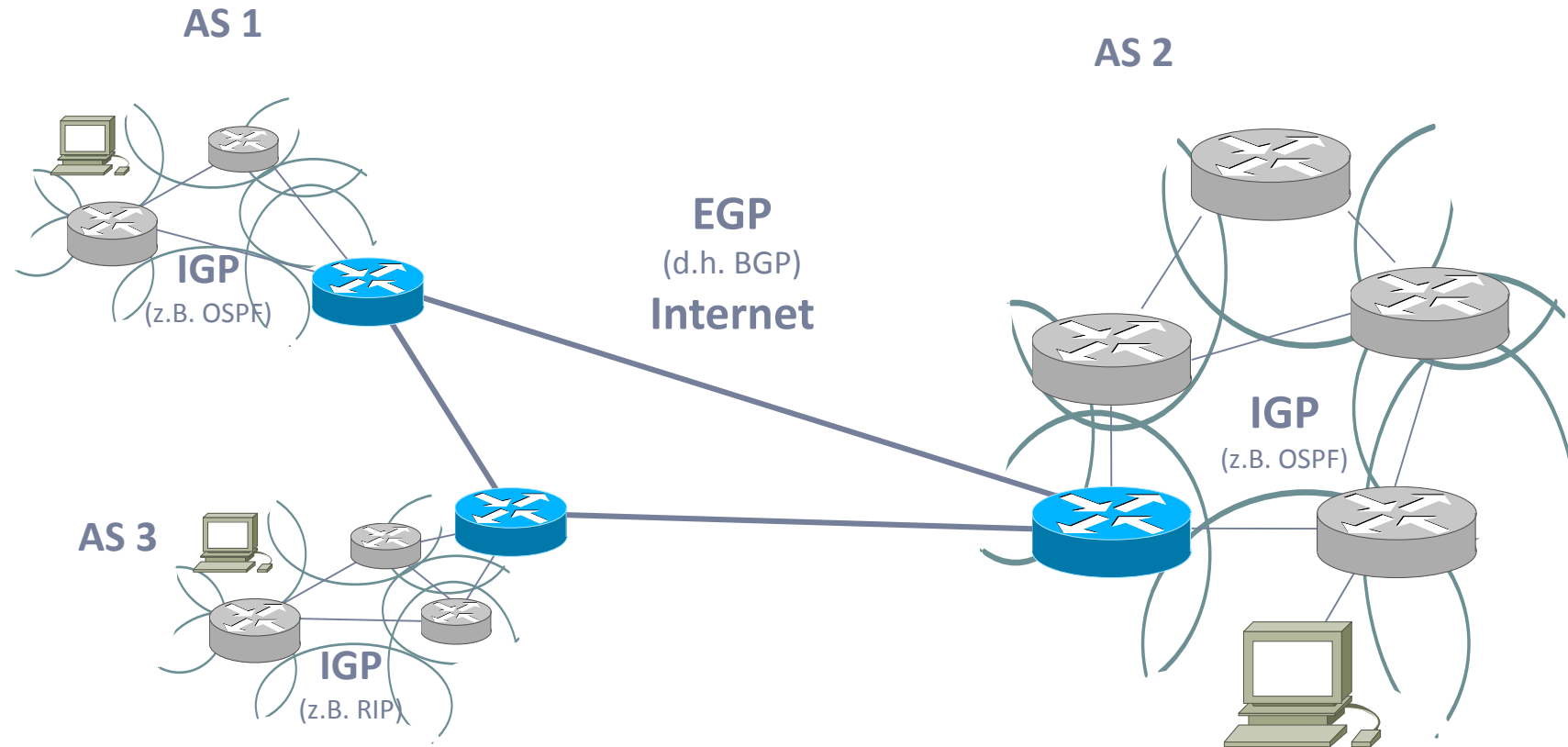
- External Gateway Protocol (EGP)

- Für dynamisches Routing zwischen verschiedenen AS (→ „Inter-net“)

- › Einziger praktischer Vertreter: BGP: Border Gateway Protocol

IGP vs. EGP

Zusammenspiel von IGP und EGP im Internet



Wie funktionieren Routingprotokolle?

Technische Unterscheidung dynamischer Routingprotokolle

- *distance vector*

- › z.B. RIP (IGP)

Internal Gateway Protocol (IGP)

- *link state*

- › z.B. OSPF (IGP)

- *path vector*

- › z.B. BGP (EGP)

External Gateway Protocol (EGP)

Distance-Vector-Protokolle

- Grobe Funktionsweise:
 - jeder Router pflegt Tabelle mit gelernten Pfaden zu versch. Zielnetzen
 - periodische Weitergabe (i.d.R. 30sek) dieser Tabelle jeweils an Nachbarrouter
 - Nachbarrouter updaten ggf. mit diesen Daten ihre Tabellen und senden Ihrerseits beim nächsten Update ihre Tabelle an Nachbarn
 - Änderungen „sprechen sich langsam im Netz rum“
 - „Distanz“ als einzige Berechnungs/Bewertungsgrundlage (Metrik) bei mehreren Routen zum gleichen Ziel
- Problem: langsame Konvergenz bei Routingänderungen

Distance-Vector-Protokolle: RIP (I)

Bekanntester Vertreter: RIP

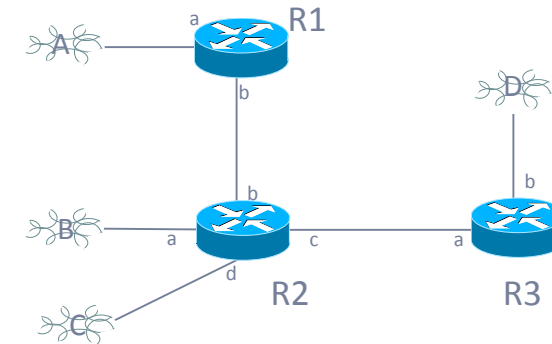
Eigenschaften

- RIP - *Routing Information Protocol*
- IGP-Einsatz, d.h. nur innerhalb von AS bzw. Intranets verwendet
- entwickelt von Ford und Fulkerson, daher auch Ford-Fulkerson Algorithmus
- definiert in RFC 1058, viele Erweiterungen
- relativ einfaches distance-Vector basiertes Protokoll
- „Hop-Count“ als einzige Metrik, d.h. keine explizite Angabe von Pfadkosten möglich

Distance-Vector-Protokolle: RIP (II)

RIP, Grobes Prinzip

- Jeder Router besitzt Routingtabelle mit „Hopcount“ als Bewertungskriterium(Metrik)
(Hopcount == Anzahl der Router, über die das Ziel erreicht werden kann)



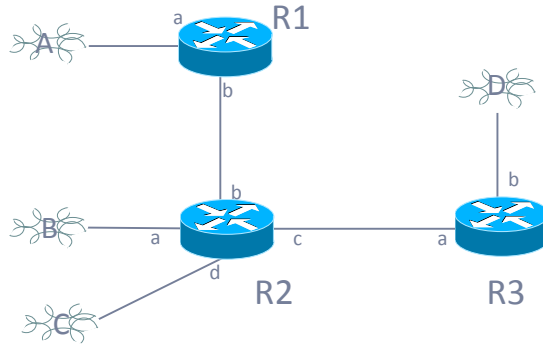
- Router schicken ihre Tabellen alle 30 sek. an ihre Nachbarn
- Jeder Router verbessert ggf. mit den Infos der Nachbarn seine Routingtabelle
- Timeout-Mechanismus:
 - nach 180 Sek. ohne neues Update: Gelernte Route wird als unbrauchbar markiert
 - nach 240 Sek. ohne Update: Gelernte Route wird gelöscht

Routingtabelle von R2

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

Distance-Vector-Protokolle: RIP, Beispiel

Ausgangsfall: Router kennen nur ihre direkten Nachbarn



R1

Ziel	Intf.	Hops
A	a	1
R2	b	1

R2

Ziel	Intf.	Hops
B	a	1
C	d	1
R1	b	1
R3	c	1

R3

Ziel	Intf.	Hops
D	b	1
R2	a	1

1. Update: Nachbarrouter tauschen ihre Routingtabellen aus

→ R2 lernt dabei Routen zum Ziel A und D, R1 und R3 wiederum zu den Zielen B,C,R3 bzw. B,C,R1

R1

Ziel	Intf.	Hops
A	a	1
B	b	2
C	b	2
R2	b	1
R3	b	2

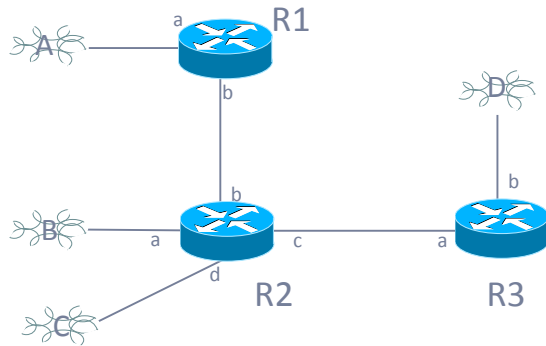
R2

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

R3

Ziel	Intf.	Hops
B	a	2
C	a	2
D	b	1
R1	a	2
R2	a	1

Distance-Vector-Protokolle: RIP, Beispiel



Ausgangspunkt nach 1. Update

R1

Ziel	Intf.	Hops
A	a	1
B	b	2
C	b	2
R2	b	1
R3	b	2

R2

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

R3

Ziel	Intf.	Hops
B	a	2
C	a	2
D	b	1
R1	a	2
R2	a	1

2. Update: Nachbarrouter tauschen wieder ihre Routingtabellen aus und updaten die Ihrige. Für R2 ergeben sich keine neuen Informationen, er hält seinen Zustand stabil. R1 und R3 lernen von R2 noch das Netz D bzw. A.

R1

Ziel	Intf.	Hops
A	a	1
B	b	2
C	b	2
D	b	3
R2	b	1
R3	b	2

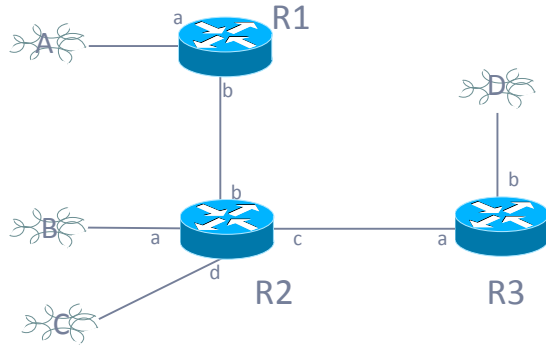
R2 [stabil]

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

R3

Ziel	Intf.	Hops
A	a	3
B	a	2
C	a	2
D	b	1
R1	a	2
R2	a	1

Distance-Vector-Protokolle: RIP, Beispiel



Ausgangspunkt nach 2. Update:

R1

Ziel	Intf.	Hops
A	a	1
B	b	2
C	b	2
D	b	3
R2	b	1
R3	b	2

R2 [stabil]

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

R3

Ziel	Intf.	Hops
A	a	3
B	a	2
C	a	2
D	b	1
R1	a	2
R2	a	1

3. Update: Ein erneuter Austausch der Routingtabellen bringt keinem Router mehr eine Änderung.

→ Nach diesem Durchgang sind die Routingtabellen auf alle Routern stabil

R1 [stabil]

Ziel	Intf.	Hops
A	a	1
B	b	2
C	b	2
D	b	3
R2	b	1
R3	b	2

R2 [stabil]

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

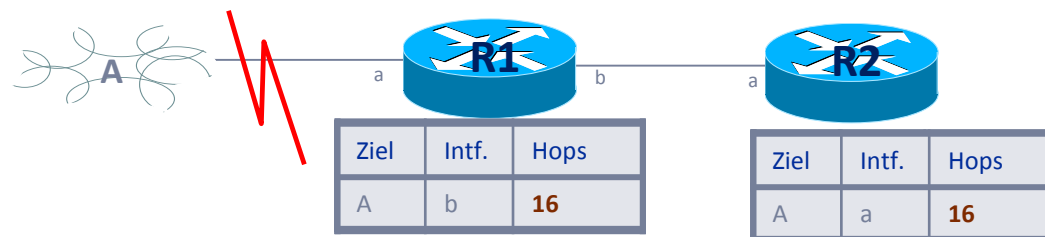
R3 [stabil]

Ziel	Intf.	Hops
A	a	3
B	a	2
C	a	2
D	b	1
R1	a	2
R2	a	1

RIP: Probleme: Wenn der Bagger....

Routingschleifen: „Counting to Infinity“

- Szenario:
 - Router R1 hat Route zu Netz A mit Distanz=1 (direkt angeschlossen)
 - Router R2 hat ebenfalls Route zu Netz A mit Distanz=2 (d.h. via R1)
 - Router haben stabilen Tabellenzustand, schicken sich Updates alle 30sek
- Fehlerfall auf R1-Seite: Verbindung auf Interface a zu Netz A geht verloren
 - R1 aktualisiert seine Routingtabelle: Pfad zu A ungültig (Hopcount 1 auf 16 gesetzt)
(Schafft es R1 sein Routingupdate vor R2 zu verschicken, ist das auch kein Problem)
 - Aber: Falls R2 **vor** R1 sein Routingupdate verschickt, wird R1 eine neue Route für A von R2 lernen (mit Distanz =3)
 - Diese Route wird R1 wieder an R2 verbreiten, R2 aktualisiert dann seine Route für A mit Distanz=4 usw...
 - Erst bei Erreichen von Hopcount 16 (nach ca. 7min!) ist die Route zu A endlich ungültig



RIP: Erweiterung

RIP, Ansätze zur Problemlösung

- **Split Horizon**
Grundidee: es macht keinen Sinn, Routen in die Richtung weiterzugeben, aus der man sie bekommen hat
- **Split Horizon with Poison Reverse**
Grundidee: sende Routinginformationen in die Richtung, aus der sie gekommen sind mit Metrik 16 (unendlich) zurück
- **Holddown**
Grundidee: akzeptiere keine Routinginformationen zu einem Ziel, für welches man selbst eben Informationen verarbeitet hat, für eine gewisse Zeit

RIP: Nachteile

Hauptnachteile von RIP bzw. Distanzvektorprotokollen

- Relativ langsame Konvergenz (Minutenbereich)
- Nur HopCount als Metrik
- Nur für kleinere Netze geeignet

→ Für größere lokale Netze mit erweiterten Anforderungen: Link-State-Protokolle

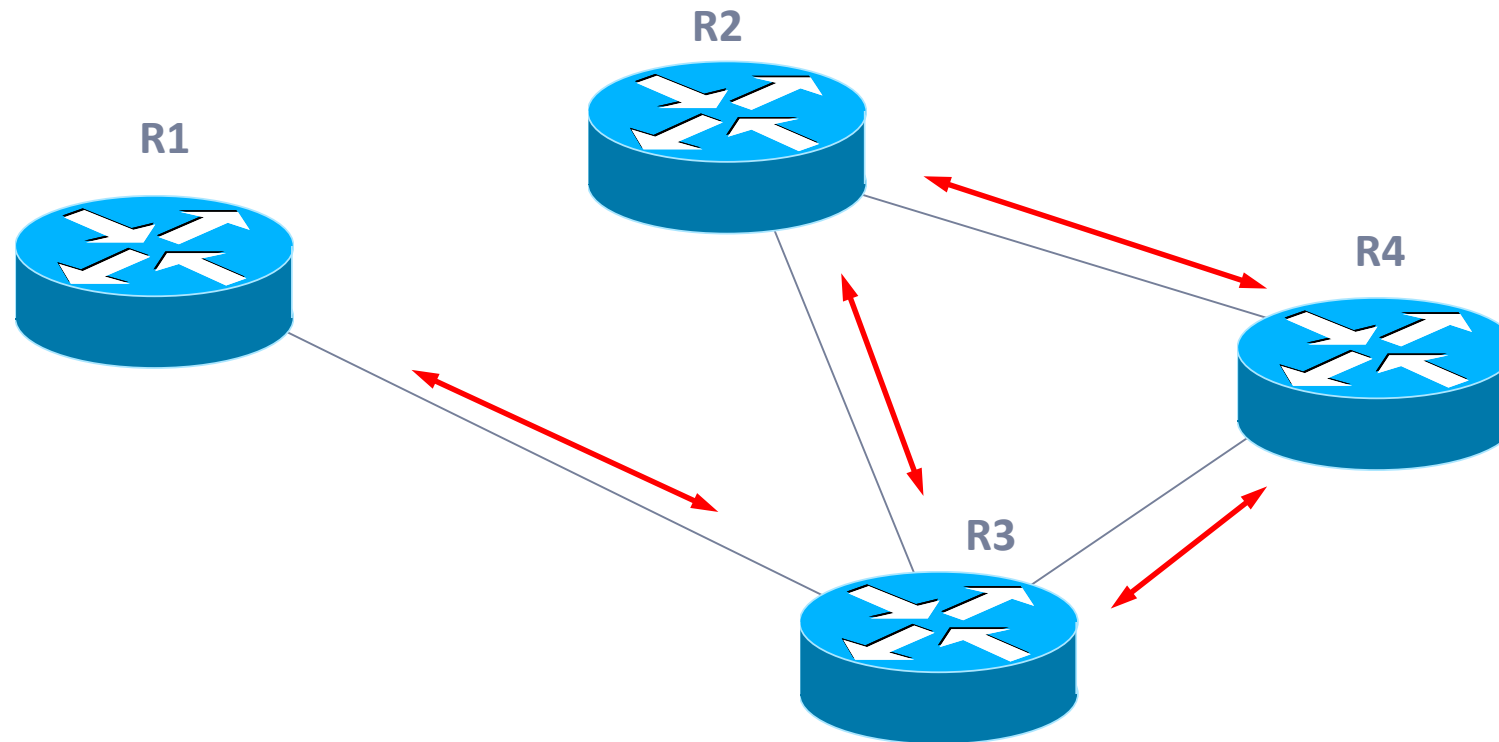
Link-State-Protokolle: Prinzip

- Benachbarte Router bauen eine permanente Verbindung („Adjazenz“) auf und tauschen asynchron Nachrichten über Veränderungen des Netzes aus
- Änderungen auf einem Router werden unmittelbar (ggf. inkrementell) an anderen Router gesendet (→ sehr schnelle Reaktion)
- Nach Konvergenz hat jeder Router eine komplette Sicht auf das gesamte Netz (als vermaschter Graph)
- Jeder Router berechnet dann auf Basis dieser Sicht eine optimale Routingtabelle (Graphentheorie, Spannbaumprinzip)

Link-State-Protokolle: Arbeitsweise (1)

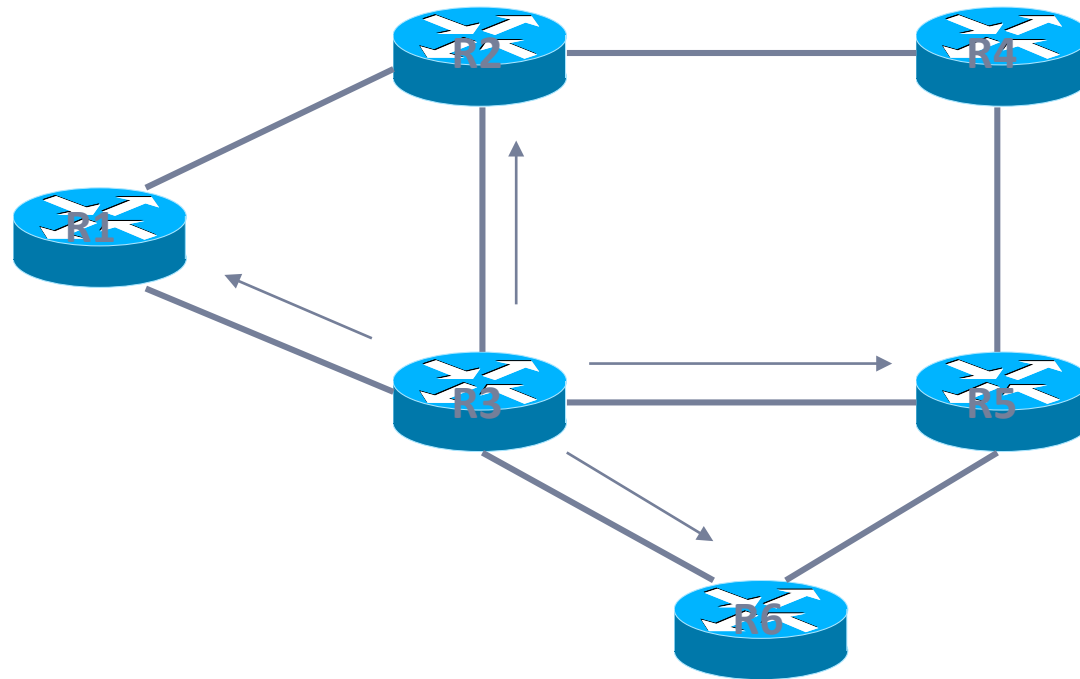
„Link-State“

Jeder Router hält permanent eine Verbindung zu seinen unmittelbaren Nachbarn offen. Darüber findet ein asynchroner Nachrichten austausch statt, über sog. LSAs: (Link State Advertisements)



Link-State-Protokolle: Arbeitsweise (2)

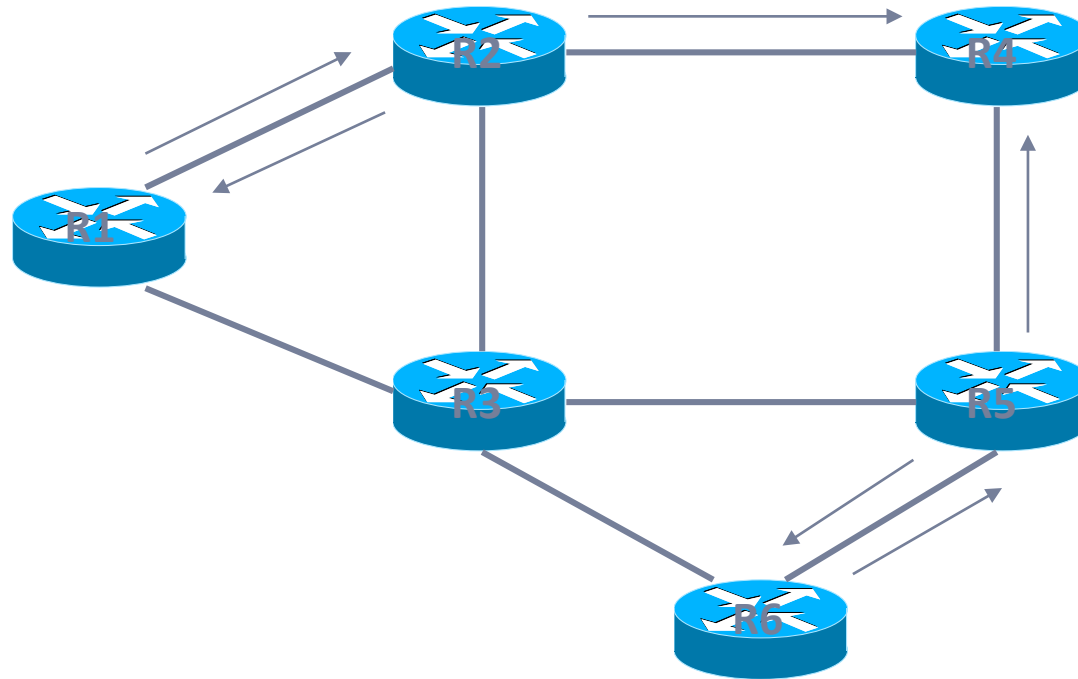
Prozedere beim Start eines Routers: „Flooding“



Bsp.: Router R3 fährt hoch. Er sendet als Erstes seine Routinginformationen (d.h. lokal angeschlossene Netze) an alle seine Nachbarn

Link-State-Protokolle: Arbeitsweise (3)

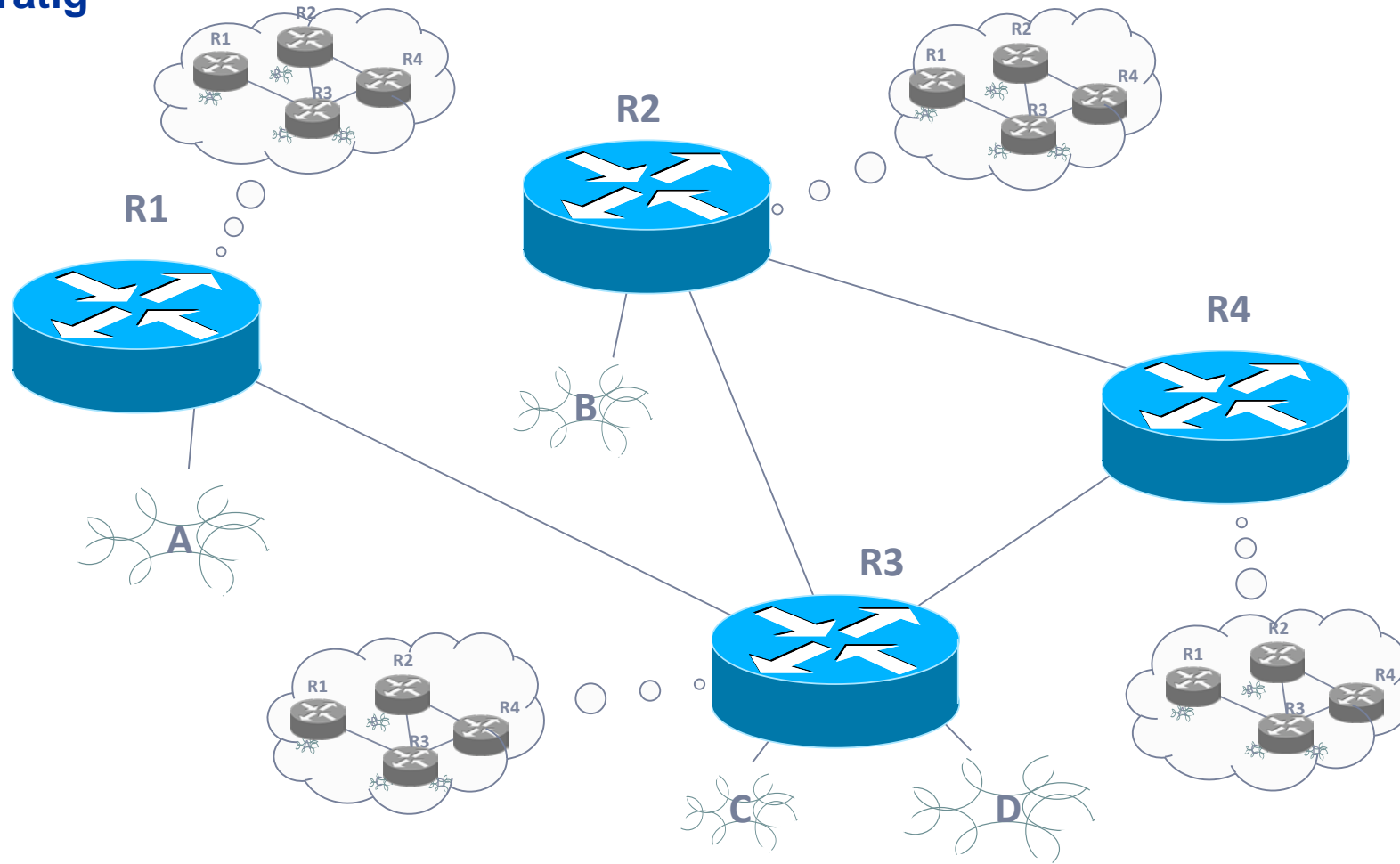
Weitergabe der Routinginformationen durch das Netz



Die Nachbarn aktualisieren augenblicklich ihre Datenbasis und schicken daraufhin Updates an ihre Nachbarn: Die Information wird „flutend“ unmittelbar durch das gesamte Netz durchgereicht (sehr schnell!)

Link-State-Protokolle: Arbeitsweise (4)

Nach kurzer Zeit hält jeder Router danach eine stets aktuelle Sicht („Graph“) des kompletten Netzwerks vorrätig



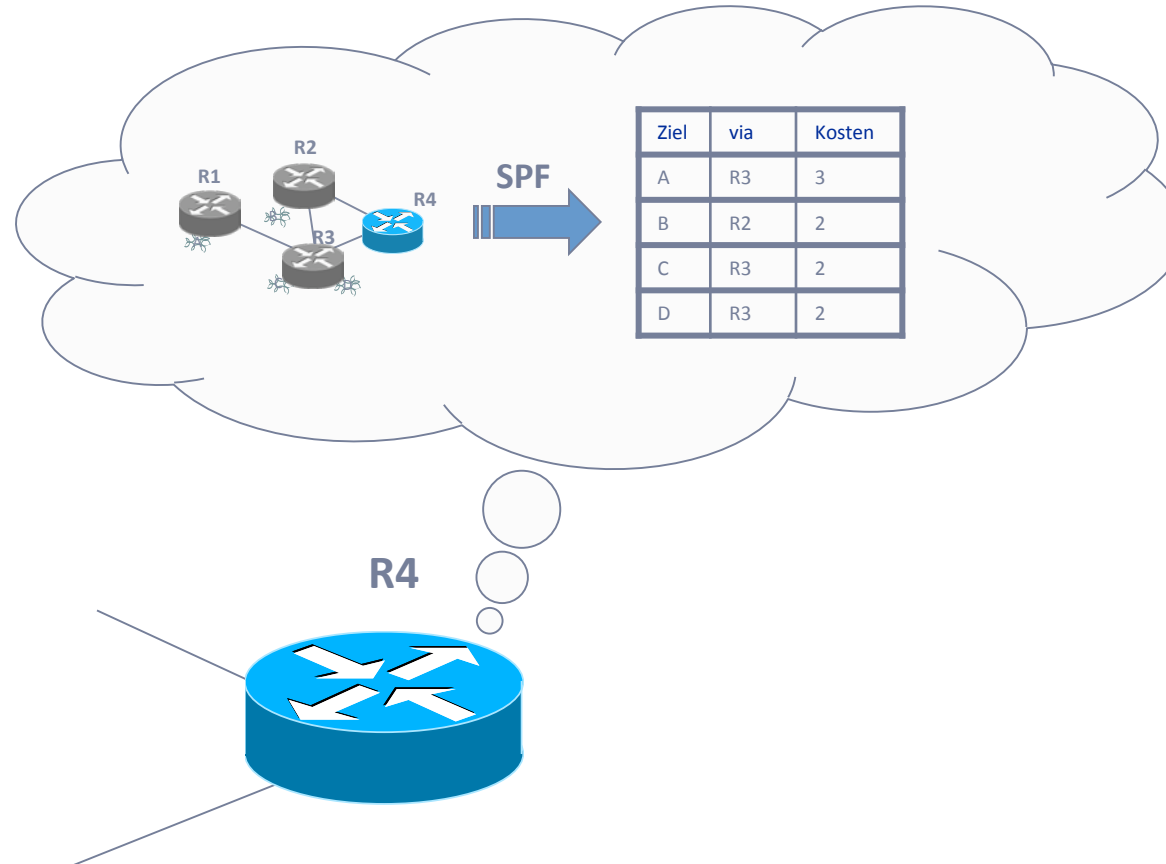
Link-State-Protokolle: SPF (1)

Danach: Jeder Router berechnet sich aus Graph seine Routingtabelle

- Grundlage: Sog. „Shortest Path“ Algorithmen
- Findet kürzeste Wege in vermaschten Graphen von gg. Start und Zielpunkt
- Jeder Router berechnet so seine optimale Routingtabelle zu allen Zielen
- Bei Topologieveränderungen: Flooding der Änderung und Neuberechnung
- Konvergenzgeschwindigkeit von nur wenigen Sekunden
- bekanntester SPF-Algorithmus: Dijkstra

Link-State-Protokolle: SPF (2)

Berechnung der Routingtabelle individuell auf jeden Router mittels SPF-Algorithmus (inkl. Kostenmaß!)



Linkstate Protokoll: OSPF(1)

- Bekanntester Vertreter: OSPF - *Open Shortest Path First*
- Das am meisten verbreitete LinkState Routingprotokoll
- entwickelt von J. Moy
- OSPFv1, v2 oder v3 (IPv6)
- IGP-Einsatz, d.h. nur innerhalb von AS verwendet

Linkstate Protokoll: OSPF(2)

- Vorteile
 - Kryptographisch abgesichert (MD5 Checksum)
 - flexible Metriken für Routingentscheidungen (z.B. Distance, Hop-Count, \$\$)
 - Wertebereich für Metrik von 1 bis 65535
 - Skalierbarkeit
 - Unterstützung von mehreren Wegen (equal path load sharing, unequal path load sharing)

Pfad-Vektor Protokolle

Situation Internet

- Für Routing zwischen Autonomen Systemen (→ Internet) sind sowohl Link-State als auch Distanzvektor-Protokolle nicht geeignet:
 - › Linkstate-Protokolle wären im Internet praktisch nicht einsetzbar („Flutorgie“)
 - › Distanz-Vektor Protokolle dagegen weisen bei größeren Netzen zu gravierende Nachteile auf (z.B. Hang zur Schleifenbildung)
- → Entwurf eines neuen Protokolls fürs „Inter-AS Routing“ (d.h. Internet): Pfad-Vektor Protokolle

Pfad-Vektor Protokolle (II)

Bsp. BGP (I)

- Prinzip wie Distanz-Vektor Protokoll (vgl. RIP)
 - Erinnerung: Distanz-Vektor: Routing-Update enthält Ziel und als Metrik zugehöriger Hopcount

Ziel	Intf.	Metrik (Hops)
A	a	3
B	b	4

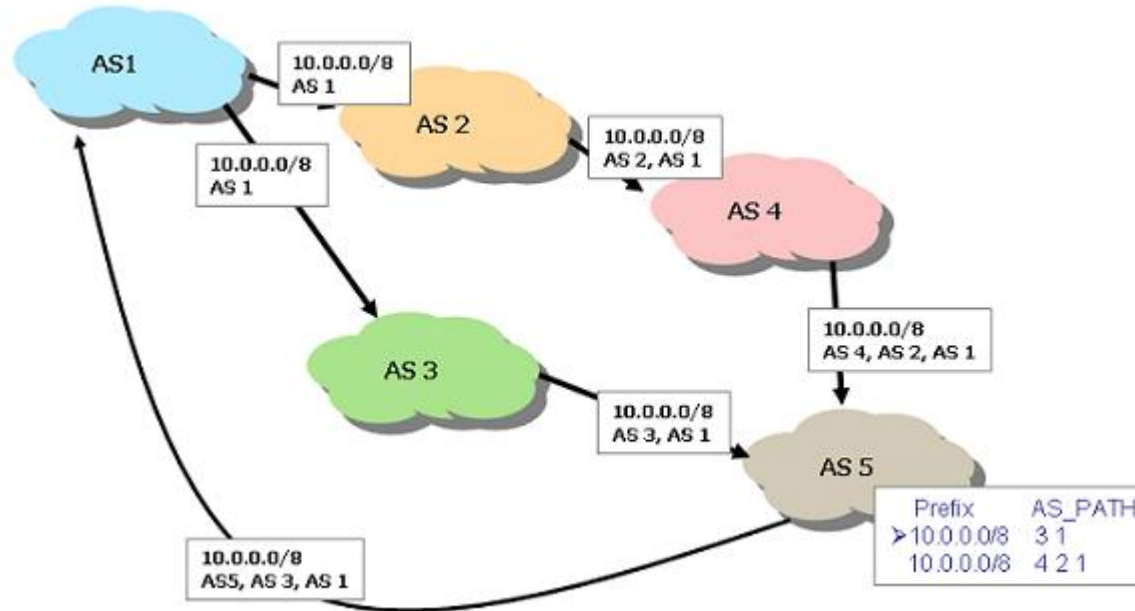
- Pfad-Vektor: Routing-Update enthält Ziel und als Metrik Pfad der bereits durchlaufenen Router (genauer: AS-Nummern) in Form einer Liste

Ziel	Intf.	Metrik (Pfad)
A	a	„R2-R3“
B	b	„R2-R3-R5“

- → Vermeidung von Routing-Schleifen
Updates werden verworfen, sobald eigener Router in Pfadliste eines Updates auftaucht.
- Einziger praktischer Vertreter: BGP

Pfad-Vektor Protokolle (III)

Bsp.: Prinzip BGP Routingupdate (ausgehend von AS1)



Quelle:
<http://routemyworld.com/wp-content/uploads/2008/12/bgpas-path.jpg>

BGP (III)

Prinzipielle Nachteile von BGP:

- (Keine Möglichkeit der Lastverteilung)
nur eine Route pro Netz wird ausgewählt
- Auswahl nur nach Anzahl AS, nicht jedoch nach Hops
Anzahl der Hops innerhalb eines AS unklar!
- keine Berücksichtigung der Link-Geschwindigkeiten
- Sicherheitsaspekte (Spoofing)
- Problem von *Route-Flaps* und *Update-Bursts*

Vielen Dank für Ihre
Aufmerksamkeit!