Protocols and Concepts at Network Layer

Complete Course on Computer Networks - Part I

Ravindrababu RAVULA • Lesson 10 • Jan 28, 2021

NOTE:

In order to find out the SID (Subnet id) or No of subnets, we must know either the Class of that network or NID

We can find out the HID even if The class of the network is not known

For example:
If Subnet mask = 255.255.255.192, And it known to be of Class A then,

We know that,
NID + SID = No of 1's
HID = No of 0's

No of 1's = 26
NID In class A = 8 bits
8 + SID = 26
SID = 18
No of subnets = 2^{18}

No of 0's = 6
HID = 6
IP/network = 2^{6}
Hosts/subnet=2^{6}-2

QUESTION:

If the subnet mask 255.255.255.128 belongs to class C, find-

1. Number of subnets
2. Number of hosts in each subnet

SOLUTION:

Given subnet mask= 255.255.255.128

= 11111111.11111111.11111111.10000000

Since 25 bits contain the value 1 and 7 bits contain the value 0, so-

- Number of NID bits + Number of Subnet ID bits = 25
- Number of HID bits = 7

Now,

- It is given that subnet mask belongs to class C.
- So, Number of NID bits = 24.

Substituting in the above equation, we get-

Number of Subnet ID bits

= 25 – 24

= 1

Number of subnets = 2^1 = 2

Since number of HID bits = 7, so-

Number of hosts per subnet = $2^7 - 2$ = 126

GATE 2008

If a class B network on the Internet has a subnet mask of 255.255.248.0, what is the maximum number of hosts per subnet?

(A) 1022

(B) 1023

(C) 2046

(D) 2047

Answer (C)

The binary representation of subnet mask is
11111111.11111111.11111000.00000000.
There are 21 bits set in subnet.
So 11 bits are left for host ids.
Total possible values of host ids is 2^11 = 2048.

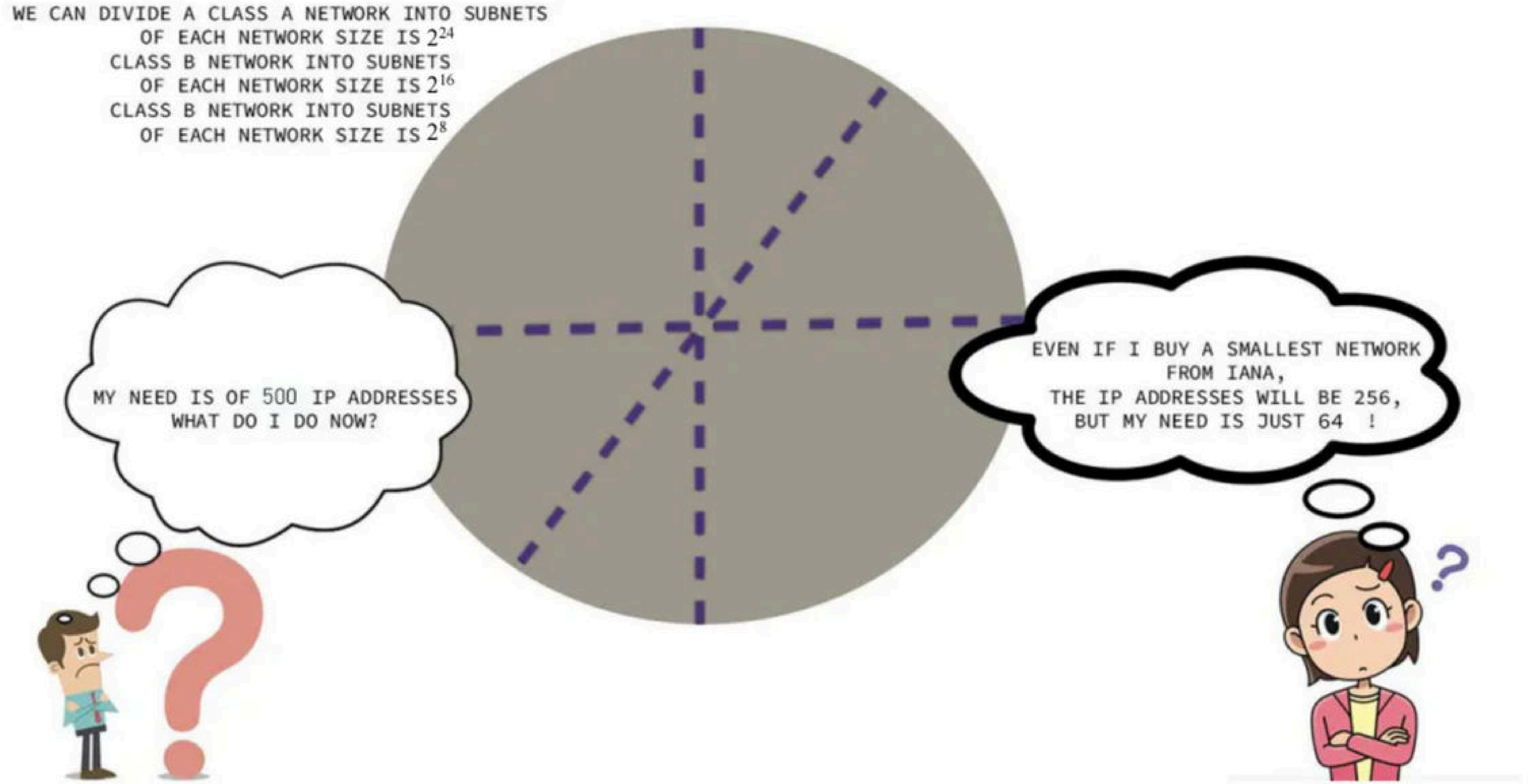
Out of these 2048 values, 2 addresses are reserved.
The address with all bits as 1 is reserved as broadcast address and address with all host id bits as 0 is used as network address of subnet.

In general, the number of addresses usable for addressing specific hosts in each network is always 2^N – 2 where N is the number of bits for host id.

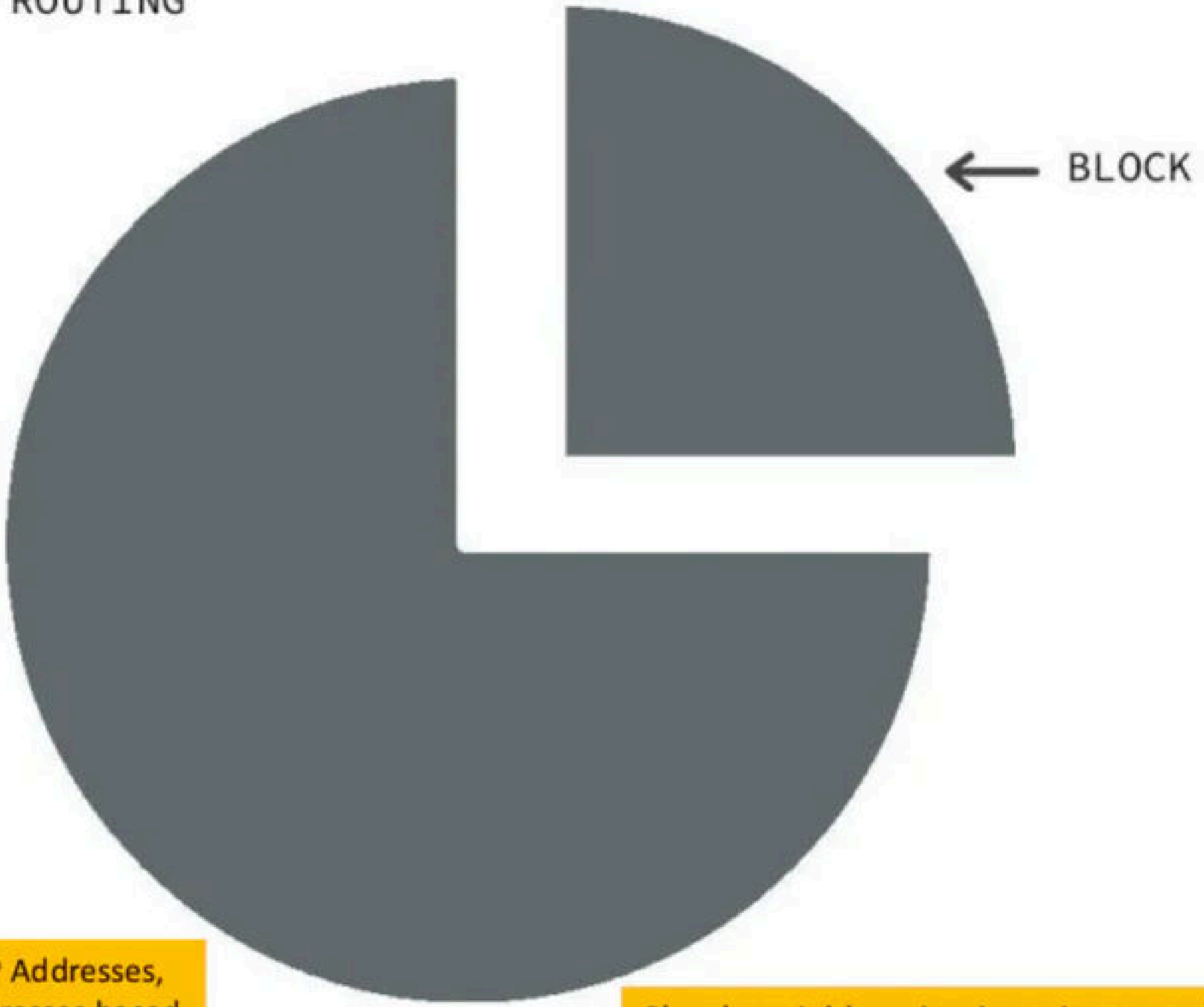
Therefore, maximum no of hosts per subnet = 2048 -2 = 2046

Computer Networks

Classless Inter Domain Routing (CIDR)


WE CAN DIVIDE A CLASS A NETWORK INTO SUBNETS
OF EACH NETWORK SIZE IS 2^{24}
CLASS B NETWORK INTO SUBNETS
OF EACH NETWORK SIZE IS 2^{16}
CLASS B NETWORK INTO SUBNETS
OF EACH NETWORK SIZE IS 2^{8}
MY NEED IS OF 500 IP ADDRESSES
WHAT DO I DO NOW?
EVEN IF I BUY A SMALLEST NETWORK
FROM IANA,
THE IP ADDRESSES WILL BE 256,
BUT MY NEED IS JUST 64 !

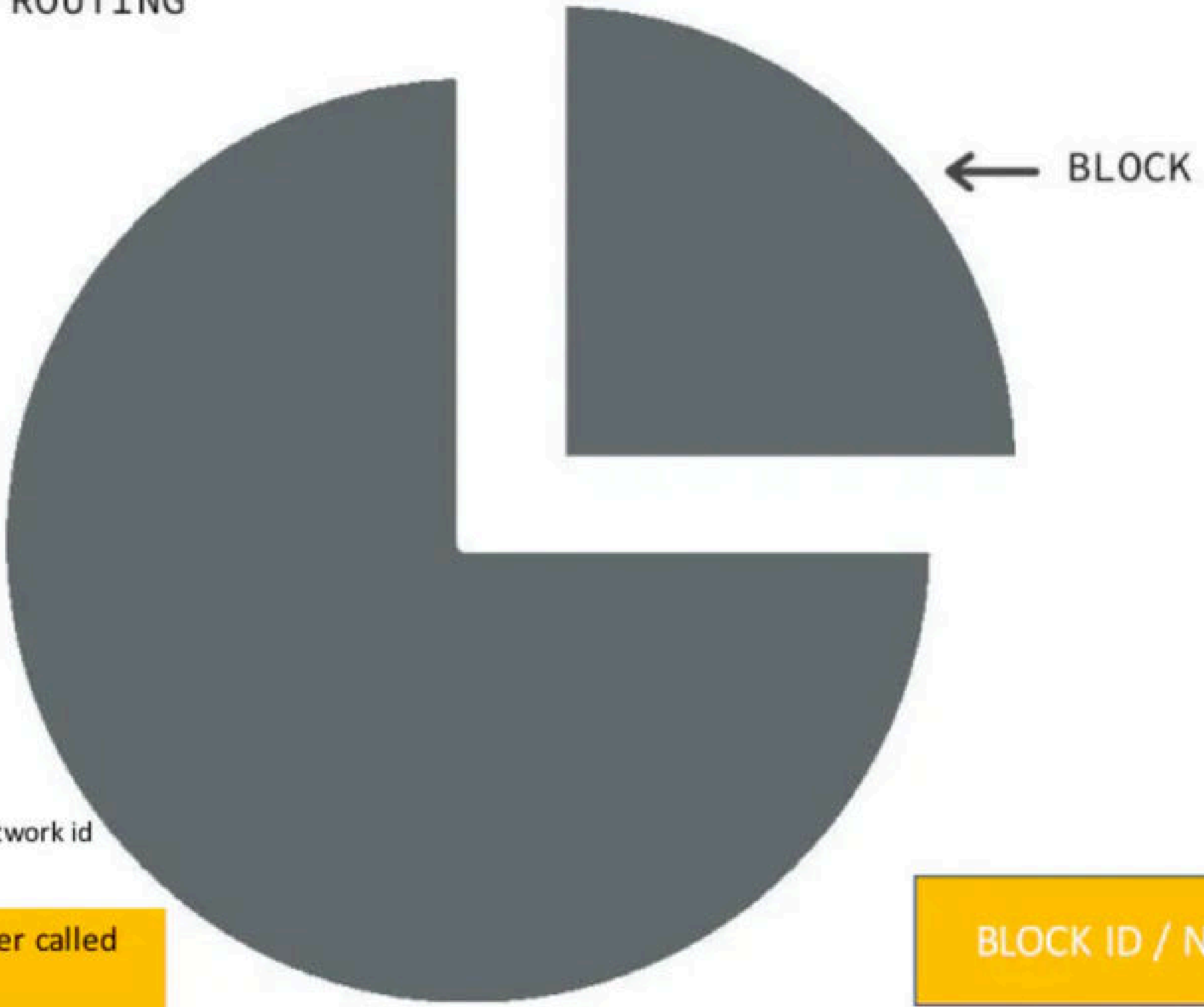
CLASSLESS INTER DOMAIN ROUTING
CIDR



When a user asks for specific number of IP Addresses, CIDR dynamically assigns a block of IP Addresses based on certain rules.
This block contains the required number of IP Addresses as demanded by the user.
This block of IP Addresses is called as a CIDR block.

Classless Addressing is an improved IP Addressing system.
It makes the allocation of IP Addresses more efficient.
It replaces the older classful addressing system based on classes.
It is also known as Classless Inter Domain Routing (CIDR)

CLASSLESS INTER DOMAIN ROUTING
CIDR



CIDR Representation:

a.b.c.d / n

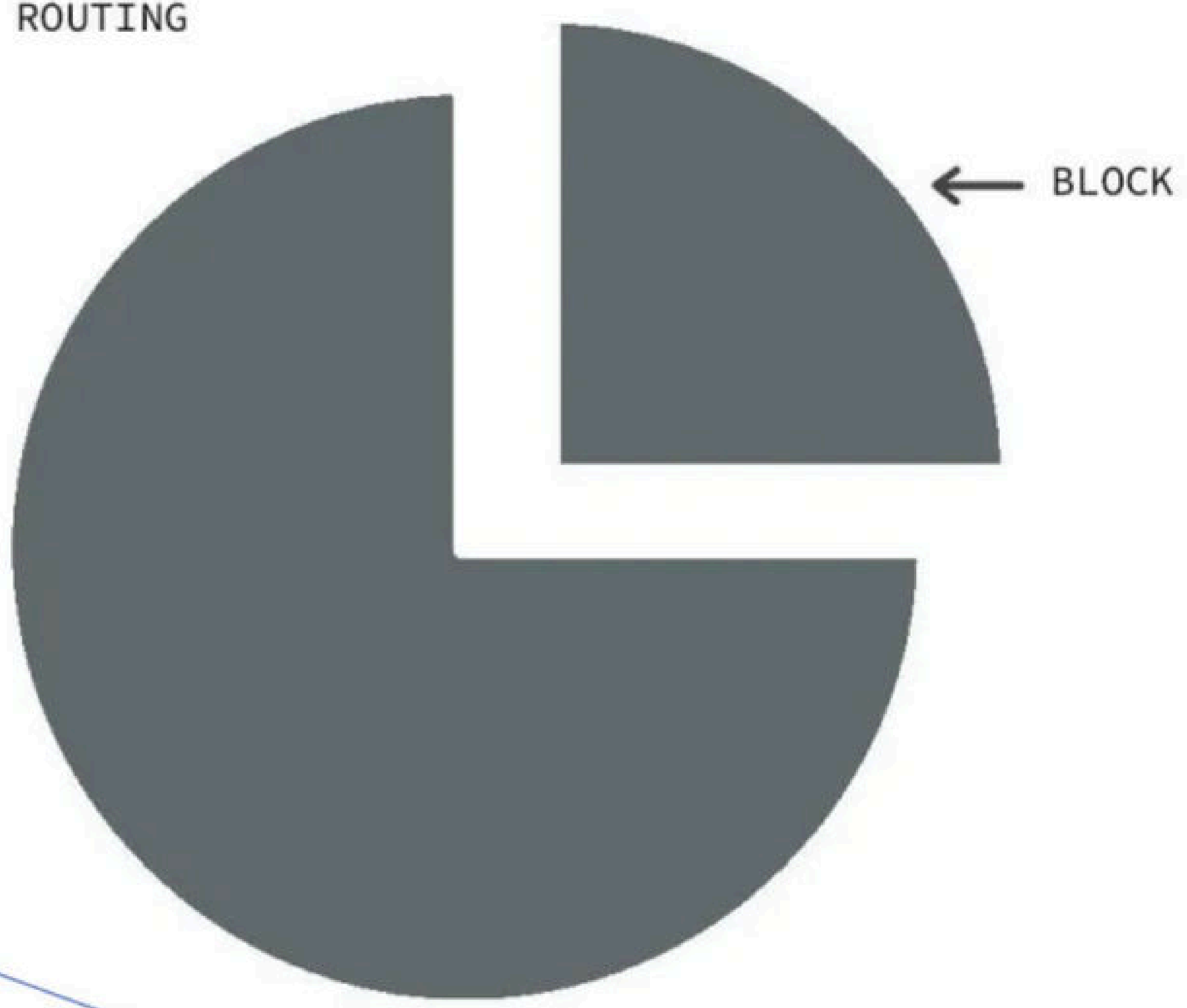
Dotted decimal representation

Block id / network id

They end with a slash followed by a number called as IP network prefix.
IP network prefix tells the number of bits used for the identification of network.
Remaining bits are used for the identification of hosts in the network.

BLOCK ID / NID	HOST ID

CLASSLESS INTER DOMAIN ROUTING
CIDR



CIDR Representation:
a.b.c.d / n

Example:
20.10.50.100/20

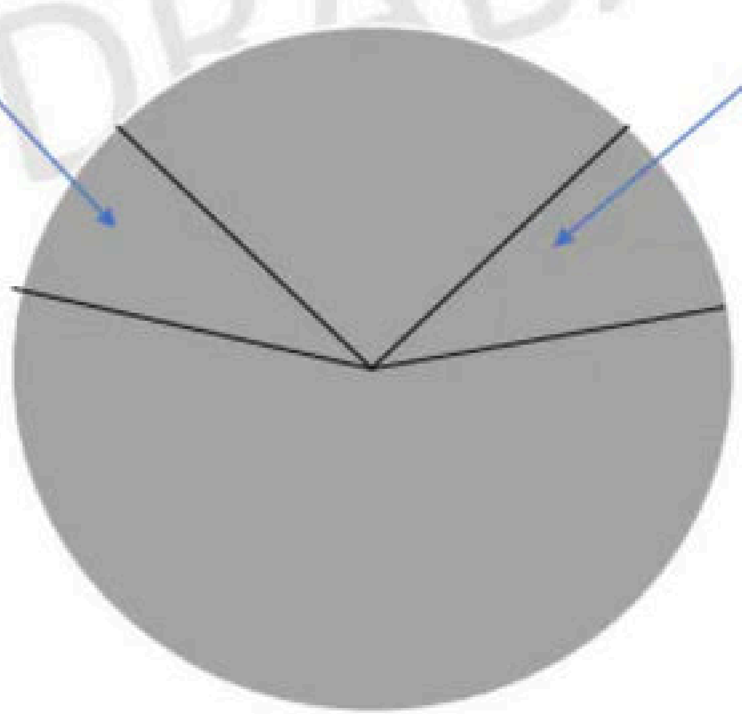
BID or NID = 20
HID =32 -20 = 12

Note: Even if the IP looks like it belongs to Class A but it is not.
Since there are no Classes in CIDR

Rules of CIDR blocks

1. **All IP addresses should be contiguous**
2. Block size should be a power of 2
3. First Ip address of the block must be evenly divisible by the size of the block.

To form a Block
You cannot take some part from here and some from here



1 1 1 1 = 15

	rem	quo
$/2^1$	1	7
$/2^2$	3	3
$/2^3$	7	1
$/2^4$	15	0

1 2 3 4 10

100

1000

2^n

NID	HID
BID	

32 IP

Block is always 2^n

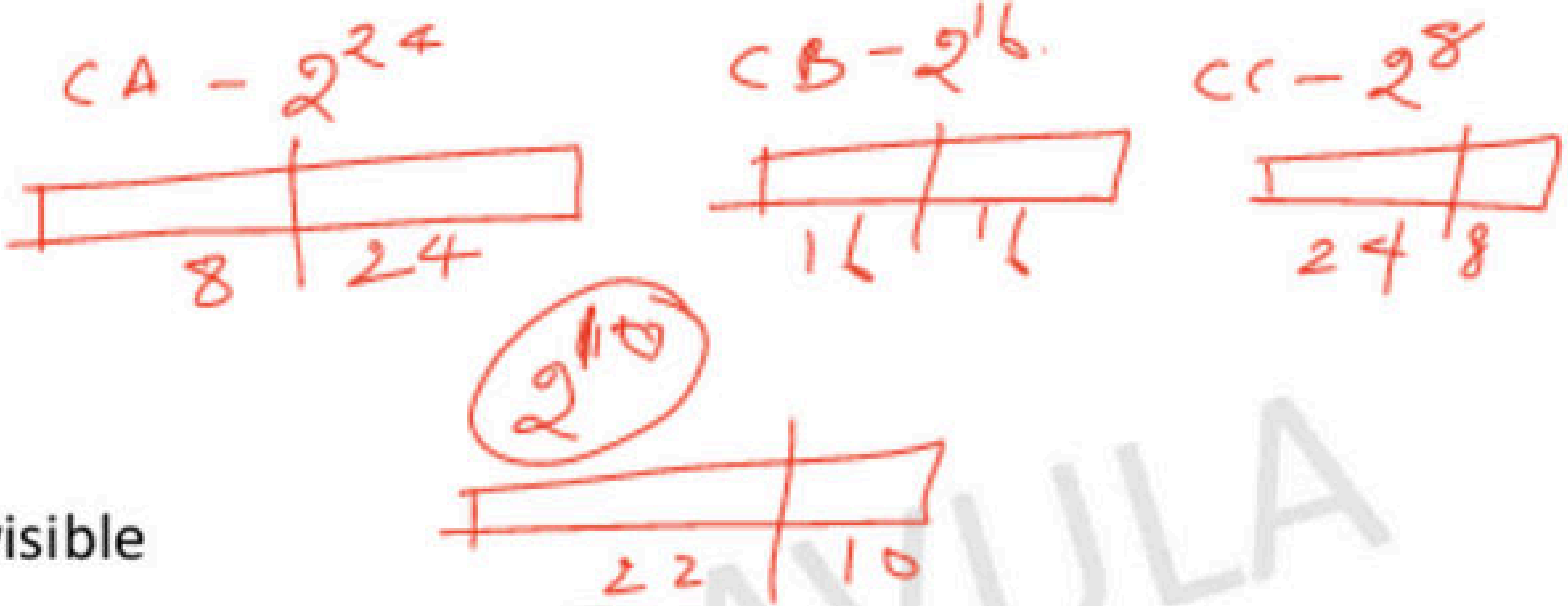
1 1 1 1 1

n bits 2^k

$\underbrace{- - - -}_{(n-k)} \mid \underbrace{- - -}_{k \text{ bits}} \rightarrow$ remainder

1 1 1 1 2^2

Rules of CIDR blocks



1.) All IP addresses should be contiguous

2.) Block size should be a power of 2

3.) First Ip address of the block must be evenly divisible by the size of the block.

If any binary pattern consisting of (m + n) bits is divided by 2^n, then-

- Remainder is least significant n bits
- Quotient is most significant m bits

So, any binary pattern is divisible by 2^n, if and only if its least significant n bits are 0.

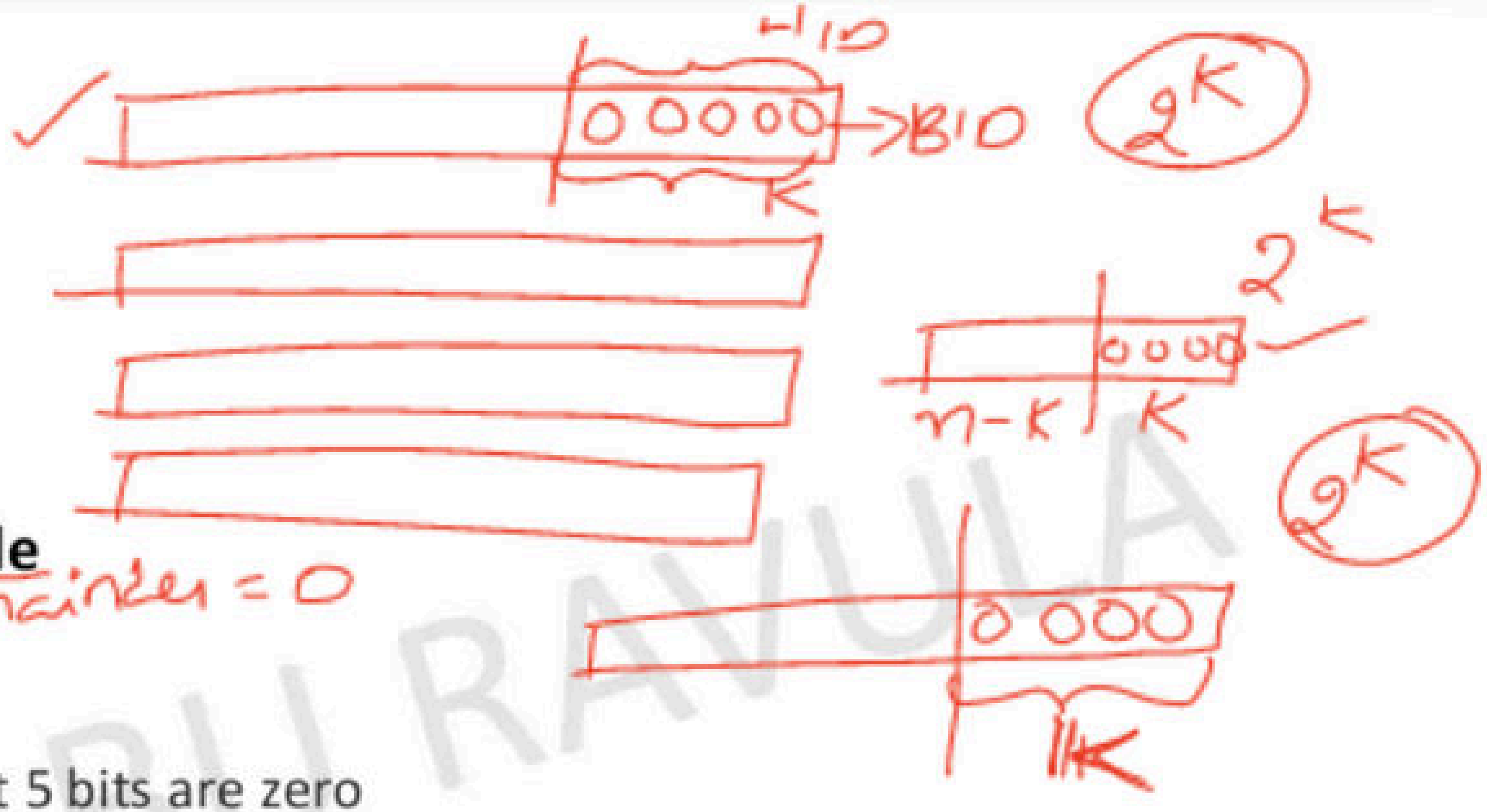
Examples-

Consider a binary pattern-

01100100.00000001.00000010.01000000

(represented as 100.1.2.64)

- It is divisible by 2^5 since its least significant 5 bits are zero.
- It is divisible by 2^6 since its least significant 6 bits are zero.
- It is not divisible by 2^7 since its least significant 7 bits are not zero.



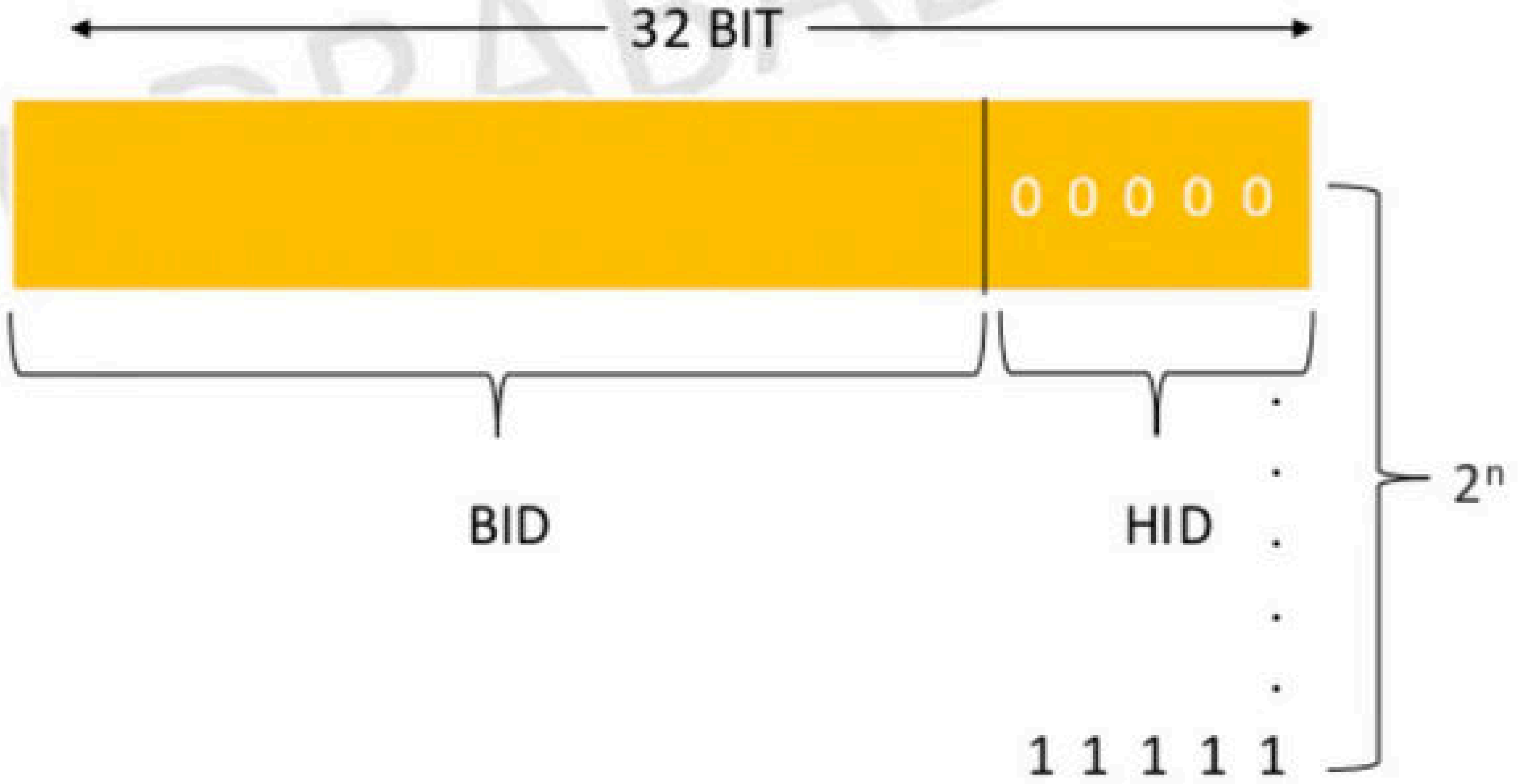
Rules of CIDR blocks

1.) All IP addresses should be contiguous
2.) Block size should be a power of 2
3.) First Ip address of the block must be evenly divisible by the size of the block.

remainder = 0

2^K

$32 = 2^5$ = least significant 5 bits are zero



We can use First IP address
As the Block id

Note : If the First Ip does not have Rule 3
Then we wont get 2^n IP addresses.

(1 - 4)

(4 - 1) + 1

Rules of CIDR blocks

1.) All IP addresses should be contiguous

2.) Block size should be a power of 2

3.) First Ip address of the block must be evenly divisible by the size of the block.

Example: 100.1.2.32
100.1.2.33
.
.
.
100.1.2.47

100.1.2.32
100.1.2.33
⋮
100.1.2.47

$16 = 2^4$

2^4

HID = 4

BID = 28

$100.1.2.0010\underbrace{\boxed{0000}}$

Valid block.

$\boxed{100.1.2.32/28}$

$\boxed{100.1.2.40/28}$

100.1.2.40/28

$\underbrace{100.1.2.0010}_{BID}\overbrace{1000}^{HID}$

0000 — 100.1.2.32 → BID

0001 — 100.1.2.33

0010

⋮

1111 — 100.1.2.47 → DBA

14 hosts

Rules of CIDR blocks

1.) All IP addresses should be contiguous
2.) Block size should be a power of 2
3.) First Ip address of the block must be evenly divisible by the size of the block.

Example: 100.1.2.32
100.1.2.33
.
.
.
100.1.2.47

<u>Rules of CIDR blocks</u>

1.) All IP addresses should be contiguous
2.) Block size should be a power of 2
3.) First Ip address of the block must be evenly divisible
by the size of the block.

Example:

100.1.2.32
100.1.2.33
.
.
.
100.1.2.47

$16 = 2^4$

Rules of CIDR blocks

1.) All IP addresses should be contiguous
2.) Block size should be a power of 2
3.) First Ip address of the block must be evenly divisible by the size of the block.

Example: 100.1.2.32
100.1.2.33
.
.
.
100.1.2.47

If any n bit number has to be divided by 2^4
Then the remainder will be last 4 Least Significant Bits

100.1.2.0010$\boxed{0000}$ / 2^4

0 is remainder
Therefore, divisible!

100.1.2.32
100.1.2.33

16 = 2^4 IP ADDRESSES

100.1.2.47

Therefore, the number of bits used for HID = 4
So, BID = 28 bits

Any above IP address can be represented in CIDR
Ex: 100.1.2.32 / 28

Where 28 represents the BID

Q.) Consider a block of IP Addresses ranging from 150.10.20.64 to 150.10.20.127.

Is it a CIDR block? If yes, give the CIDR representation.

150.10.20.64 ✓

.65

.66

⋮

.127

(i) Con

(ii) $\text{size} = 2^n \quad 64 = 2^6$ ✓

$150.10.20.01\boxed{000000}$

2^6

HID = 6

NID = 26

$\boxed{150.10.20.64/26}$ ✓

Q.) Consider a block of IP Addresses ranging from 150.10.20.64 to 150.10.20.127.

Is it a CIDR block? If yes, give the CIDR representation.

Rules of CIDR blocks

1.) All IP addresses should be contiguous
2.) Block size should be a power of 2
3.) First Ip address of the block must be evenly divisible by the size of the block.

Solution:

Rule-01:

According to Rule-01, all the IP Addresses must be contiguous.
Clearly, all the given IP Addresses are contiguous.
So, Rule-01 is satisfied.

Rule-02:

According to Rule-02, size of the block must be presentable as 2^n.
Number of IP Addresses in given block = 127 – 64 + 1 = 64.
Size of the block = 64 which can be represented as 2^6.
So, Rule-02 is satisfied.

Q.) Consider a block of IP Addresses ranging from 150.10.20.64 to 150.10.20.127.

Is it a CIDR block? If yes, give the CIDR representation.

Rules of CIDR blocks

1.) All IP addresses should be contiguous
2.) Block size should be a power of 2
3.) First Ip address of the block must be evenly divisible by the size of the block.

Rule-03:

According to Rule-03, first IP Address must be divisible by size of the block.
So, 150.10.20.64 must be divisible by 26.
150.10.20.64 = 150.10.20.01000000 is divisible by 26 since its 6 least significant bits are zero.
So, Rule-03 is satisfied.
Since all the rules are satisfied, therefore given block is a CIDR block.
CIDR Representation-
We have-

Size of the block = Total number of IP Addresses = 26.
To have 26 total number of IP Addresses, 6 bits are required in the Host ID part.
So, Number of bits in the Network ID part = 32 – 6 = 26.
Thus,
CIDR Representation = 150.10.20.64 / 26

Q.)Given the CIDR representation 20.10.30.35 / 27. Find the range of IP Addresses in the CIDR block.

20 . 10 . 30 . 35

32 + 2 + 1

4

HID

20 . 10 . 30 . 001 | 00011

NID & BID

00000 — 20.10.30.32

00001 — 20.10.30.33

⋮

11111 — 20.10.30.63

Q.)Given the CIDR representation 20.10.30.35 / 27. Find the range of IP Addresses in the CIDR block.

Solution:
Given CIDR representation is 20.10.30.35 / 27.

It suggests-

27 bits are used for the identification of network.
Remaining 5 bits are used for the identification of hosts in the network.

Given CIDR IP Address may be represented as-

00010100.00001010.00011110.00100011 / 27

So,

First IP Address = 00010100.00001010.00011110.00100000 = 20.10.30.32
Last IP Address = 00010100.00001010.00011110.00111111 = 20.10.30.63

Thus, Range of IP Addresses = [20.10.30.32 , 20.10.30.63]

Q.) Given the CIDR representation 100.1.2.35 / 20. Find the range of IP Addresses in the CIDR block.

100.1.2.35/20

HID

100.1.0000 0010.00100011

0000.00000000

0000.00000001

NID & BID

1111.11111111

100.1.0.0

100.1.15.255

valid

Conti

Size

(iii)

12

100.1.0000 $\boxed{0000.00000000}$

100.1.0.0 ... 100.1.0.255 } 256

100.1.1.0 ... 100.1.1.255 } 256

100.1.2.0 ... 100.1.2.255 } 256

100.1.15.0 ... 100.1.15.255 } 256

256×16

$= 2^8 \times 2^4 = 2^{12}$

(i) Con

(ii) 2^{12}

(iii) $I/2^{12}$

Q.) Given the CIDR representation 100.1.2.35 / 20. Find the range of IP Addresses in the CIDR block.

Solution-

Given CIDR representation is 100.1.2.35 / 20.

It suggests-

20 bits are used for the identification of network.

Remaining 12 bits are used for the identification of hosts in the network.

Given CIDR IP Address may be represented as-

01100100.00000001.00000010.00100011 / 20

So,

First IP Address = 01100100.00000001.00000000.00000000 = 100.1.0.0

Last IP Address = 01100100.00000001.00001111.11111111 = 100.1.15.255

Thus, Range of IP Addresses = [100.1.0.0 , 100.1.15.255]

$\frac{2^7}{2}$ 2^6 20.30.40.10/25 — 1/2, 1/2

20.30.40.0 to 20.30.40.127 $\frac{128}{2} = 64$

20.30.40.0 | 0001010

20.30.40.0 | 0000000

BID

1111111

SID

20.30.40.0 | $\boxed{\ }$ - - - - - -

20.30.40.0 to 20.30.40.63/26

20.30.40.64 to 20.30.40.127/26

$\underbrace{20.30.40.0\ \boxed{0}}_{25+1}$ 000000 … 111111

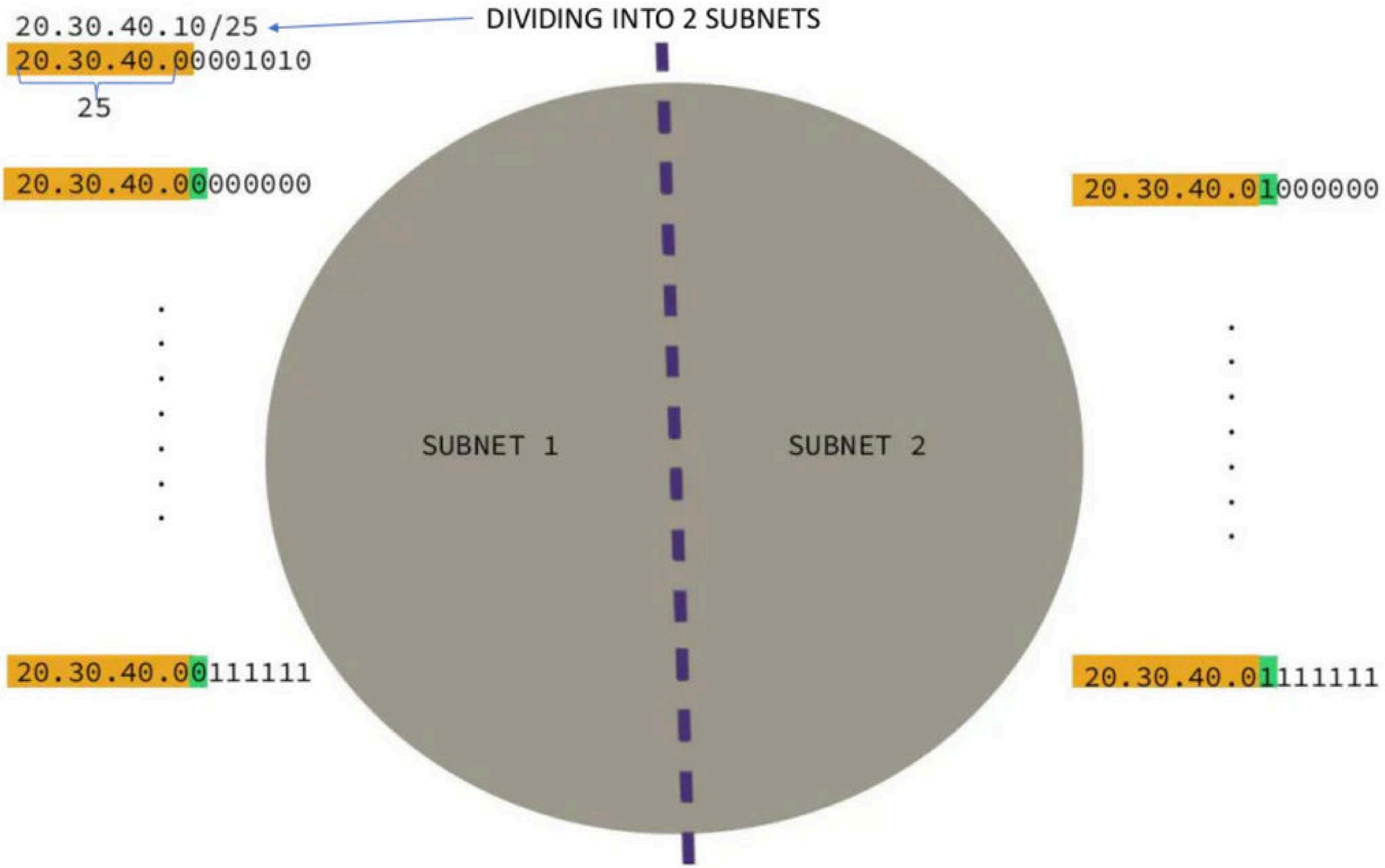
20.30.40.0 $\boxed{1}$ 000000 … 111111

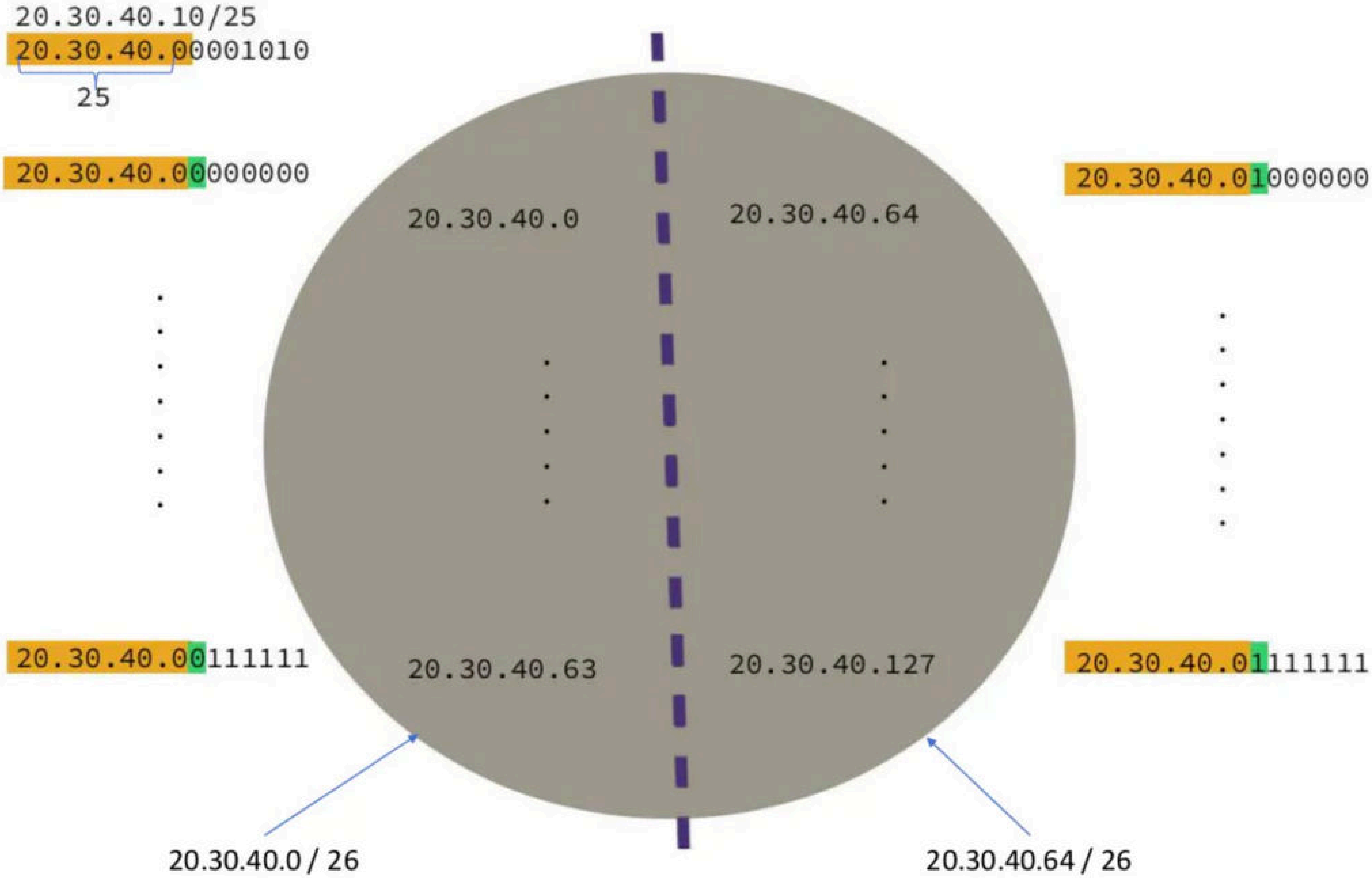
(0-63) (64-127)

Computer Networks

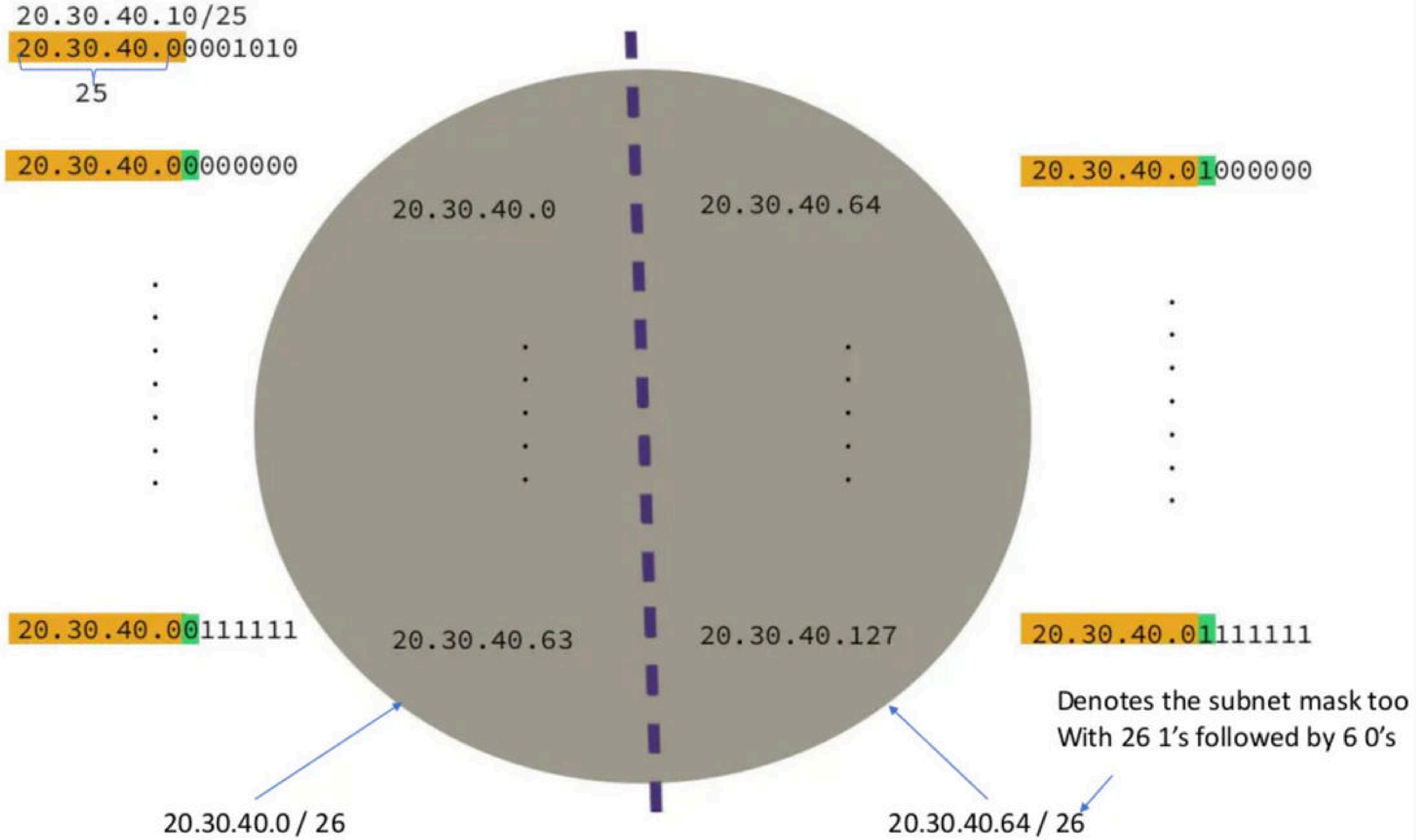
Subnetting in CIDR, VLSM in CIDR

a.b.c.d/n


20.30.40.10/25
DIVIDING INTO 2 SUBNETS
20.30.40.00001010
25
20.30.40.00000000
20.30.40.01000000
SUBNET 1
SUBNET 2
20.30.40.00111111
20.30.40.01111111



20.30.40.10/25
20.30.40.00001010
25
20.30.40.00000000
20.30.40.00111111
20.30.40.0
20.30.40.63
20.30.40.64
20.30.40.127
20.30.40.01000000
20.30.40.01111111
20.30.40.0 / 26
20.30.40.64 / 26



20.30.40.10/25
20.30.40.00001010
25
20.30.40.00000000
20.30.40.0
20.30.40.64
20.30.40.01000000
20.30.40.00111111
20.30.40.63
20.30.40.127
20.30.40.01111111
Denotes the subnet mask too
With 26 1's followed by 6 0's
20.30.40.0 / 26
20.30.40.64 / 26

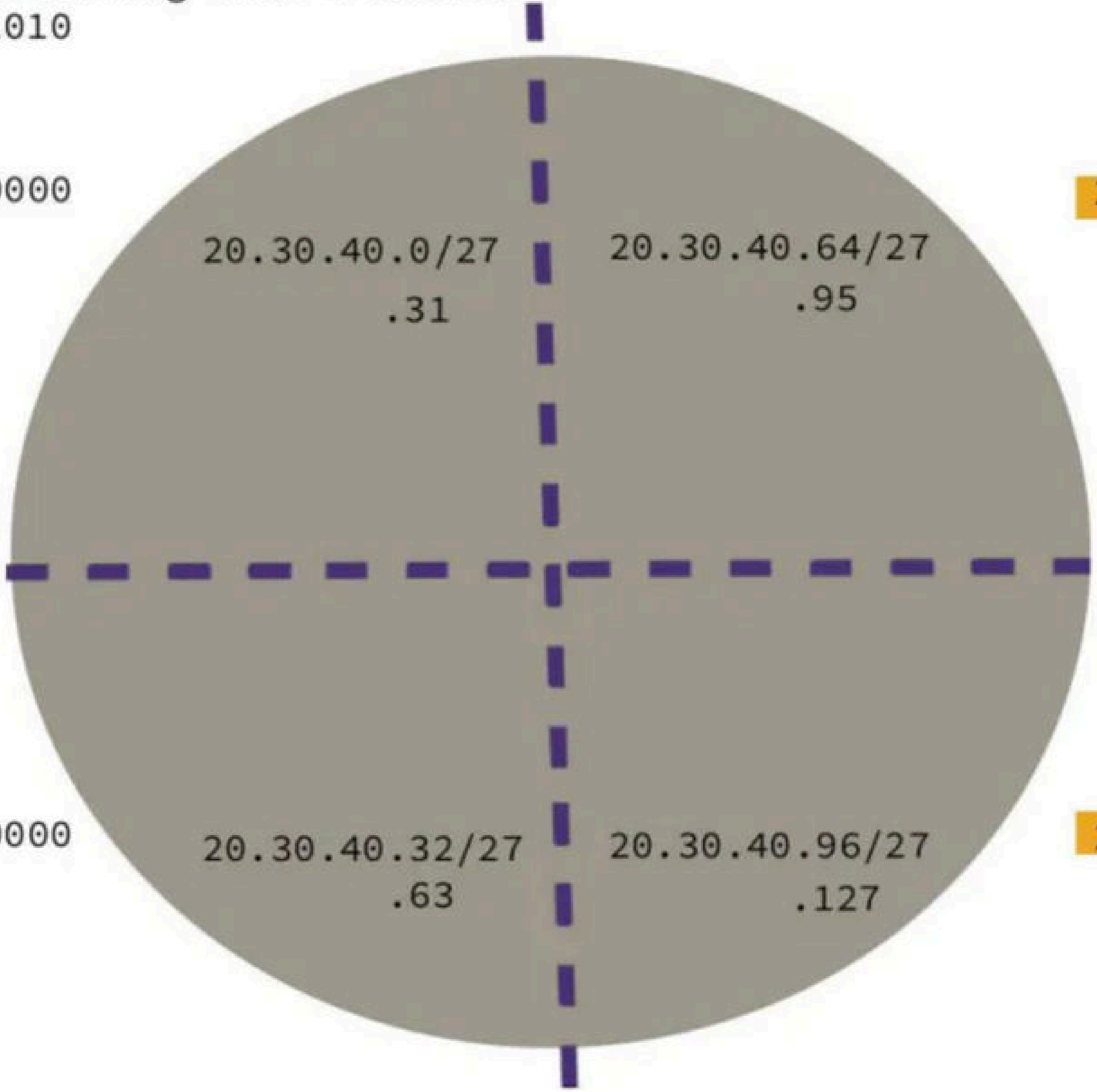
20.30.40.10/25 ←Diving into 4 subnets

20.30.40.00001010

25

20.30.40.00000000

20.30.40.01000000



20.30.40.00100000

20.30.40.01100000

FROM THE PREVIOUS 2 EXAMPLES

DIVING THE NETWORK INTO 2 SUBNETS:

/25 -> 2^7

/26 -> 2^6 /26 -> 2^6

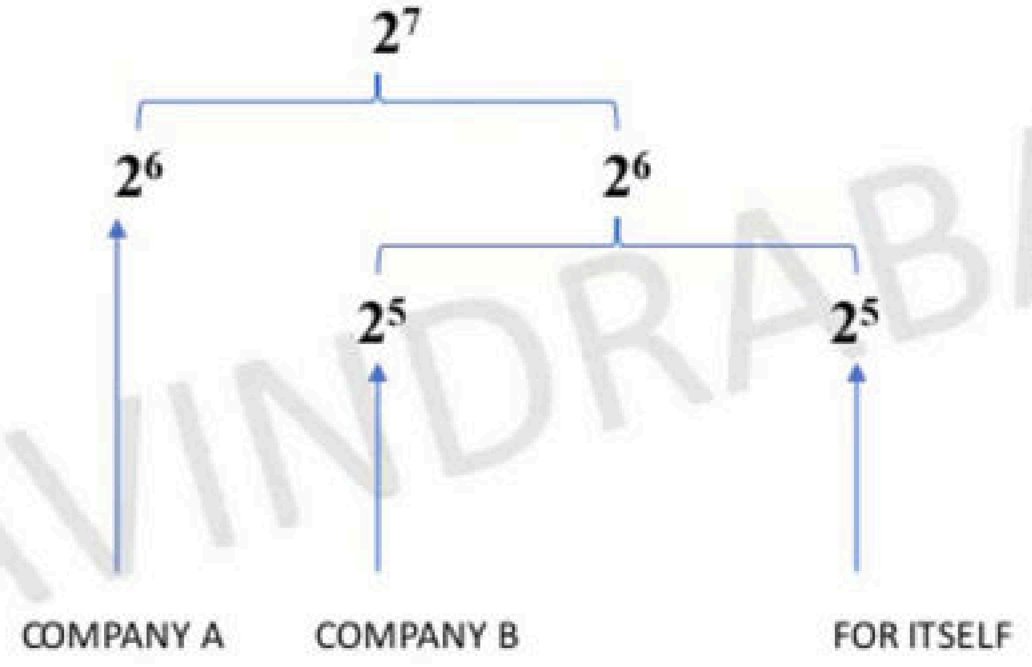
DIVING THE NETWORK INTO 4 SUBNETS:

/25 -> 2^7

/27 -> 2^5 /27 -> 2^5 /27 -> 2^5 /27 -> 2^5

VLSM IN CIDR BLOCKS

LET US ASSUME A SCENARIO IN WHICH AN ISP HAS A BLOCK 20.30.40.10/25 AND THEY WISH TO DIVIDE IT INTO 3 SUBNETS IN A FOLLOWING WAY:



20.30.40.10/25 Diving into 3 subnets

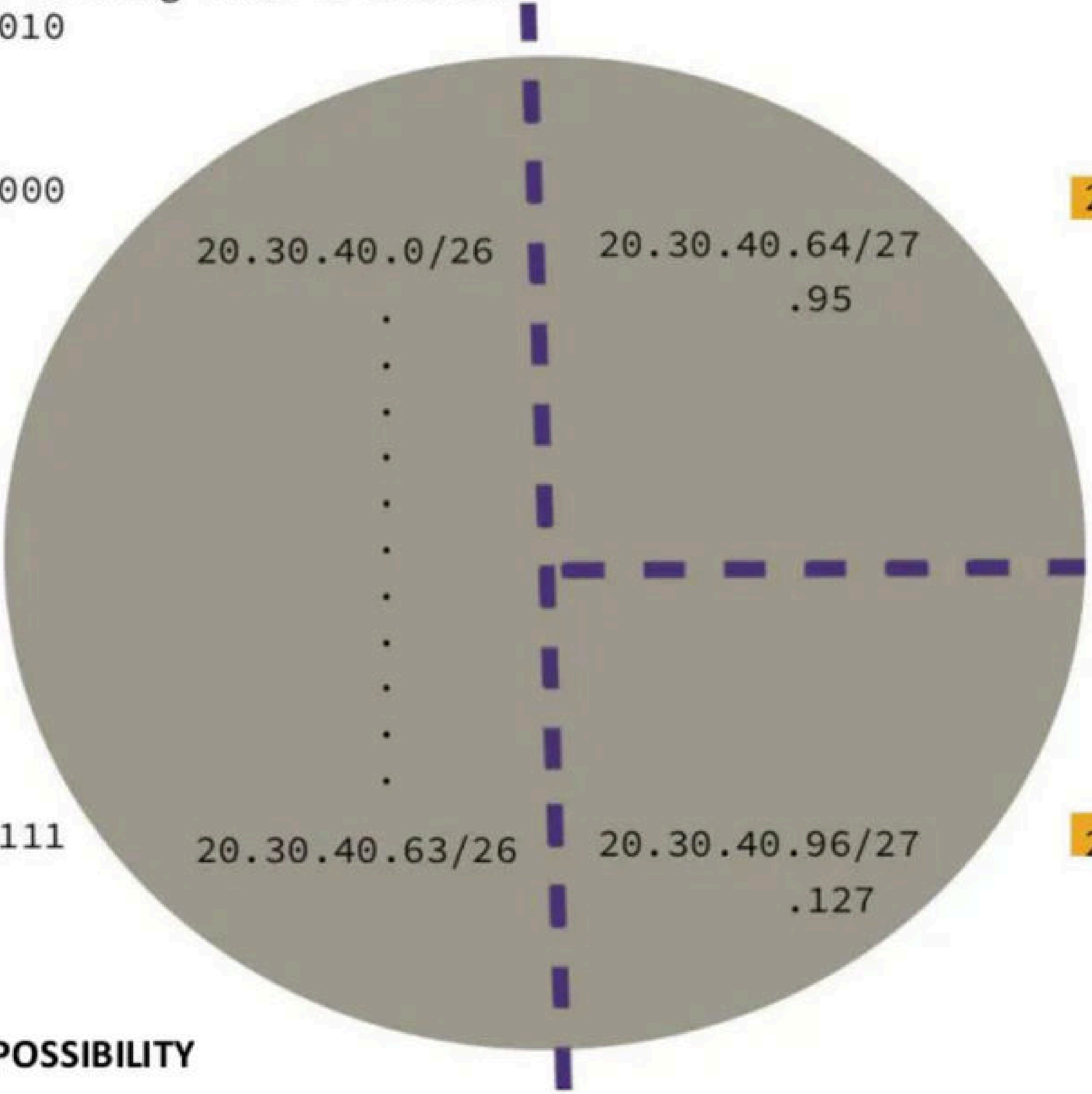
20.30.40.00001010

25

20.30.40.00000000

.
.
.

20.30.40.00111111

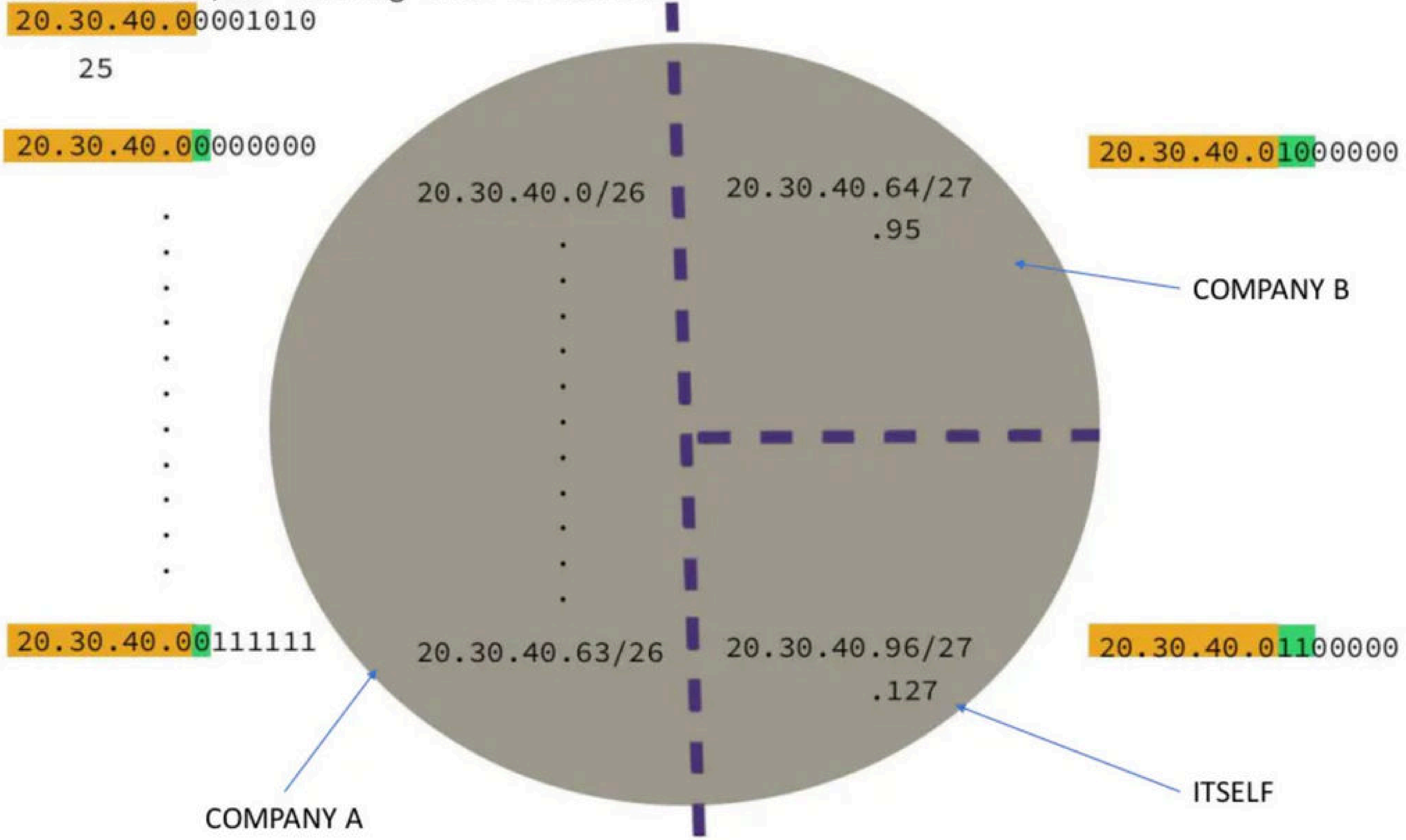


20.30.40.01000000

20.30.40.01100000

THIS COULD BE ONE POSSIBILITY

20.30.40.10/25 Diving into 3 subnets



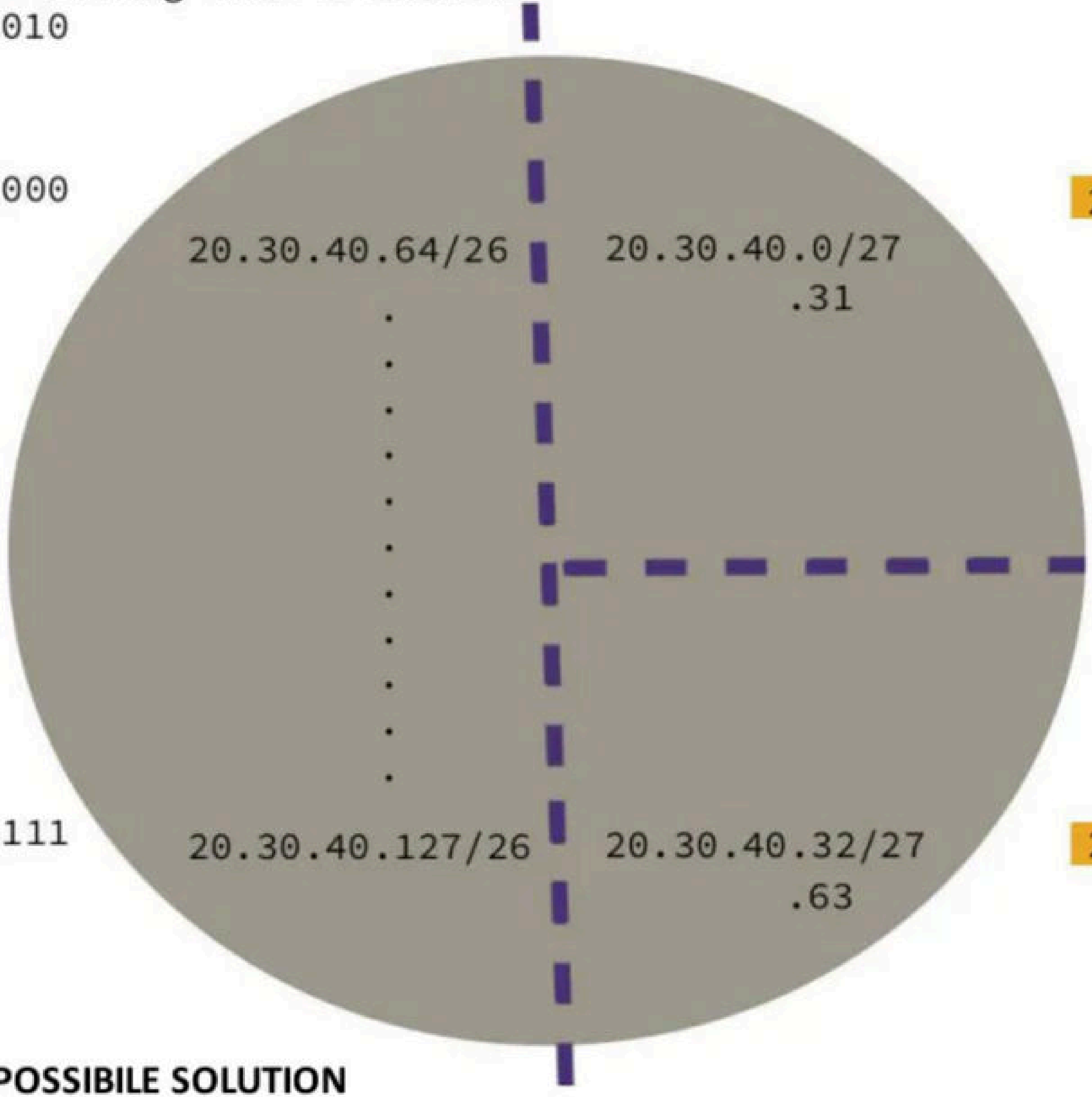
20.30.40.10/25 ← Diving into 3 subnets

20.30.40.00001010

25

20.30.40.01000000

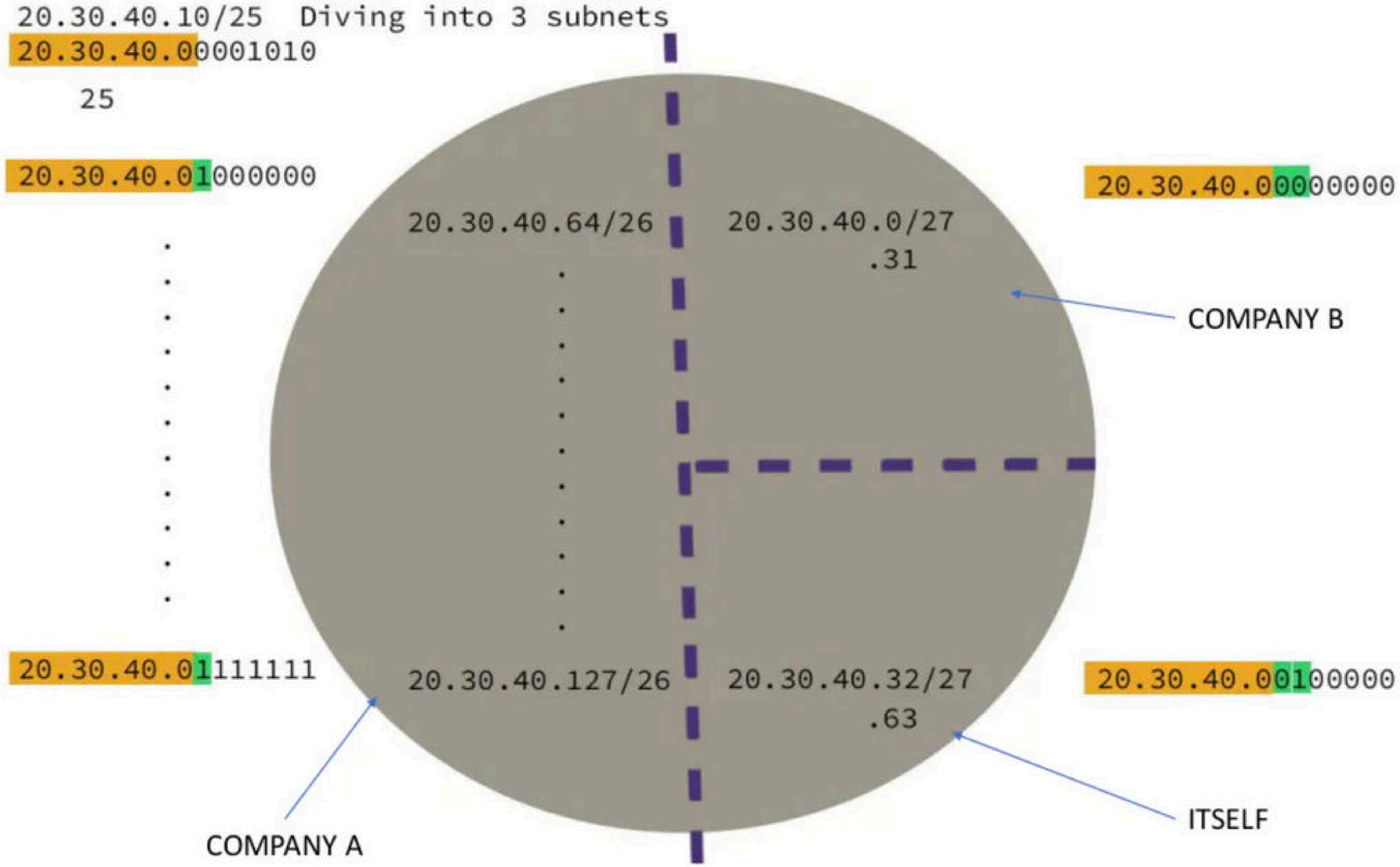
20.30.40.01111111



20.30.40.00000000

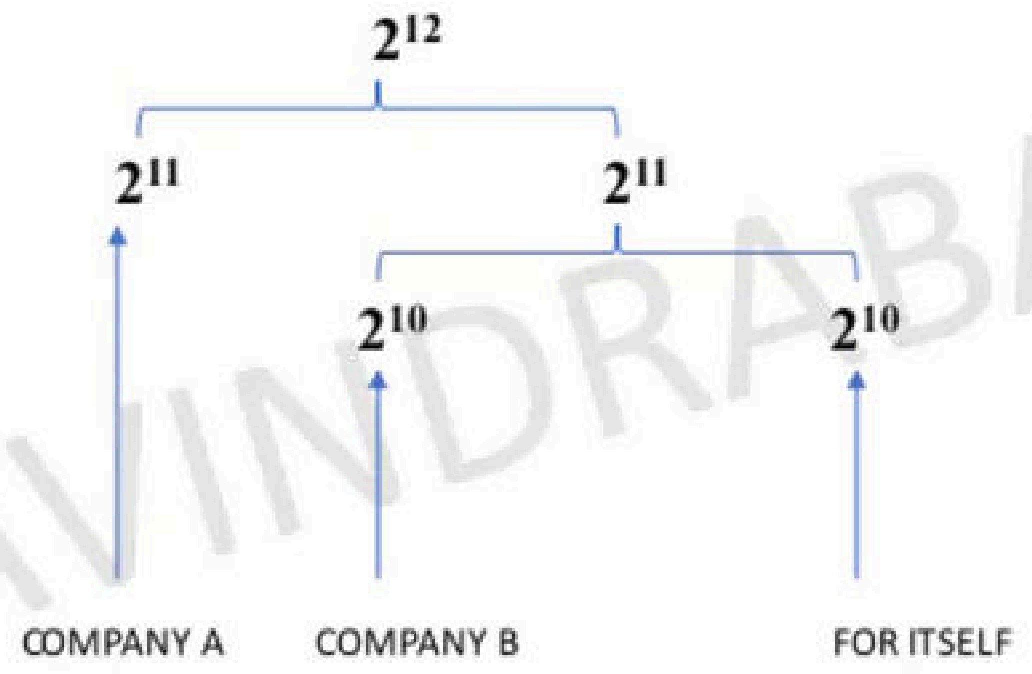
20.30.40.00100000

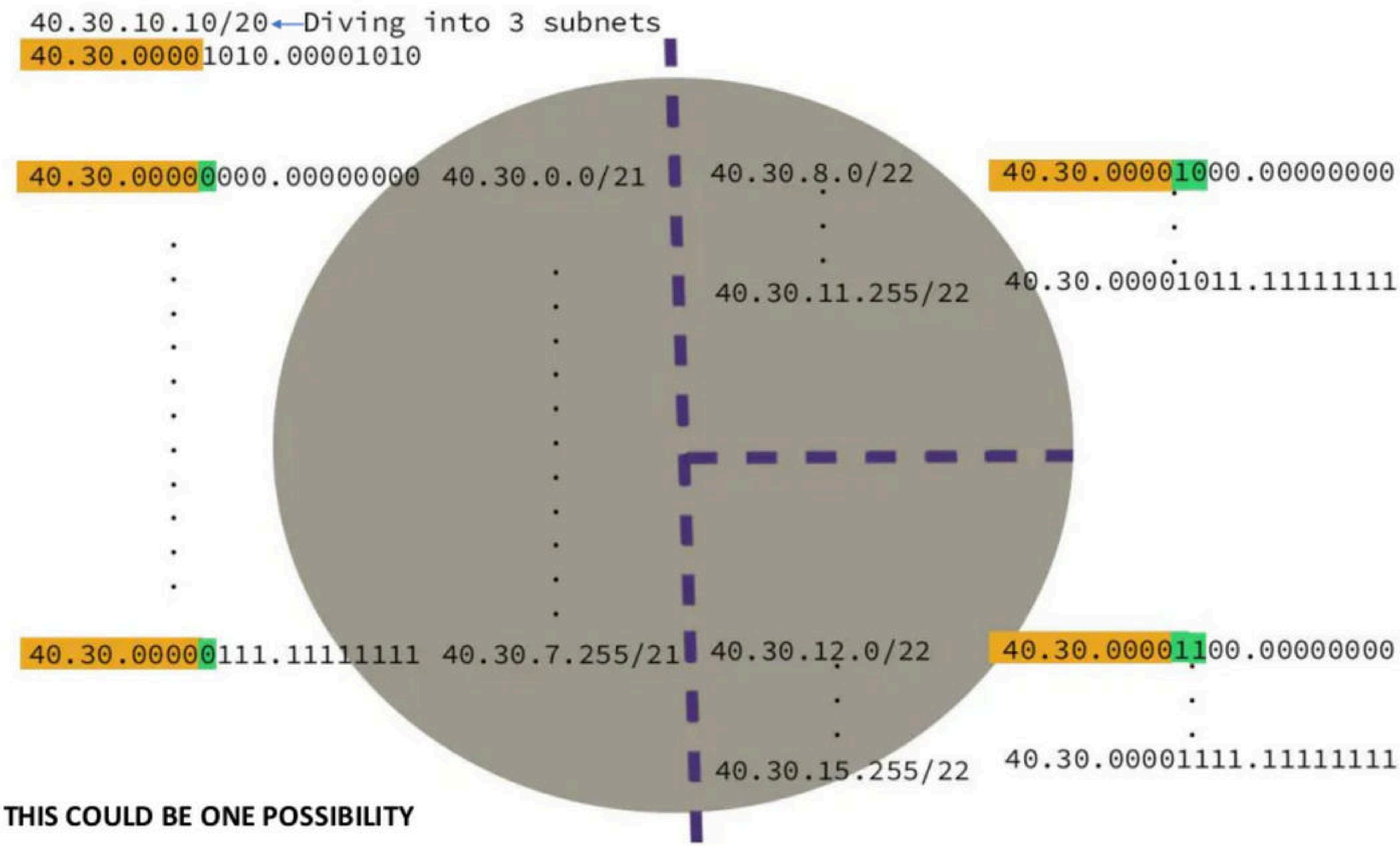
THIS COULD BE ANOTHER POSSIBILE SOLUTION

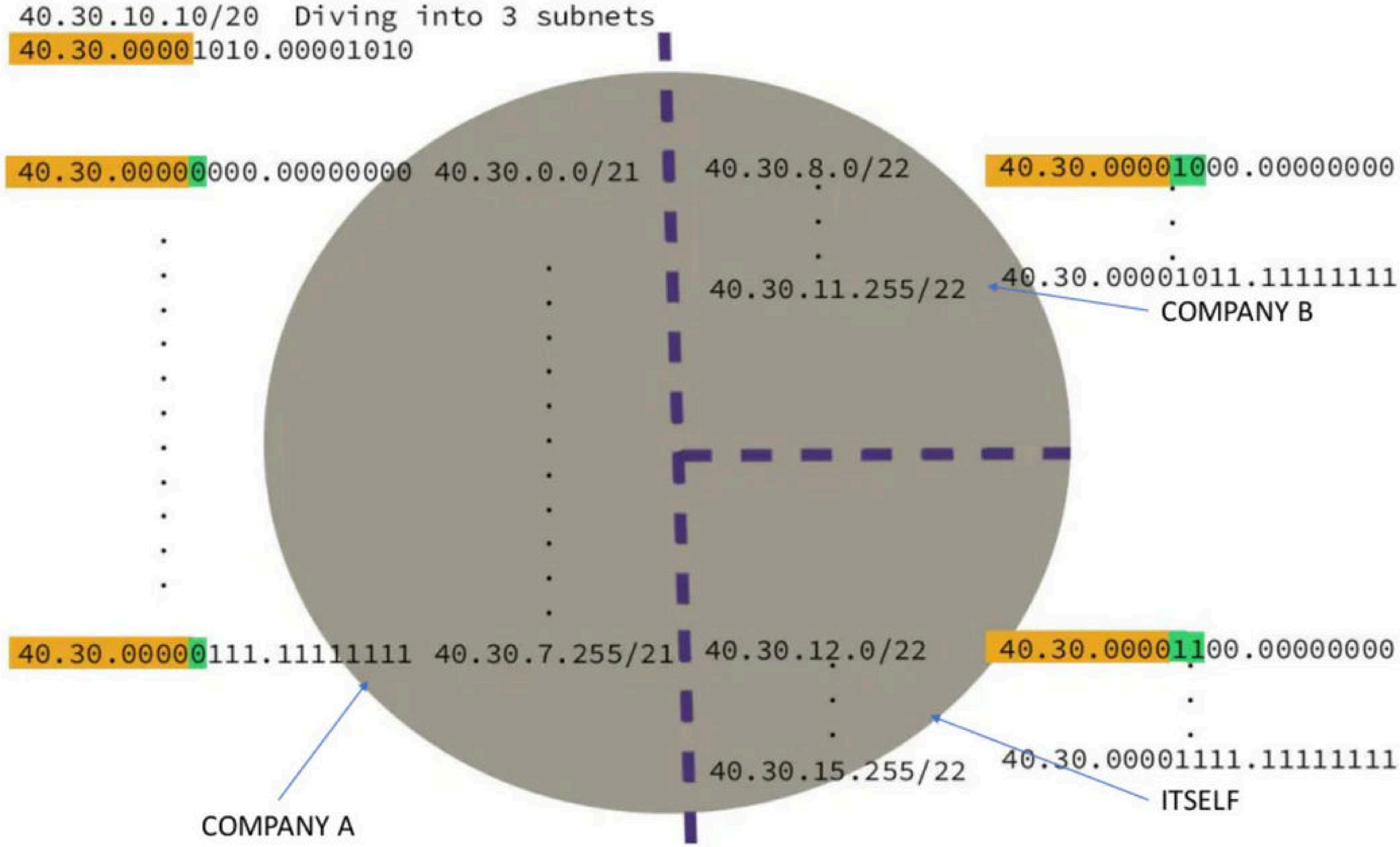

20.30.40.10/25 Diving into 3 subnets
20.30.40.00001010
25
20.30.40.01000000
20.30.40.01111111
20.30.40.64/26
20.30.40.127/26
20.30.40.0/27
.31
20.30.40.32/27
.63
20.30.40.00000000
20.30.40.00100000
COMPANY B
ITSELF
COMPANY A

LET'S CHECK OUT ONE MORE EXAMPLE

LET US ASSUME A SCENARIO IN WHICH AN ISP HAS A BLOCK 40.30.10.10/20 AND THEY WISH TO DIVIDE IT INTO 3 SUBNETS IN A FOLLOWING WAY:

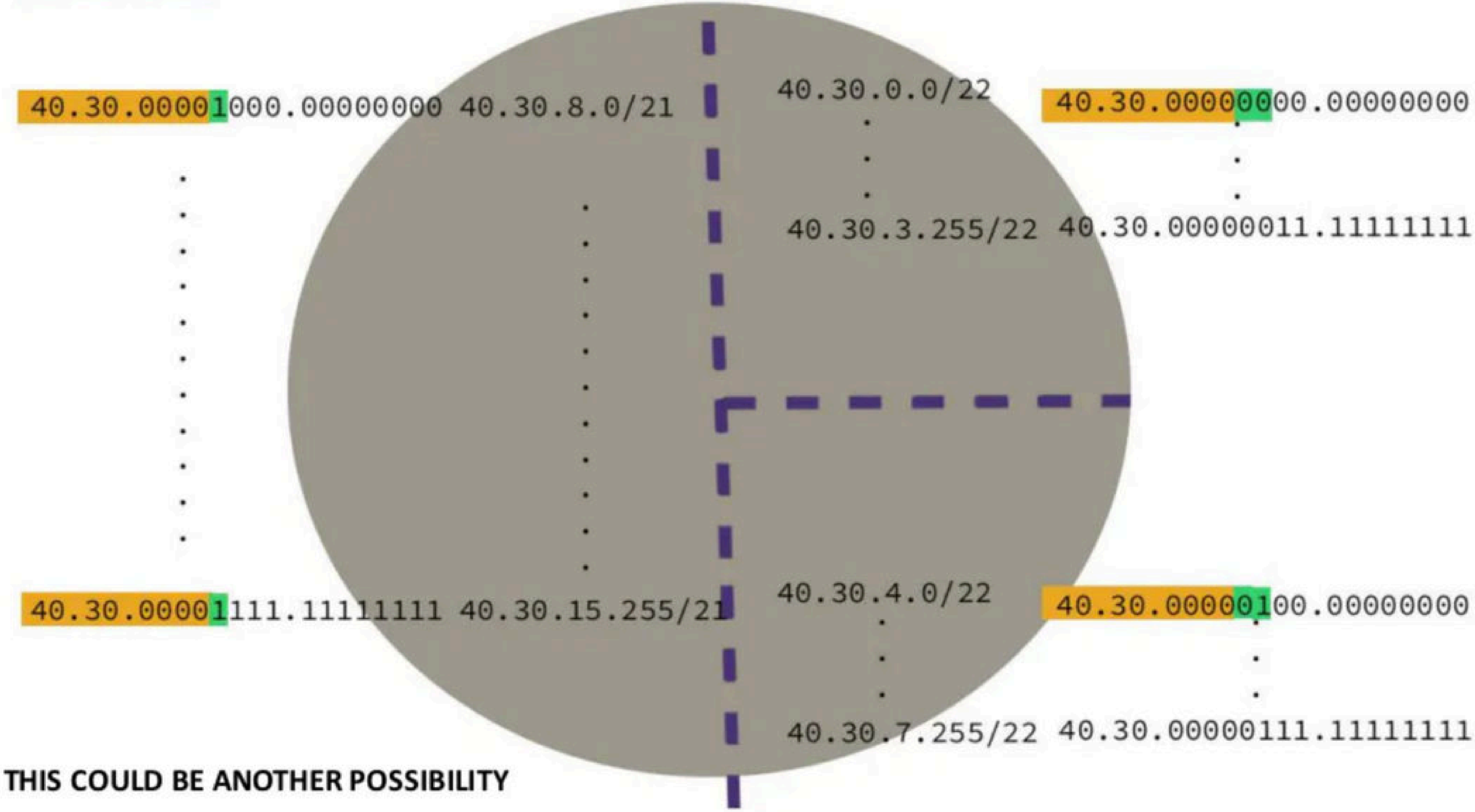



40.30.10.10/20 ← Diving into 3 subnets
40.30.00001010.00001010
40.30.00000000.00000000 40.30.0.0/21
40.30.00000111.11111111 40.30.7.255/21
40.30.8.0/22
40.30.11.255/22
40.30.00001000.00000000
40.30.00001011.11111111
40.30.12.0/22
40.30.15.255/22
40.30.00001100.00000000
40.30.00001111.11111111
THIS COULD BE ONE POSSIBILITY


40.30.10.10/20 Diving into 3 subnets
40.30.00001010.00001010
40.30.00000000.00000000 40.30.0.0/21
40.30.00000111.11111111 40.30.7.255/21
COMPANY A
40.30.8.0/22
40.30.11.255/22
40.30.00001000.00000000
40.30.00001011.11111111
COMPANY B
40.30.12.0/22
40.30.15.255/22
40.30.00001100.00000000
40.30.00001111.11111111
ITSELF

40.30.10.10/20 ← Diving into 3 subnets

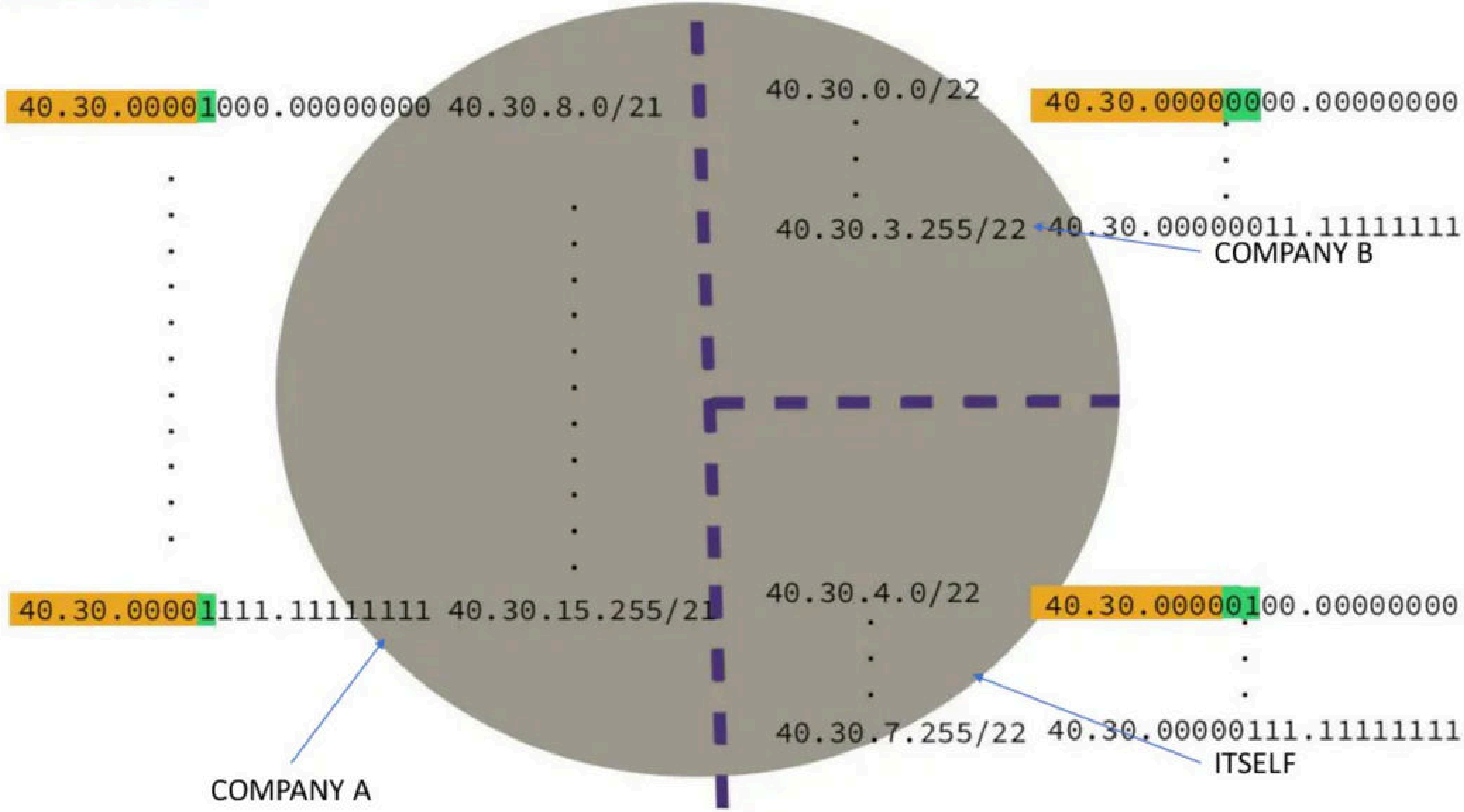
40.30.00001010.00001010



THIS COULD BE ANOTHER POSSIBILITY

40.30.10.10/20 Diving into 3 subnets

40.30.00001010.00001010



Computer Networks

Supernetting OR Aggregation

In subnetting, a single network is divided into multiple smaller subnetworks.

In Supernetting/Aggregation, multiple networks are combined into a bigger network termed as a Super network or Supernet.

Rules for Aggregation :

1. All the Networks should be contiguous.
2. The block size of every networks should be equal and must be in form of 2^n.
3. First Network id should be exactly divisible by whole size of supernet.

Points to be checked before aggregation :

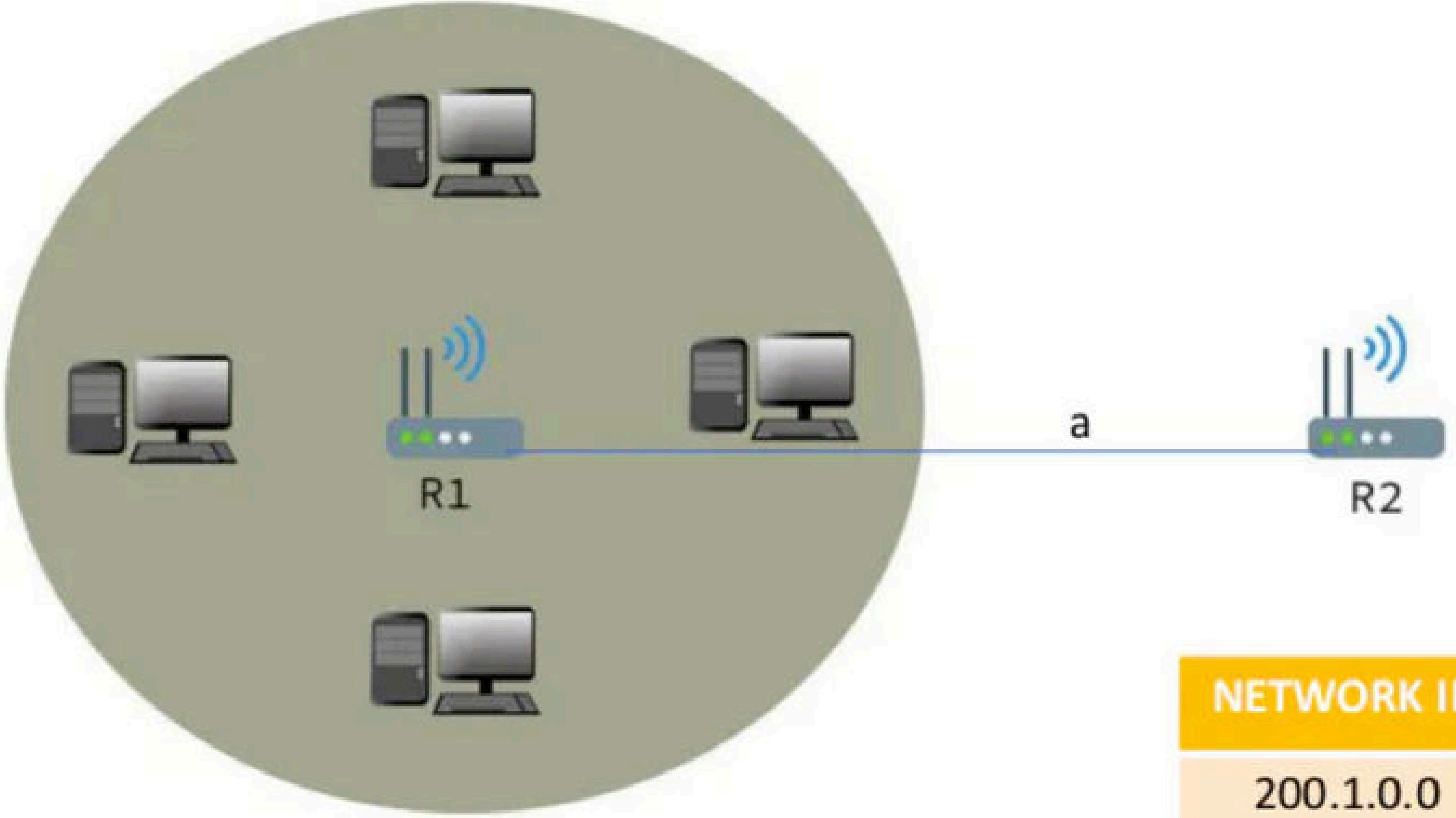
1. All the Networks should be contiguous.
2. The block size of every networks should be equal and must be in form of 2^n.
3. First Network id should be exactly divisible by whole size of supernet.

Example:

200.1.0.0 / 24
200.1.1.0 / 24
200.1.2.0 / 24
200.1.3.0 / 24

Total size of the supernet = $4 \times 2^8 = 2^{10}$
Diving the First Network id with 2^{10}
Has 0 in last 10 bits. Therefore Divisible.

All are satisfied, hence we can move forward with aggregation.



Instead of having 4 entries
We want just one entry in R2
having a single supernet id

ROUTING TABLE AT R2

NETWORK ID	SUBNET MASK	INTERFACE
200.1.0.0	255.255.255.0	a
200.1.1.0	255.255.255.0	a
200.1.2.0	255.255.255.0	a
200.1.3.0	255.255.255.0	a

Finding the Supernet mask:

It is 32 bit number

In Supernetting, the Number of 1's represent the Fixed part
And Number of 0's represent the Variable part

Fixed	Variable
200.1.000000	00.00000000
200.1.000000	01.00000000
200.1.000000	10.00000000
200.1.000000	11.00000000

Replacing the fixed part with 1's and variable with 0's
We get the supernet mask,

$\boxed{255.255.252.0}$

Finding the Supernet mask:

It is 32 bit number

In Supernetting, the Number of 1's represent the Fixed part
And Number of 0's represent the Variable part

Fixed	Variable
200.1.000000	00.00000000
200.1.000000	01.00000000
200.1.000000	10.00000000
200.1.000000	11.00000000

Replacing the fixed part with 1's and variable with 0's
We get the supernet mask,

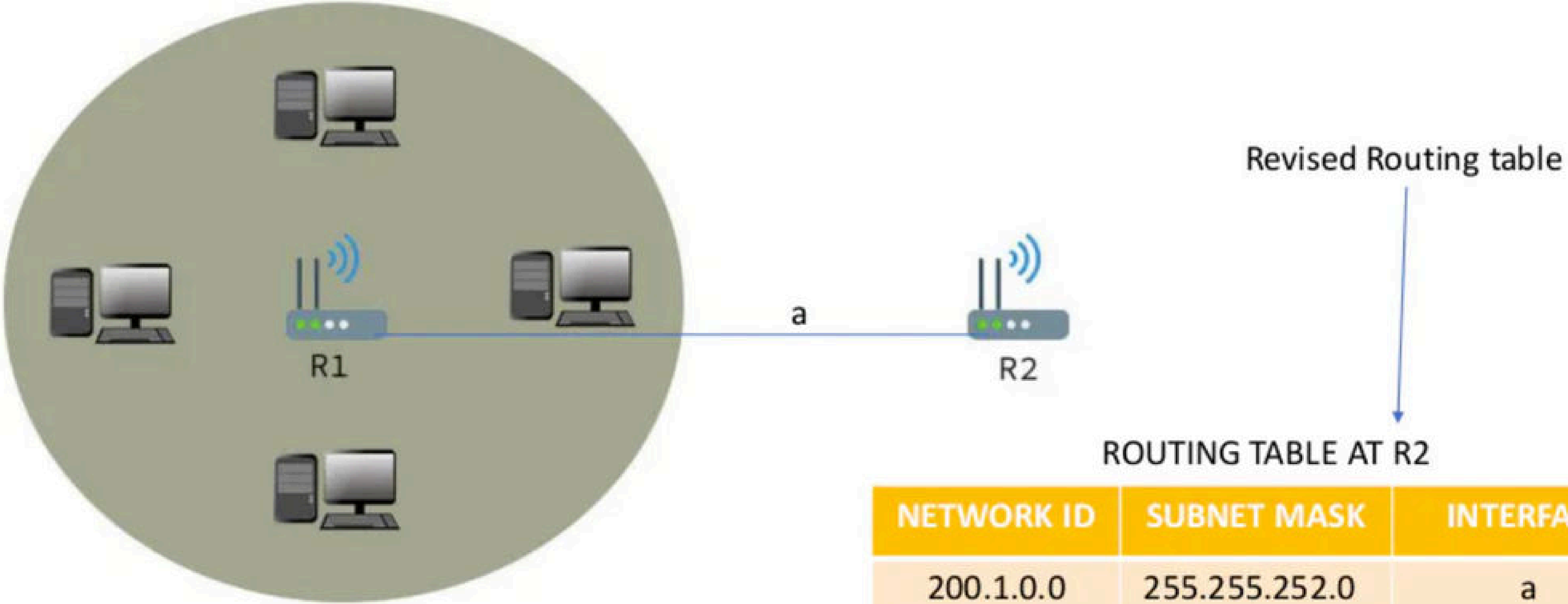
255.255.252.0

SHORTCUT TO FIND THE SUPERNET ID AND SUPERNET MASK

If the network id's follow the Rules then,
The first IP address is always the Supernet id.

Add the network size of all the networks given
The total denotes the host if part of the subnet mask .
From the example:
Network size of all = $2^8 + 2^8 + 2^8 + 2^8 = 2^{10}$
Therefore the subnet mask contains
10 bits in host id part.
And the network id part will contain 22 bits.

That is what is the subnet mask



ROUTING TABLE AT R2

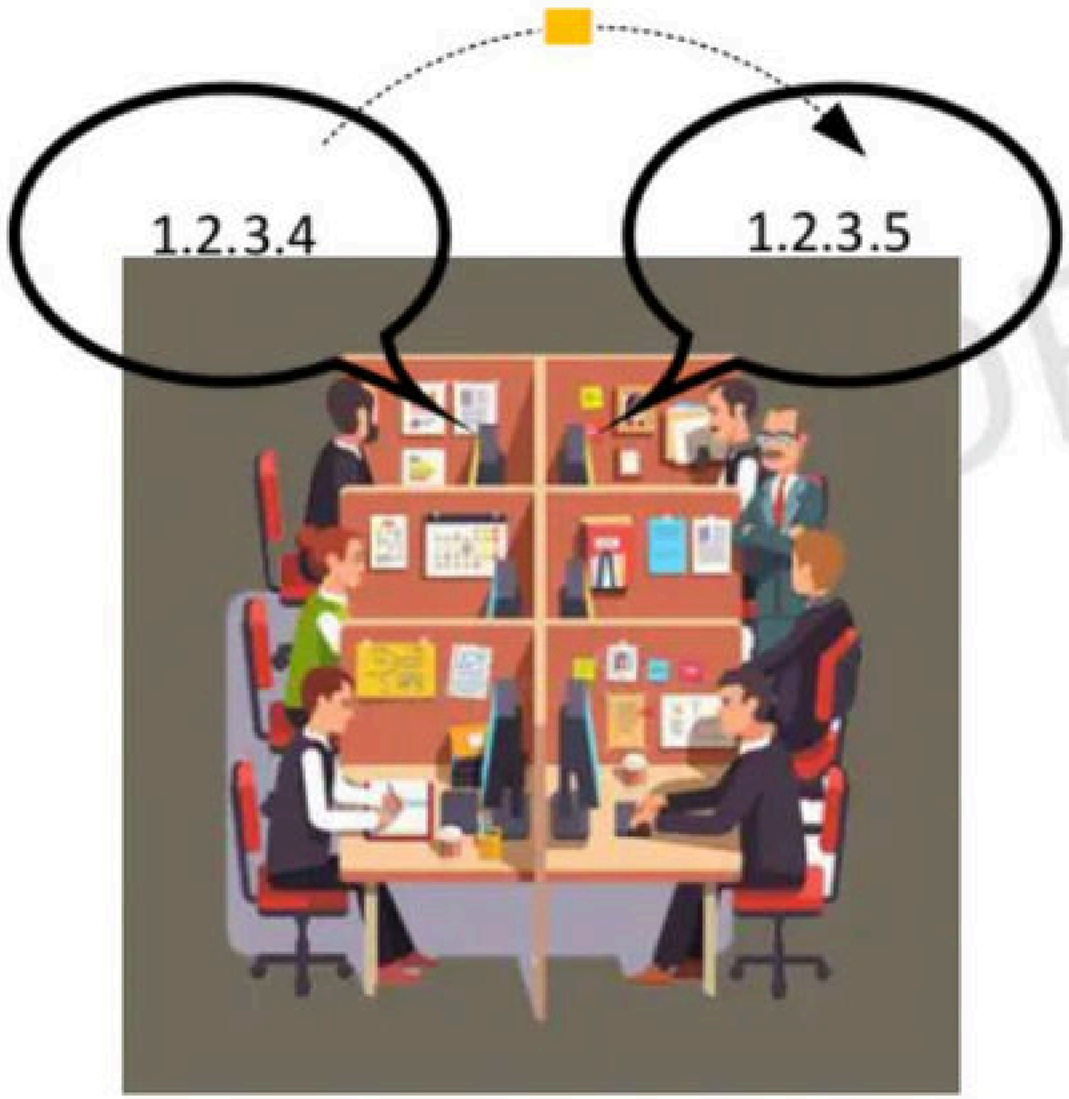
NETWORK ID	SUBNET MASK	INTERFACE
200.1.0.0	255.255.252.0	a

Note : In case of R1, we need all the 4 entries

Computer Networks

Private IP addresses

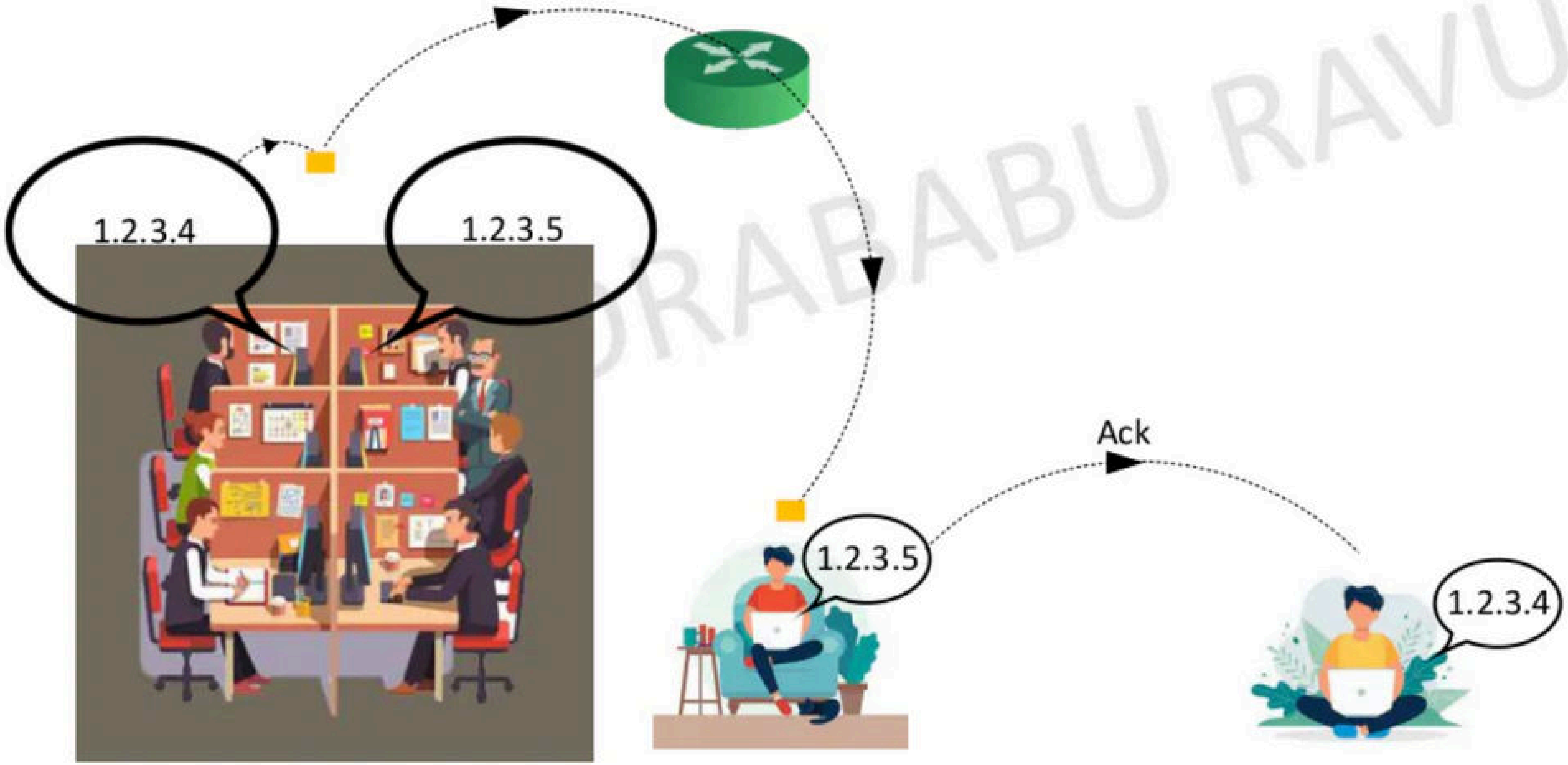
Imagine there is a software company, in which there are 6 PCs and the company has 2 criteria:
1.)No internet access / No communication with the outside world
2.)All the employees can communicate between each other



For this you can set a TCP/IP network and
Assign any IP address to the hosts

This allocation remains safe only till the packet is sent within the network

Suppose internet was accessed and the packet sent
from 1.2.3.4 To 1.2.3.5 escaped onto the internet



IANA has come to a solution to the problem and introduced **Private IP address**
So if one wishes to use Ips only within the Private Network, Private IPs should be assigned and not the Public IPs

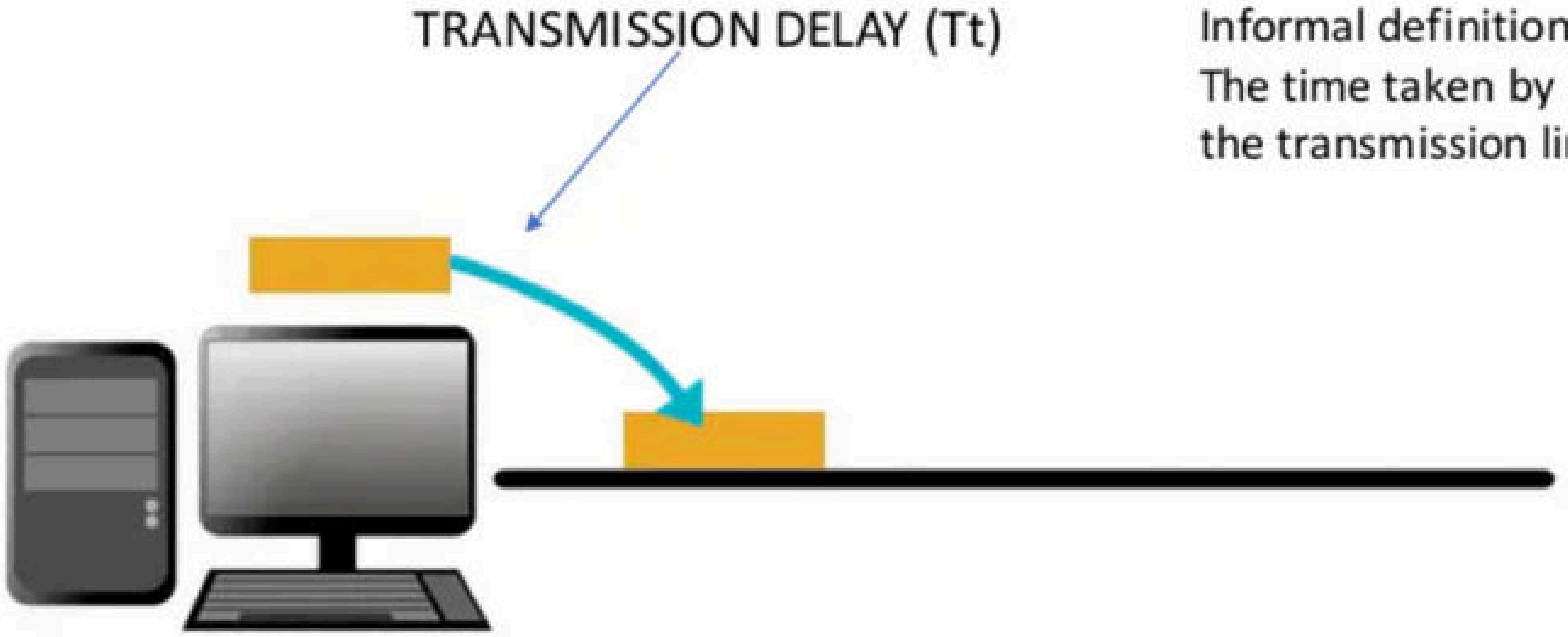
PRIVATE IP RANGE

- 10.0.0.0 to 10.255.255.255
- 172.16.0.0 to 172.31.255.255
- 192.168.0.0 to 192.168.255.255

So, due to this what happens is even if the packet goes outside the private network the router discards it because It was send from a Private IP.

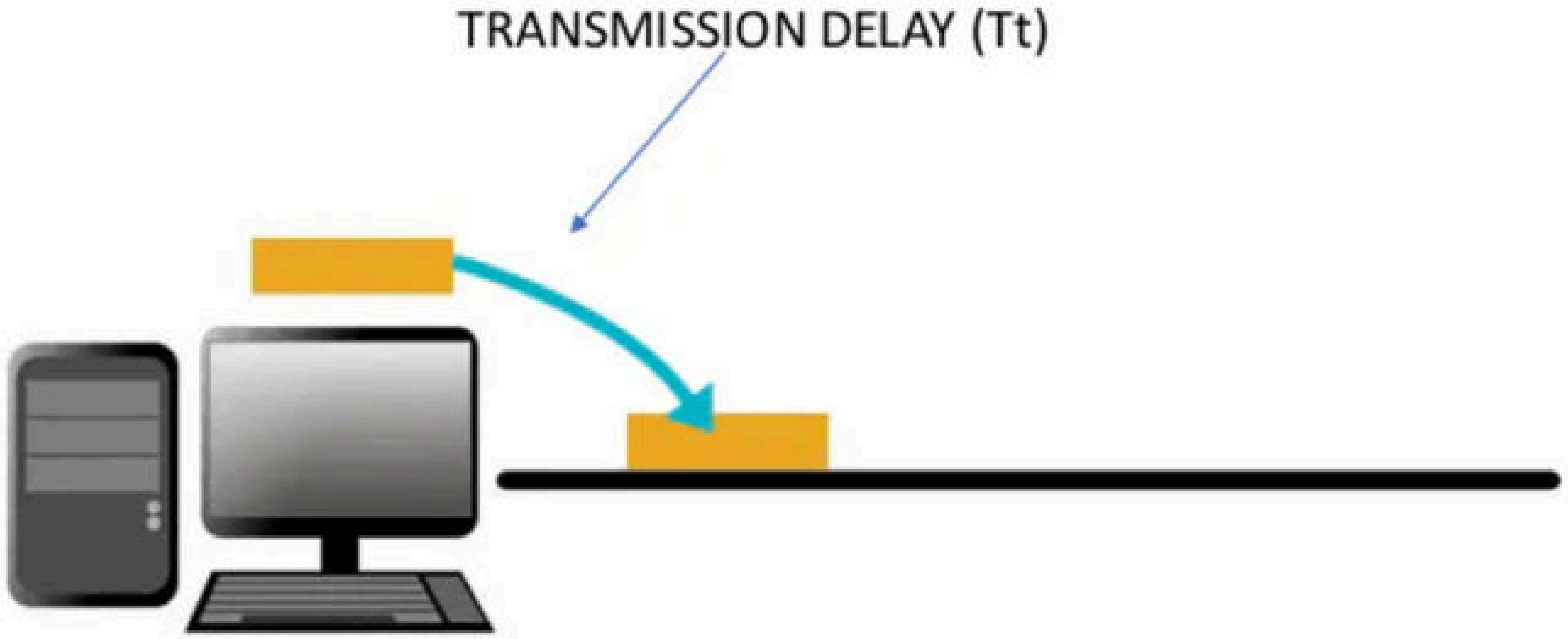
Computer Networks

Delays in Computer Networks



Informal definition:

The time taken by the host to put the data packet into the transmission link is called Transmission Delay



If Bandwidth is 1 bps
And data to be sent is 10 bits
The Tt would be 10 sec

So, in general we can say that,

$$Tt = \frac{\text{Length of data (bits)}}{\text{Bandwidth(bps)}} = \frac{L}{B} \text{ sec}$$

Consider these 2 examples

Example 1 : If L = 1000 bits and BW = 1 Kbps
Then

$$Tt = \frac{L}{B} = \frac{1000}{1000} = 1 \text{ sec}$$

Example 2 : If L = 1 KB and BW = 1 Kbps
Then

$$Tt = \frac{L}{B} = \frac{1024}{1000} = 1.024 \text{ sec}$$

Consider these 2 examples

Example 1 : If L = 1000 bits and BW = 1 Kbps
Then

$$Tt = \frac{L}{B} = \frac{1000}{1000} = 1 \text{ sec}$$

Example 2 : If L = 1 KB and BW = 1 Kbps
Then

$$Tt = \frac{L}{B} = \frac{1024}{1000} = 1.024 \text{ sec}$$

REMEMBER

Whenever data has **K**,
K= 1024

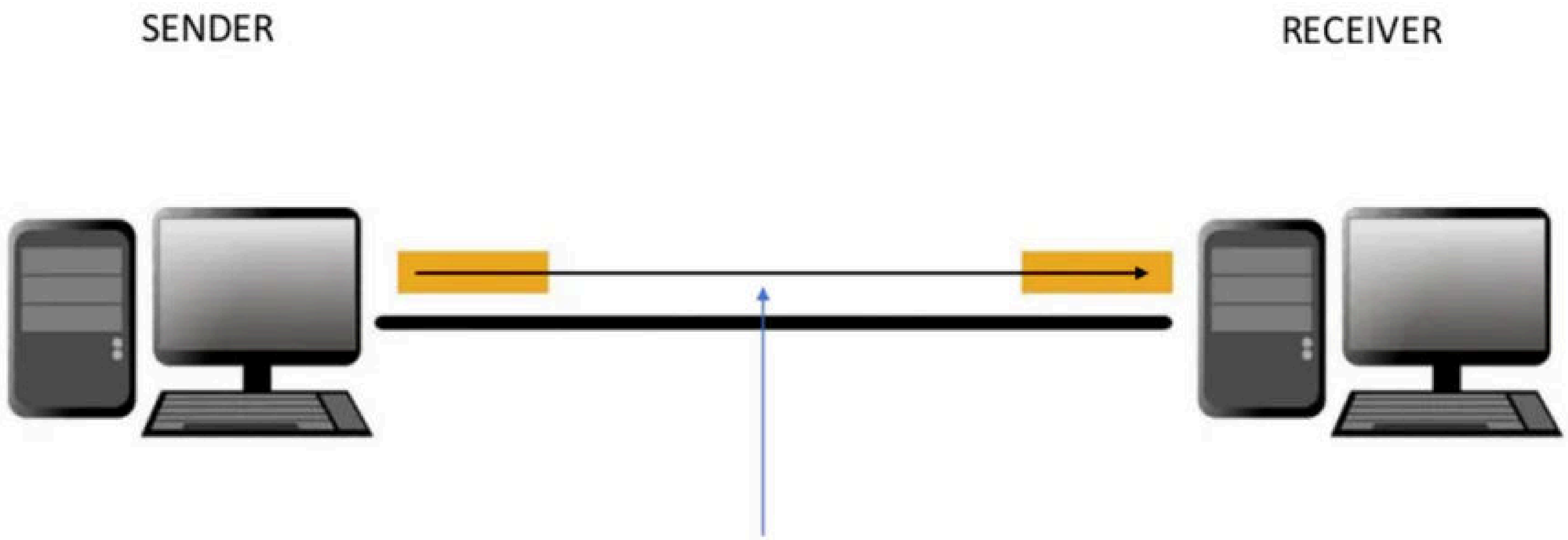
Whenever data has **M**,
M=1024 X 1024

Whenever data has **G**,
G=1024 X 1024 X 1024

Whenever Bandwidth has **K**,
K= 1000 = 10^3

Whenever Bandwidth has **M**,
M= 10^6

Whenever Bandwidth has **G**,
G= 10^9



PROPAGATION DELAY (Tp)

Time taken for one bit to travel from sender to receiver end of the link is called as propagation delay.

$$T_p = \frac{\text{distance of the link (d)}}{\text{Velocity(v)}} \text{ seconds}$$

Generally the link nowadays is optical fibre and
We know that the speed at which the light travels
is 3×10^8 m/s.

In optical fibre the light travels with 70% of the original speed of light(3×10^8 m/s)
=>$v = 0.7 \times 3 \times 10^8$ m/s = $\underline{2.1 \times 10^8 \text{ m/s}}$

Generally the velocity will be given in the question but, in questions where velocity is not given, we need to take it as 2.1×10^8 m/s

Example: d = 2.1 km and v = 2.1×10^8 m/s

$$Tp = \frac{\text{distance of the link (d)}}{\text{Velocity(v)}}$$

$$Tp = \frac{2.1}{2.1 \times 10^8}$$

$$= \frac{2.1 \times 1000}{2.1 \times 10^8}$$

$$Tp = 10^{-5} \text{ sec}$$

Converting 10^{-5} sec to μsec

We know that μ is 10^{-6}

Step 1 :Divide and multiply by 10^{-6}

$$10^{-5} \times \frac{10^{-6}}{10^{-6}}$$

Step 2 :Combine

$$10^{-5} \times \frac{10^{-6}}{10^{-6}} = 1 \times 10^{-6}$$

= 1 μsec

Time taken by the processor to process the data packet is called as **processing delay**

Time spent by the data packet waiting in the queue before it is taken for execution is called as **queuing delay**.

NOTE :THESE TWO DELAYS DEPEND UPON THE TYPE OF PROCESSOR USED AND CANNOT BE EVALUATED AND THEREFORE ARE CONSIDERED AS ZERO IN ALMOST ALL NUMERICALS