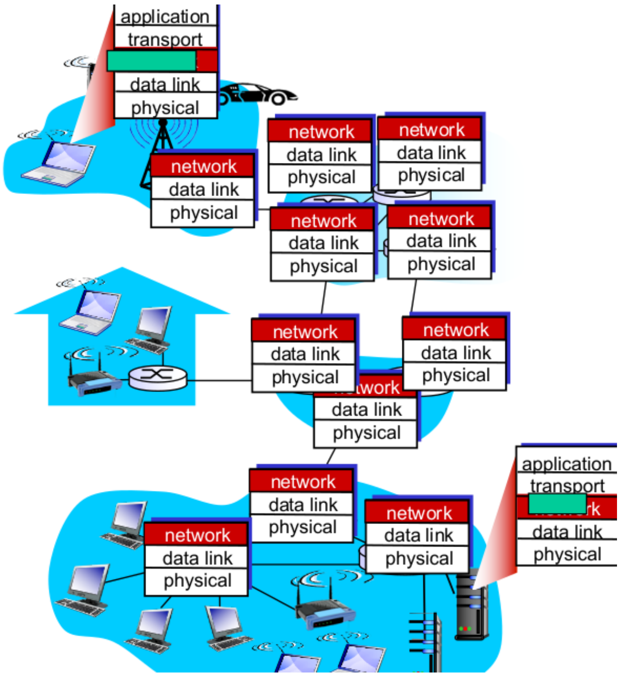
Network Layer

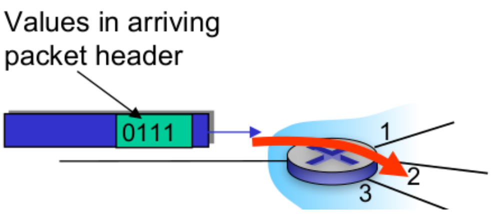
- Transport segment from sending to receiving host
- On sending side encapsulates segments into datagrams
- On receiving side, delivers segments to transport layer
- Network layer protocols in every host, router, interm. node
- Router examines header fields in all IP datagrams passing through it



Layer Functions:
* Forwarding – Wohin Packete geleitet werden: Jedes Gerät besitzt eine Forwarding Tabelle: Wohin wird das Packet weitergeleitet.
* Routing – Planung im Allgemeinen: Definiert das Forwarding auf dem Gerät

Data Plane

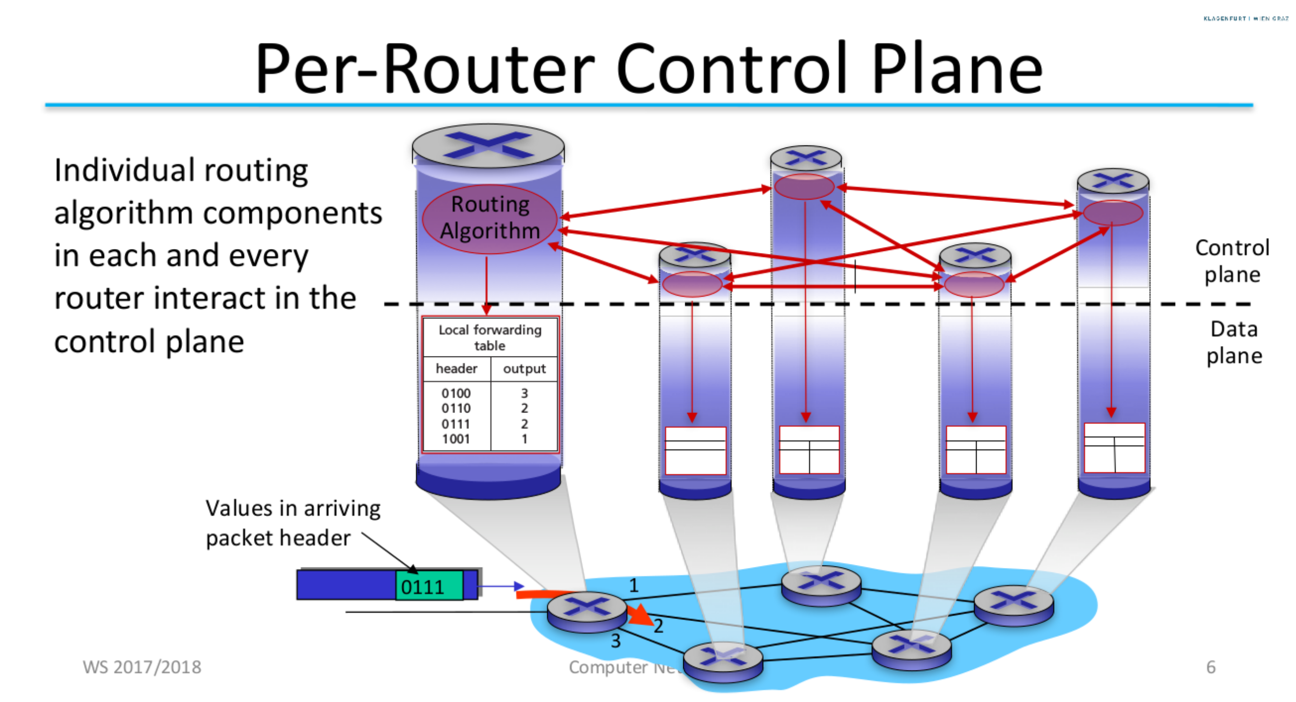
- Local, per-router function
- Determines how datagram arriving on router input port is forwarded to router output port
- Forwarding function



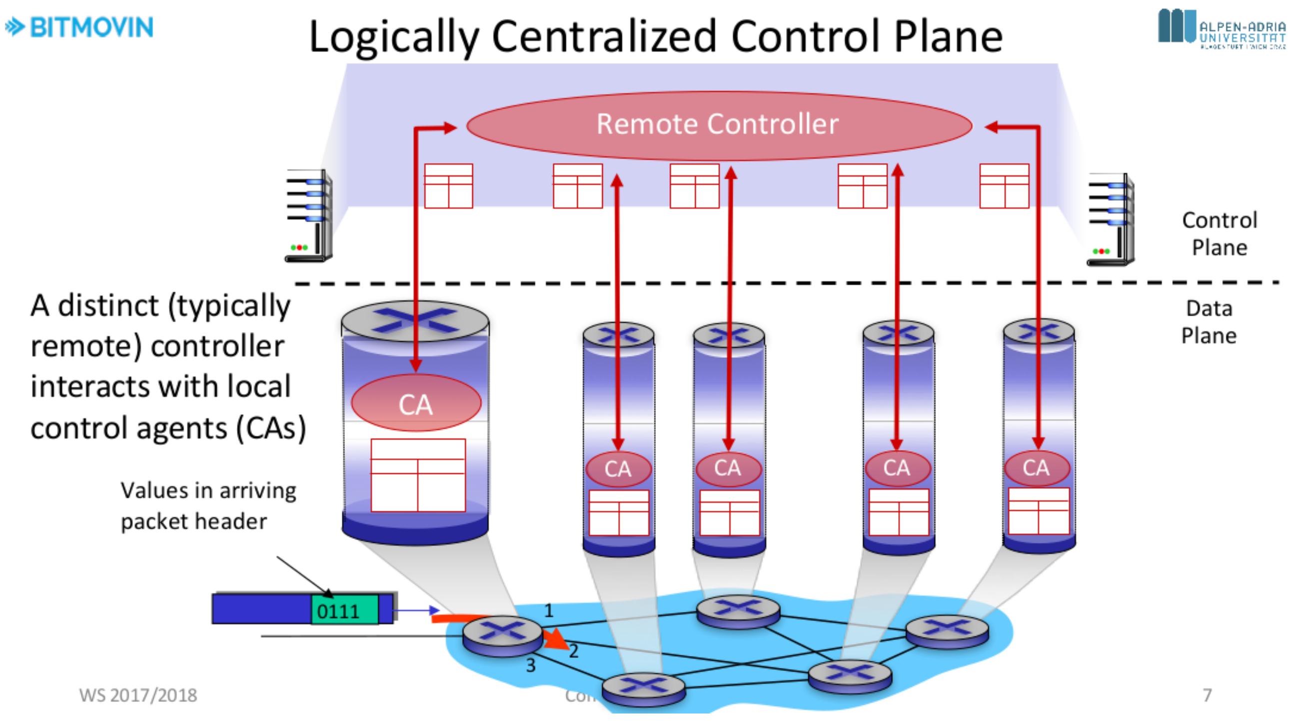
Control Plane

- Network-wide logic
- Determines how datagram is routed among routers along end-end path from source host to destination host
- Two control plane approaches:
  - Traditional routing algorithms: implemented in routers
  - Software-Defined Networking (SDN): implemented in (remote) servers

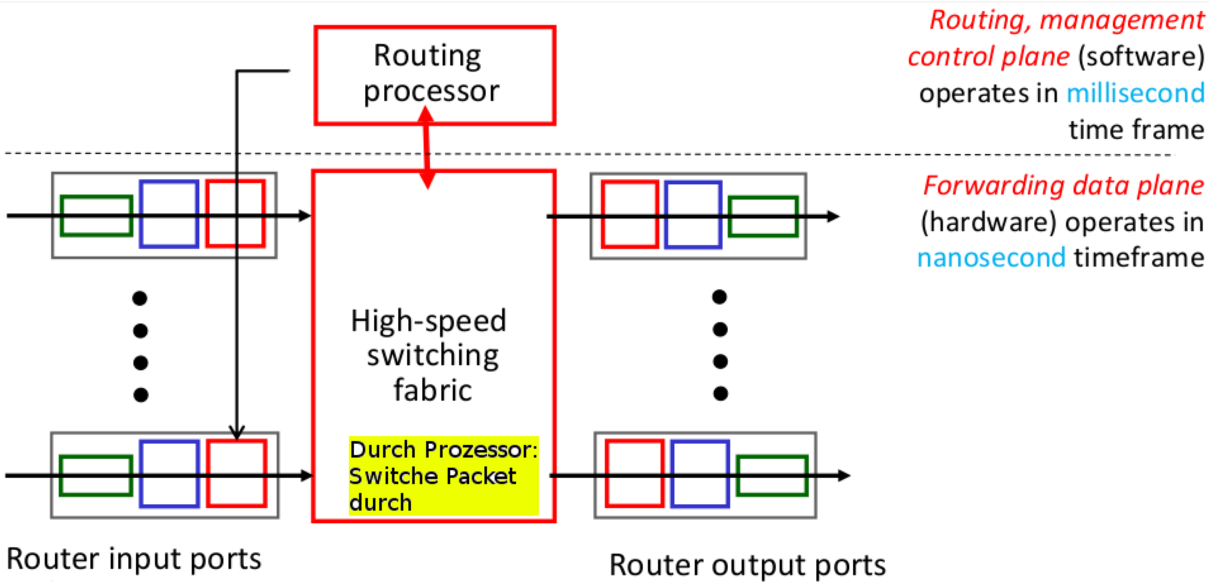
SDN: Auf Server implementiert, auf Router spezifiziert

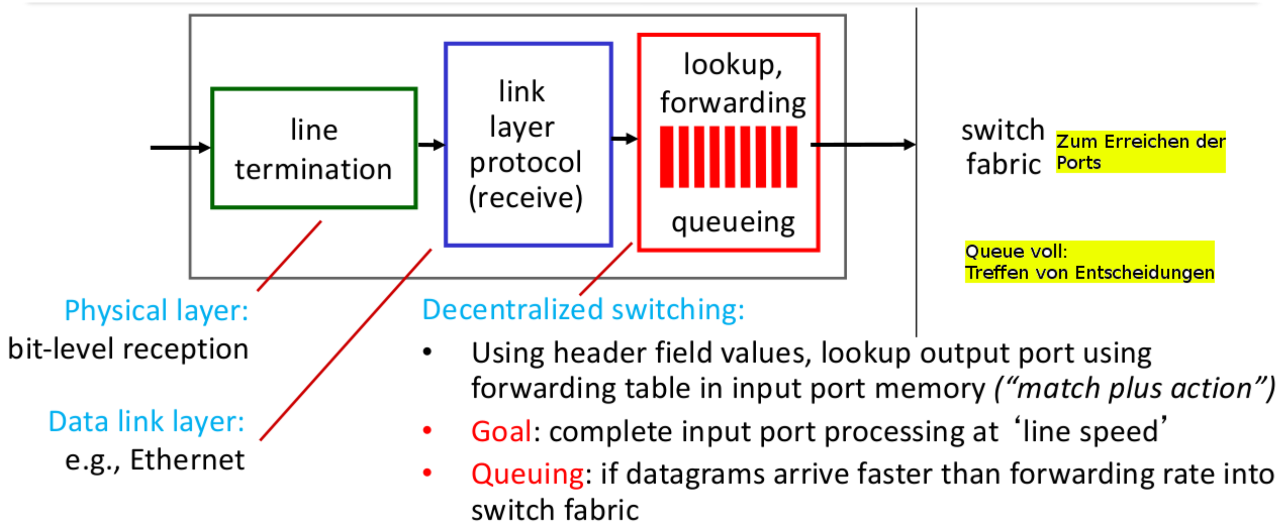


Weiterer Implementierungsalgorithmus für’s Routing:



Architektur eines Routers (X):





Lookup Tabellen decken nicht alles ab. Generell gelten Default-Gateways.

BITMOVIN

Longest Prefix Matching

ALPEN-ADRIA UNIVERSITÄT

When looking for forwarding table entry for given destination address, use *longest* address prefix that matches destination address.

Destination Address Range	Link interface
11001000 00010111 00010*** *********	0
11001000 00010111 00011000 *********	1
11001000 00010111 00011*** *********	2
otherwise	3

Examples:

Basierend auf Matches:

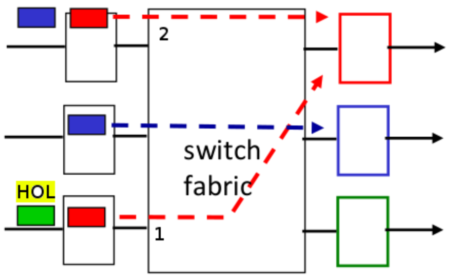
DA: 11001000 00010111 00010110 10100001 Which interface?

DA: 11001000 00010111 00011000 10101010 Which interface?

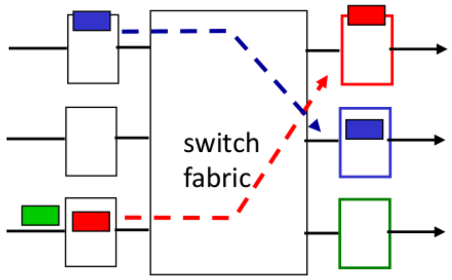
Interface: 1

Input Port Queuing

- Fabric slower than input ports combined -> queueing may occur at input queues
  - Queueing delay and loss due to input buffer overflow!
- Head-of-the-Line (HOL) blocking: queued datagram at front of queue prevents others in queue from moving forward



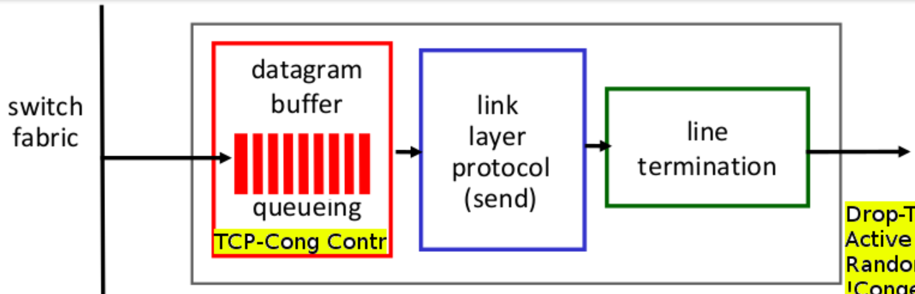
Output port contention:
only one red datagram can be transferred.
lower red packet is blocked



One packet time later:
green packet experiences
HOL blocking

Output Ports

This slide in HUGELY important!



Drop-Tail:
Active Queue Management
Random Early Det.
!Congestion!

- Buffering required when datagrams arrive from fabric faster than the transmission rate

Datagram (packets) can be lost due to congestion, lack of buffers

FCFS/FIFO/RR,Priority;
Weighted Fair Queueing

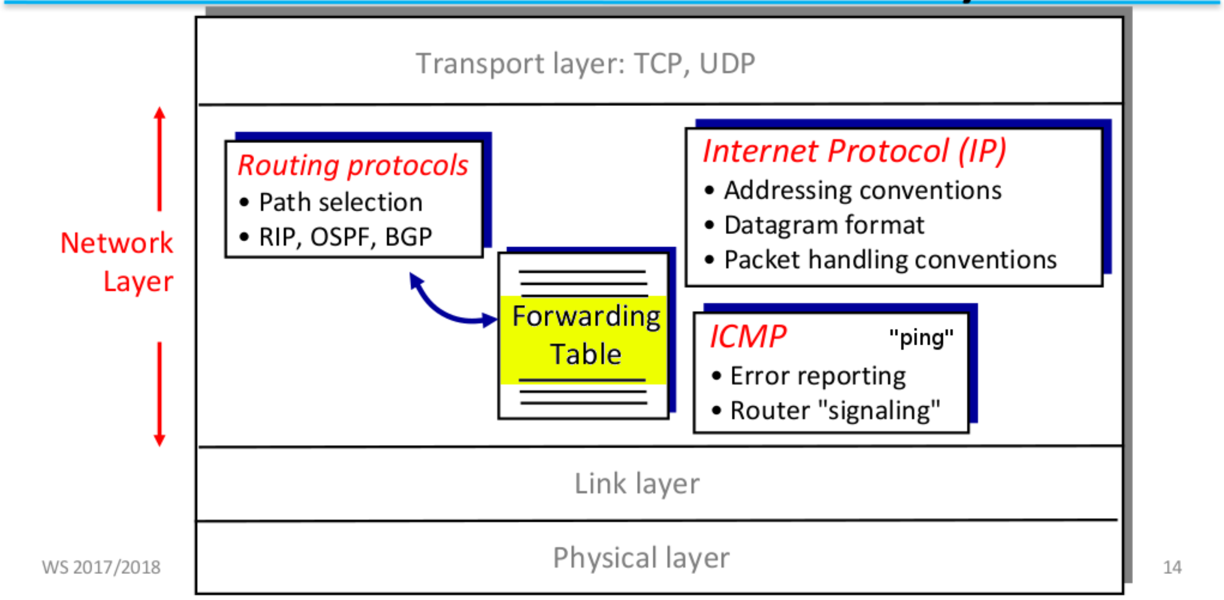
- Scheduling discipline chooses among queued datagrams for transmission

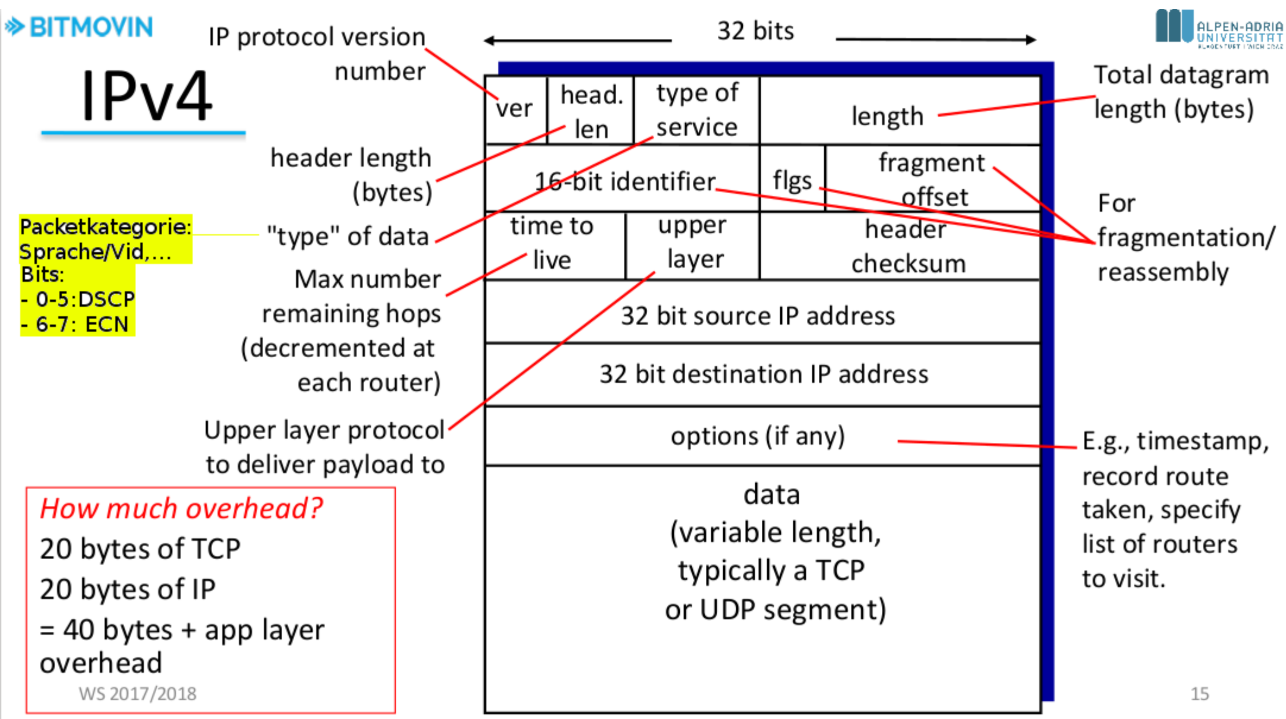
Priority scheduling – who gets best performance, network neutrality

- Net Neutrality: Alle Packete werden alle werden normal weitergeleitet, ohne dass andere Packete bevorteiligt werden – Via Klassifizierungen werden Prioritäten gesetzt.- Wichtigkeit: Wann welches Packet weitergeleitet wird.

* Congestion Control: Datenbufferladungen bis hin zu Buffer bloat

The Internet Network Layer

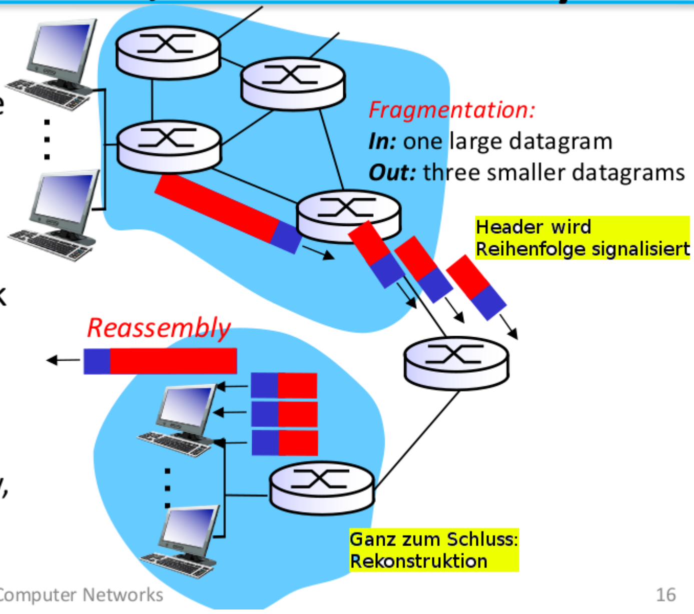




- Time to live: Maximallänge, bis das Packet verworfen wird um Netzwerk zu entlasten
- Upper Layer Protocol (Tcp = 6, UDP = 17)
- Header Checksum: Every router computes checksum

IP Fragmentation, Reassembly

- Network links have MTU (max. transfer unit) – largest possible link-level frame
  - Different link types, different MTUs
- Large IP datagram divided ("fragmented") within network
  - One datagram becomes several datagrams
  - "Reassembled" only at final destination
  - IP header bits used to identify, order related fragments



IP Fragmentation, Reassembly

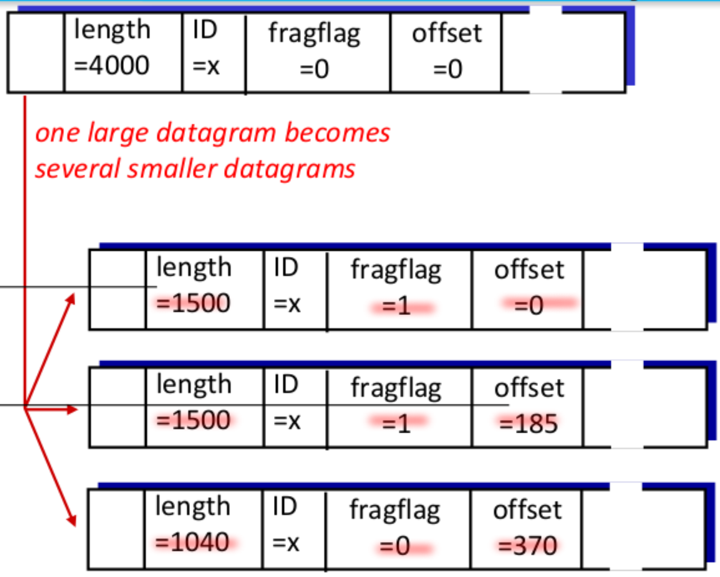
Example:

4000 byte datagram

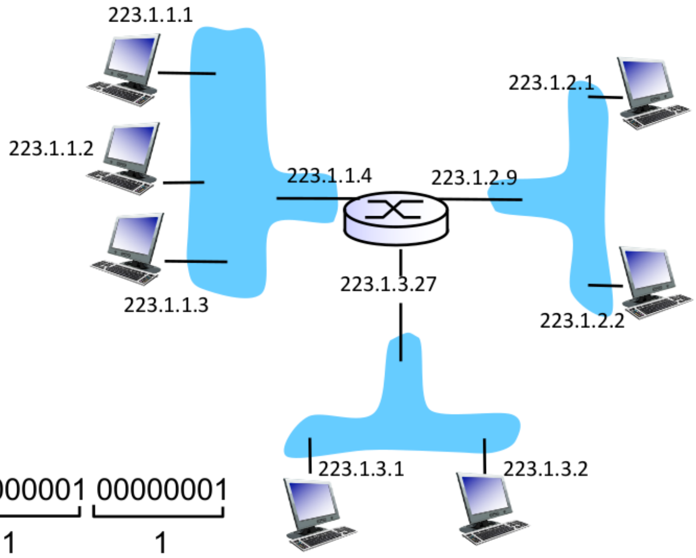
MTU = 1500 bytes

1480 bytes in data field

offset = 1480/8



- IP address: 32-bit identifier for host, router interface
- Interface: connection between host/router and physical link
  - Router's typically have multiple interfaces
  - Host typically has one or two interfaces (e.g., wired Ethernet, wireless 802.11)
- IP addresses associated with each interface



$$223.1.1.1 = \underbrace{11011111}_{223}\ \underbrace{00000001}_{1}\ \underbrace{00000001}_{1}\ \underbrace{00000001}_{1}$$

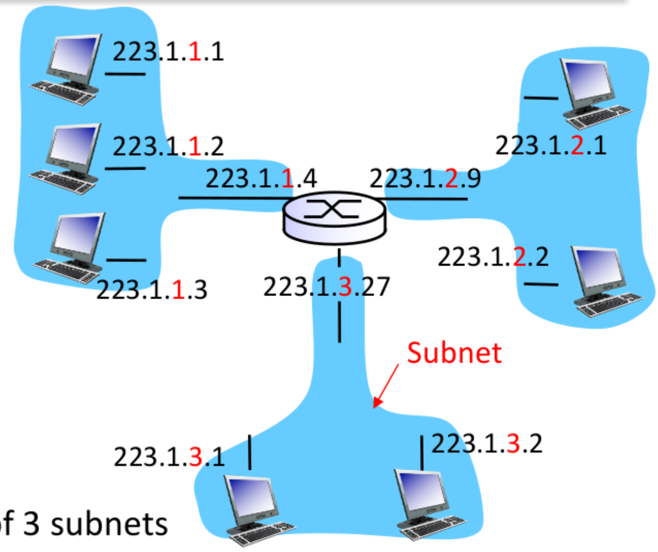
? 1040

- IP-Interfaces nicht unbedingt unikat

- Verbunden via Ethernetswitch oder WiFi- Konstruktion:

Subnets

- IP address
  - Subnet part – high order bits
  - Host part – low order bits
- What's a subnet?
  - Device interfaces with same subnet part of IP address
  - Can physically reach each other without intervening router



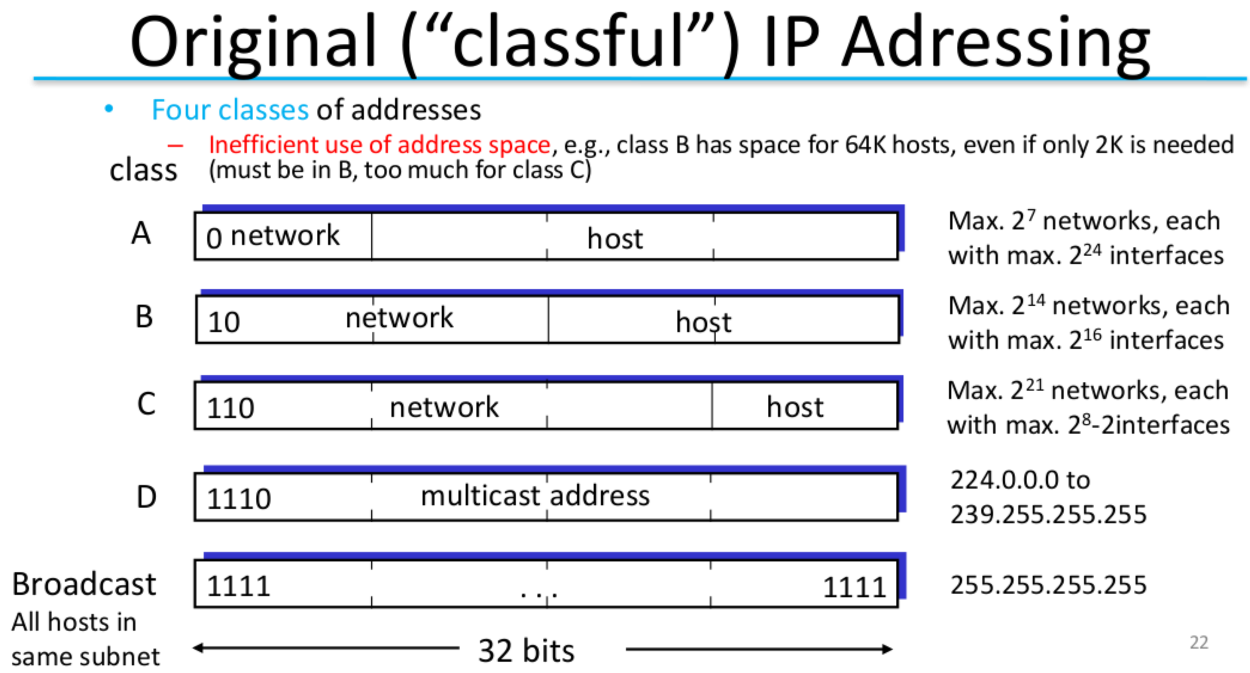
Network consisting of 3 subnets

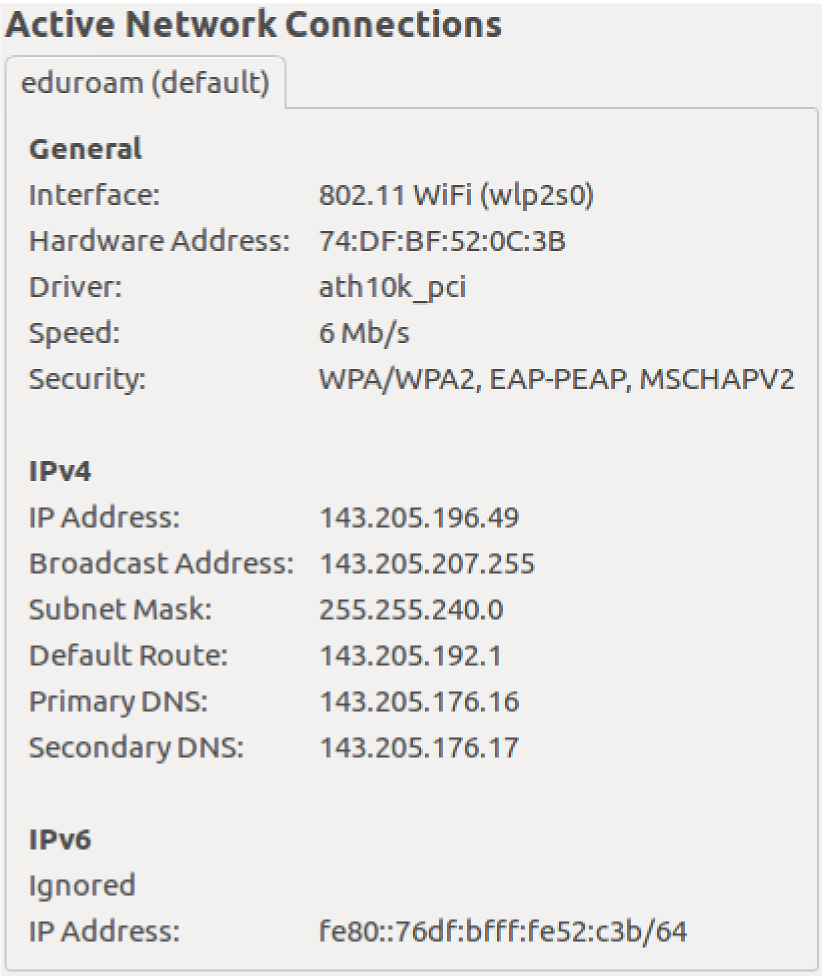
- 3 Subnets: xxx.x.1, 2, 3

223.1.1.1	=	11011111	00000001	00000001	00000001
		223	1	1	1

→ Erste 24 Bit definieren (Sub)-Net.

- Subnet-Mask: /24: Wieviele mögliche Hostadressen in einem Netzwerk verfügbar sind. Immer 2 Abziehen wegen 2 Default-IPs.
- 2048 mögliche IP-Adressen im AAU-Eduroam. Errechenbar aus Subnet-Mask <=> /21





→ Klasse B Netzwerk. Via Vergleich von IP-Addressen.
Berechnen:
Subnet mask →
255.255.248.0 in Binär: 11111111.1111111.**11111**000.00000000
Subnet Part: 11111 = 21
Host Part = 000.00000000 = 11
2^11 = 2048 mögliche IP-Addressen im /21 Subnet

→ 143.205.187.0 / 255
Späteres Cambridge Beispiel:
32-21 = 11 => 2^11 = 2048 IP Adressen; 2048 / 256 = 8 ↔ 0..7
x.24.0 und x.24.7

Pro Netzwerkbereich je 2 Adressen weniger: Broadcast & x

IP Addressing: CIDR

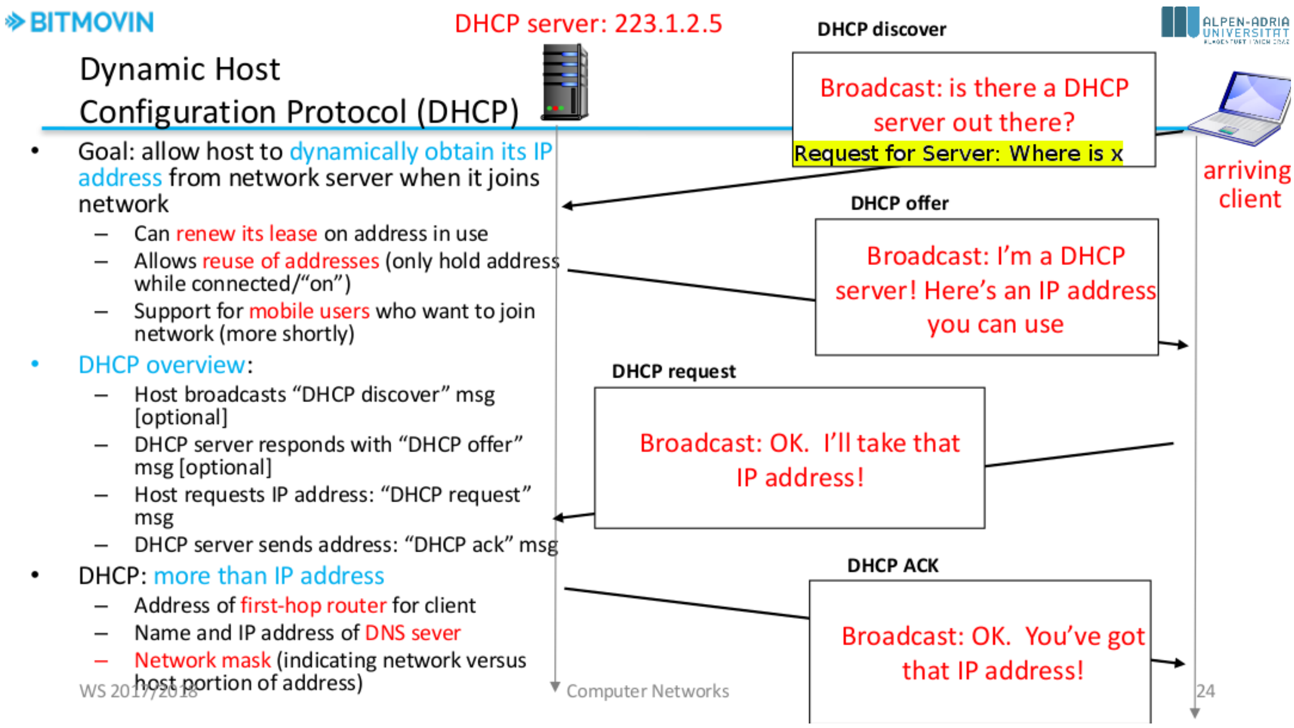
CIDR: Classless InterDomain Routing

- Subnet portion of address of arbitrary length
- Address format: a.b.c.d/x, where x is # bits in subnet portion of address

200.23.16.0/23 — subnet part: 11001000 00010111 0001000 | host part: 0 00000000

University	First address	Last address	How many	Written as
Cambridge	194.24.0.0	194.24.7.255	2048	194.24.0.0/21
Edinburgh	194.24.8.0	194.24.11.255	1024	194.24.8.0/22
(Available)	194.24.12.0	194.24.15.255	1024	194.24.12/22
Oxford	194.24.16.0	194.24.31.255	4096	194.24.16.0/20

Dynamic Host Configuration Protocol – DHCP:
- In IPV4: Adresse, SubnetMask, DefaultRouter.
- Ping www ↔ Löst nach zu suchenden Domains an, welche man im DNS angibt.



Sample as above:
DHCP-Discover
Src: 0.0.0.0., 68
dest: 255.255.255.255, 67 ↔ Broadcast: JEDER Rechner im Netzwerk bekommen diese Anfrage
yiaddr: 0.0.0.0
transaction ID: 654

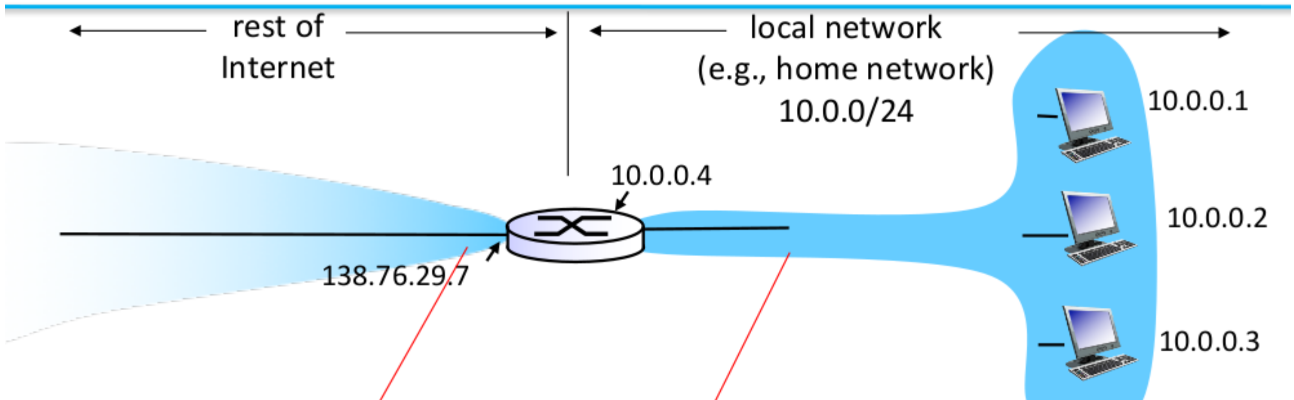
DCHP-Offer:
src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddrr: 223.1.2.4
transaction ID: 654
lifetime: 3600 sec

DHCP-Request
src: 0.0.0.0, 68
dest: Broadcast
yiaddrr as above
Transaction id: 655
lifetime 3600s

DHCP-ACK
src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddrr: 223.1.2.4
ID: 655, Lifetime 360s

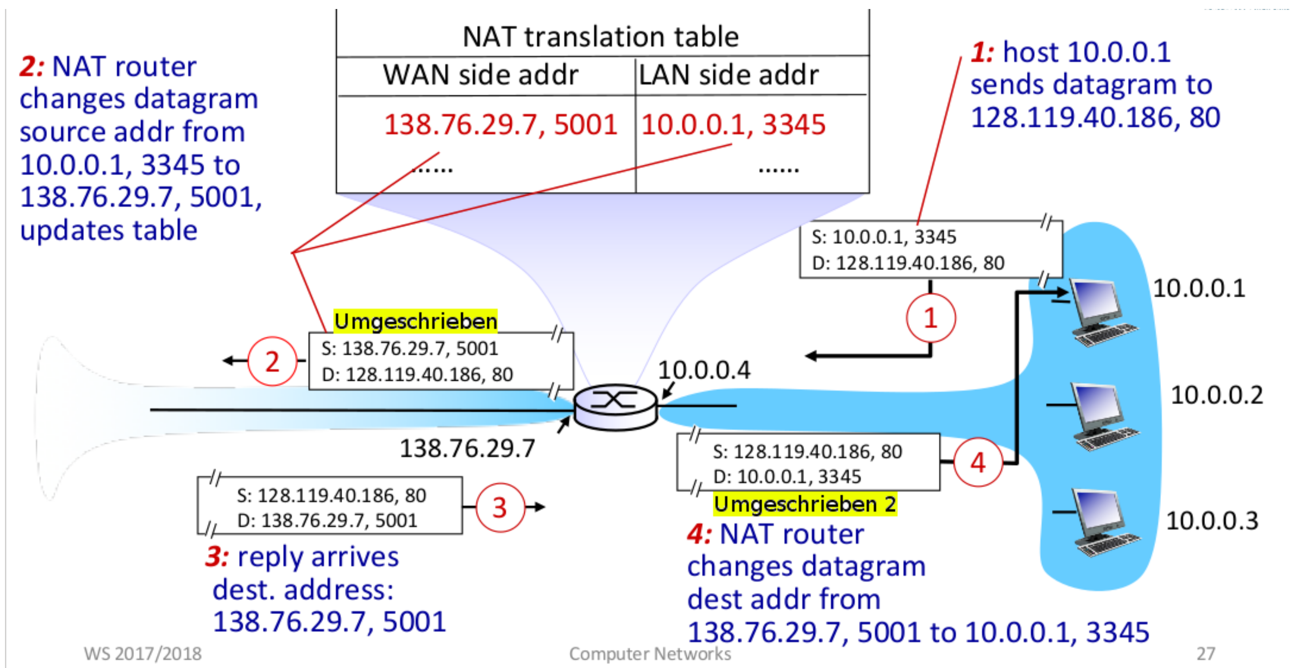
- ICANN verwaltet DNS-Adressen

NAT: Network Address Translation



all datagrams *leaving* local network have *same* single source NAT IP address: 138.76.29.7, different source port numbers

datagrams with source or destination in this network have 10.0.0/24 address for source, destination (as usual)



- 16-bit port-number field
  - 60,000 simultaneous connections with a single LAN-side address!
- NAT is controversial
  - Routers should only process up to layer 3
  - Address shortage should be solved by IPv6
  - Violates end-to-end argument
    - NAT possibility must be taken into account by app designers, e.g., P2P applications
- NAT traversal
  - What if client wants to connect to server behind NAT

Direkter Gerätzugriff ist jedoch möglich.

IPv6: Motivation

- Initial motivation IPv4: 143.205.122.10
  - 32-bit address space soon to be completely allocated
- Additional motivation
  - Header format helps speed processing/forwarding
  - Header changes to facilitate Quality of Service (QoS)

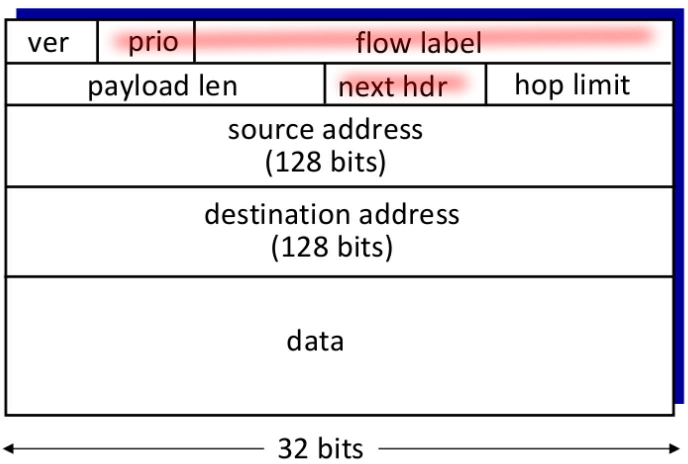
- IPv6 datagram format
  - 128-bit address space
  - Fixed-length 40 byte header
  - No fragmentation allowed

IPv6:2001:0db8:85a3:08d3:1319:8a2e:0370:7344

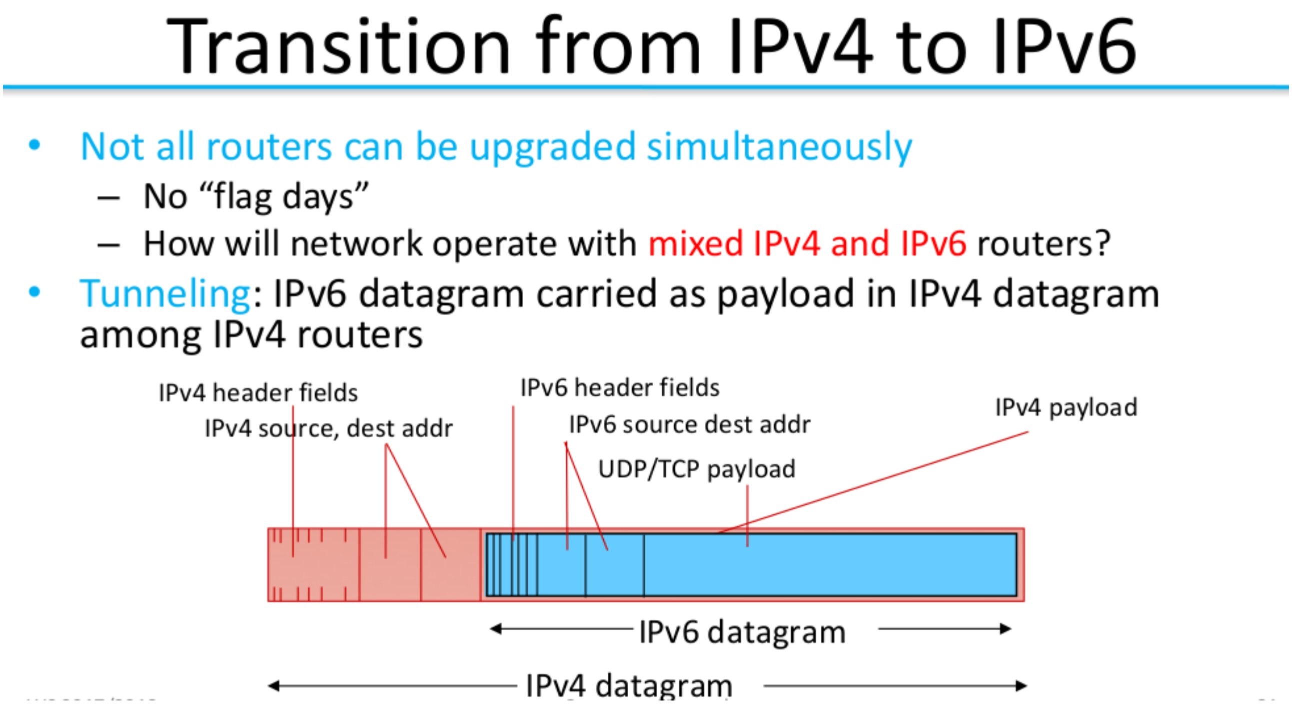
“IPv5 failed.”, existiert aber.

IPv6 datagram format

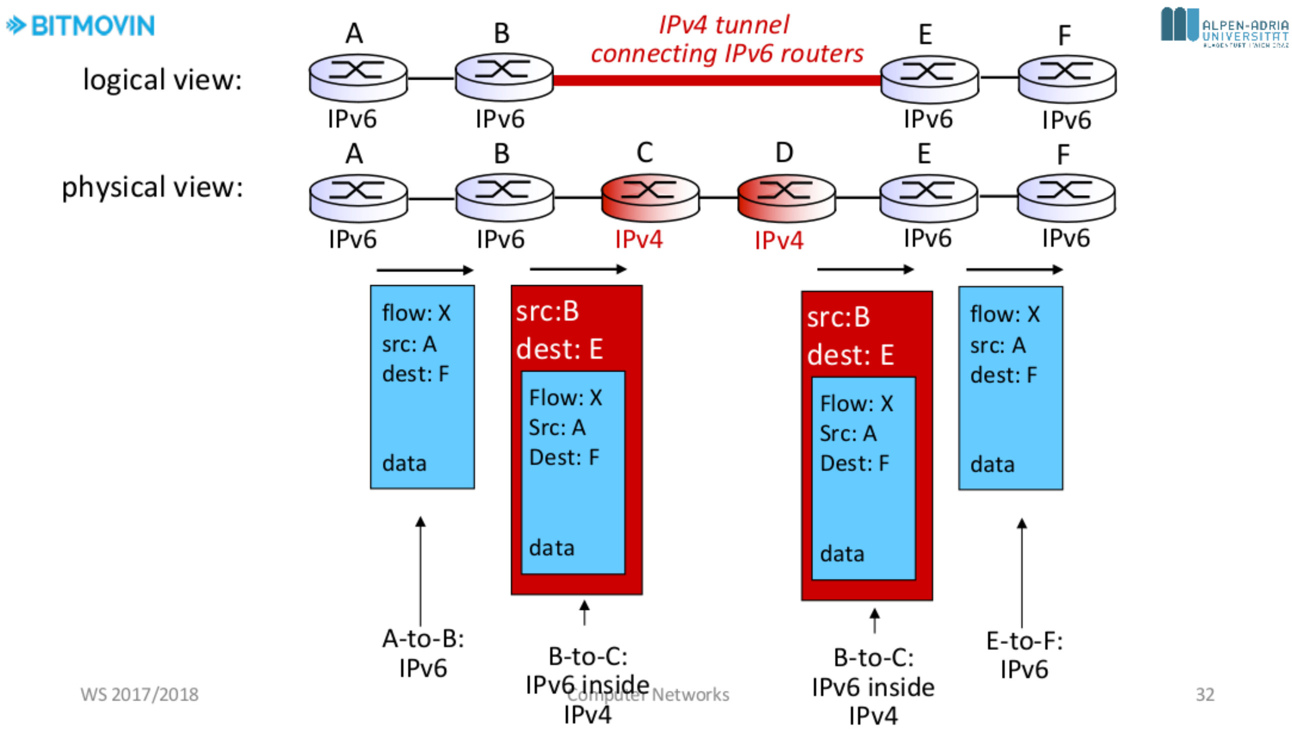
- New header fields
  - Priority: identify priority among datagrams in flow
  - Flow label: identify datagrams in same “flow” (concept of “flow” not well defined)
  - Next header: identify upper layer protocol for data
- Other changes
  - Checksum: removed entirely to reduce processing time at each hop
  - Options: allowed, but outside of header, indicated by “Next Header” field
  - IPv6 stateless auto-configuration vs. stateful DHCPv6
    - IPv6: Neighbor Discovery Protocol (NDP, ND)
    - DHCPv6: IPv6 equivalent to DHCP for IPv4



- Checksum: Entfernt, um Verarbeitungssschritt zu beschleunigen. <=> Beschädigte Packete werden dennoch komplett überwiesen wo dann Datacorruption spez. Wird



Nach wie vor V4 und V6. Radikaler Wechsel ist unmöglich ↔ Tunneling ↔ V6 wird in V4 encapsulated.



→ Es gehen keine Informationen verloren
Tunnelling existiert auch bei VPN ↔ Verschlüsselt Coms an außerhalb des Netzwerkes.
Payload Verschlüsseln ↔ Übertragung

=> Router können keine Packet-Inspection (PI) ausführen.