



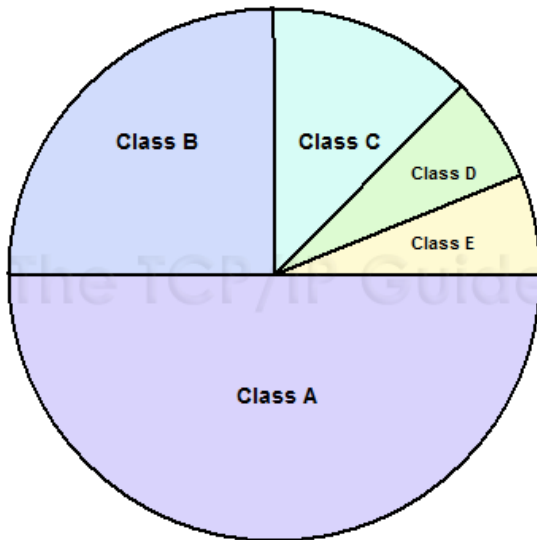
# IP Adressklassen

# IP Adressklassen

Es gibt fünf Klassen von Adressen, die sich darin unterscheiden, wie viele Hosts adressiert werden können

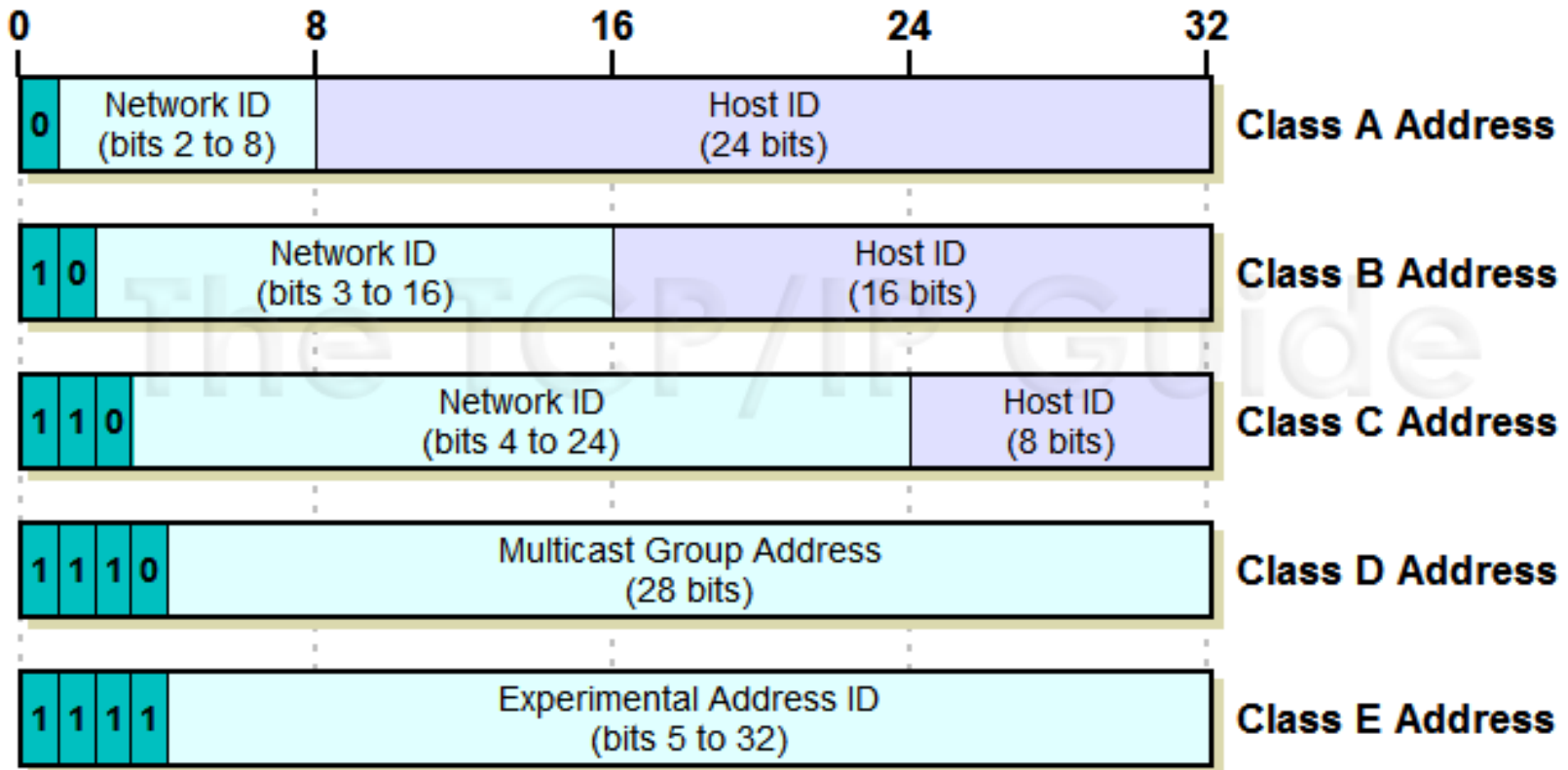
- Klasse A: sehr große Netzwerke, zum Beispiel Großkonzerne, Provider, Militär und Universitäten
- Klasse B: mittlere bis große Unternehmen, Universitäten, Provider
- Klasse C: kleinere Unternehmen, Provider
- Klasse D: Multicast Adressen
- Klasse E: reserviert für Entwicklungs- und Testzwecke

# Übersicht Adressklassen



- Die Grafik zeigt wie die Netze aufgeteilt wurden
- Dies geschah anhand des 1. Bit der IP Adresse
- Daraus ergeben sich folgende Bereiche
  - Klasse A
    - 1.0.0.0 bis 126.255.255.255
  - Klasse B
    - 128.0.0.0 bis 191.255.255.255
  - Klasse C
    - 192.0.0.0 bis 223.255.255.255
  - Klasse D
    - 224.0.0.0 bis 239.255.255.255:
  - Klasse E
    - 240.0.0.0 bis 255.255.255.255
- Heute verwenden wir den Klassen Begriff nur mehr um die Größe des jeweiligen Netzes zu beschreiben

# Classful Addressing



# Öffentliche und private IP Adressen

- Private IP Adressen
  - Benutzt zur Datenübertragung innerhalb privater Netze
  - Können vom Netzbetreiber innerhalb gewisser Regeln frei gewählt werden
- Öffentliche IP Adressen
  - Internet Corporation for Assigned Names and Numbers (ICANN)
  - Internet Assigned Numbers Authority (IANA)
    - Unterabteilung der ICANN
    - ARIN (Nordamerika), RIPE (Europa), APNIC (Asien), LACNIC (Lateinamerika), AfriNIC (Afrika)
  - regional Internet registries (RIRs)

# Private Adressen

- Anders als öffentliche Adressen sind private nicht weltweit einmalig
- Sie sind niemandem zugeteilt, und jeder darf sie frei verwenden
- Natürlich muss eine private Adresse im lokalen Netz dennoch eindeutig also einmalig sein
- Bereiche

Klasse	Start	Ende
Klasse A	10.0.0.0	10.0.0.0
Klasse B	172.16.0.0	172.31.0.0
Klasse C	192.168.0.0	192.168.255.0

# IPv4 Adresse Struktur

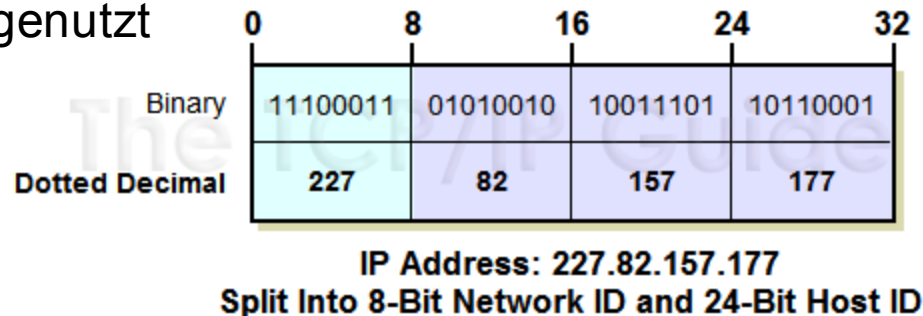
- Aus der Tabelle erkennen wir es gibt also einen Bereich der sich nicht verändert z.B. 10. bei der A Klasse und einen der sich verändert

IP-Adressraum	<b>XXX.XXX.XXX.XXX</b>	Klasse A
IP-Adressraum	<b>XXX.XXX.XXX.XXX</b>	Klasse B
IP-Adressraum	<b>XXX.XXX.XXX.XXX</b>	Klasse C

- Der fett gedruckte Teil ist statisch, darf als nicht verändert werden sonst befindet man sich in einem anderen Netz!
- Der normal gedruckte Teil kann „frei“ vergeben werden
  - Erste IP reserviert für „NetzID“
  - Letzte IP reserviert für „Broadcast Adresse“

# IPv4 Adresse Struktur

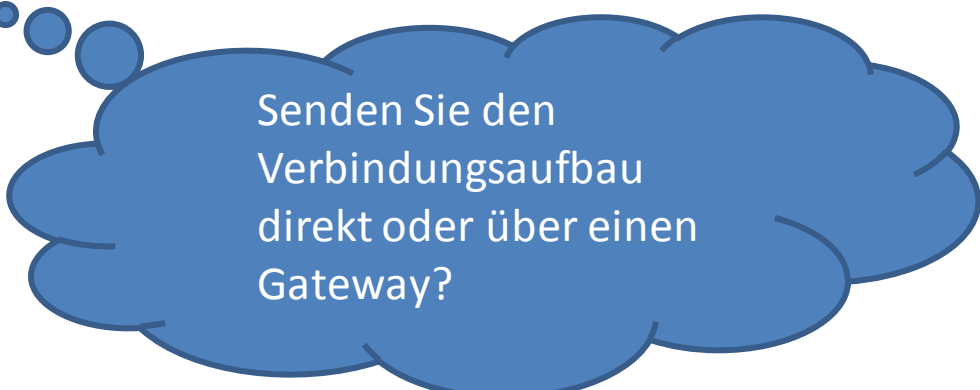
- Eine IP Adresse besteht also aus 2 Teilen
  - **Network Identifier (Network ID)**
    - Identifikations ID des Netzwerk
    - Gewisse Anzahl an Bits startend beim äußersten linken Bit
    - Wird auch *network prefix* genannt
  - **Host Identifier (Host ID)**
    - Der Rest der Bits wird zur eindeutigen Identifikation des jeweiligen Hosts genutzt





# Wozu muss man das überhaupt Klassen?

- Um die Grenzen unseres eigenen Netzes zu kennen
- Wozu muss man das überhaupt wissen?
  - Um zu erkennen ob die IP Adresse des Empfängers sich im selben Netz befindet und man ihr somit eine Nachricht direkt schicken kann oder ob man stattdessen diese über den Gateway senden muss
- Bsp.
  - Ihre IP Adresse ist 192.168.1.15 Sie wollen ihren Emailserver mit IP 192.168.2.20 erreichen

A blue thought bubble with a tail pointing towards the example text.

Senden Sie den Verbindungsaufbau direkt oder über einen Gateway?

# Classful Addressing

- Die strikte Klasseneinteilung hat einige Nachteile
  - Mangelnde interne Adresse Flexibilität
    - Großen Organisationen werden großes, „monolithische“ Blöcke der Adressen zugewiesen, die nicht gut zur Struktur ihrer internen Netze passen
  - Ineffiziente Verwendung des Adressbereiches
    - Die Unterteilung in nur 3 Klassen (A, B und C) werden IP Adressen vergeudet
  - Starke Zunahme der Routingtabelleneinträge

## Lösung: Subnetting

*Anmerkung:*

*Verwendet man herkömmliche Klassen sprechen wir von einer Netzmaske, werden Klassen weiter unterteilt von einer Subnetzmaske – kein Unterschied in der Funktion!*



# Netzmaske / Subnetzmaske

# Netzmaske

- Wurde im August 1985 eingeführt da man immer mehr Netze benötigte und mit der Classfull Variante viele IP Adressen verschwendete
- Für die Zuteilung der Subnetzmaske war damals noch der Router zuständig, den man mittels ICMP Mask request nach der Subnetzmaske fragte
- *Anmerkung: Damals wurde die IP Adresse sofern nicht statisch konfiguriert mittels RARP (Juni 84) zugewiesen, BOOTP (Sep. 85) bzw. DHCP (Okt. 93) wurden danach erfunden*

# Netzmaske

- Antwort auf die Emailserver Frage: Wir wissen es (noch) nicht!
- Da wir heutzutage nicht mehr die ersten Bit zur Klasseneinteilung verwenden brauchen wir eine zusätzliche Information um zu erkennen um welche Klasse es sich bei einer IP Adresse handelt
- Die Netzmaske erlaubt die Unterteilung der IP Adresse in Host- und Netzwerkteil
- Dies erfolgt durch die **UND-Verknüpfung (Konjunktion)** der IP Adresse mit der Netzmaske
  - Das Ergebnis ist immer die NetzID

# Beispiel Netzmaske des Absender

- Wir berechnen das Beispiel der vorletzten Folie

IP Adresse Eigene	Dezimal	192	168	1	15
	Binär	11000000	10101000	00000001	00001111
Netzmaske Eigene	Dezimal	255	255	255	0
	Binär	11111111	11111111	11111111	00000000
UND Gatter		11000000	10101000	00000001	00001111
		11111111	11111111	11111111	00000000
Ergebnis		11000000	10101000	00000001	00000000

Die NetzID zu unserer IP Adresse ist **192.168.1.0**

Hat die Ziel-IP Adresse dieselbe NetzID befindet sie sich im selben Netz andernfalls nicht! • • •

Woher weiß man die Netzmaske der Ziel-IP?

# Beispiel Netzmaske des Empfängers

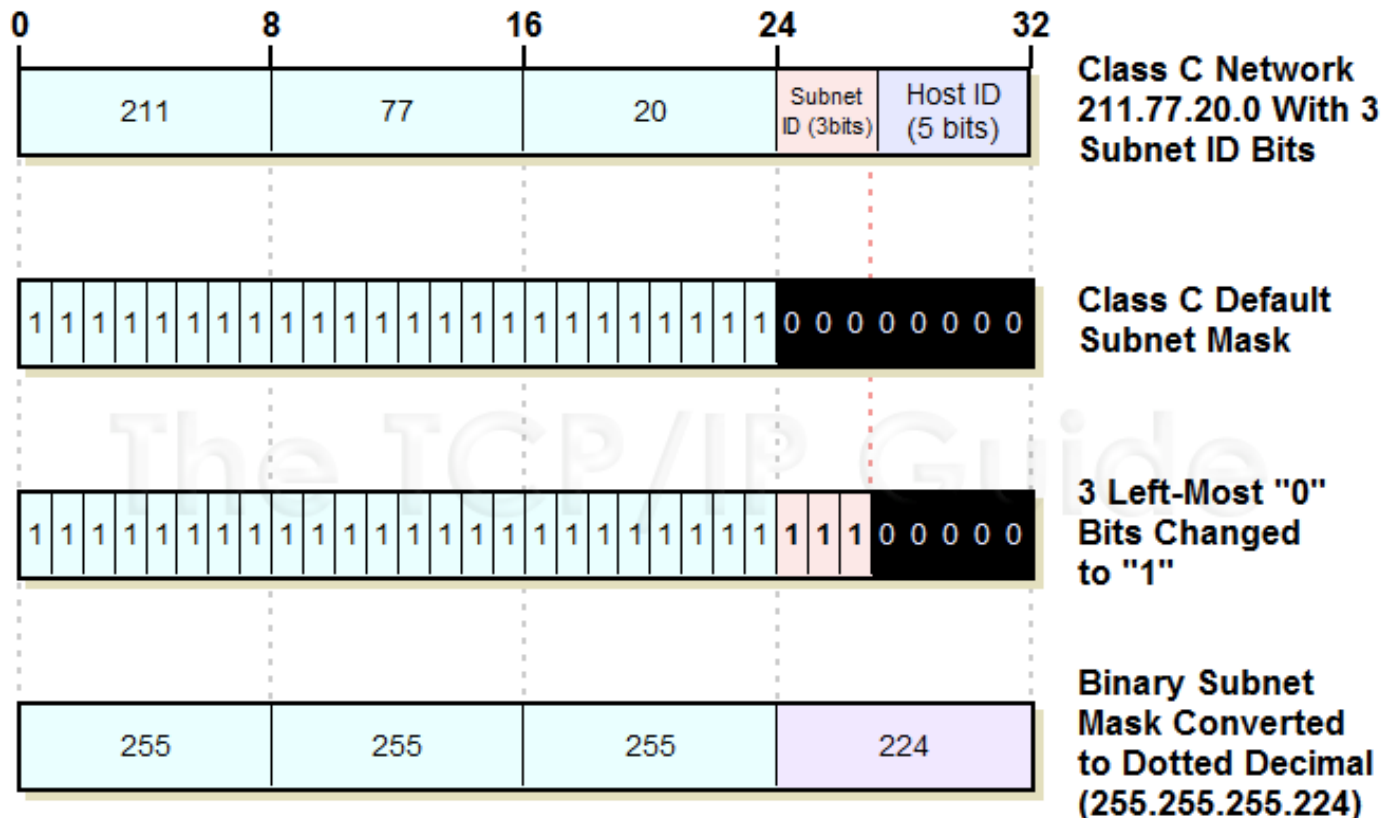
- Antwort: Sofern die Ziel IP im selben Netz wie wir sind, muss sie unweigerlich dieselbe Netzmaske haben, also verwenden wir unsere eigene zur Berechnung!

IP Adresse Mailserver	Dezimal	192	168	2	20
	Binär	11000000	10101000	00000010	00010100
Netzmaske Eigene	Dezimal	255	255	255	0
	Binär	11111111	11111111	11111111	00000000
UND Gatter		11000000	10101000	00000010	00010100
		11111111	11111111	11111111	00000000
Ergebnis		11000000	10101000	00000010	00000000

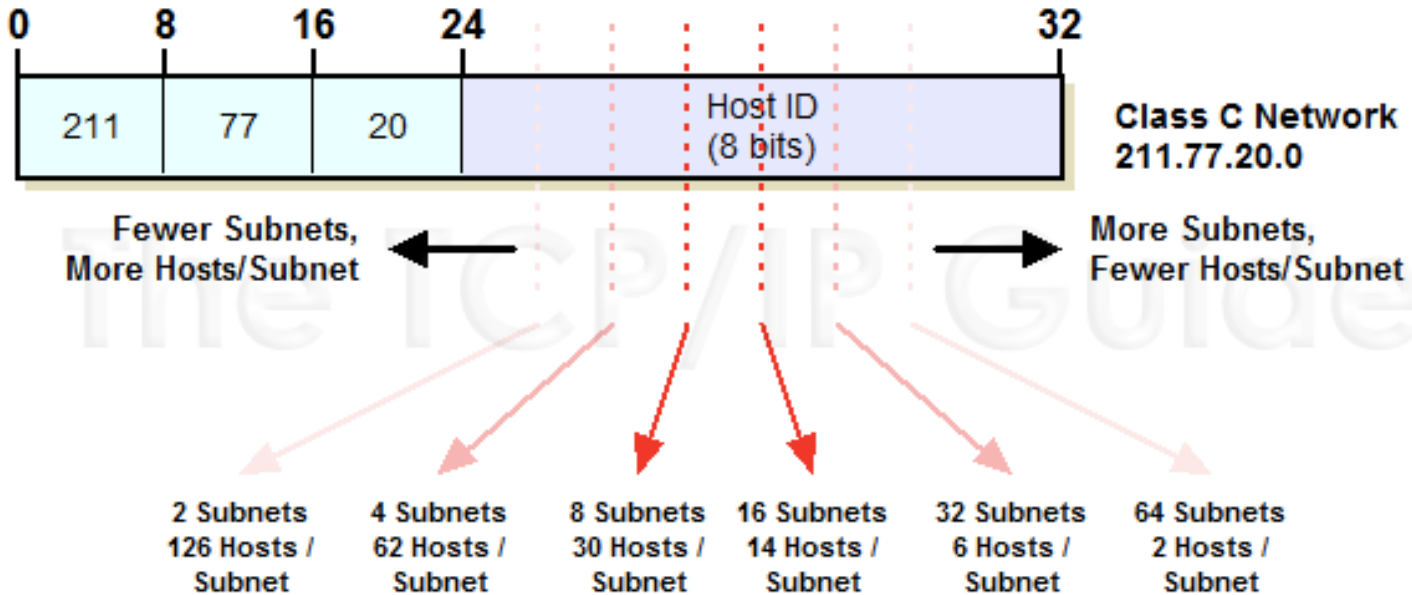
- Die NetzID zu unserer IP Adresse ist **192.168.2.0**  
**Diese beiden IP Adressen sind also nicht im selben Netz, eine Kommunikation ist nur über den Gateway möglich**

# Berechnung Subnetzmaske

- Klassen können wir wie gesagt noch weiter unterteilen, wir sprechen dann von einer Subnetzmaske







Mögliche Hosts pro Subnet:

**Max. Hosts – 2<sup>n</sup>**

Abzüge:  
1 für NetzId  
1 für Broadcast Adresse

# Subnetting Beispiel 1

Slashnotation die  
ersten x Bit der  
Subnetzmaske sind  
1 der Rest 0

IP Adressbereich: 192.168.10.0/27

Subnetzmaske: 255.255.255.224

Mögliche Subnetze: 8

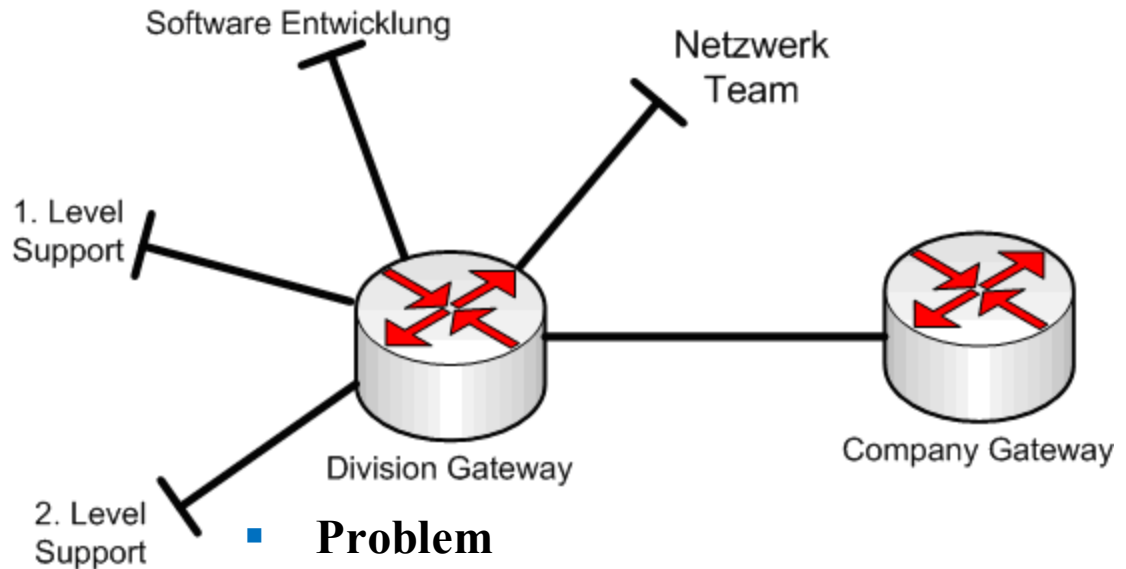
Maximale Hosts pro Subnetz: 30 (32-2)

*Anmerkung: Alle Subnetze mussten gleich groß  
sein!*

SubnetzID	Subnetz Adresse	Host	Broadcast
0	192.168.10.0	.1 - .30	.31
1	192.168.10.32	.33 - .62	.63
2	192.168.10.64	.65 - .94	.95
3	192.168.10.96	.97 - .126	.127
4	192.168.10.128	.129 - .158	.159
5	192.168.10.160	.161 - .190	.191
6	192.168.10.192	.193 - .222	.223
7	192.168.10.224	.225 - .254	.255

# Subnetting Beispiel 2

Der Konzern Netzwerkadmin stellt ihrer Niederlassung das Subnetz 192.168.10.0/24 zur Verfügung.



- Wir benötigen  
 $10+70+31+20=131$  Adressen
  - Netzwerkteam 10 IP
  - Softwareentwicklung 70 IP
  - 1. Level Support 31 IP
  - 2. Level Support 20 IP

- **Problem**

Alle Subnetze müssen gleich groß sein und die SE muss mit 70 Hosts reinpassen  
Das geht sich einfach nicht aus obwohl wir eigentlich nur 131 Adressen brauchen!

- **Lösung**

- Variable Length Subnet Mask (VLSM)



# Variable Length Subnet Mask (VLSM)

# VLSM Lösung

## von Subnetting Beispiel 2

- Es gibt verschiedene Möglichkeiten Subnetze richtig zu berechnen hier ein selbstentwickelter Ansatz
1. Wir notieren uns als Hilfswerte in der ersten Zeile die Slashnotation und darunter die jeweilige Anzahl der Hosts des Hostteils (ohne Abzüge von NetzID und Broadcast). In unserem Beispiel brauchen wir eine Klasse von der Größe von einer C Klasse also starten wir bei /24 bis runter zu /30 (darunter ist nicht notwendig):

Slash Notation	/24	/25	/26	/27	/28	/29	/30
Maximale Hosts	256	128	64	32	16	8	4

Anmerkung: erweitern sie den linken Bereich jeweils passend zum aktuellen Beispiel also /23, /22 usw.

# VLSM Lösung von Subnetting Beispiel 2

2. Wir starten bei der Berechnung von VLSM Netzen immer mit dem größten Netz und arbeiten uns bis zum kleinsten vor



a) Softwareentwicklung 70 IP

Wir suchen in der Hilfsliste zwischen welchen Bereichen unsere 70+2 IP Adressen (+1 für NetzID & +1 für Broadcast Adresse) einzuordnen sind

/24	/25	/26	/27	/28	/29	/30
256	128	64	32	16	8	4



70+2

IP Adressen

Und nehmen die nächst größere Subnetzgruppe hier also /25  
Nun wissen wir welches Subnetz wir verwenden müssen

# VLSM Lösung

## von Subnetting Beispiel 2

- Als SubnetzID nehmen wir unsere Startadresse mit der gerade ermittelten Subnetzmaske: 192.168.10.0/25
- Zur Broadcast Berechnung nehmen wir die nächste mögliche SubnetzID (aktuelle SubnetzID + Anzahl der Hosts an der zugehörigen Slash Notation in unserer Hilfsliste) und ziehen 1 davon ab. (die letzte Broadcast ist immer die IP vor der nächsten SubnetzID)
  - $0 + 128 - 1 = 127$
- Die möglichen Hosts sind immer -2 zu berechnen wegen Broadcast und SubnetzID
  - $128 - 2 = 126$

	Softwareentw.	1. Level Support	2. Level Support	Netzwerkteam
SubnetzID	192.168.10.0/25			
Broadcast	192.168.10.127/25			
Mögliche Hosts	126			

# VLSM Lösung

## von Subnetting Beispiel 2

### b) 1. Level Support 31 IP

Wir wählen nun das nächst kleinere Netzwerk hier 1. Level Support

Der Hostteil der IP Adresse der SubnetzID ist einfach und haben wir gerade berechnet, die letzte SubnetzID + maximale Hosts d.h.

$0+128=128$ . Nun brauchen wir die passende Slashnotation

/24	/25	/26	/27	/28	/29	/30
256	128	64	32	16	8	4



31+2

IP Adressen

Achtung nie auf die +2 vergessen (SubnetzID+Broadcast). Wir nehmen wieder die nächst größere Slashnotation, also /26



# VLSM Lösung

## von Subnetting Beispiel 2

Nun nehmen wir wieder unsere SubnetzID fügen die maximalen Hosts dazu um die nächste SubnetzID zu bekommen, von der ziehen wir 1 ab für unsere aktuelle Broadcast.

- $128+64-1=191$

Die möglichen Hosts sind

- $64-2=62$

Das ergibt folgende Einträge:

	Softwareentw.	1. Level Support	2. Level Support	Netzwerkteam
SubnetzID	192.168.10.0/25	192.168.10.128/26		
Broadcast	192.168.10.127/25	192.168.10.191/26		
Mögliche Hosts	126	62		

# VLSM Lösung von Subnetting Beispiel 2

## c) 2. Level Support 20 IP

Wir wählen wieder das nächst kleinere Netzwerk hier 2. Level Support

Wir berechnen wieder die nächste gültige SubnetzID indem wir letzte SubnetzID+max. Hosts oder auch letzter Broadcast+1 berechnen.

–  $128+64=192$  bzw.  $191+1=192$

/24	/25	/26	/27	/28	/29	/30
256	128	64	32	16	8	4



20+2

IP Adressen

Wir vergessen nicht auf unser +2 (SubnetzID+Broadcast) und nehmen wieder die nächst größere Slashnotation, also /27

# VLSM Lösung

## von Subnetting Beispiel 2

Wir kennen nun den Ablauf schon, aktuelle SubnetzID + maximale Hosts dazu um die nächste SubnetzID zu bekommen, von der ziehen wir 1 ab für unsere aktuelle Broadcast.

- $192+32-1=223$

Die möglichen Hosts sind

- $32-2=30$

Das ergibt folgende Einträge:

	Softwareentw.	1. Level Support	2. Level Support	Netzwerkteam
SubnetzID	192.168.10.0/25	192.168.10.128/26	192.168.10.192/27	
Broadcast	192.168.10.127/25	192.168.10.191/26	192.168.10.223/27	
Mögliche Hosts	126	62	30	

# VLSM Lösung

## von Subnetting Beispiel 2

d) Netzwerkteam 10 IP

Für das letzte Subnetz berechnen wir wieder die nächste gültige SubnetzID indem wir letzte SubnetzID+max. Hosts oder auch letzter Broadcast+1 berechnen.

–  $192+32=224$  bzw.  $223+1=224$

/24	/25	/26	/27	/28	/29	/30
256	128	64	32	16	8	4



10+2

IP Adressen

Wir vergessen nicht auf unser +2 (SubnetzID+Broadcast) und nehmen wieder die nächst größere Slashnotation, also /28

# VLSM Lösung

## von Subnetting Beispiel 2

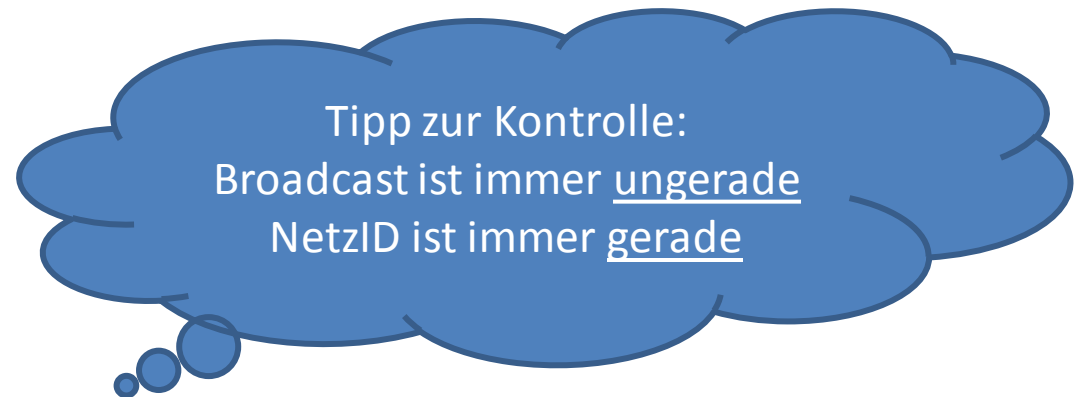
Wie immer, aktuelle SubnetzID + maximale Hosts dazu um die nächste SubnetzID zu bekommen, von der ziehen wir 1 ab für unsere aktuelle Broadcast.

- $224 + 16 - 1 = 239$

Die möglichen Hosts sind

- $16 - 2 = 14$

Das ergibt folgende Einträge:



	Softwareentw.	1. Level Support	2. Level Support	Netzwerkteam
SubnetzID	192.168.10.0/25	192.168.10.128/26	192.168.10.192/27	192.168.10.224/28
Broadcast	192.168.10.127/25	192.168.10.191/26	192.168.10.223/27	192.168.10.239/28
Mögliche Hosts	126	62	30	14

# VLSM Beispiel 1

Unser Unternehmen wurde von einem Konzern gekauft und wir der DACH Division untergliedert. Für die VPN Anbindung unseres Netzes ins Konzernnetz müssen wir unsere internen IP Adressen auf den uns zugewiesenen Bereich umstellen.

Nachdem wir die benötigten IP Adressen für unsere Abteilungen inklusive unserer geplanten Zuwachsraten für die nächsten 5 Jahre gemeldet haben

- Management 6 IP
- Service 130 IP
- Buchhaltung & HR 7 IP
- Server 4 IP

wurde uns der Bereich 10.0.18.0/23 zugewiesen.

Erstellen Sie die Subnetze wie wir es gelernt haben, Ergebnis können Sie zur Selbstkontrolle im Moodle prüfen.

# VLSM Beispiel 2

- Eine Firma bekommt von Ihrem ISP den IP Adressenbereich 204.45.222.0/24 zugewiesen. Die Firma hat 6 Subnetze. Vier Subnetze (S1, S2, S3, S4) sind sehr kleine mit jeweils 10 Hosts, S5 beinhaltet 50 Hosts und S6 100 Hosts.
- Finden Sie eine Lösung für dieses Subnetting Problem.

Erstellen Sie die Subnetze wie wir es gelernt haben, Ergebnis können Sie zur Selbstkontrolle im Moodle prüfen.

# Lösung VLSM Beispiel 2

1. Wir benötigen ein Netz für mind. 100 Hosts und berechnen  $2^7 - 2 = 126$  dh. 7 Bits werden für die Hosts benötigt bleibt 1 Bit für die SubnetzID.

Subnetze:

- 204.45.222.0/25
- 204.45.222.128/25

2. Wir nehmen das 2. Subnetz und benötigen mind. 50 Hosts und berechnen  $2^6 - 2 = 62$

Subnetze:

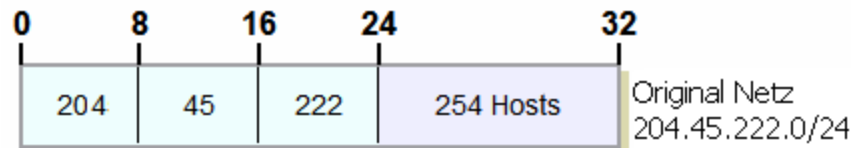
- 204.45.222.128/26
- 204.45.222.192/26

3. Nun nehmen wir wieder das 2. Subnetz und benötigen 4 weitere Netze dh. Wir nehmen 2 Bits der verfügbaren 6 für die SubnetzID somit verbleiben  $2^4 - 2 = 14$

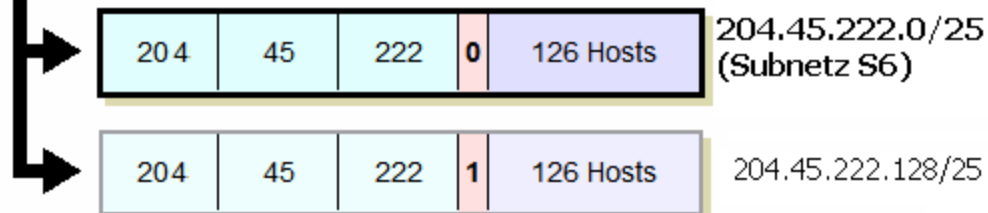
Subnetze:

- 204.45.222.192/28
- 204.45.222.208/28
- 204.45.222.224/28
- 204.45.222.240/28

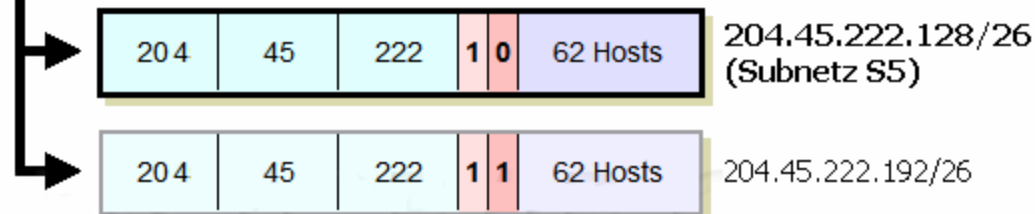




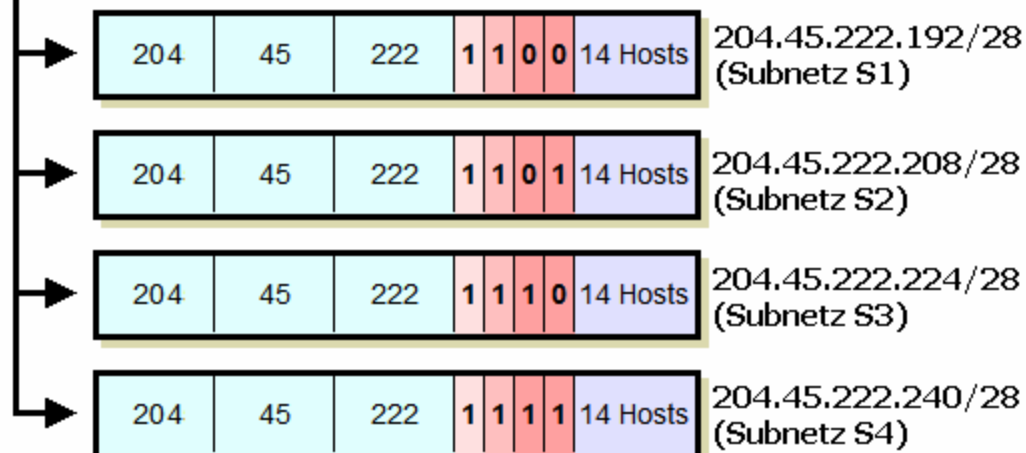
Erste Unterteilung: /24 Netz in /25 Subnetze



Zweite Unterteilung 204.45.222.128/25  
in 2 /26 Subnetze



Dritte Unterteilung 204.45.222.192/26 in  
vier /28 Subnetze



# VLSM Beispiel 3

- Folgende Aufgabe braucht ein wenig mehr Überlegung, lassen Sie sich Zeit um darüber nachzudenken!
- Ein ISP erhält folgenden IP Adressenbereich 79.94.0.0/15
- Das Management entscheidet, dass der Bereich in 2 gleich große Teile unterteilt werden soll. Zunächst wird nur die erste Hälfte des Bereiches verwendet, die 2. Hälfte wird für eine spätere Verwendung aufgespart
- Die erste Hälfte wird in vier /18 er Blöcke unterteilt
  - Jeder dieser Blöcke soll in Teilblöcke für unterschiedlich große Kunden unterteilt werden
  - Ein Block für Teilblöcke zu je 510 Hosts, einer für 254 Hosts, einer für 126 und einer für 62 Hosts

Erstellen Sie jeweils das 1. und letzte Subnetz für die 4 Blöcke wie wir es gelernt haben, Ergebnis können Sie zur Selbstkontrolle im Moodle prüfen.

# Quellen

- Kozierok, Charles M. “Welcome to The TCP/IP Guide!”, [www.tcpipguide.com](http://www.tcpipguide.com). Accessed 6 May 2020.