

# Routing-Protokolle

**Prof. Dr. Norbert Pohlmann**

Fachbereich Informatik

Verteilte Systeme und Informationssicherheit



# Inhalt

---

- **Ziele, Einordnung und Übersicht**
- **Router**
- **Routing-Verfahren**
- **Routing-Protokolle**
- **Autonome Systeme / Internet**
- **Ablauf**
- **Zusammenfassung**

- **Ziele, Einordnung und Übersicht**
- Router
- Routing-Verfahren
- Routing-Protokolle
- Autonome Systeme / Internet
- Ablauf
- Zusammenfassung

# Routing-Protokolle

## → Ziele

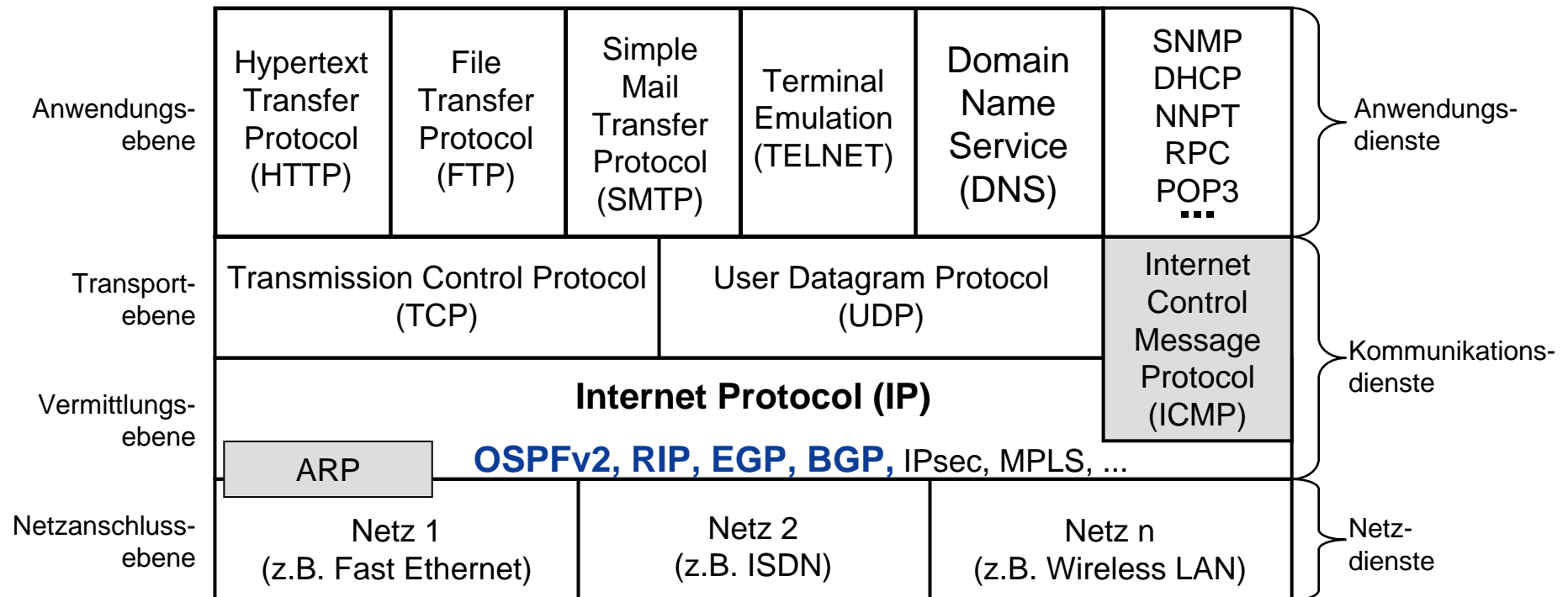
---

- Gutes Verständnis für die Notwendigkeit und die Möglichkeiten eines Teils des Konfigurationsmanagements im Bereich des Netzwerkmanagements.
- Guten Sachverstand und Überblick für die wichtigsten Routing-Verfahren und -Protokolle in den TCP/IP-Kommunikationsarchitekturen.
- Erlangen der Kenntnisse über die Aufgaben, Prinzipien und Mechanismen von Routing-Algorithmen und -Protokollen.

# Routing-Protokolle

## → Einordnung

### Internet-Protokollstack



# Inhalt

---

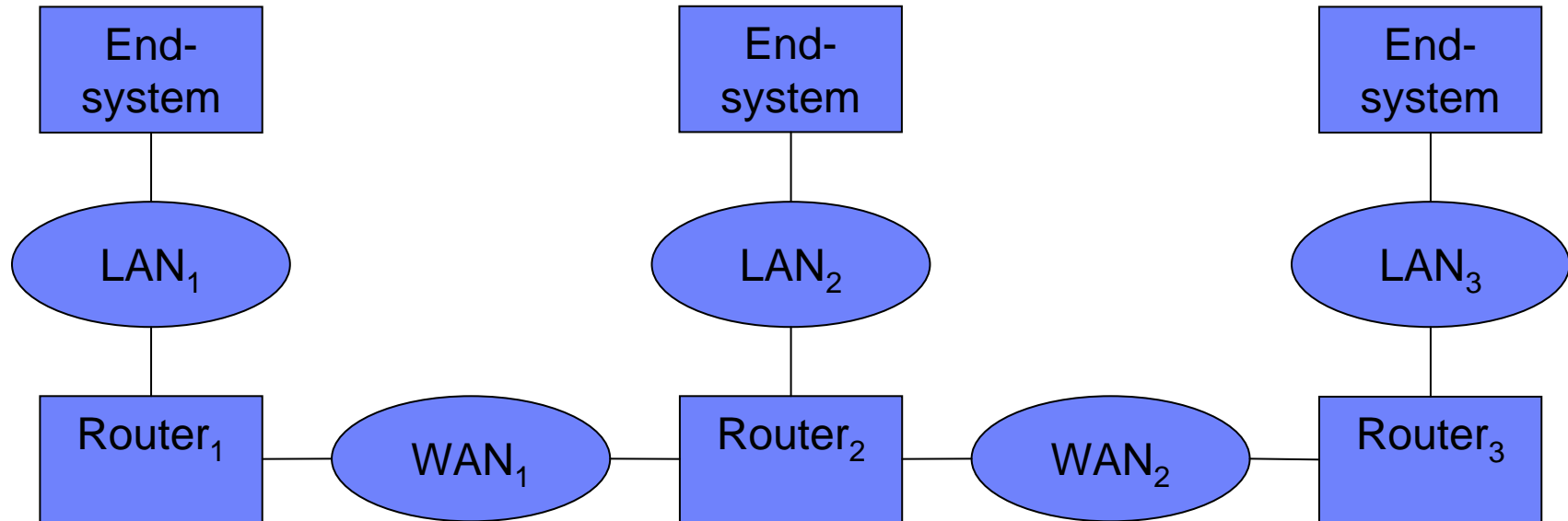
- Ziele, Einordnung und Übersicht

## ■ Router

- Routing-Verfahren
- Routing-Protokolle
- Autonome Systeme / Internet
- Ablauf
- Zusammenfassung

# Router

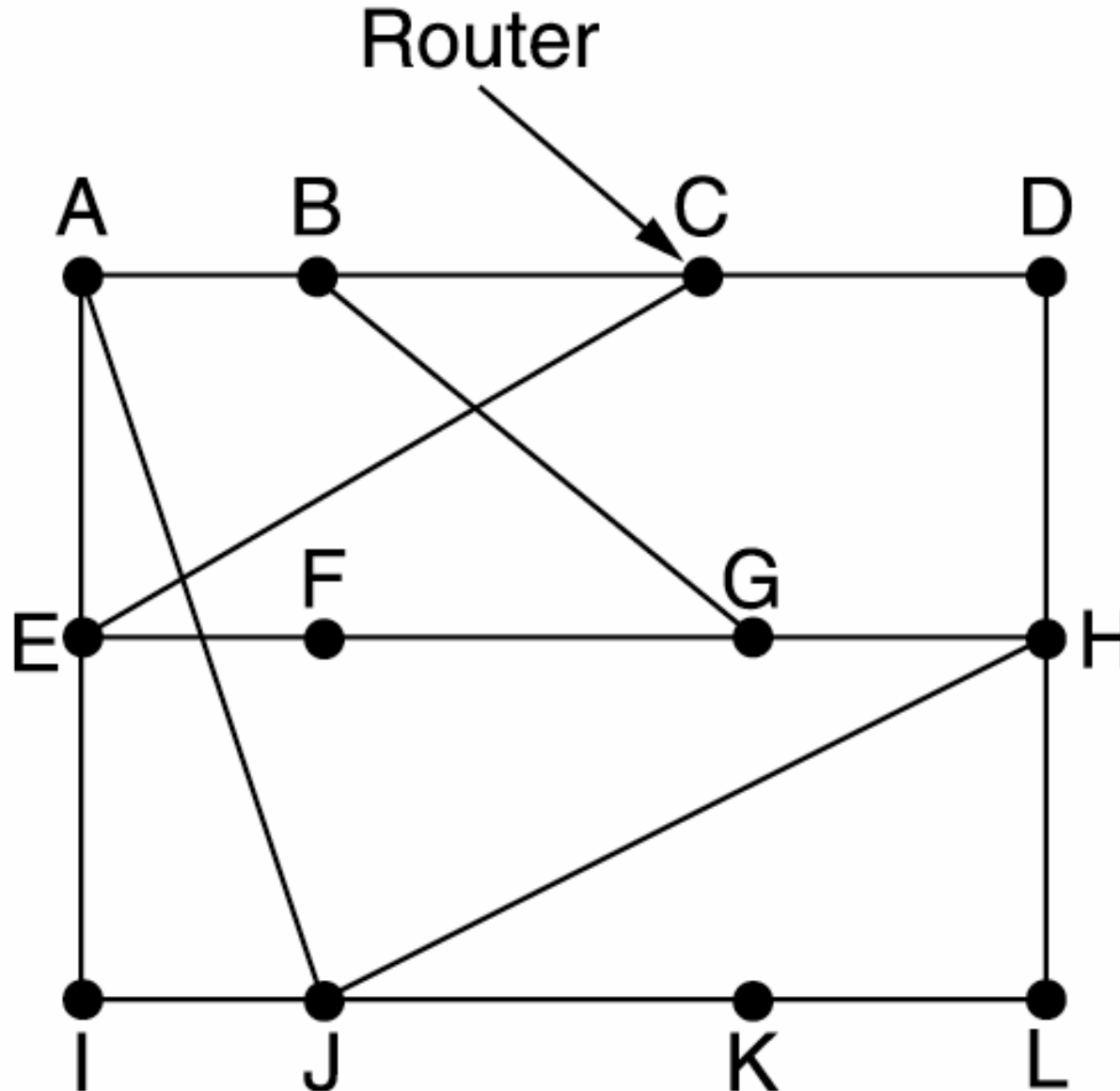
## → Einsatzzweck



- Ein Router koppelt das Netz mit unterschiedlichen Netzadressen auf der Vermittlungsschicht zu entfernten Endsystemen über ein oder mehrere WANs.
- Mit Hilfe von IP-Adressen und den dazugehörigen Routing-Protokollen baut ein Router die dazugehörigen Routing-Tabellen auf.
- Bei den meisten Routern geschieht dieser Aufbau automatisch nach dem Einschalten des Routers.

# Router

→ Beispiel eines Teilnetzes





# Router

## → Aufgaben

---

- Kopplung über Schicht 3 des OSI-Referenzmodells
- **Wegewahl** anhand weltweit eindeutiger, hierarchischer Netzwerkadressen
  - meist IP-Adressen
  - Adressen werden in speziellen Tabellen gehalten
  - durch Hierarchie weltweite Zuordnung der Adressen möglich
  - Voraussetzung für optimale Wegewahl
- Segmentieren und Reassemblieren von Schicht-3-Datenpaketen zur Anpassung an unterschiedliche maximale Paketgrößen auf Schicht 2

# Router

---

- **Weitere Aufgaben:**

- Sicherheitsmechanismen zur Regelung von Netzzugriffen abhängig von der Netzwerkadresse.

**Firewalling:** anhand von Quell- bzw. Zieladressen kann über eine Weiterleitung der Pakete entschieden werden (Packet-Filter-Funktionen)

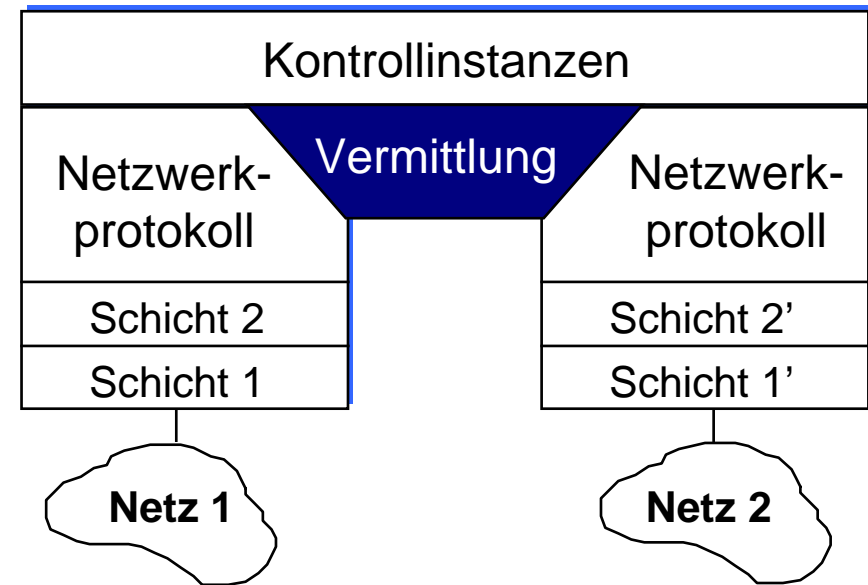
- automatische Begrenzung von Schicht-2-Broadcasts

- Router sind eine der leistungsfähigsten Netzwerkkomponenten mit Datendurchsätzen im **Multi-Gigabit-Bereich**

# Router

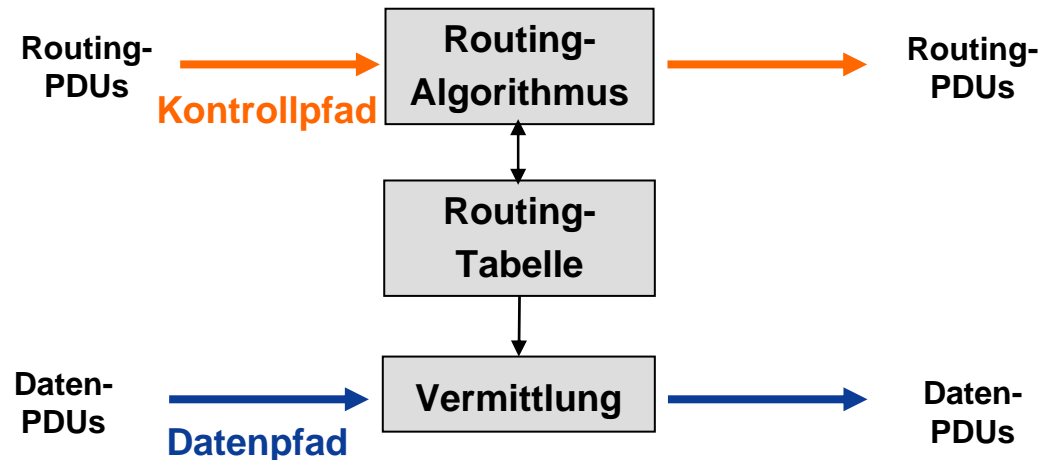
## → Architektur - wesentliche Merkmale

- Für jedes Netzwerk eine eigene Schicht-1- und Schicht-2-Instanz
- Netzwerkprotokoll ist in der Regel für alle Netzwerke gleich (z.B. IP-Router)
- Wegwahl anhand der global eindeutigen Netzwerkadressen
- Vermittlungskomponente verbindet die Netzwerkprotokollinstanzen; sie realisiert die Weiterleitungsfunktion
- Kontrollinstanzen implementieren beispielsweise Routing-Protokolle, Protokolle zur Fehleranzeige und Managementprotokolle (ICMP)



# Router

## → Kontroll- und Datenpfad

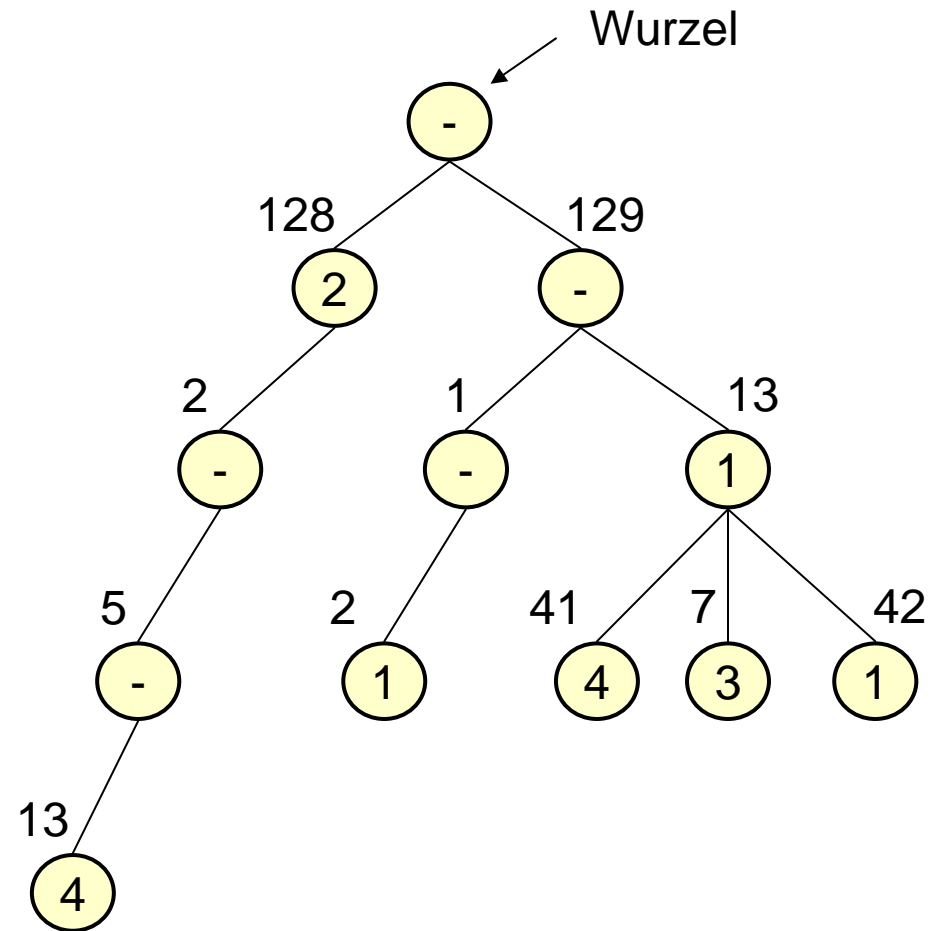


- **Datenpfad auf Netzwerkschicht**
- **Kontrollpfad darüber**
- Gewinnung von Routinginformationen durch das **Routing-Protokoll**
- **Routing-Algorithmus** verwaltet die Routing-Tabelle (Eintragen, Ändern, Löschen von Einträgen)
- **Routing-Tabelle** enthält Routinginformationen
- Wegewahl bei der Vermittlung wird anhand der Routing-Tabelle durchgeführt

# Router

## → Prinzip einer Routing-Tabelle

Zieladresse	Ausgang
129.13.*.*	1
128.*.*.*	2
129.1.2.*	1
129.13.41.*	4
129.13.42.*	1
129.13.7.*	3
128.2.5.13	4



- Beispiel einer vereinfachten Routing-Tabelle:
  - Suche nach dem am besten passenden **Präfix**
  - Weiterleitung der Pakete an die den jeweiligen Präfixen zugeordneten Ausgänge

# Inhalt

---

- Ziele, Einordnung und Übersicht
- Router

## ■ Routing-Verfahren

- Routing-Protokolle
- Autonome Systeme / Internet
- Ablauf
- Zusammenfassung

# Router

## → Routing-Verfahren (1/3)

---

### ■ Aufgaben des Routingverfahrens

- Treffen der Entscheidung, auf welcher Übertragungsleitung ein eingehendes Paket weitergeleitet werden soll

### ■ Ziele

- niedrige bis mittlere Paketverzögerungszeit
- hoher Netzdurchsatz

### ■ Ansatzpunkt

- Übertragen eines Pakets von einem Quellrechner zu einem Zielrechner über den Weg mit den geringsten „Kosten“.
- **Mögliche Metriken** sind z.B.: **Anzahl der Hops**, **finanzielle Kosten**, **Verlässlichkeit**, **Durchsatz**, **Bandbreite** und **Verzögerung**.

# Router

## → Routing-Verfahren (2/3)

---

- **Ort des Routing-Verfahrens:**

- **Zentral** in einem Netzkontrollzentrum (Routing Control Center, RCC)
- **Dezentral**, d.h. verteilt auf die Vermittlungsknoten (Router)

- **Dynamik des Routing-Verfahrens:**

- **Nicht adaptiv (statisch):** die Routing-Tabellen in den Vermittlungsknoten bleiben über längere Zeit konstant (verglichen mit Verkehrsänderungen)
- **Adaptiv (dynamisch):** Routing-Entscheidungen hängen vom Zustand des Netzes ab (Topologie, Lastverhältnisse), und passen sich schnell an.

Für die Wegewahl sind Veränderungen der Topologie oder die aktuelle Auslastung einer Leitung entscheidend.

Für die Berücksichtigung des momentanen Netzzustandes ist ein Informationsaustausch zwischen den Knoten notwendig.



# Router

## → Routing-Verfahren (3/3)

---

### ■ Zeitpunkt der Wegwahlentscheidung:

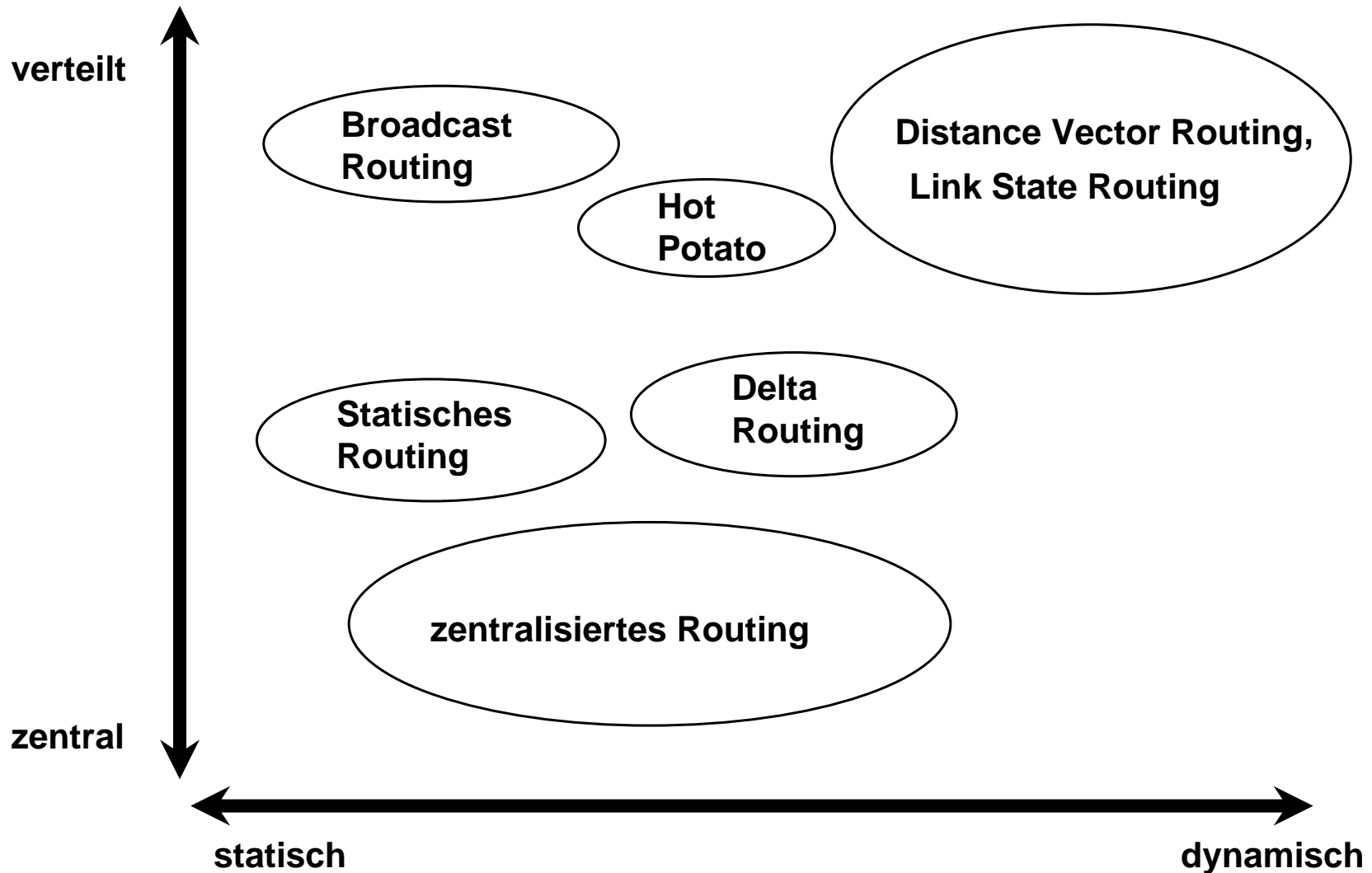
- Datenübertragung mit Hilfe einzelner Datagramme: erneute Wegewahl für jedes einzelne IP-Paket
- Verbindungsorientierte Datenübertragung (z.B. Telefonnetz): einmalige Wegewahl, alle nachfolgende Datenpakete folgen diesem Weg

### ■ Art der Informationsbeschaffung:

- Jeder **Knoten entscheidet isoliert** (*Isolated Routing*, hierunter fallen z.B. Broadcast Routing, Hot Potato, ...) ohne Informationsaustausch mit anderen Routern
- Benachbarte **Knoten tauschen Informationen aus**
- Informationen betreffen nicht nur lokale Bereiche, sondern sind übergreifend und **werden zentral gesammelt**

# Router

## → Übersicht der Routing-Verfahren



# Statisches Routing (1/2)

---

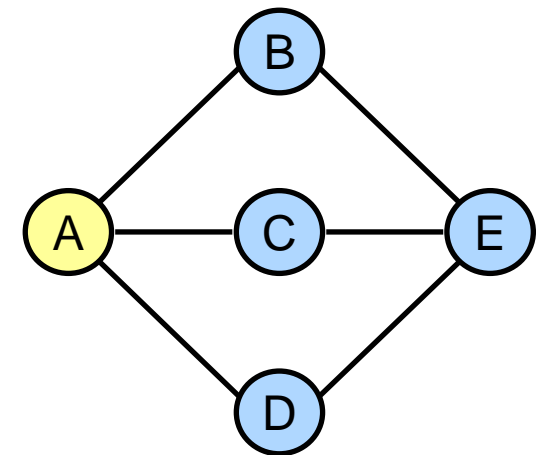
- Statische Routing ist ein einfaches nicht adaptives Verfahren, welches von vielen kleinen stabilen IP-Netzen benutzt wird.
- **Verfahren:**
  - Jeder Knoten unterhält eine Tabelle mit einer Zeile für jeden möglichen Zielknoten.
  - Der komplette Inhalt dieser Tabellen wird statisch in jedem Router konfiguriert.
  - Eine Zeile enthält z.B. n Einträge, welche die beste, zweitbeste, etc. Übertragungsleitung für dieses Ziel ist, zusammen mit einer **relativen Gewichtung entsprechend der Kapazität** der jeweiligen Leitung.
  - Vor der Weiterleitung eines Pakets wird eine Zufallszahl gezogen und eine der Alternativen anhand der Gewichtung ausgewählt

# Statisches Routing (2/2)

- Beispiel:
  - Ziehen einer Zufallszahl  $x$ , mit  $0.99 \geq x \geq 0.00$
  - Falls  $x < 0.6$  dann Weiterleiten nach B
  - Falls  $0.9 \geq x \geq 0.6$  dann Weiterleiten nach C  
sonst Weiterleiten nach D

Tabelle in Knoten A

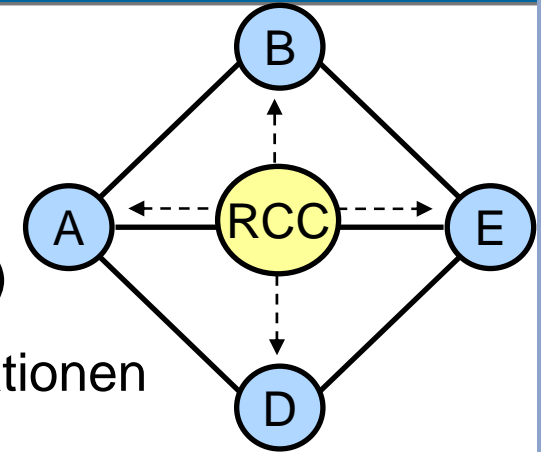
Ziel	1. Wahl		2. Wahl		3. Wahl	
	Kn	Gew	Kn	Gew	Kn	Gew
E	B	0,6	C	0,3	D	0,1
⋮						
⋮						



- Statisches Routing ist ein mächtiges Werkzeug, um das Routing-Verhalten in einem Netzwerk präzise zu kontrollieren.
- Wenn jedoch regelmäßige Änderungen in der Topologie auftreten, kann der hierfür erforderliche Aufwand der manuellen Neukonfiguration ein statisches Routing undurchführbar machen.

# Zentralisiertes Routing

- Ist ein adaptives Routingverfahren.
- **Verfahren:**
  - Im Netz gibt es ein Routing Control Center (RCC)
  - Jeder Knoten sendet periodisch Zustandsinformationen an das RCC, z.B.:
    - Liste aller aktiven Nachbarn
    - Aktuelle Längen der Warteschlangen
    - Umfang an Verkehr, der seit dem letzten Bericht abgewickelt wurde
  - RCC sammelt diese Zustandsinformationen und berechnet aufgrund dieser Kenntnis über das gesamte Netz die optimalen Wege zwischen allen Knoten (z.B. kürzeste Wege)
  - Jeder Knoten trifft seine Routing-Entscheidungen anhand der ihm zugewiesenen Routing-Tabelle vom RCC



# Zentralisiertes Routing

---

## ■ Vorteile:

- Im Prinzip hat das RCC die vollständige Übersicht, und damit können perfekte Entscheidungen getroffen werden.
- Die Router müssen keine aufwendigen Routing-Berechnungen durchführen.

## ■ Nachteile:

- Für große Netze dauert die Routing-Berechnung sehr lang.
- Der Ausfall des RCCs lähmt das ganze Netz, dadurch ist eine Redundanz durch Back-up Rechner dringend erforderlich.
- Es sind globale Inkonsistenzen möglich, da Router nahe dem RCC neue Routing-Tabellen wesentlich früher erhalten, als weiter entfernte Knoten.
- Eine starke Belastung des RCCs durch die zentrale Funktion.

# Hot Potato

---

- Hot Potato ist ein isoliertes Routing, d.h. es findet keinerlei Austausch mit den Nachbarknoten bezüglich der Wegwahlinformation statt.
- **Verfahren:**
  - Jeder Knoten versucht, eingehende Pakete so schnell wie möglich weiterzuleiten (bildhaft, wie eine heiße Kartoffel).
  - Die Übertragungsleitung mit der kürzesten Warteschlange wird für die Weiterleitung genutzt.
  - Hierbei kann es jedoch zu sehr großen Umwegen kommen (kürzeste Warteschlange ist i.d.R. nicht der kürzeste Weg!).
- **Varianten:**
  - Auswahl der besten Übertragungsleitung nach statischem Verfahren, solange deren Warteschlange unter einer bestimmten Schwelle bleibt.
  - Auswahl der Übertragungsleitung mit kürzester Warteschlange, falls deren statisches Gewicht nicht zu niedrig ist.

# Broadcast Routing (1/2)

## → Fluten (Flooding)

---

- Broadcast Routing wird auch Fluten (Flooding) genannt.
- Ist ein statisches, isoliertes und einfaches Routingverfahren, d.h. auch hier findet keinerlei Austausch mit den Nachbarknoten bezüglich der Wegwahlinformation statt.
- **Verfahren:**
  - Jedes eingehende Paket wird auf allen Übertragungsleitungen (Ausgangsleitungen) weitergeleitet, außer auf derjenigen, auf der es eingetroffen ist.
  - Probleme:
    - Beim Flooding werden viele Paketduplikate erstellt (Paketflut).
    - Diese Zahl kann unendlich sein (z.B. durch Zirkulieren), wenn nicht Maßnahmen ergriffen werden, um diese einzudämmen.



# Broadcast Routing (2/2)

---

## ■ Maßnahmen zur Eindämmung der Flut:

- Erkennung von Duplikaten durch die Numerierung der Pakete.
- Kontrolle der Lebensdauer eines Pakets durch Zählen der zurückgelegten Teilstrecken (hops).
  - Initialisierung eines hop-Zählers im Paket durch minimale bzw. maximale Weglänge zwischen Quelle und Ziel.
  - Dekrementieren des Zählers um 1 in jedem Knoten.
  - Bei Erreichen von 0 Verwerfung des Pakets.

## ■ Varianten:

- Selektives Fluten bedeutet, dass Weiterleitung nicht auf allen, sondern nur auf einigen Ausgangsleitungen durchgeführt wird.
- Random Walk bedeutet die zufällige Auswahl einer Ausgangsleitung.

# Distance Vector Routing

## → Übersicht

---

- Distance Vector Routing ist ein **verteiltes, adaptives Routing**.
- Jeder Knoten tauscht periodisch Routing-Informationen mit jedem Nachbarn aus.
- Typischerweise unterhält jeder Knoten eine Routing-Tabelle, die für jeden anderen Knoten im Netz einen Eintrag enthält, beispielsweise über:
  - bevorzugte Übertragungsleitungen
  - Schätzung über Zeit oder Entfernung zu diesem Knoten.
- Distance Vector Routing wurde früher als RIP (Routing Information Protocol) im Internet viel benutzt.

# Distance Vector Routing

## → Verfahren

---

- Jeder Router speichert eine Tabelle mit den besten Entfernungen (z.B. Anzahl der hops, Verzögerung in ms) zu jedem Ziel und dem dazugehörigen Ausgang.
- Diese Tabelle wird periodischer mit den Nachbarn austauscht.
- **Bewertung:**
  - Die Ansprüche an den Router sind sehr gering, da er nur den jeweiligen nächsten Hop kennen muss.
  - Allerdings stabilisieren sich die Routing-Tabellen im Netz nur sehr langsam, wenn sich Veränderungen in den Routern ergeben („count-to-infinity“-Problematik).

# Distance Vector Routing

## → Beispiel (1/4)

- Als Beispiel nehmen wir an, dass die Übertragungszeit als Maß verwendet wird und der Router die Übertragungszeit zu allen seinen Nachbar-Router kennt (mit Hilfe von ECHO-Paketen).
- Einmal alle T s sendet jeder Router an jeden Nachbarn eine Liste mit den geschätzten Übertragungszeiten zu jedem Ziel.
- Das heisst auch, jeder Router empfängt eine Liste seiner Nachbarn.
- Nehmen wir an, dass eine Tabelle gerade vom **Nachbarn-Router X** eingetroffen ist, wobei **Xy** die Schätzung von **X** ist, wie lange der Weg zu **Router Y** dauert.
- Wenn der Router weiß, dass die Übertragung zu **X m** ms dauert, weiß er auch, dass er **Router Y** über **X** in **Xy+m** ms erreichen kann.
- Wenn man dies für jeden Nachbarknoten berechnet, kann ein Router herausfinden, welche Schätzung am besten zu sein scheint, und diese mit der zugehörigen Leitung dann in seiner neuen Routing-Tabelle verwendet.

# Distance Vector Routing

## → Beispiel (2/4) - Liste der geschätzten Übertragungszeiten

- Die ersten vier Spalten zeigen die von den Nachbarn von Router J eingegangenen Übertragungsvektoren.
- J selber schätzt die Übertragungszeit:
  - zu A auf 8 ms,
  - zu I auf 10 ms,
  - zu H auf 12 ms und
  - zu K auf 6 ms.

To	A	I	H	K
A	0	24	20	21
B	12	36	31	28
C	25	18	19	36
D	40	27	8	24
E	14	7	30	22
F	23	20	19	40
G	18	31	6	31
H	17	20	0	19
I	21	0	14	22
J	9	11	7	10
K	24	22	22	0
L	29	33	9	9

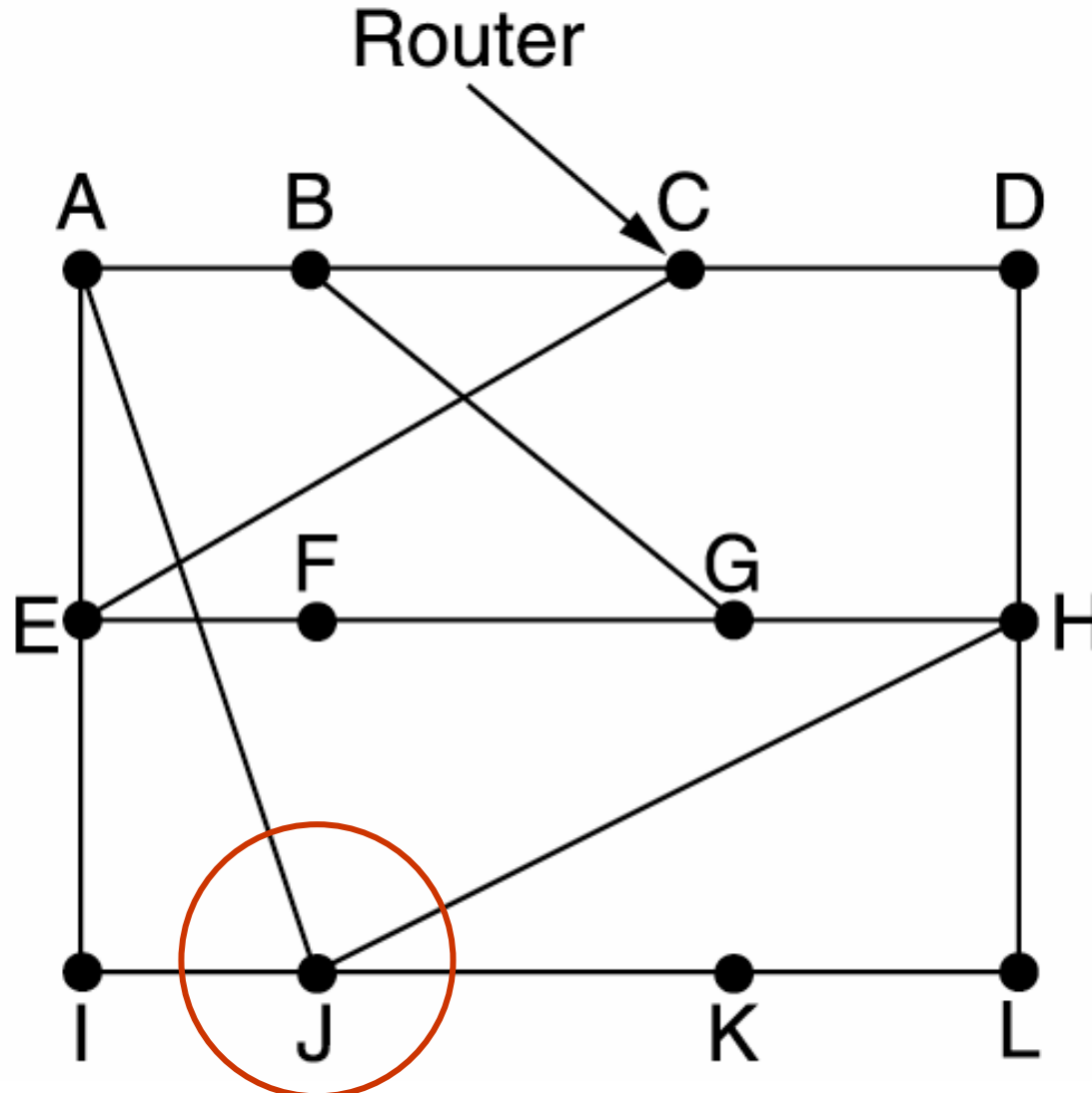
	JA	JI	JH	JK
Verzög.				
ist	8	10	12	6

Von den vier Nachbarn von J erhaltene Vektoren

# Distance Vector Routing

## → Beispiel (3/4)

- Beispiel-Teilnetz



# Distance Vector Routing

## → Beispiel (4/4) Berechnung des neuen Weg zu Router G

- J weiss, dass er A in 8 ms erreichen kann.
- A behauptet, G in 18 ms erreichen zu können.
- Daraus schließt J, dass er mit einer Übertragungszeit von 26 ms zu G rechnen muss, wenn er das Paket für G nach A überträgt.
- Weitere Berechnungen:
  - über I 41 (31+10) ms
  - **über H 18 (6+12) ms**
  - über K 37 (31+6) ms
- Über H ist der schnellste Weg zu G, aus diesem Grund trägt J in seine Routing-Tabelle H ein.
- Die gleiche Berechnung wird für alle anderen Ziele durchgeführt.

Neu geschätzte Übertragungsverzögerung von J

To	A	I	H	K	↓ Leitung	
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K

JA	JI	JH	JK	
Verzög. ist	Verzög. ist	Verzög. ist	Verzög. ist	Neue Routing-Tabelle für J
8	10	12	6	

Von den vier Nachbarn von J erhaltene Vektoren

# Distance Vector Routing

## → Count-to-Infinity-Problem (1/6)

---

- Distance Vector Routing funktioniert theoretisch, hat in der Praxis aber einen großen Nachteil.
- Der Algorithmus führt zwar zur richtigen Lösung, schafft das aber sehr langsam.
- Insbesondere reagiert er schnell auf gute Nachrichten, aber sehr träge auf schlechte Nachrichten.
- Nehmen wir an, der beste Weg eines Routers zu Ziel X ist lang.
- Meldet Nachbar A beim nächsten Mal plötzlich eine kürzere Übertragungszeit zu X, schaltet der Router einfach um, indem er Leitungen A verwendet, um den Datentransfer an X zu senden.
- Die gute Nachricht wird mit einem einzigen Austausch im Vektor verarbeitet.



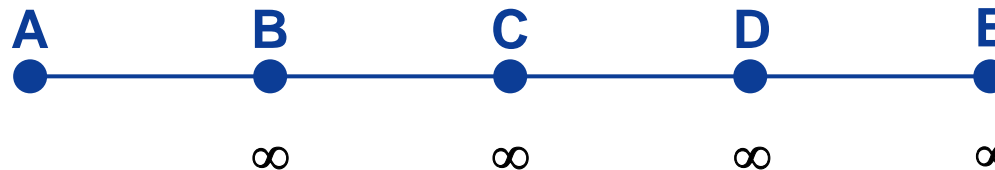
# Distance Vector Routing

## → Count-to-Infinity-Problem (2/6)

- Um zu sehen, wie schnell sich gute Nachrichten verbreiten, betrachten wir das (lineare) Teilnetz mit fünf Knoten, bei dem das Maß für die Übertragung die Anzahl der Teilstecken ist.



- Wir nehmen an, dass A anfänglich nicht in Betrieb ist und dies alle anderen Router wissen.
- Mit anderen Worten, sie haben alle die Übertragung nach A als unendlich registriert.



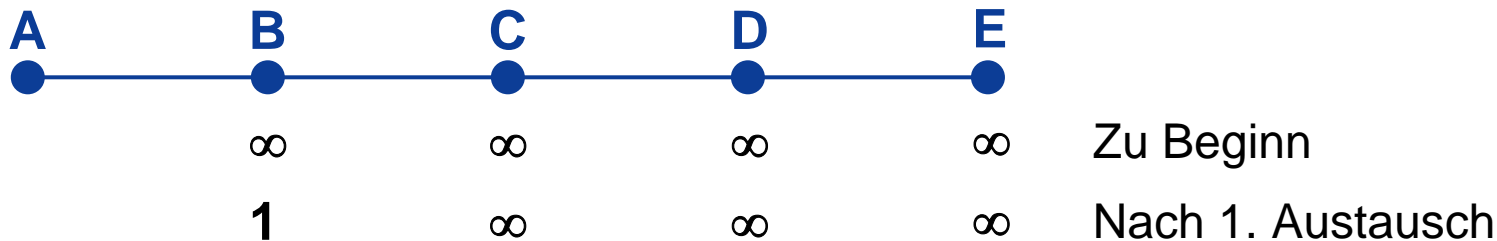
Zu Beginn

- Schaltet sich A wieder zu, erfahren das die anderen Router über den Vektoraustausch.
- Wir nehmen an, dass alle Router gleichzeitig den Vektoraustausch durchführen.

# Distance Vector Routing

## → Count-to-Infinity-Problem (3/6)

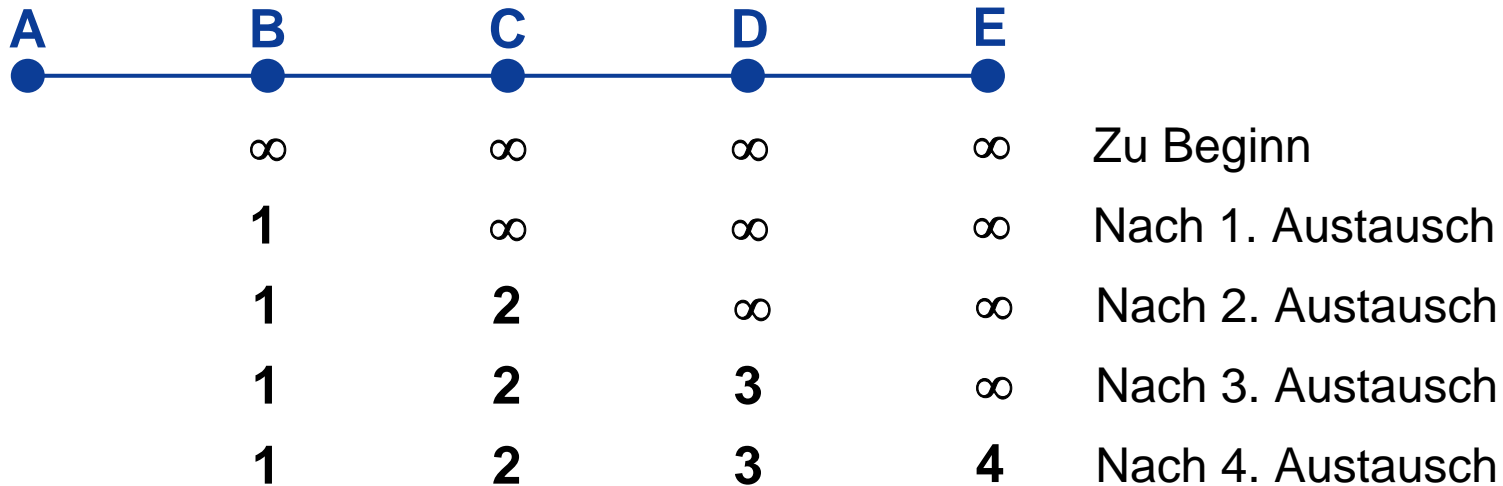
- Zum Zeitpunkt des ersten Austausches erfährt B, dass sein linker Nachbar eine Übertragungszeit von 1 zu A hat.
- B trägt in seine Routing-Tabelle ein, dass A **eine** Teilstrecke nach links entfernt liegt.
- Alle anderen Router glauben noch, dass A ausgeschaltet ist.
- An diesem Punkt sehen die Einträge in der Routing-Tabelle für A noch folgendermaßen aus:



- Beim nächsten Mal erfährt C, dass B einen Pfad von Länge 1 zu A hat.
- Folglich aktualisiert C seine Routing-Tabelle auf 2, was D und E aber erst später erfahren.

# Distance Vector Routing

## → Count-to-Infinity-Problem (4/6)

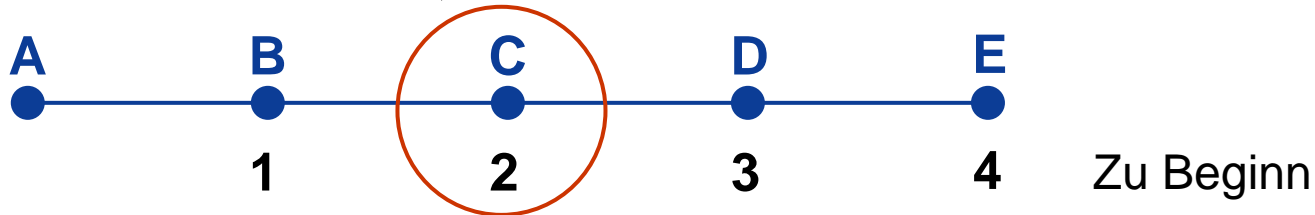


- Die gute Nachricht verbreitet sich in der Geschwindigkeit von einer Teilstrecke pro Austausch.
- In einem Teilnetz, dessen längster Pfad eine Länge von N Teilstrecken hat, tauschen alle innerhalb von N die Nachrichten über neu zugeschaltete Router und Leistungen aus.

# Distance Vector Routing

## → Count-to-Infinity-Problem (5/6)

- Was passiert nun, wenn A ausfällt oder die Leitung zwischen A und B unterbrochen wird, was aus der Sicht von B das Gleiche ist?



- Beim ersten Paketaustausch hört B überhaupt nichts von A.
- C teilt aber mit, dass er einen Pfad der Länge 2 zu A hat.
- B weiß nicht, dass dieser Pfad über ihn, B, zu A führt.
- Deshalb denkt B, er könne A über C mit einer Pfadlänge von 3 erreichen und aktualisiert seine Tabelle.
- Beim nächsten Austausch erfährt C, dass alle seine Nachbarn behaupten, einen Pfad der Länge 3 zu A zu haben.
- Er wählt einen aus und setzt seine Entfernung zu A auf 4.
- Diese Spiel setzt sich weiter fort, bis irgendwann alle Router den Wert  $\infty$  für die Entfernung zu A in der Tabelle haben.
- Wie lange dies geschieht, hängt vom numerischen Wert von  $\infty$  ab.

# Distance Vector Routing

## → Count-to-Infinity-Problem (6/6)

A	B	C	D	E	
●	●	●	●	●	
	1	2	3	4	Zu Beginn
	3	2	3	4	Nach 1. Austausch
	3	4	3	4	Nach 2. Austausch
	5	4	5	4	Nach 3. Austausch
	5	6	5	6	Nach 4. Austausch
	7	6	7	6	Nach 5. Austausch
	7	8	7	8	Nach 6. Austausch
	...	...	...	...	...
	$\infty$	$\infty$	$\infty$	$\infty$	Nach $\infty$ . Austausch

- Bei diesem Beispiel wird deutlich, dass sich schlechte Nachrichten langsam verbreiten.
- Die Zahl der benötigten Übertragungen hängt davon ab, auf welchen Wert  $\infty$  gesetzt wurde.
- Daher wird  $\infty$  auf den längsten Pfad+1 gesetzt.
- Als Metrik kann nicht die Verzögerung verwendet werden, da es hier keine definierte Obergrenze gibt.

# Link State Routing

## → Übersicht (1/2)

---

- Link State Routing ist ein verteiltes und adaptives Routing-Verfahren.
- Der Name Link State (Zusand der Verbindung) beschreibt die Tatsache, dass bei diesem Routing-Verfahren jeder Router die Topologie des kompletten Netzes kennt, an das er angeschlossen ist.
- Damit kann ein Router den besten Weg eines Datenpaketes selbst ermitteln.
- Dazu müssen die Router die Informationen des Netzes in geeigneter Weise in ihrer Datenbank speichern.
- Das Link State Routing stellt im Gegensatz zum Distance Vector Routing **sehr hohe Ansprüche an die Rechnerleitung und Speicherressourcen der Router**.
- Dafür stabilisiert sich die Routing-Tabelle bei einer Veränderung aber sehr viel schneller.

# Link State Routing

## → Übersicht (2/2)

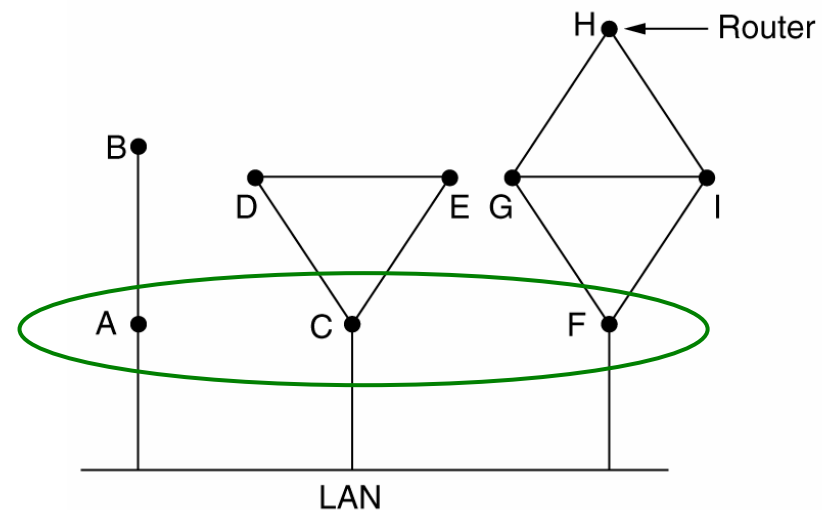
---

- Das Link State Routing besteht aus fünf Schritten:
  - **1.** Die **Nachbarn und deren Netzadressen ermitteln** (Hello-Paket)
  - **2.** Die **Übertragungszeit oder die Kosten** zu jedem seiner Nachbarn **messen** (ECHO-Paket)
  - **3.** Ein **Paket zusammenstellen**, in dem alles steht, was gelernt wurde
  - **4.** Dieses **Paket an alle anderen Router senden** (Flooding)
  - **5.** Den **kürzesten Pfad zu allen anderen Routern berechnen** (SPF-Algorithmus (Shortest Path First Algorithmus) von E. W. Dijkstra)

# Link State Routing

## → 1. Ermittlung der Nachbar-Router (1/2)

- Nach dem Start muss ein Router als erstes ermitteln, wer seine Nachbarn sind.
- Hierzu sendet er ein spezielles HELLO-Paket auf jede Punkt-zu-Punkt-Leitung.
- Der Router am anderen Ende muss eine Antwort zurücksenden, durch die er seine Identität bekannt gibt.
- Diese Namen müssen global eindeutig sein.
- Sind zwei oder mehrere Router an ein LAN angeschlossen, ist die Situation etwas komplizierter.

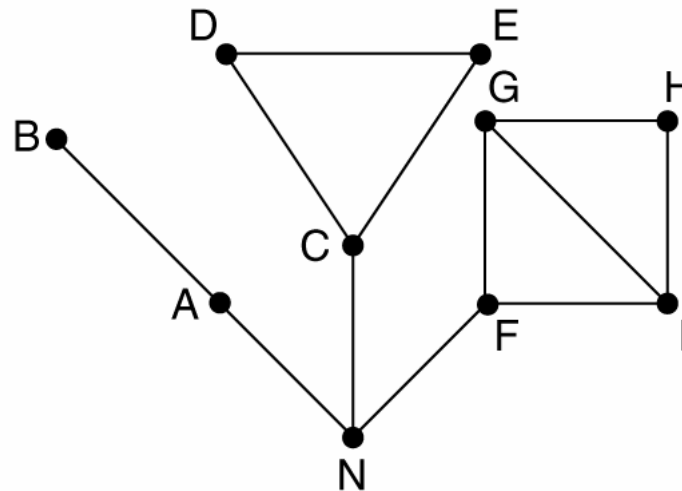




# Link State Routing

## → 1. Ermittlung der Nachbar-Router (2/2)

- Eine Möglichkeit, das LAN zu modellieren, ist, es selbst als Knoten zu betrachten.



- Hier wird ein neuer künstlicher Knoten N eingeführt, an dem A, C und F angeschlossen sind.
- Die Tatsache, dass man im LAN von A nach C gelangen kann, ist hier durch den Pfad A-N-C dargestellt.

# Link State Routing

## → 2. Ermittlung der Leitungskosten

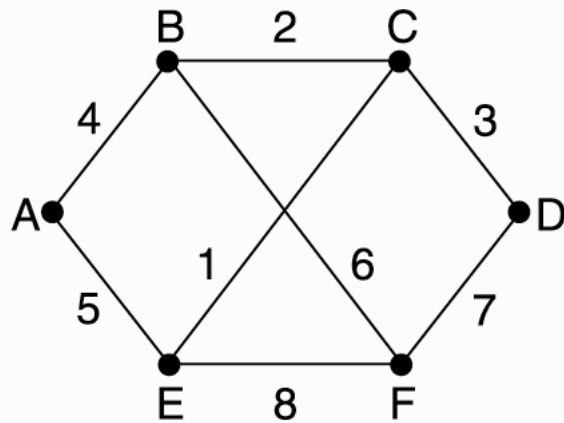
---

- Das Link State Routing setzt voraus, dass jeder Router die Übertragungszeit zu seinen Nachbarn kennt oder zumindestens einen guten Schätzwert dafür hat.
- Der direkteste Weg, diese Übertragungszeit zu ermitteln, ist das Aussenden eines speziellen ECHO-Pakets.
- Die andere Seite muss es sofort wieder zurücksenden.
- Durch Messen der Hin- und Rückreisezeit, geteilt durch zwei, kann der sendende Router die Übertragungszeit vernünftig abschätzen.
- Bessere Ergebnisse werden erzielt, wenn der Test mehrmals wiederholt und dann der Durchschnitt gebildet wird.

# Link State Routing

## → 3. Link-State-Pakete erstellen (1/2)

- Nachdem die erforderlichen Informationen für den Austausch erfasst wurden, muss jeder Router im nächsten Schritt ein Paket mit allen Daten zusammenstellen.
- Das Paket beginnt mit der Identität des Senders, gefolgt von einer Folgenummer und dem Alter sowie einer Liste der Nachbarn mit den entsprechenden Verzögerung.



Link - State - Pakete

A		B		C		D		E		F	
Folge-Nr.		Folge-Nr.		Folge-Nr.		Folge-Nr.		Folge-Nr.		Folge-Nr.	
Alter		Alter		Alter		Alter		Alter		Alter	
B	4	A	4	B	2	C	3	A	5	B	6
E	5	C	2	D	3	F	7	C	1	D	7
		F	6	E	1			F	8	E	8

# Link State Routing

## → 3. Link-State-Pakete erstellen (2/2)

---

- Das Erstellen von Link-State-Paketen ist einfach.
- Der schwierige Teil ist die Frage, wann sie erstellt werden sollen.
- Eine Möglichkeit ist, sie periodisch aufzubauen, d.h. in regelmäßigen Abständen.
- Eine Alternative ist die Erstellung, wenn ein bestimmtes wichtiges Ereignis eintritt, z.B. der Ausfall oder die Aufnahme einer Leistung oder eines benachbarten Routers oder eine erhebliche Änderung der Eigenschaften.

# Link State Routing

## → 4. Link-State-Pakete verteilen (1/3)

- Der schwierigere Teil des Algorithmus ist die zuverlässige Verteilung der Link-State-Pakete.
- Bei der Verteilung und Installation der Pakete ändern die Router, die sie als erstes erhalten, ihre Wege.
- Folglich werden in verschiedenen Routern eventuell unterschiedliche Versionen der Topologie verwendet, was zu Inkonsistenzen, Schleifen, unerreichbaren Rechnern und anderen Problemen führen kann.
- Das Basiskonzept des Verteilungsalgorithmus ist die Anwendung von Flooding zur Verteilung der Link-State-Pakete.
- Jedes Paket enthält eine Folgenummer, die bei jedem neu ausgesendeten Paket erhöht wird.
- Die Router vermerken alle Paare (Quell-Router, Folgenummer), die sie sehen.
- Kommt ein neues Link-State-Paket an, wird es mit der Paketliste verglichen.

# Link State Routing

## → 4. Link-State-Pakete verteilen (2/3)

- Ist es neu, wird es an alle Leitungen außer derjenigen, auf der es eingegangen ist, ausgegeben.
- Ist es ein Duplikat, wird es verworfen.
- Ist die Folgenummer eines Paketes niedriger als das höchste bisher erfasste, wird das Paket als veraltet abgelehnt.
- Um z.B. ausgefallene Router berücksichtigen zu können, wird das Alter eines Pakets mit aufgenommen (z.B. 60s) und pro Sekunde um 1 reduziert.
- Ist das Alter gleich null, werden die Informationen im Router verworfen.
- Kommt ein neues Paket alle zehn Sekunden an, „veraltet“ ein Paket nur, wenn ein Router ausfällt.
- Das Feld Alter wird von jedem Router auch während des anfänglichen Flooding-Verfahrens um 1 reduziert, um sicherzustellen, dass kein Paket verloren geht und ewig weiterlebt (ein Paket mit dem Alter null wird verworfen).

# Link State Routing

## → 4. Link-State-Pakete verteilen (3/3)

- Als Schutz vor Fehlern auf dem Routern/den Router-Leitungen werden alle Link-State-Pakete bestätigt.
- **Beispiel eines Paketpuffer für den Router B**

muss übertragen werden → Sende-Flags    ACK-Flags ← muss bestätigt werden

Quelle	Folge-Nr.	Alter	Sende-Flags			ACK-Flags			Daten
			A	C	F	A	C	F	
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	← da, von A und B gekommen, muss nur an C gesendet werden
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

- Jede Zeile entspricht einem kürzlich angenommenen, aber noch nicht voll verarbeiteten Link-State-Paket.
- In der Tabelle werden der Ursprung des Paketes, die Folgennummer, das Alter und Sende- und Bestätigungs-Flags erfasst.

# Link State Routing

## → 5. Berechnung neuer Wege (1/5)

- Nachdem ein Router eine vollständige Menge an Link-State-Paketen angesammelt hat, kann er den Graphen für das Teilnetz aufbauen, weil alle Verbindungen vertreten sind.
- Die Verbindungen werden zweimal dargestellt, je einmal für eine Richtung.
- Die zwei Werte können getrennt verwendet werden, oder aber man berechnet einen Durchschnitt davon.
- Jetzt kann mit einem passenden Algorithmus der kürzeste Pfad zu allen möglichen Zielen ermittelt werden.
- Die Ergebnisse werden in die Routing-Tabelle eingeführt und der Normalbetrieb aufgenommen.
- Bei einem Teilnetz mit  $n$  Routern, die je  $k$  Nachbarn haben, ist der zum Ablegen der Eingangsdaten erforderliche Speicher proportional zu  $kn$ .
- Bei großen Teilnetzen kann der Speicherplatz aber auch die Berechnungszeit problematisch sein.
- Das **OSPF-Protokoll** ist das am häufigste verwendete Routing-Protokoll im Internet und setzt einen Link-State-Algorithmus ein.



# Link State Routing

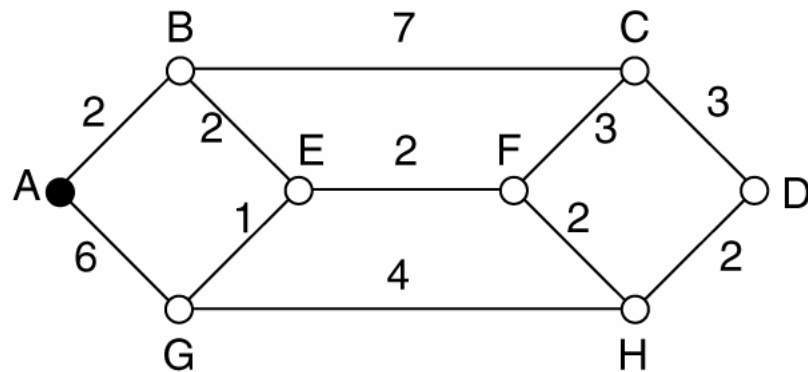
## → 5. Berechnung neuer Wege (2/5)

- Berechnung des kürzesten Pfades nach dem SPF-Algorithmus (Shortest Path First Algorithmus) von E. W. Dijkstra.
- Jeder Knoten wird mit seiner Entfernung vom Quellknoten auf dem besten bekannten Pfad beschriftet.
- Anfangs ist kein Pfad bekannt, so dass alle Knoten die Bezeichnung „unendlich“ tragen.
- Je weiter der Algorithmus fortschreitet und Pfade gefunden werden, desto mehr kann sich die Beschriftung ändern und jeweils besser Pfade anzeigen.
- Eine Beschriftung kann provisorisch oder permanent sein.
- Zunächst sind alle Beschriftungen provisorisch.
- Wird festgestellt, dass eine Beschriftung den kürzestmöglichen Pfad von Quelle zu einem Knoten angibt, wird sie permanent gesetzt und danach nie mehr geändert.

# Link State Routing

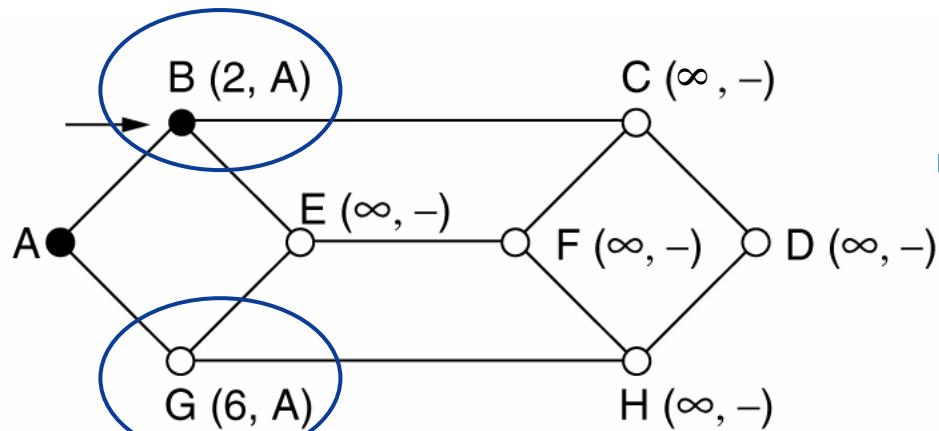
## → 5. Berechnung neuer Wege (3/5)

- Gesucht wird der kürzeste Weg von A nach D.



(a)

- Im Graphen sind die Gewichtung in Entfernung (1,2,3,...) eingetragen.
- Als erstes wird der Knoten A als permanent markiert.

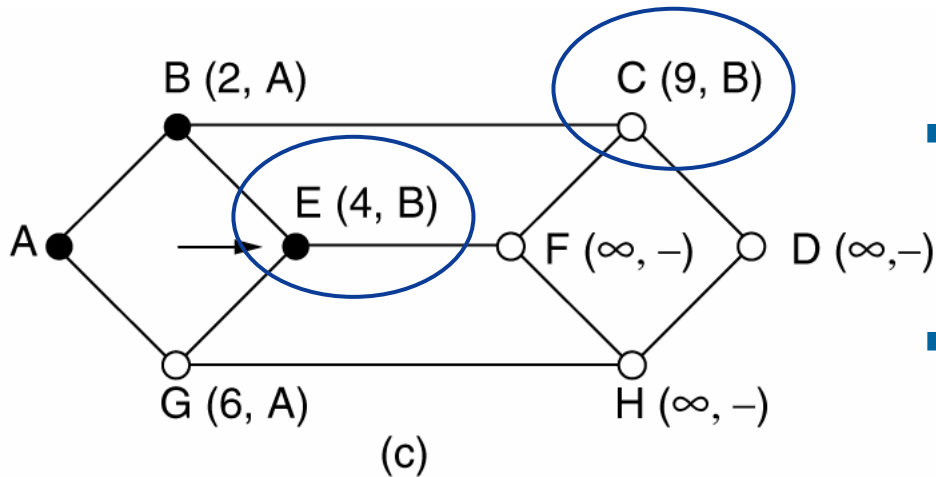


(b)

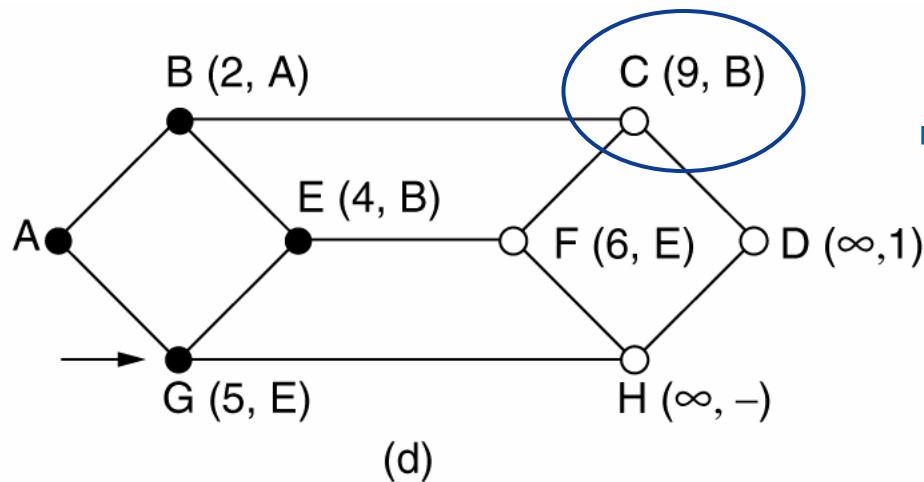
- Nun wird von A jeder benachbarte Knoten mit der Entfernung beschriftet.

# Link State Routing

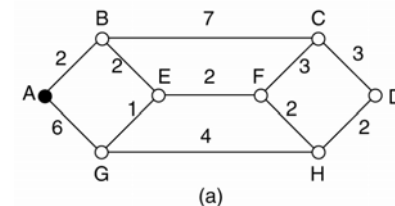
## → 5. Berechnung neuer Wege (4/5)



- Nun wird von B jeder benachbarte Knoten mit der Entfernung beschriftet.
- Da der Weg von A nach E über B der kürzeste ist, wird dieses als permanent markiert.

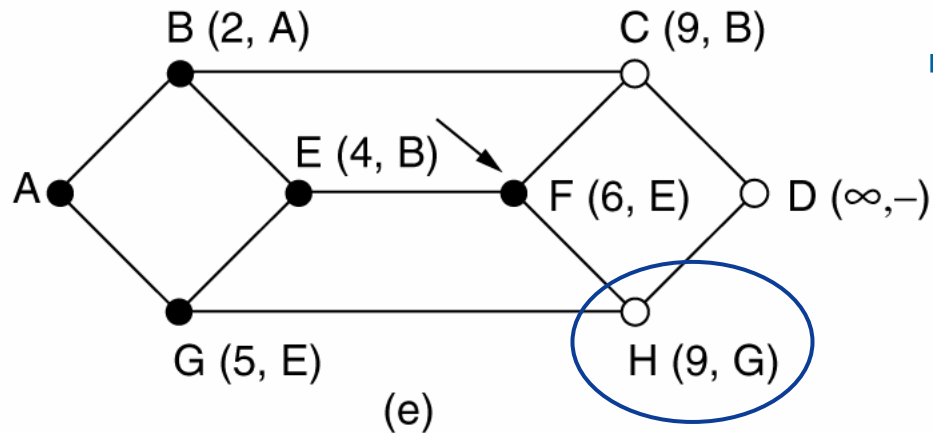


- Nun wird von E jeder benachbarte Knoten mit der Entfernung beschriftet.

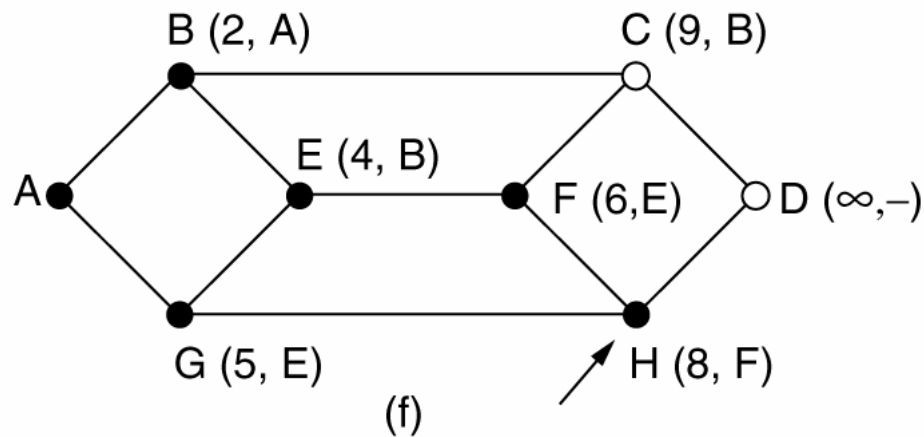


# Link State Routing

## → 5. Berechnung neuer Wege (5/5)



- Nun wird von G jeder benachbarte Knoten mit der Entfernung beschriftet.



- usw.
- Die Komplexität des Algorithmus ist im besten Fall  $O(n \cdot \log n)$ , wobei  $n$  die Anzahl der Verbindungen ist.

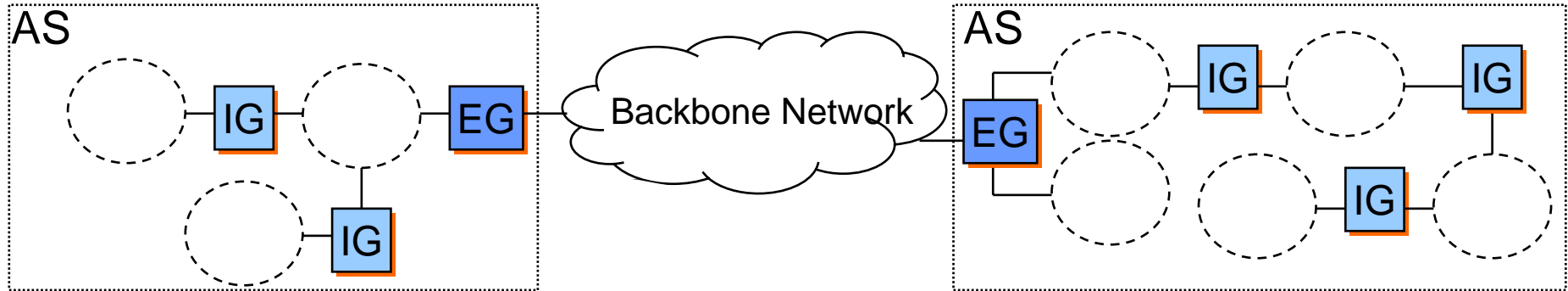
# Inhalt

---

- Ziele, Einordnung und Übersicht
- Router
- Routing-Verfahren
- **Routing-Protokolle**
  - Autonome Systeme / Internet
  - Ablauf
  - Zusammenfassung

# Routing-Protokolle

## → Übersicht (1/2) - Definitionen



### ■ **Autonomes System (AS):**

- Netz(e) unter einheitlicher Verwaltung.
- Ein AS kann aus sehr vielen Netzen bestehen, die wiederum intern mit Routern verbunden sind.

### ■ **Interior Gateway (IG):**

- Interner Router eines autonomen Systems (AS)

### ■ **Exterior Gateway (EG):**

- Router am Rande eines autonomen Systems (AS)

# Routing-Protokolle

## → Interior Gateway Protocol (IGP)

---

- Grundsätzlich unterscheidet man zwei Arten von Routing-Protokollen.
  - **Interior Gateway Protocol (IGP)**
  - **Exterior Gateway Protocol (EGP)**
- Ein **Interior Gateway Protocol (IGP)** wird innerhalb eines Netzes (autonomen Systems) eingesetzt.
- Das bekannteste IGP war das **Router Information Protocol (RIP)**, welches aufgrund seiner Einschränkungen und Probleme durch das **Open Shortest Path First (OSPF)** abgelöst wird.
- Ein internes Gateway-Protokoll muss lediglich Pakete so effizient wie möglich von der Quelle zum Ziel befördern.
- Es muss sich um keinerlei besondere Regeln kümmern.

# Routing-Protokolle

## → Exterior Gateway Protocol (EGP)

- Zur Kommunikation **zwischen Routern** von autonomen Netzen wird ein **Exterior Gateway Protocol (EGP)** eingesetzt.
- Das seit 1984 bestehende Protokoll EGP wurde mittlerweile weitgehend durch das **Border Gateway Protocol (BGP)** abgelöst.
- Bei Exterior Gateways müssen besondere Regeln beachtet werden.
- Z.B. ist ein Unternehmen nicht willens, Pakete auf dem Transit von einem fremden autonomen System zu einem anderen fremden autonomen System zu befördern, obwohl es auf dem kürzesten Weg zwischen den beiden fremden Systemen liegt.
- Andererseits ist es vielleicht bereit, Transitverkehr für seine Nachbarn oder bestimmte andere autonome Systeme zu übernehmen, falls sie für diesen Dienst bezahlen.
- Telekommunikationsgesellschaften stellen ihre Übertragungsdienste gerne zur Verfügung, nicht aber fremden Parteien.



# BGP (Border Gateway Protocol)

## → Überblick, Aufgaben

- Die Aufgabe von BGP ist, die Organisation und Durchführung der Wegewahl zwischen autonomen Systemen (AS) unter Berücksichtigung besonderer politischer, wirtschaftlicher oder sicherheitsbezogener Regeln.
- Beispiele für besondere Regel:
  - Datenverkehr soll nie durch bestimmte autonome System fließen.
  - Vom Pentagon ausgehender Datenverkehr darf nie über Irak übertragen werden.
  - Albanien darf nur durchquert werden, wenn es keine Alternative zum Ziel gibt.
  - Datenverkehr, der bei IBM beginnt und endet, darf nicht über Microsoft führen.
  - usw.
- Regeln und Maßnahmen werden normalerweise manuell in jedem BGP-Router konfiguriert (oder mit Hilfe eines Scripts aufgenommen).
- Sie sind nicht Teil des Protokolls.

# Inhalt

---

- Ziele, Einordnung und Übersicht
- Router
- Routing-Verfahren
- Routing-Protokolle
- **Autonome Systeme / Internet**
  - Ablauf
  - Zusammenfassung

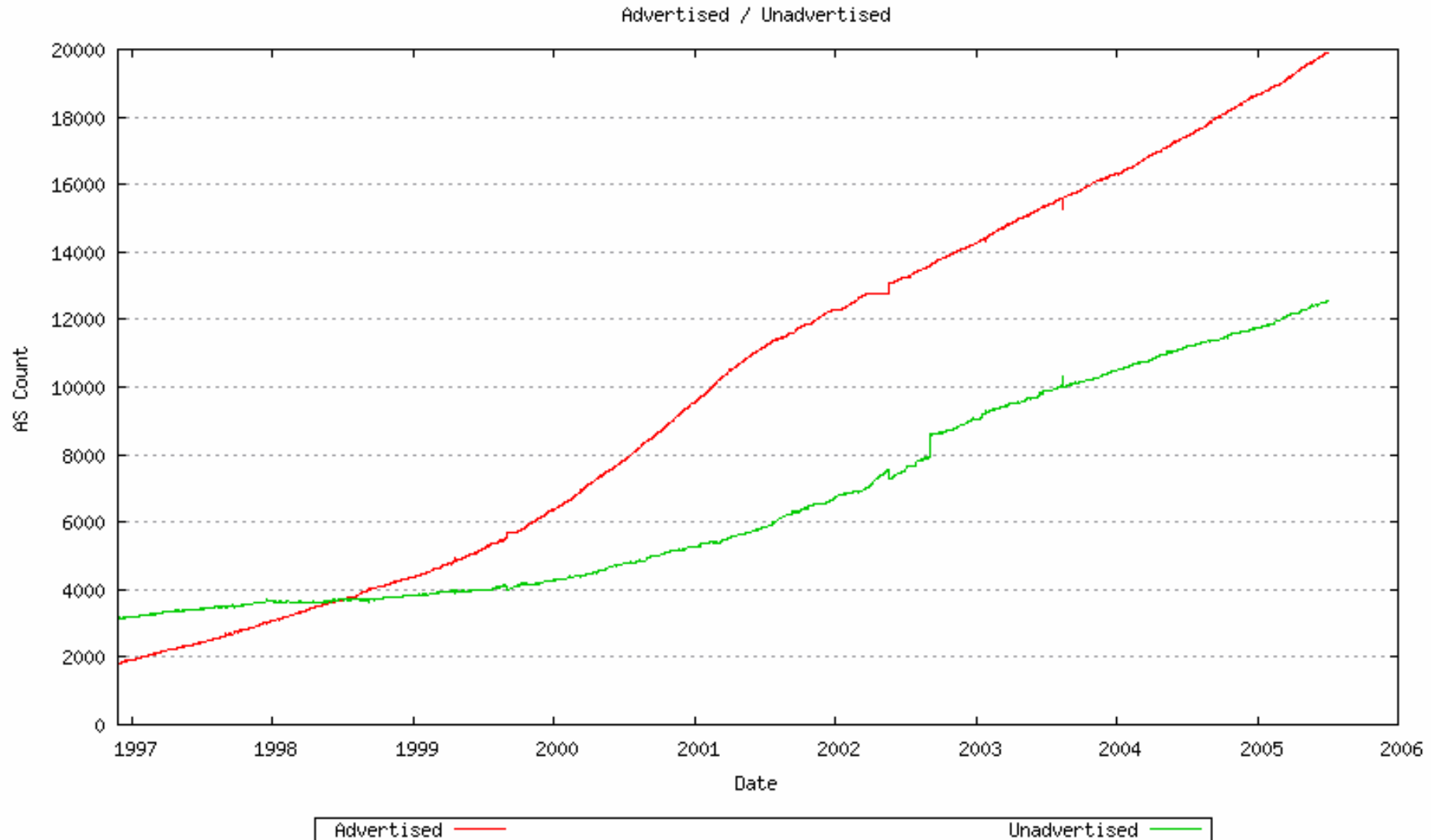
# Autonome Systeme

## → Übersicht (1/9)

- Netzwerken, die das Internet bilden, werden autonomen Systeme genannt.
- Dies bedeutet, dass ein Betreiber eines solchen autonomen Netzwerks, salopp gesagt, sich internet-fähige Netzwerktechnik kauft und sich von seiner regional zuständigen Vergabestelle für IP-Adressen einen Block von IP-Adressen zuteilen lässt, mit der er sein Netzwerk adressiert.
- Damit sein autonomes System eindeutig unterscheidbar ist, erhält er zusätzlich eine so genannte AS-Nummer, die sein autonomes System im gesamten Internet eindeutig kennzeichnet.
- Diese AS-Nummer ist gleich mehrfach der Schlüssel im weltweiten Routing zwischen autonomen Systemen.
- Zum einen repräsentiert die AS-Nummer nämlich das Netzwerk selbst, also beispielsweise alle Netzwerke, die ein Internet Provider betreibt.
- Das können die Netze sein, die er selbst für seine Server benötigt, aber auch Netzwerke und Adressblöcke sein, die er an Kunden weitergibt, die zwar IP-Adressen benötigen, aber kein eigenes, autonomes System betreiben müssen.

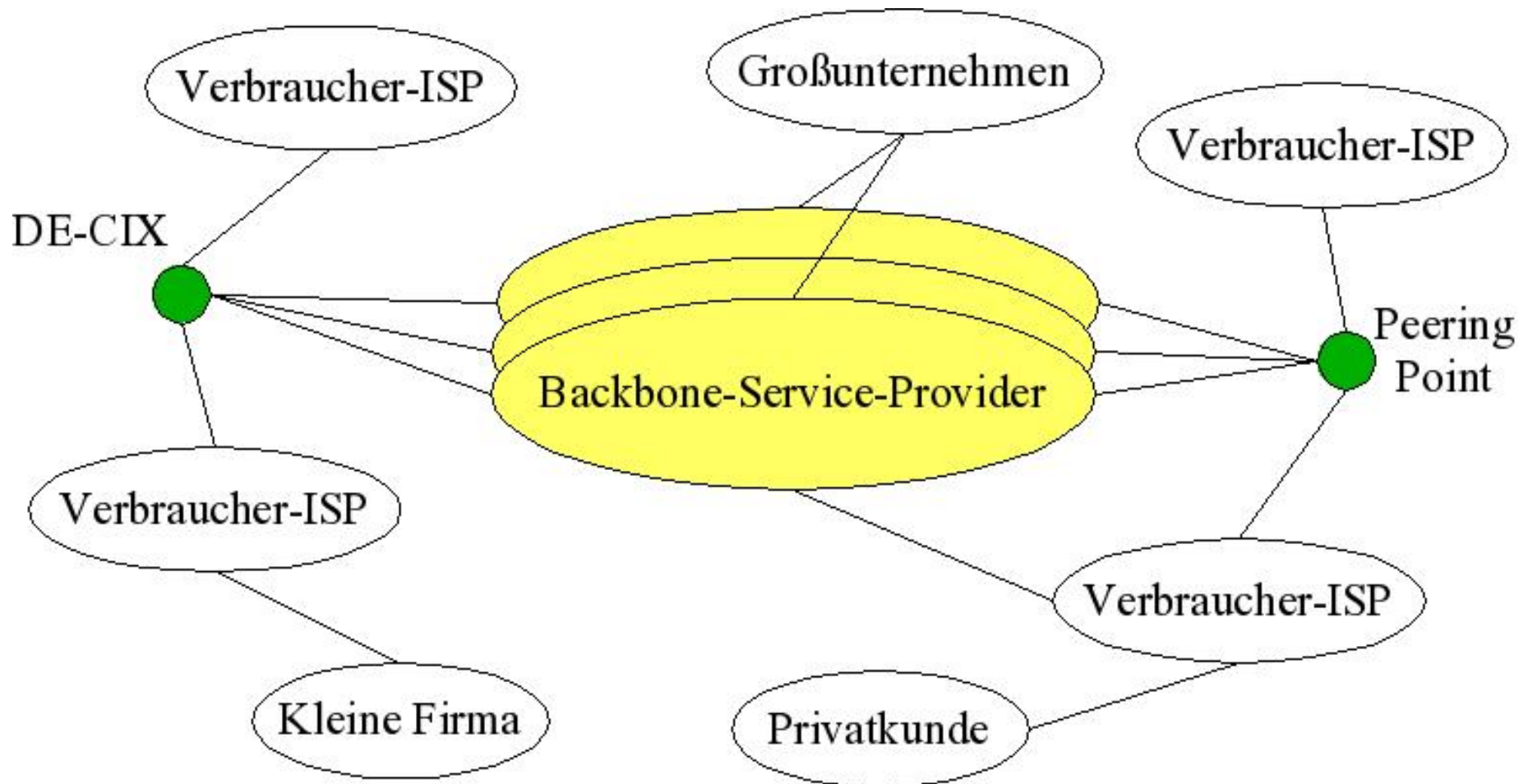
# Autonome Systeme

## → Übersicht (2/9)



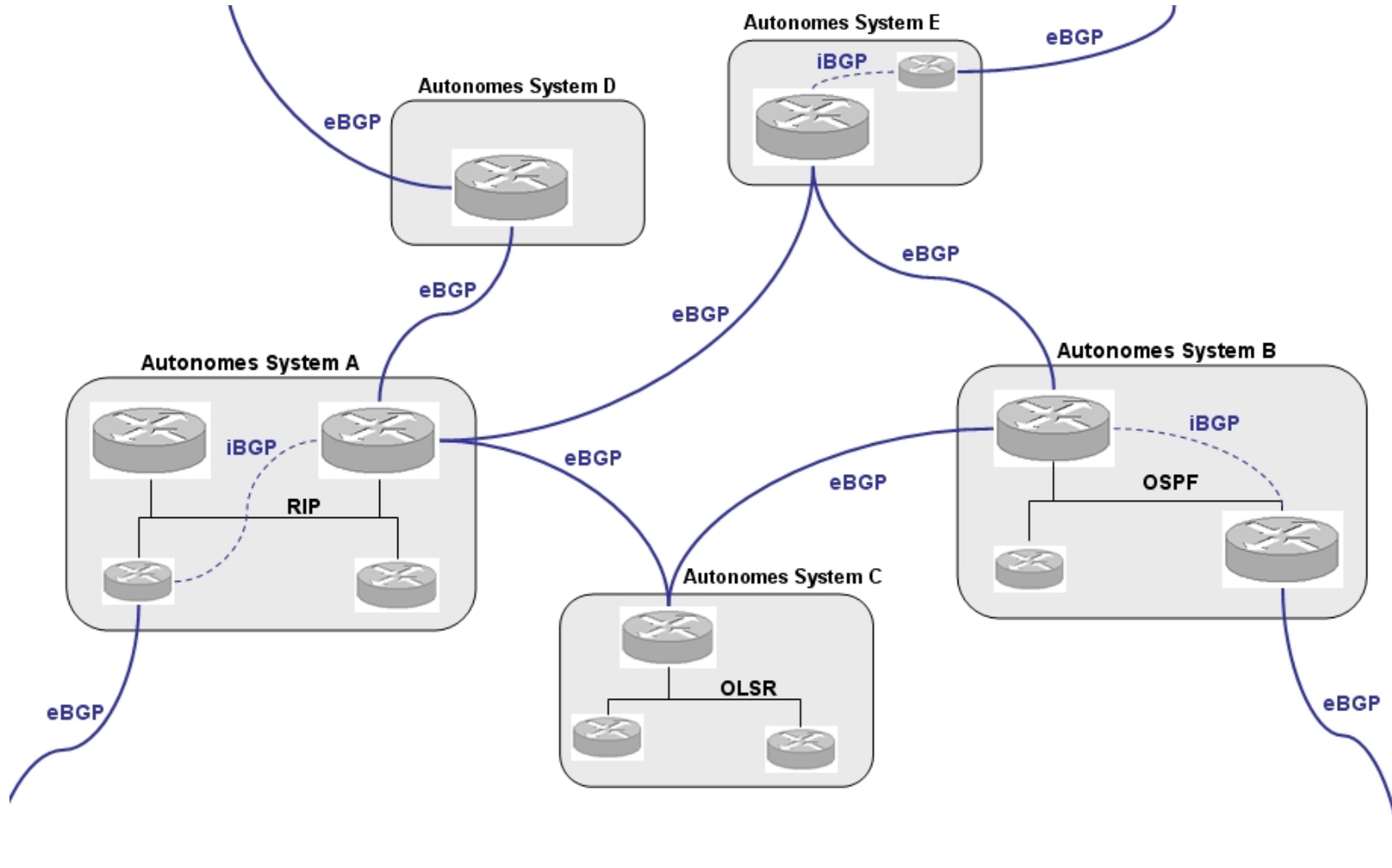
# Autonome Systeme

## → Übersicht (3/9)



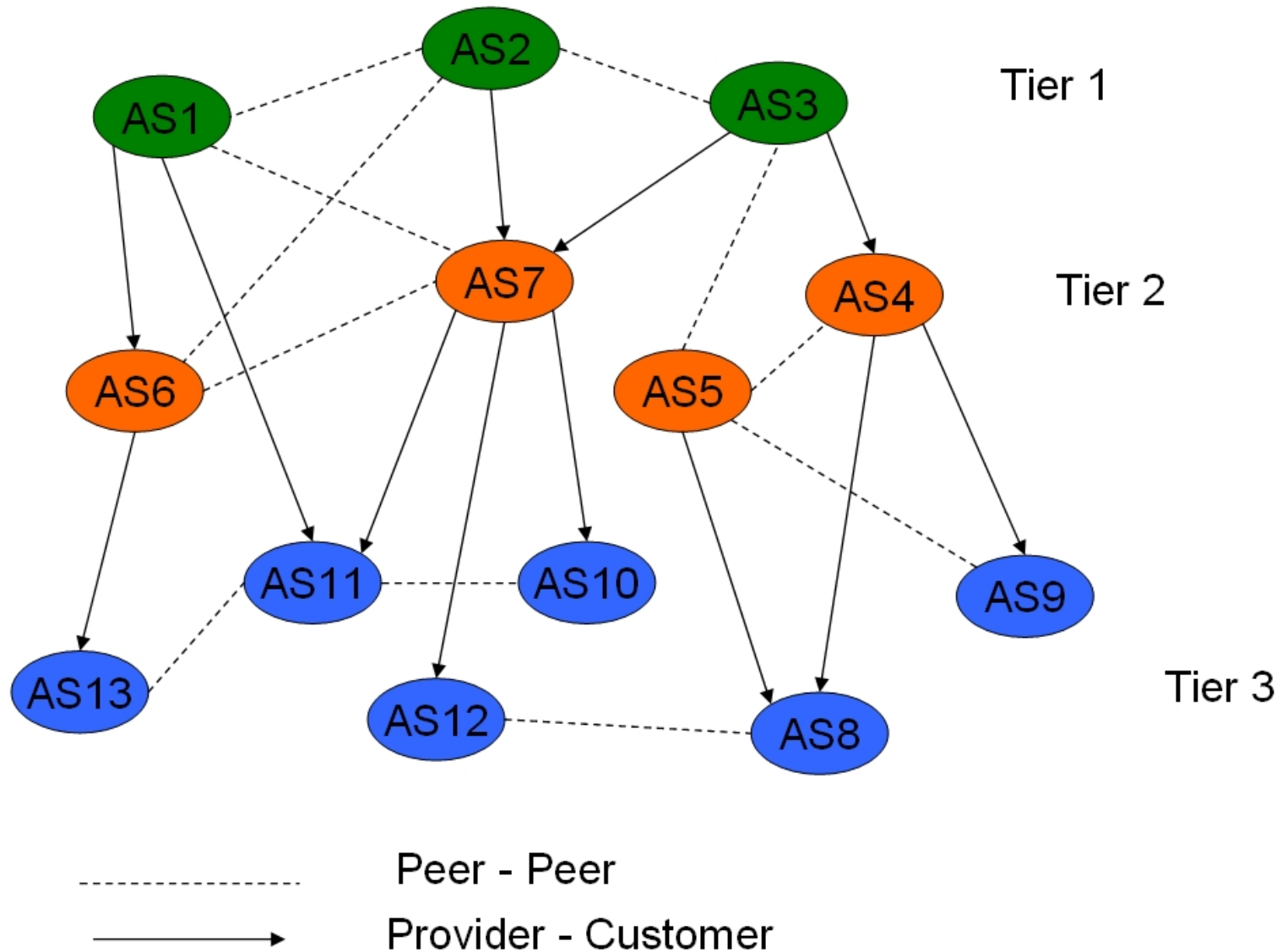
# Autonome Systeme

## → Übersicht (4/9)



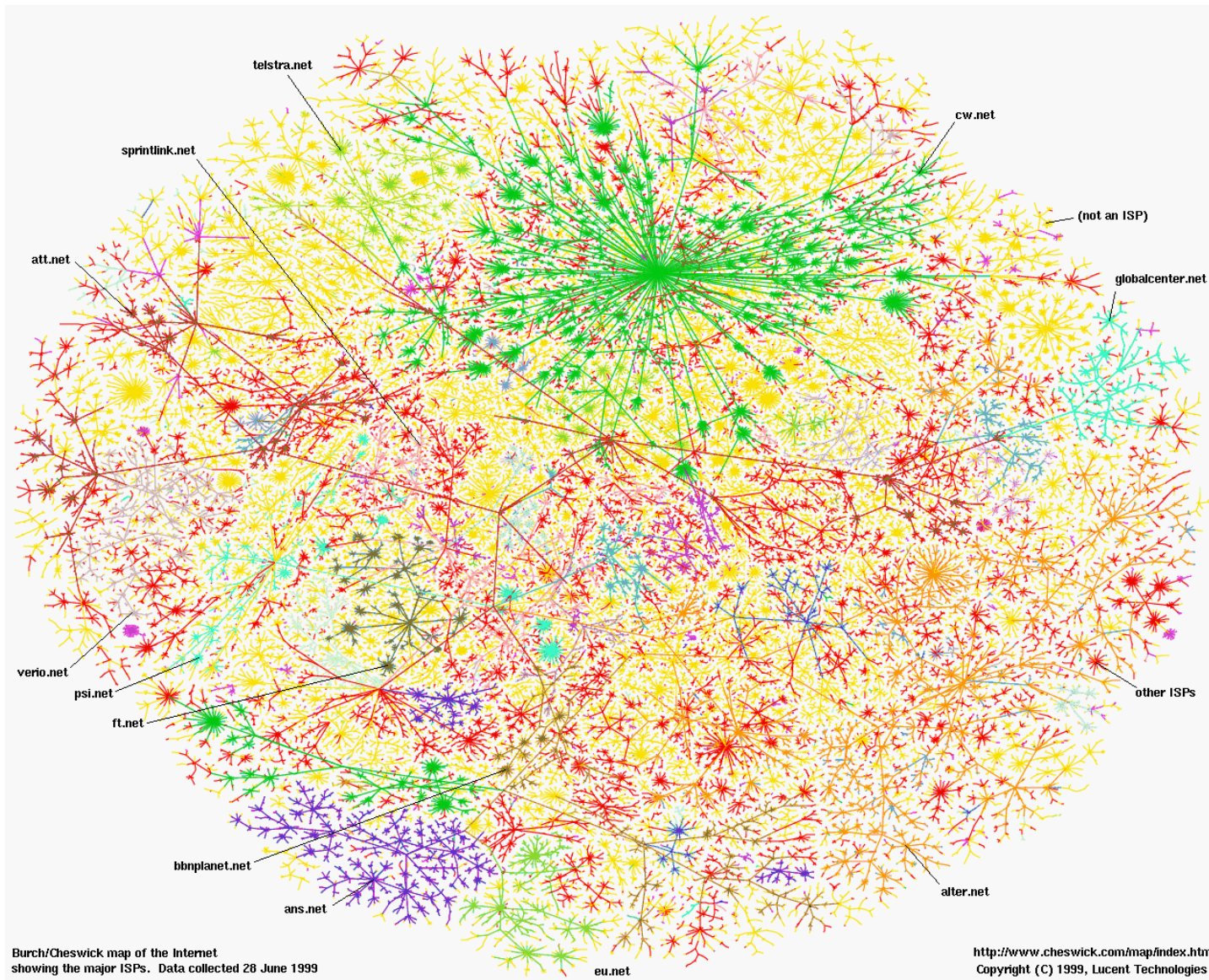
# Autonome Systeme

## → Übersicht (4/9)



# Autonome Systeme

## → Übersicht (6/9)





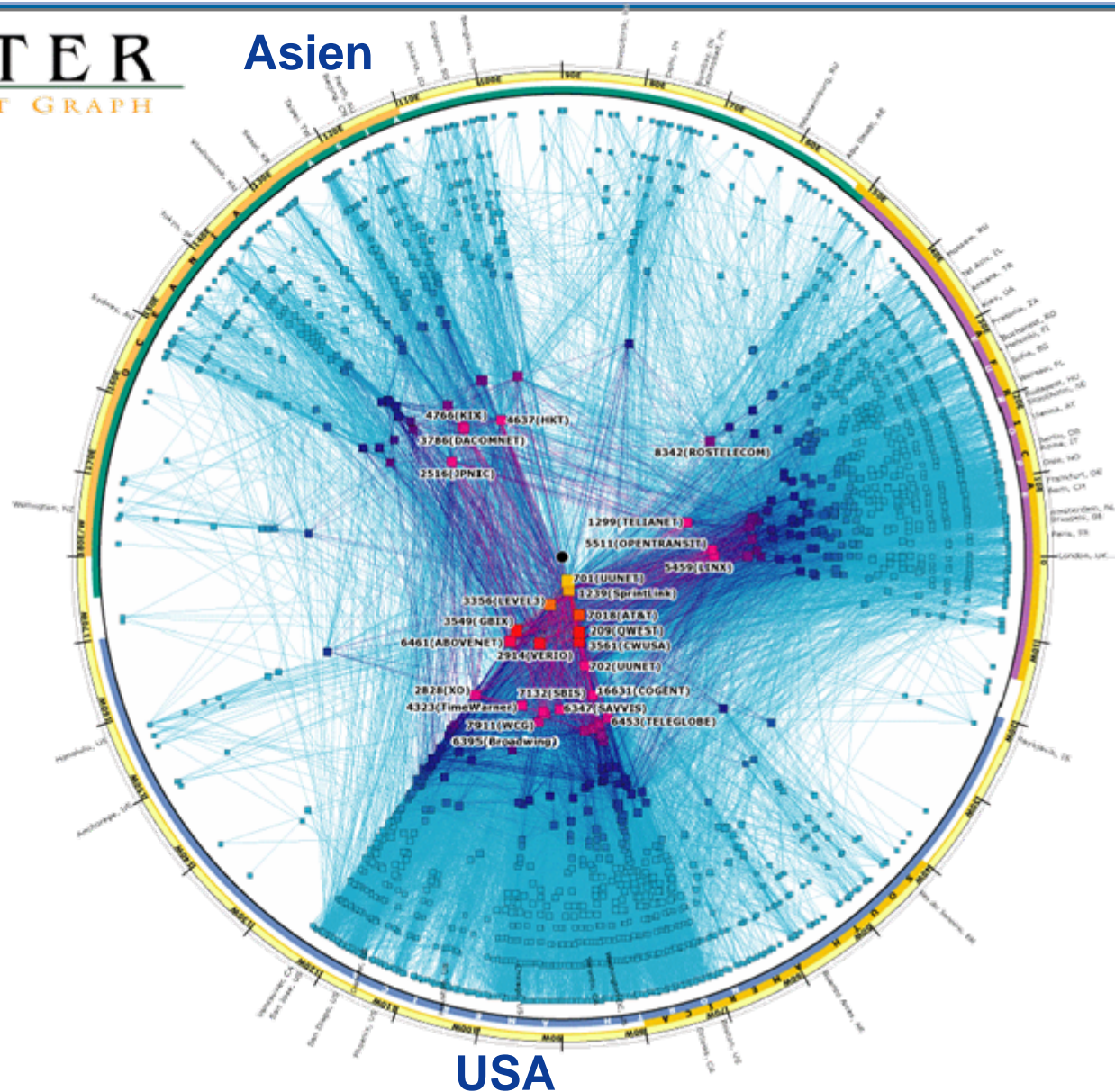
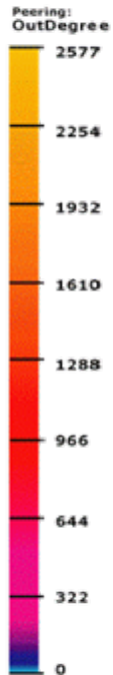
# Autonome Systeme

## → Übersicht (7/9)

SKITTER  
AS INTERNET GRAPH

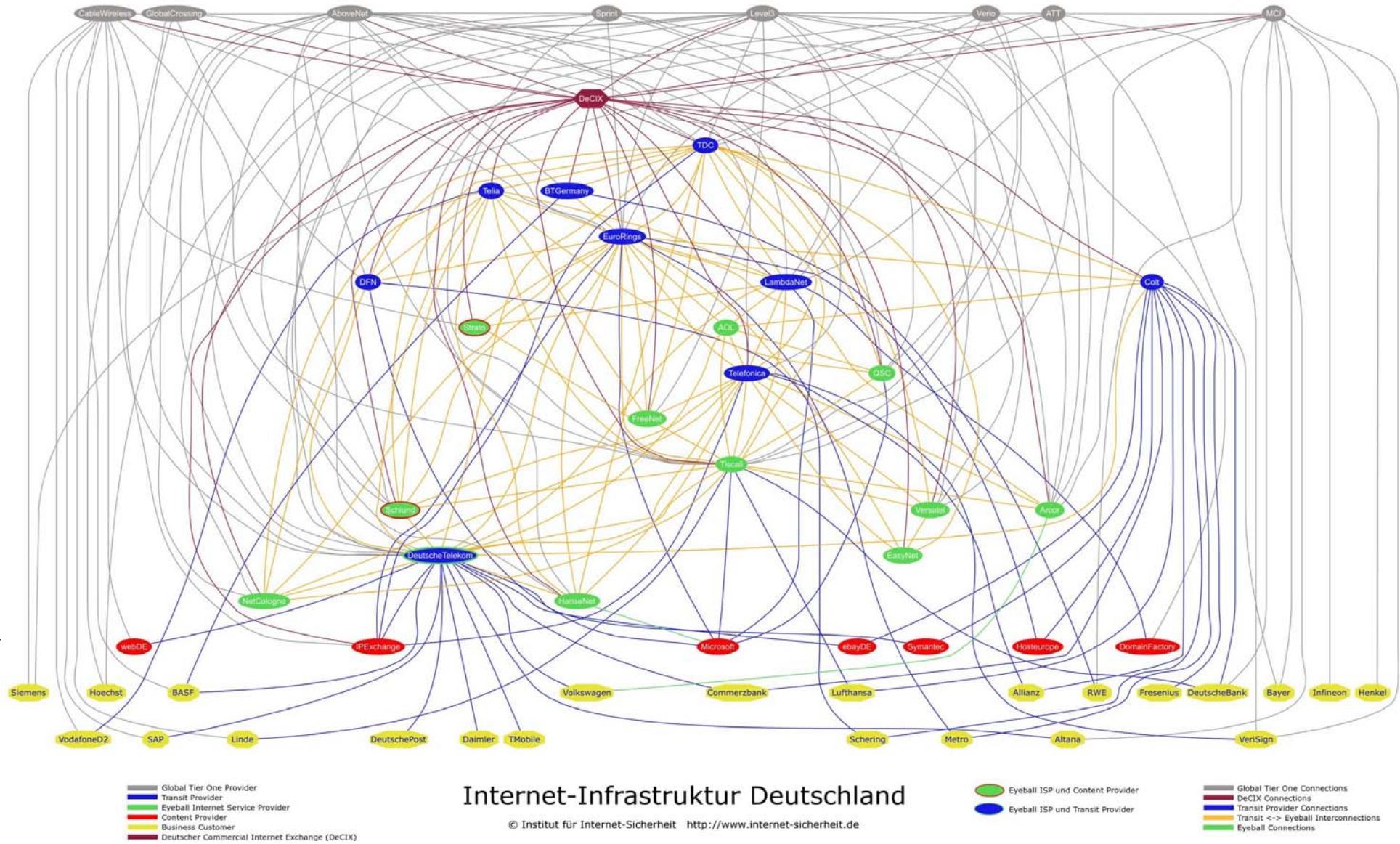
Asien

Europa



# Autonome Systeme

## → Übersicht (8/9)



### Internet-Infrastruktur Deutschland

© Institut für Internet-Sicherheit <http://www.internet-sicherheit.de>

# Autonome Systeme

## → Übersicht (9/9)

---

- Die AS-Nummer ermöglicht es demnach, verschiedene IP-Subnetze eines Netzbetreibers zusammenzufassen.
- Diese Zusammenfassung dient vor allem dem Zweck, das Routing auf oberster Ebene, also zwischen autonomen Systemen, dramatisch zu vereinfachen.
- Nicht jedes autonome System muss den Weg für alle einzelnen IP-Subnetze kennen, sondern jedes autonome System muss die Möglichkeit haben, bei Bedarf durch Abfragen festzustellen, zu welchem autonomen System die gewünschte IP-Adresse gehört, um dann die Datenübertragung an das entsprechende autonome System weiterzuleiten, das die Ziel-IP-Adresse beinhaltet.

# Maschen im Netz

## → Übersicht (1/2)

---

- In der Netzwerktechnik gibt es ein grundlegendes Gesetz:
  - Je schneller eine Datenübertragung von A nach B erfolgt, desto besser.
  - Jeder Netzbetreiber ist deshalb daran interessiert, so schnell wie möglich die Daten vom Absender zum Empfänger zu übertragen.
  - Damit macht er den Absender und Empfänger der Datenübertragung "glücklich", aber auch sich selbst, da er seine Netzwerkressourcen effizienter nutzen kann.
- Zwar besteht das Internet nicht nur aus einem Netz eines Netzbetreibers, sondern aus vielen Netzen vieler Netzbetreiber, dennoch gilt auch im Internet der Grundsatz "Schneller ist besser".

# Maschen im Netz

## → Übersicht (2/2)

---

- Zusätzlich gilt aber noch ein weiterer Grundsatz:
  - Je weniger Netzübergänge, desto besser.
  - Der Übertragung von Daten kostet Geld.
  - Diese Kosten entstehen durch die Kosten für die Infrastruktur, also für Standleitungen und Gerätschaften.
- Damit ein Netzbetreiber möglichst wenig Kosten mit der Übertragung von Daten hat, ist er daran interessiert,
  - den zu übertragenden Datenverkehr möglichst lange in seinem Netzwerk zu führen und
  - falls Absender und Empfänger in unterschiedlichen Netzwerken liegen, möglichst mit wenigen Netzübergängen zu übertragen.
- Dazu gibt es in dezentralen Netzwerken zwei verschiedene Möglichkeiten.

# Maschen im Netz

## → Private Peering (1/2)

- Bei einem Private Peering schaffen zwei Netzbetreiber einen Übergang zwischen ihren Netzwerken.
- Dazu legt meist ein Netzbetreiber eine Standleitung zu einem Gebäude des anderen Netzbetreibers.
- Seltener trifft man sich auf "neutralem" Boden, um dort einen Übergang zu schaffen.
- Der Übergang selbst wird mit einem Router geschaffen, an dessen beiden Seiten jeweils das Netzwerk der beiden Netzbetreiber angeschlossen ist.
- Über Routing-Einträge in beiden autonomen Systemen wird dann der Datenaustausch über diesen Netzübergang entsprechend gesteuert.
- Hat ein Netzbetreiber auf diese Weise mehrere Netzübergänge für sein Netzwerk aufgebaut, kann er über diese Netzübergänge sehr differenziert den Datenverkehr ins Internet steuern.

# Maschen im Netz

## → Private Peering (2/2)

- Sind beispielsweise die Übertragungskosten nach Nordamerika bei Netzanbieter A und B (mit denen er beide Private Peerings betreibt) unterschiedlich, kann er den Datenverkehr entsprechend über den günstigeren Netzanbieter abwickeln und den teureren Netzbetreiber als "Havarieanbindung" in seinem Routing berücksichtigen, also für den Fall, falls der Netzübergang zum günstigeren Netzanbieter nicht funktionieren sollte.
- Der Nachteil freilich liegt auch auf der Hand:
  - Für jedes Private Peering zu einem anderen Netzbetreiber ist ein weiterer Router und eine Standleitung notwendig.
  - Zudem können es sich gerade kleine Internet Provider oft finanziell nicht leisten, mehrere Private Peerings zu betreiben.

# Maschen im Netz

## → Internet Exchange Points (1/3)

- Der Gegensatz zum Private Peering sind die so genannten Internet Exchange Points, oft auch als Metropolitan Area Exchanges (MAE) oder Commercial Internet Exchanges (CIX) bezeichnet.
- Sie sind am ehesten vergleichbar mit Flughäfen:
  - In einem Flughafen haben Sie in der Abflughalle für gewöhnlich Schalter der verschiedensten Fluggesellschaften, eben denen, die von diesem Flughafen Flüge anbieten.
  - Der Flughafen selbst bietet (meist) keine Flüge selbst an, sondern vermietet diese Schalter und weitere Flughafenressourcen an die Fluggesellschaften, die dann ihre Dienstleistungen an ihre Kunden verkaufen.
- Internet Exchange Points sind neutrale Austauschpunkte, zu denen Internet Provider eigene Standleitungen legen können.
- Diese Anbindungen werden dann innerhalb des Exchange Points auf ein zentrales, lokales Netzwerk gelegt, an das alle anderen, teilnehmenden Internet Provider ebenfalls geschaltet sind.



# Maschen im Netz

## → Internet Exchange Points (2/3)

- Die Internet Provider können nun untereinander Abkommen zum Datenaustausch im Internet Exchange Point treffen und diese Abkommen durch gemeinsame Routing-Einträge von und zu ihren autonomen Systemen steuern.
- Sie können also den Datenverkehr zwischen ihren Netzwerken direkt im Internet Exchange Point miteinander austauschen und müssen diesen Datenverkehr nicht über teurere Dritte oder über andere, eventuell teurere Anbindungen austauschen.
- Die Idee, die hinter Internet Exchanges steckt, ist einfach und bestechend zugleich: Anstatt zu jedem anderen Provider eine separate Standleitung zu schalten, wird mit der Verbindung zu einem Internet Exchange Point die Möglichkeit geboten, mit der gleichen Standleitung gleichzeitig direkte Verbindungen zu vielen anderen Internet Providern zu schaffen.

# Maschen im Netz

## → Internet Exchange Points (3/3)

- Internet Exchange Points haben im Internet unterschiedliche Ausrichtungen bzw. "Größenklassen":
- In frühen Jahren des Internet waren Exchange Points noch eine nationale oder gar kontinentale Angelegenheit und die großen Backbone-Betreiber quasi unter sich.
- Durch die immer stärker werdende Verzweigung des Internet stiegen aber im Laufe der Zeit auch immer stärker die Datenmengen an, die möglichst effizient unter den immer mehr Netzbetreibern ausgetauscht werden mussten.
- Deswegen entwickeln sich Ländern mit hoher Internet-Nutzung neben den großen Exchange Points auch kleinere Austauschpunkte, die oft nur für regionale oder städtische Netzanbieter interessant sind, gerade aber dieser Gruppe die Möglichkeit bietet, über so einen Exchange Point mehrere Verbindungen zu anderen Netzbetreibern und letztendlich ins Internet zu realisieren und Redundanzen zu schaffen.

# Der Dienstleister macht das Netz

## → Übersicht (1/3)

---

- Da wir nun wissen, dass das Internet nicht ein homogenes Netzwerk ist und auch nicht von einem Betreiber allein "verwaltet" wird, ist auch der "wirtschaftliche" Aufbau klar:
- Die Kunden bezahlen für Ihren Internet-Zugang bei Ihren Internet Provider, der wiederum Geld in seine Infrastruktur und eine Internet-Anbindung investiert.
- Letztendlich sorgen also die Kunden mit Ihren Ausgaben für Ihren Internet-Zugang nicht nur dafür, dass es Ihrem Internet Provider gut geht, sondern dass das Internet überhaupt existiert.
- Grob unterscheiden lassen sich folgende Gruppen von Dienstleistern, die das Internet bilden:

# Der Dienstleister macht das Netz

## → Carrier (2/3)

- Netzbetreiber, also Dienstleister, die eine eigene Netzinfrastruktur innerhalb des Internet betreiben und Übertragungskapazitäten an Kunden verkaufen, werden in der Telekommunikation als Carrier bezeichnet (aus dem englischen Verb "to carry", was so viel bedeutet wie "tragen").
- Große Carrier beschränken sich meist auf dieses Kerngeschäft allein, betreiben also allein ihr Netzwerk und die Verbindungen in andere Netze und verkaufen Übertragungsleistungen für dieses Netzwerk an andere Unternehmen, die diese Leistungen dann an andere Endkunden verkaufen, gelegentlich auch direkt an Großkunden.
- Da Carrier auf diese Weise die Hauptlast des Datenverkehrs im Internet tragen, bilden ihre Netzwerke das Rückgrat, das so genannte Backbone, des Internet.
- Beispiele sind: T-Systems/T-Com

# Der Dienstleister macht das Netz

## → Internet Service Provider (3/3)

---

- Diese Gruppe der Dienstleister, kurz auch Internet Provider oder ISP genannt, sind diejenigen, die Internet-Dienstleistungen an Endkunden verkaufen.
- Das kann ein Full-Service sein, es gibt aber auch ISP, die sich beispielsweise nur auf Internet-Verbindungen (Access Provider) oder auf das Webhosting (Webhosting Provider) spezialisieren.
- ISP beschränken sich meist auf diese Arbeit an der "Kundenfront" und beziehen ihre eigene Internet-Anbindung über Carrier.
- Beispiele sind: T-Online, ...

# Der Dienstleister macht das Netz

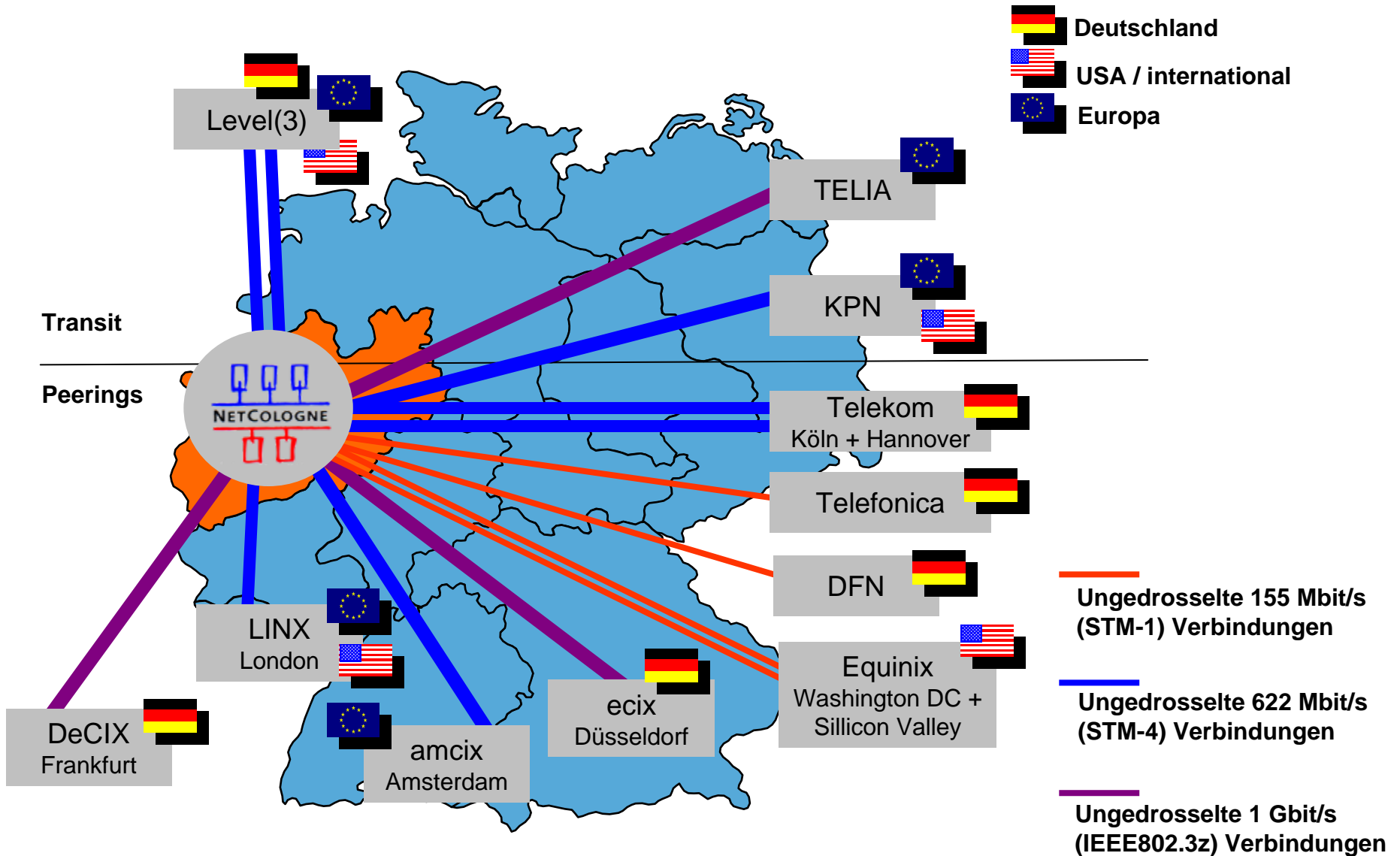
## → Institutionelle Netzbetreiber (4/4)

---

- Dazu gehören beispielsweise Forschungs- und Universitätsnetzwerke, die aus öffentlichen Mitteln gefördert und unterhalten werden.
- Diese Zugänge sind auch meist weitgehend nichtkommerziell und dürfen überhaupt nicht oder nur in sehr beschränktem Masse gewerblich genutzt werden.
- Beispiele sind: DFN, ...

# NetColone

## → Beispiel: Internet Anbindung



# Inhalt

---

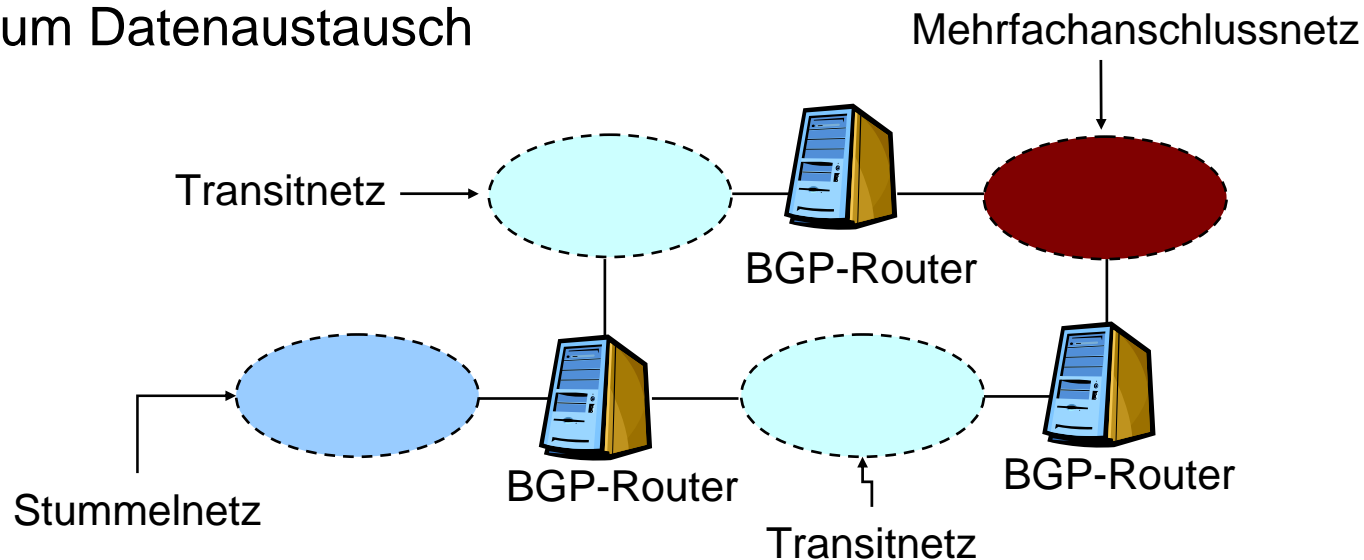
- Ziele, Einordnung und Übersicht
- Router
- Routing-Verfahren
- Routing-Protokolle
- Autonome Systeme / Internet
- **Ablauf**
- Zusammenfassung



# BGP (Border Gateway Protocol)

## → Grundlagen

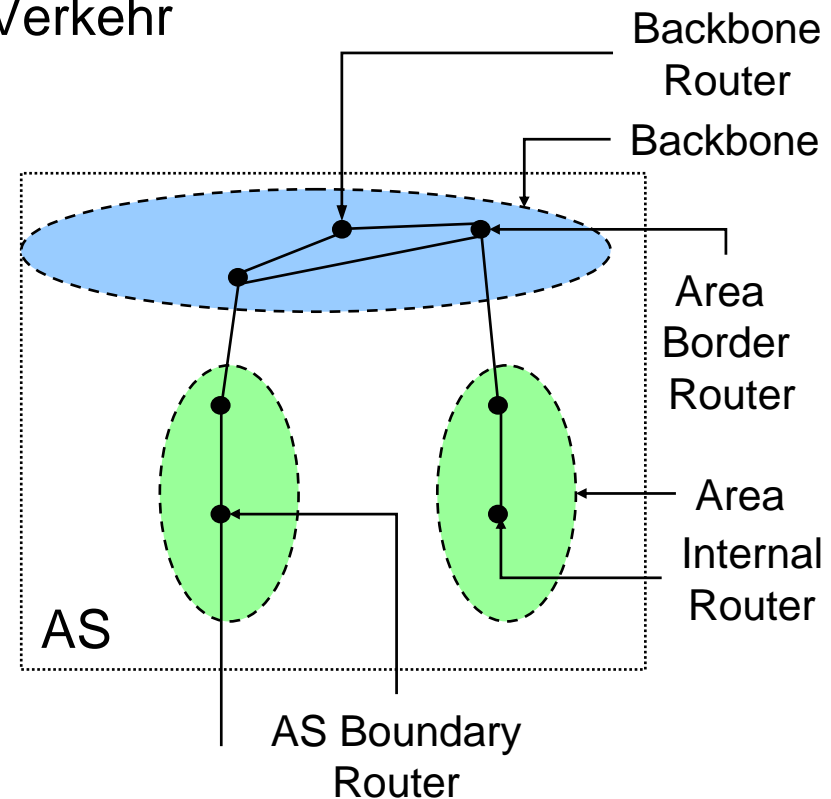
- BGP = Border Gateway Protocol
- offener Standard, entwickelt von der IETF - RFC 1771
- Distance Vector Routing
- Exakter Pfad zum Zielknoten wird gespeichert
- BGP-Router tauschen komplette Pfade aus
- Routing innerhalb der AS wird nicht betrachtet
- TCP zum Datenaustausch



# OSPF

## → Grundlage

- Verbessertes Link State Routing
- Basiert auf der Berechnung eines kürzesten Pfades von der Quelle zum Ziel
- Hierarchische Aufteilung des AS
- Intra-Area-, Inter-Area-, Inter-AS-Verkehr



# OSPF

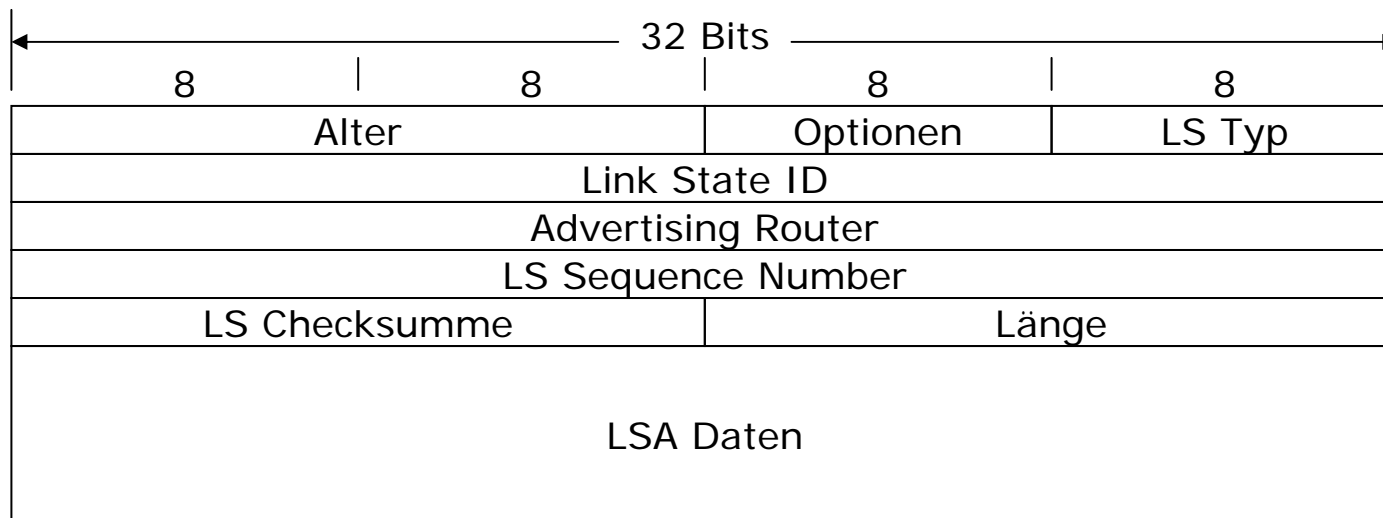
## → Überblick

---

- OSPF = Open Shortest Path First
  - benutzt Dijkstra's SPF Algorithmus
  - offener Standard, entwickelt von der IETF
  - aktuell: Version 2, RFC 2328
- Interior Gateway Protocol zum Routing innerhalb eines AS
- **Vorteile:**
  - Schneller Datenbankabgleich bei Topologie-Änderungen (adaptive)
  - Geringe Anfälligkeit gegenüber fehlerhaften Routing-Informationen
  - Unterstützt unterschiedlichste Metriken: z.B. Anzahl Hops, Verzögerung
  - Lastausgleich zum Zielknoten (Berücksichtigung verschiedener Wege)
  - Unterstützt große Netzwerke
  - Unterstützt verschiedene Wege

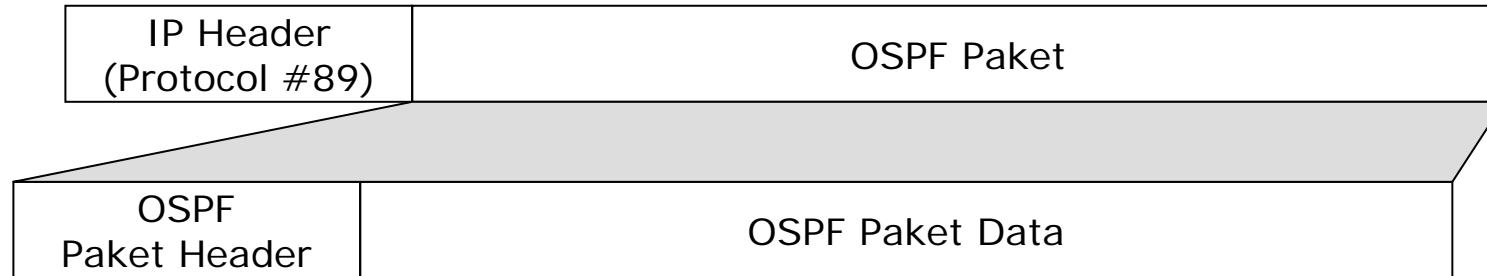
# Link State Advertisements (LSA)

- Router verschicken LSA für jeden ihrer Links
- beschreiben damit den für sie sichtbaren Teil der Topologie
- alle LSA's zusammen ergeben die Link State Datenbank



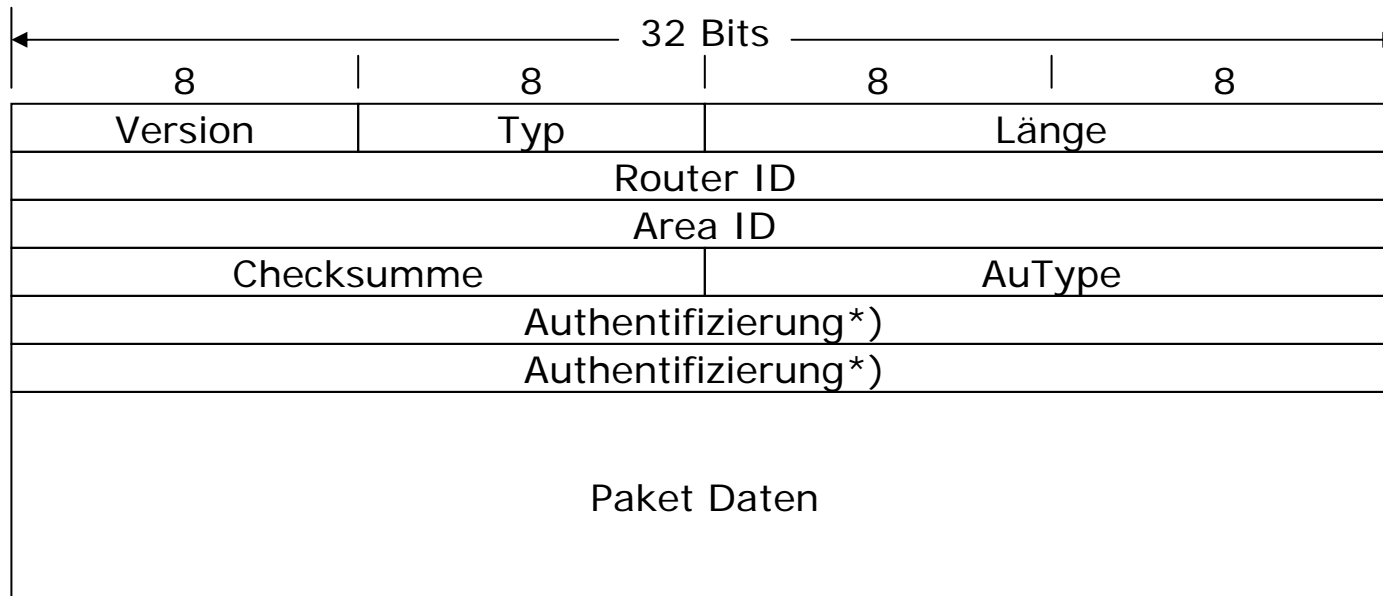
- verschiedene Arten von LSA's zur Implementierung der einzelnen Features

# OSPF Pakete

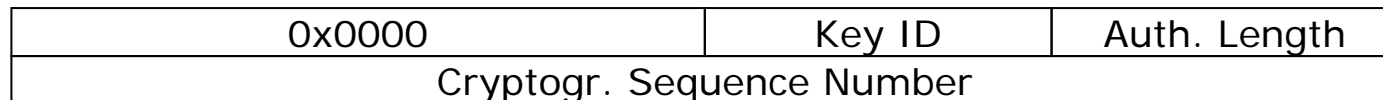


- 5 Pakettypen:
  - Hello
  - Database Description
  - Link State Request
  - Link State Update
  - Link State Acknowledgement
- Übertragung über IP, Protokoll #89 (Portnummer 89)
- Übertragung direkt an Nachbarn oder über Multicast Adressen
- OSPF Pakete werden nur zwischen Nachbarn im Netz ausgetauscht, niemals außerhalb ihres Ursprungs-Netzes weiter gerouted (TTL=1)

# OSPF Header



\*) falls AuType = 2:



- Die Authentisierung wird durch den AuType beschrieben.
- Einfache Authentikation, besteht aus Passwörtern (8 Zeichen)
- Es ist auch eine kryptographische Authentikation möglich.

# Identifizieren von Nachbarn

---

- Nachbarn sind die Router, mit denen ein Router direkt Informationen austauscht.
- periodisches Senden von Hello-Paketen über alle Interfaces
  - Hello Protokoll
  - Auch zum Erkennen fehlerhafter Links
- Sicherstellen mehrerer Voraussetzungen
  - bidirektionale Kommunikation
  - Übereinstimmung verschiedener Parameter
  - Erkennen und Aushandeln verschiedener Erweiterungen
  - in bestimmten Netzwerktypen zusätzliche Funktion

# Datenbank-Synchronisation

---

- zentraler Punkt, da alle Router identische Datenbanken (DB) haben müssen
- 2 Arten von Synchronisation:
  - Initiale Synchronisation, wenn das Nachbarschafts-Verhältnis aufgebaut wird
  - Kontinuierliche Synchronisation, um die Konsistenz der Datenbanken zu gewährleisten (flooding)



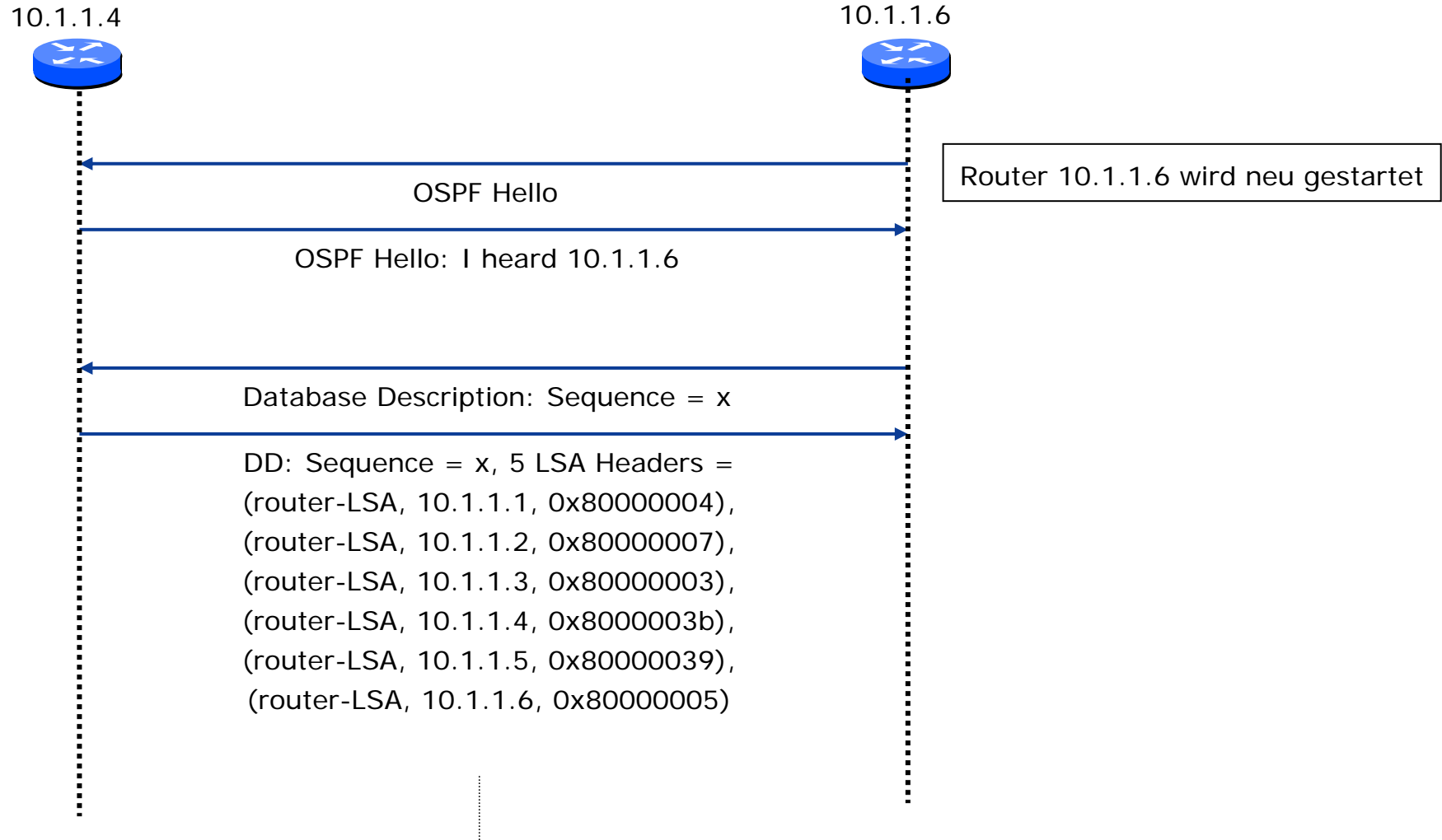
# Initiale Synchronisation

---

- Explizite Übertragung der gesamten Datenbank, wenn Nachbarschaftsverhältnis aufgebaut wird
- abhängig vom Netzwerk-Typ
- Synchronisation, wenn bi-direktionale Kommunikation besteht
- Senden aller LS Header aus der eigenen LS-DB an den Nachbarn
  - OSPF Database Description Pakete
- Flooding aller zukünftigen LSA's über die Verbindung
- immer nur ein DD Paket zur selben Zeit, Senden des nächsten Pakets erst nach Acknowledge durch Senden eines entsprechenden DD Pakets des Nachbarn
- bestimmen, welche LSA's in der eigenen DB fehlen und Anfordern dieser Pakete mit Link State Request Paketen
- Nachbar „flooded“ diese in Link State Update Paketen
- danach bereit für Daten-Verkehr über die Verbindung (fully adjacent)

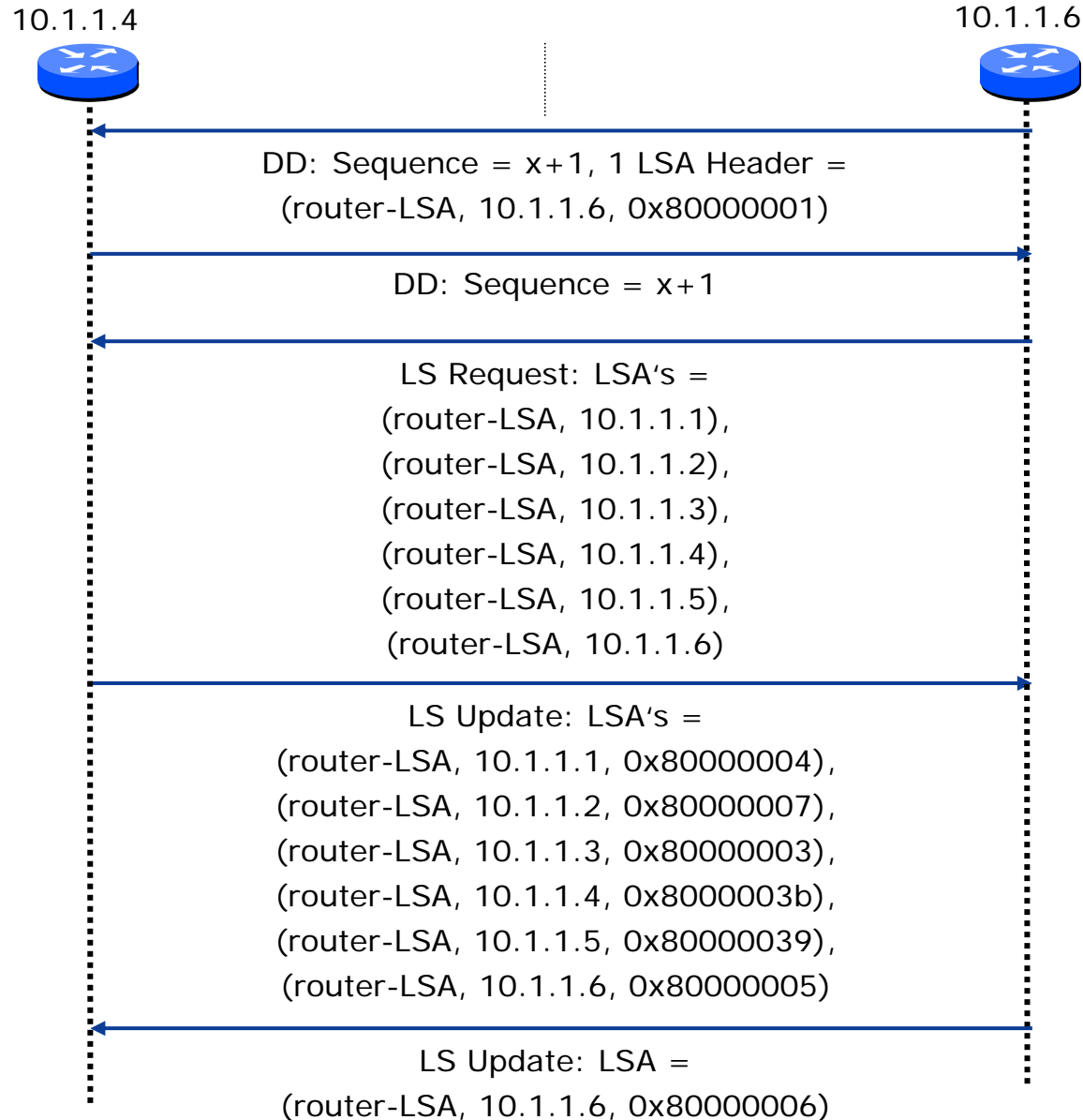
# Beispiel

## → Aus der Sicht von Knoten 10.1.13 (1/2)

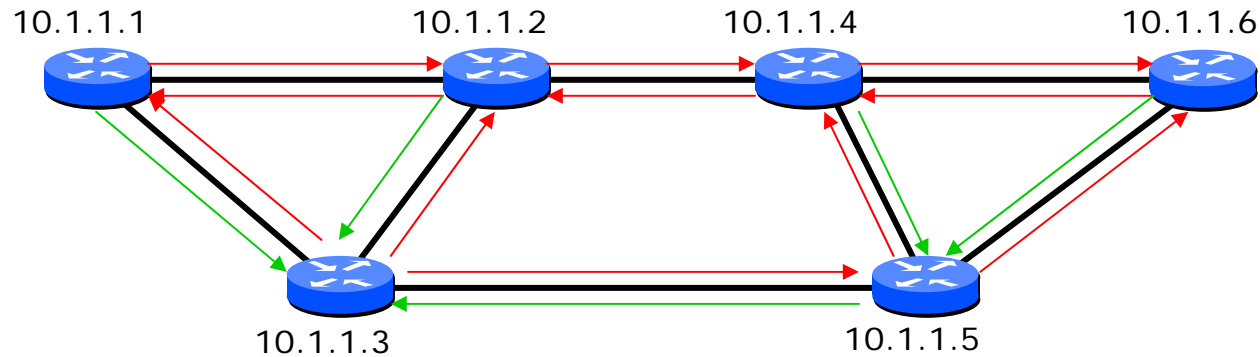


# Beispiel

## → Aus der Sicht von Knoten 10.1.13 (2/2)



# Reliable Flooding



- 10.1.1.3 schickt LS Update
- gleiche Kopie eines LSA ist implizites Ack
- generell: delayed Ack's
- alle LSA's müssen implizit oder explizit acknowledged werden

# Robustheit des Flooding

---

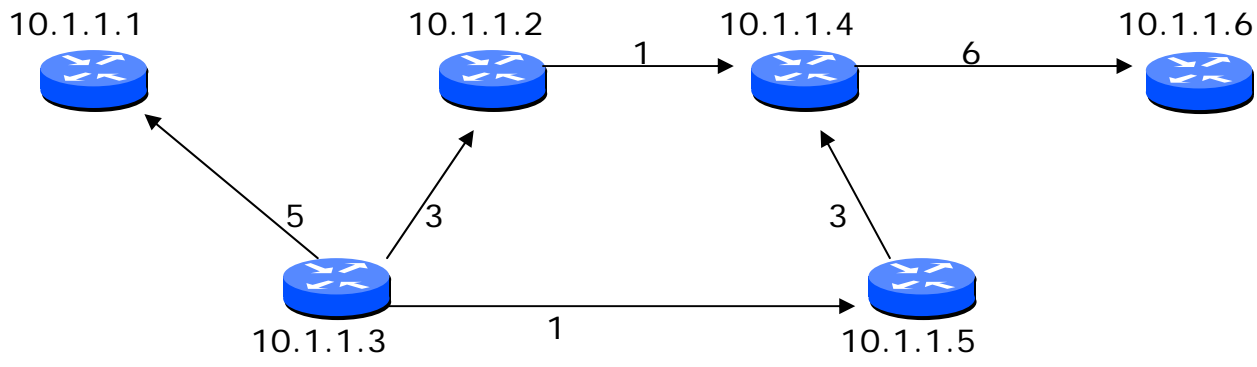
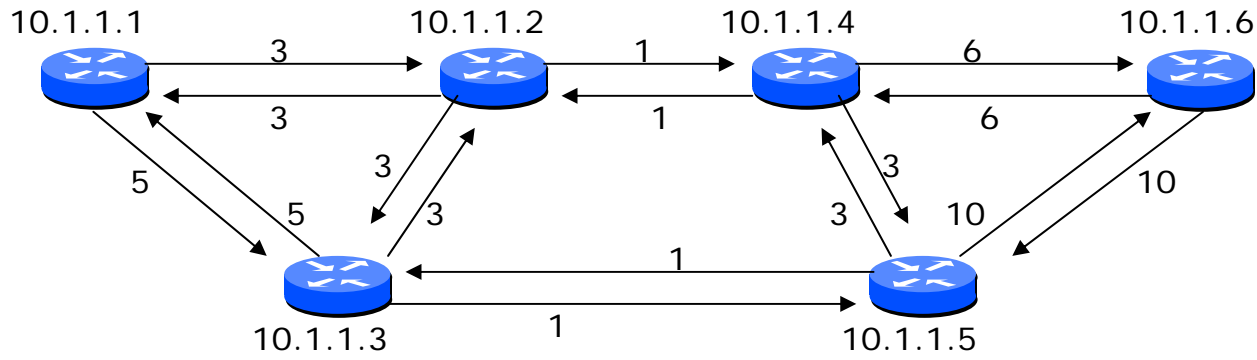
- Flooding über alle Links
  - einzelne gestörte Links stören nicht die Synchronisation
- LSA-Refreshes nach 30 Minuten, um Fehler in Router-Datenbanken zu korrigieren
- Übertragungsfehler werden durch Checksumme erkannt
  - solche LSA's nicht acknowledge
- LS Alterung zwingt zum regelmäßigen Austausch der LSA's in den Datenbanken
  - nach max. 1h ist auf jeden Fall aktuelle LSA in der Datenbank
- LSA's können höchstens alle 5s aktualisiert werden
- keine Annahme von LSA's, die weniger als 1s vorher bereits angenommen wurden

# Berechnung der Routing Tabelle

---

- Link State Datenbank ist ein gerichteter Graph mit Kosten für jeden Link
- Dijkstra's SPF Algorithmus zur Berechnung der kürzesten Pfade zu allen Zielen gleichzeitig:
  - addiere Router zum Shortest-Path-Tree
  - füge alle Nachbarn zur Candidate List hinzu
  - addiere den Router mit den kleinsten Kosten zum Tree
  - füge die Nachbarn dieses Routers an die Candidate List an
    - wenn noch nicht vorhanden
    - wenn Kosten niedriger als im bisherigen Listeneintrag
  - wiederhole, bis die Candidate List leer ist
- Laufzeit  $O(n \cdot \log(n))$

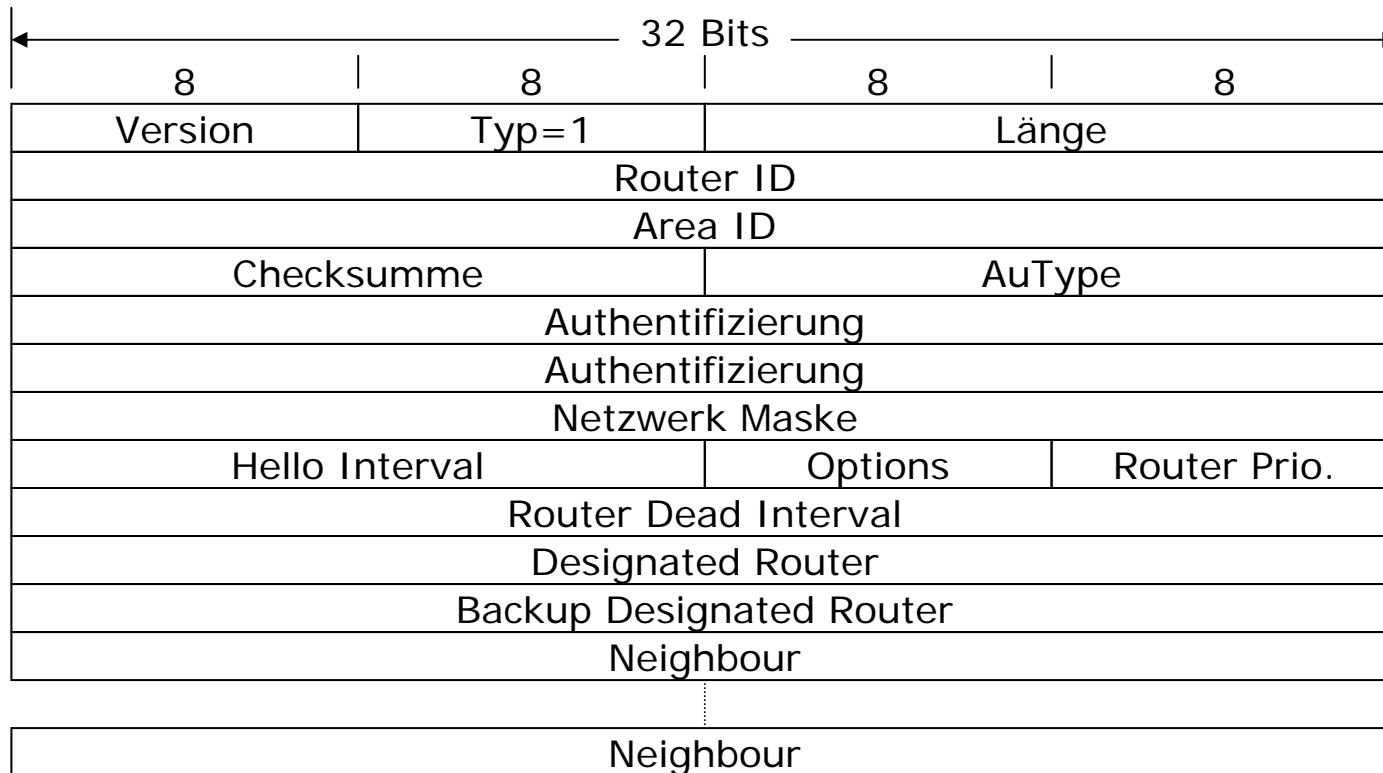
# Beispiel



- 10.1.1.1 (10, 10.1.1.2)
- 10.1.1.2 (3, 10.1.1.4)
- 10.1.1.4 (5, 10.1.1.6)
- 10.1.1.6 (11, 10.1.1.5)

# OSPF Pakettypen

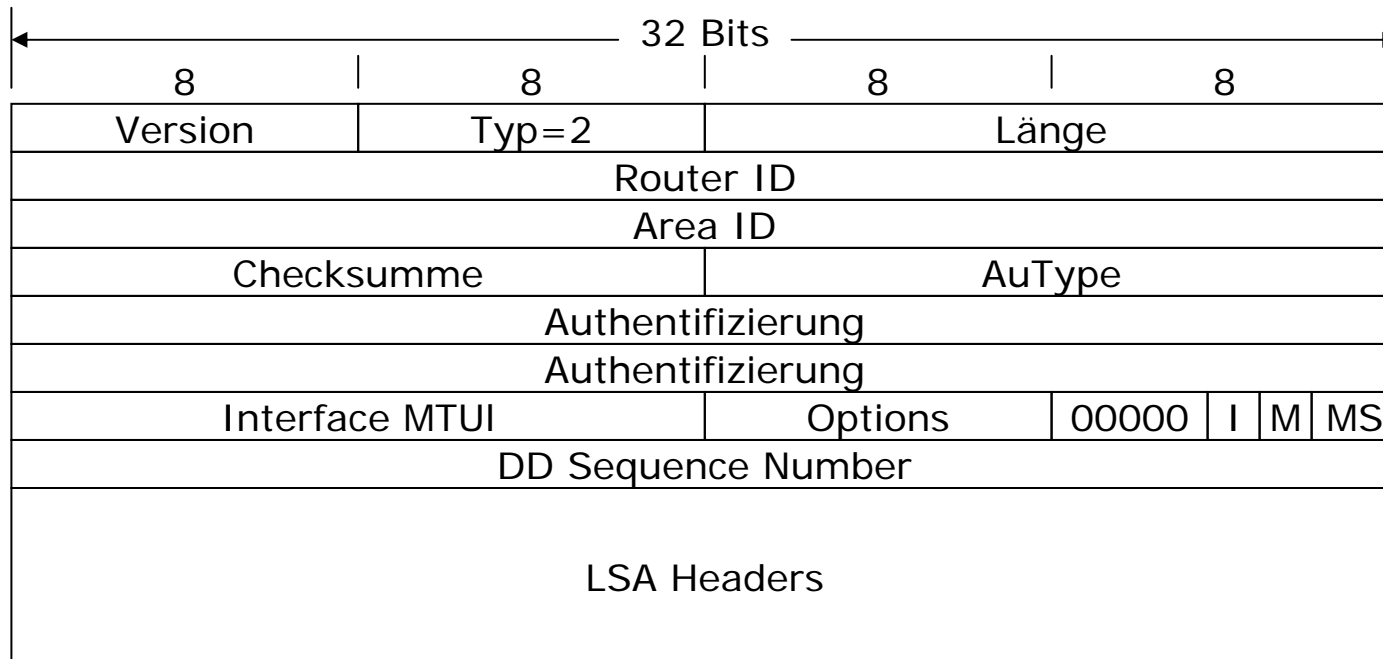
## → Hello Paket





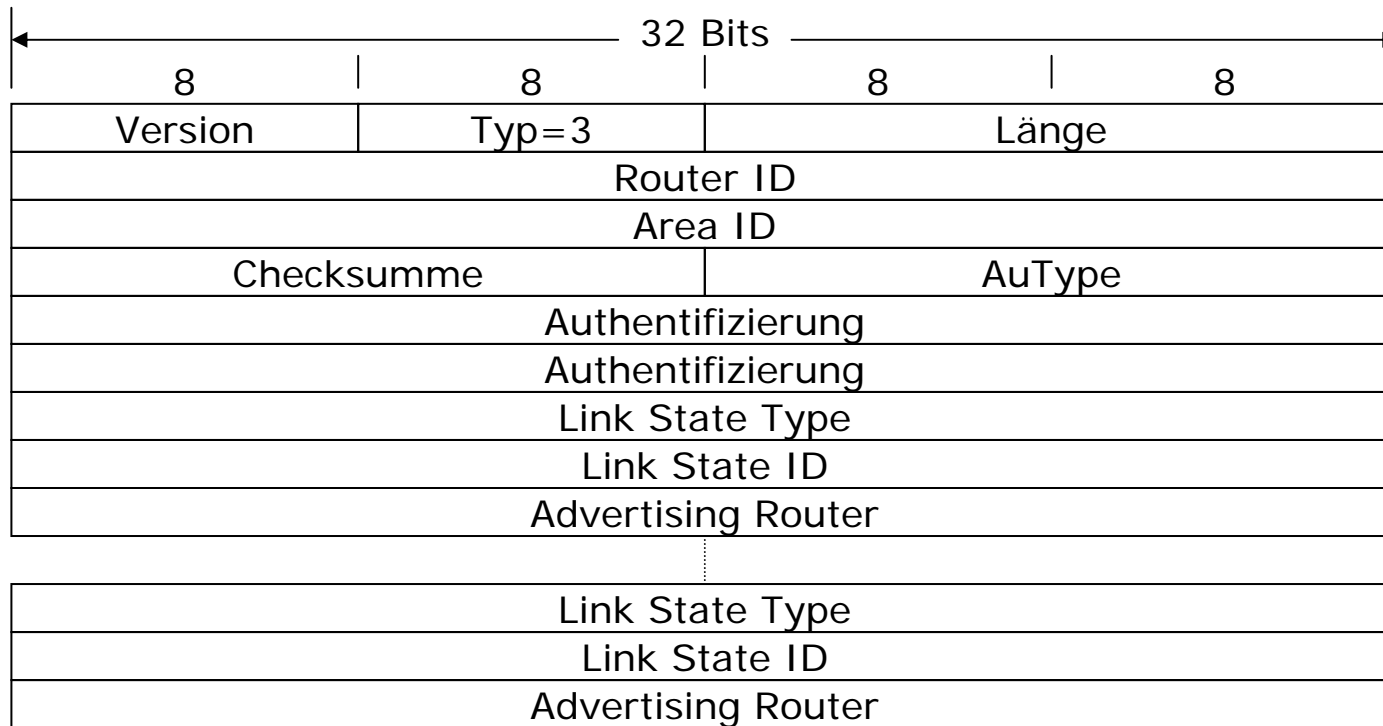
# OSPF Pakettypen

## → Database Description Paket



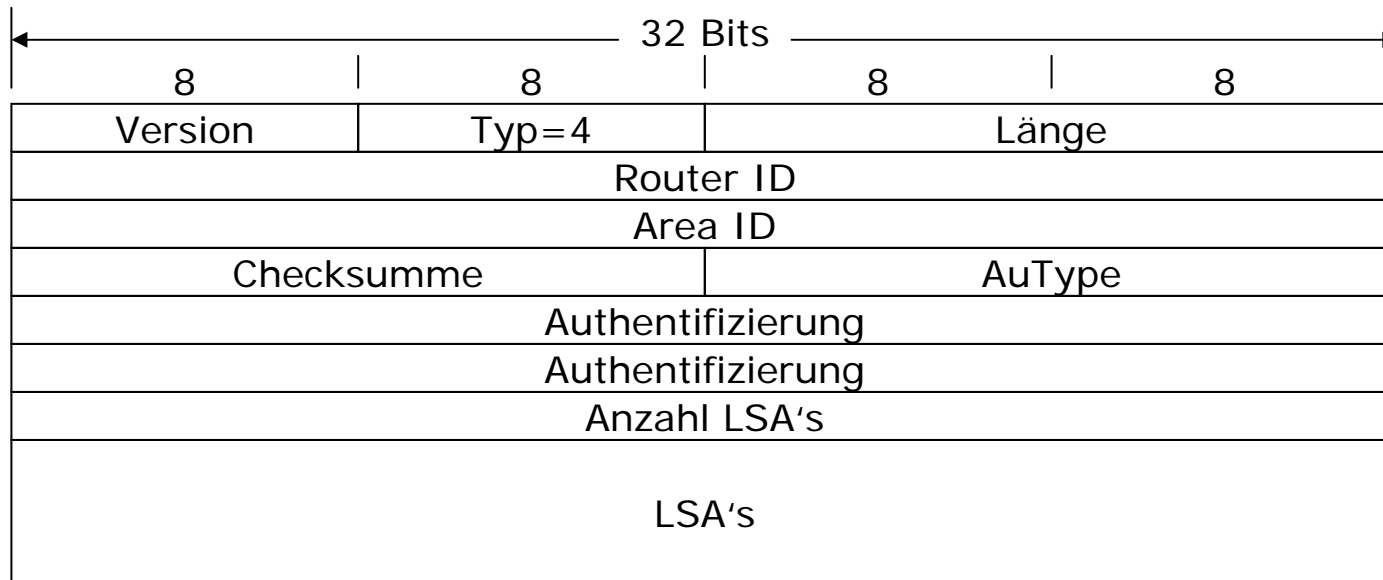
# OSPF Pakettypen

## → Link State Request



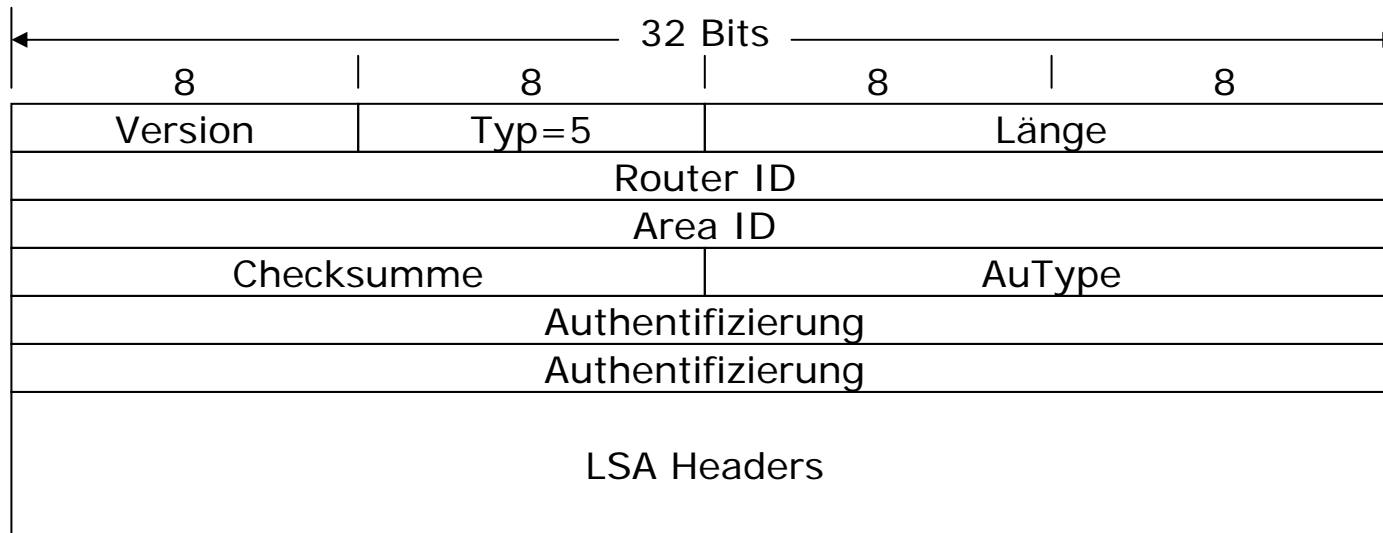
# OSPF Pakettypen

## → Link State Update



# OSPF Pakettypen

## → Link State Acknowledgement



# Inhalt

---

- Ziele, Einordnung und Übersicht
- Router
- Routing-Verfahren
- Routing-Protokolle
- **Zusammenfassung**

# Routing-Protokolle

## → Zusammenfassung

---

- Routing ist ein Teil des Konfigurationsmanagements im Bereich des Netzwerkmanagements.
- Statisches Routing ist ein mächtiges Werkzeug, um das Routing-Verhalten in einem Netzwerk präzise zu kontrollieren.
- Wenn jedoch regelmäßige Änderungen in der Topologie auftreten, kann der hierfür erforderliche Aufwand der manuellen Neukonfiguration ein statisches Routing undurchführbar machen.
- Das Border Gateway Protocol (BGP) ist das wichtigste Routing-Protokoll zur Kommunikation zwischen Routern von autonomen Systemen.
- Das Open Shortest Path First Protocol (OSPF) ist das wichtigste Routing-Protokoll zur Kommunikation innerhalb eines autonomen Systems und arbeitet mit dem Link State Routing.

# Routing-Protokolle

**Vielen Dank für Ihre Aufmerksamkeit**

**Fragen ?**

[norbert.pohlmann@informatik.fh-gelsenkirchen.de](mailto:norbert.pohlmann@informatik.fh-gelsenkirchen.de)

