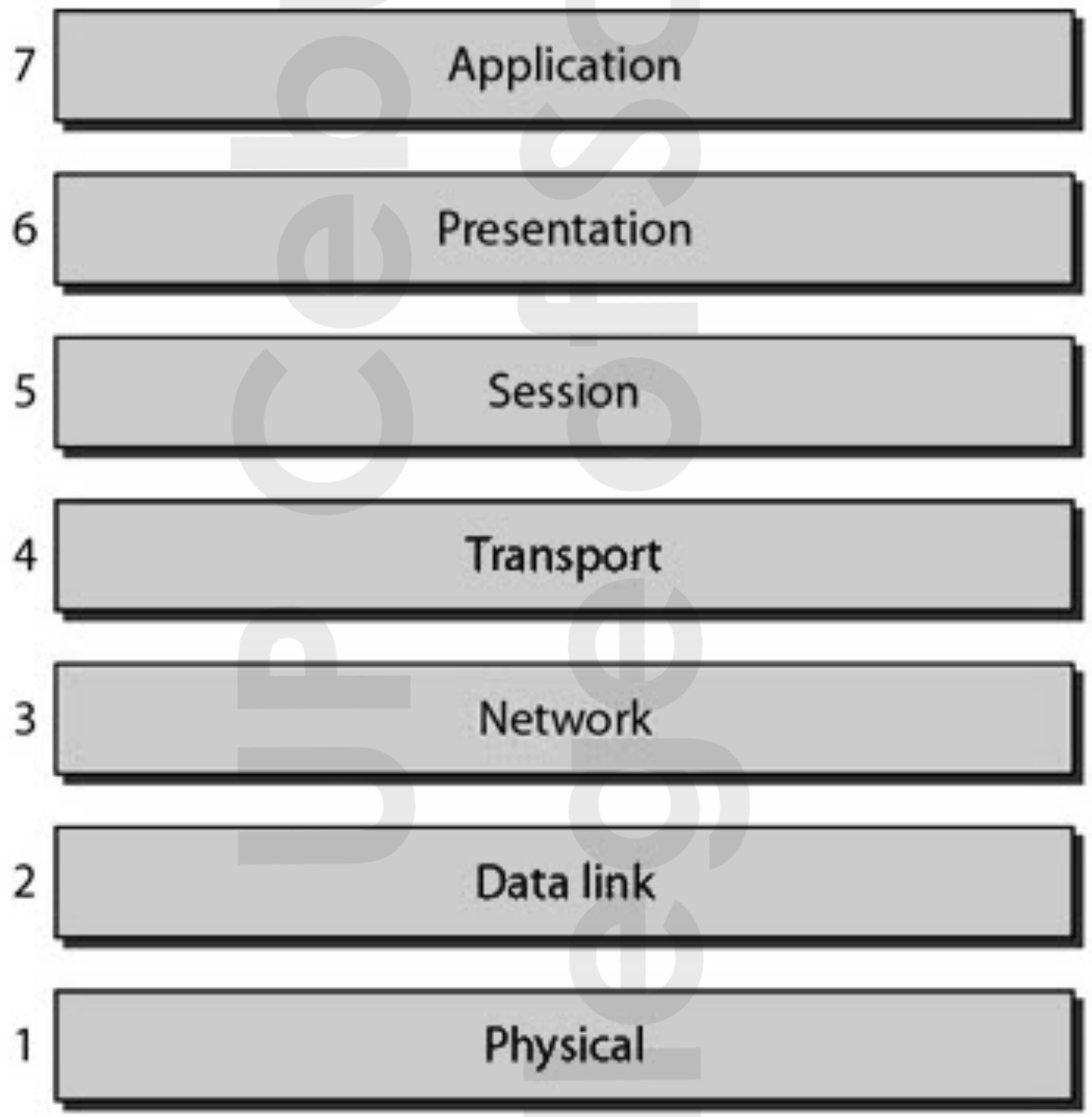
The Network Layer

Prof. Dhong Fhel K. Gom-os

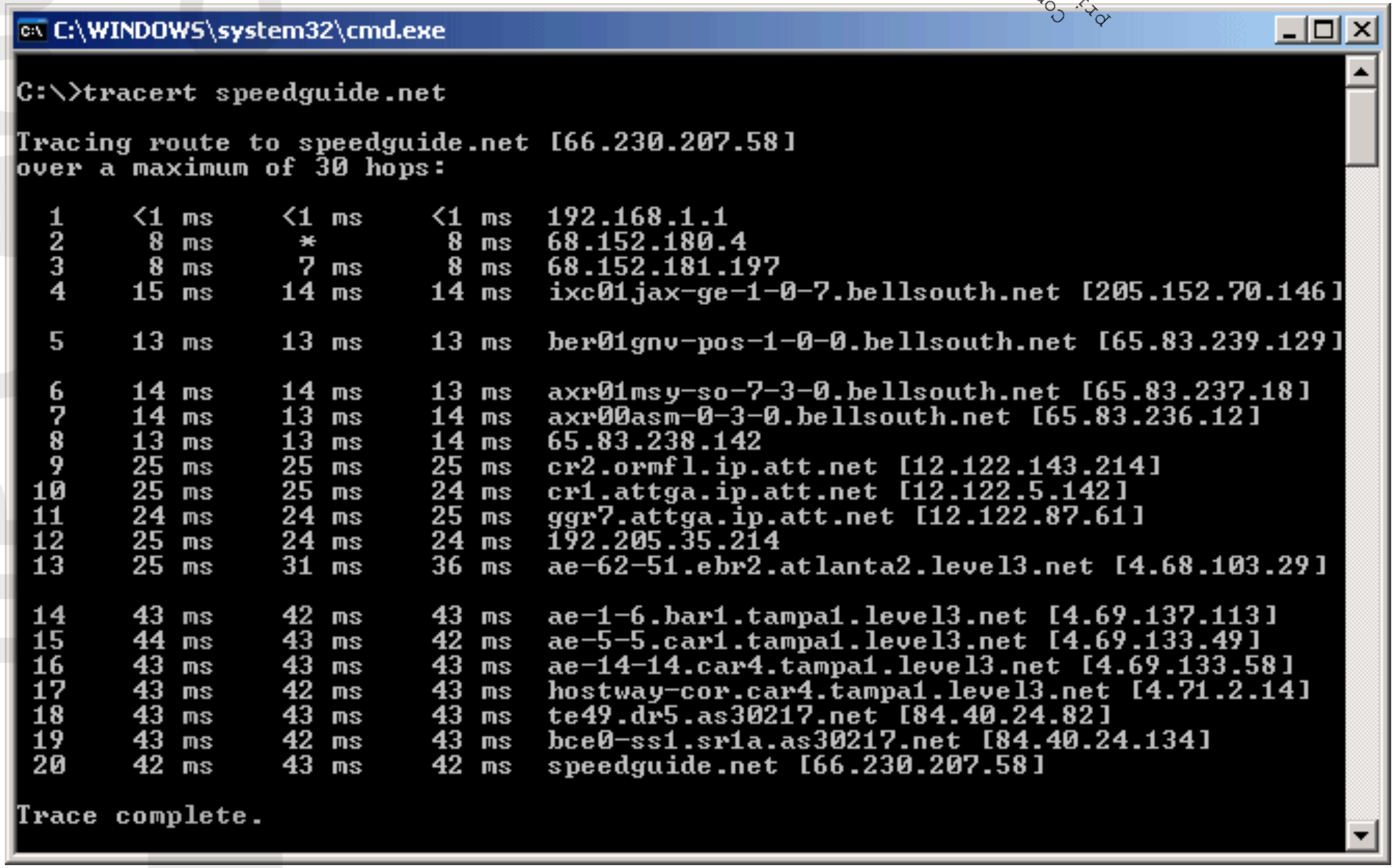

7 Application
6 Presentation
5 Session
4 Transport
3 Network
2 Data link
1 Physical

Specific Responsibilities of the Data Link Layer

1. **Framing** - It divides the stream of bits received from the network layer into manageable data units called *frames*.
2. **Addressing** - It adds a header to the frame to define the addresses of the sender and receiver of the frame.
3. **Flow Control** - If the rate at which the data are absorbed by the receiver is less than the rate at which data are produced in the sender, the data link layer imposes a flow control mechanism to avoid overwhelming the receiver.
4. **Error Control** - It adds reliability to the physical layer by adding mechanisms to detect and retransmit damaged, duplicate, or lost frames.
5. **Media Access Control** - When two or more devices are connected to the same link, data link layer protocols are necessary to determine which device has control over the link at any given time.

Network Layer

- Responsible for the source-to-destination delivery of a packet, possibly across multiple networks (links).
- Whereas the data link layer oversees the delivery of the packet between two systems on the same network (link), the network layer ensures that each packet gets from its point of origin to its final destination.



```
C:\WINDOWS\system32\cmd.exe

C:\>tracert speedguide.net

Tracing route to speedguide.net [66.230.207.58]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  192.168.1.1
  2     8 ms     *        8 ms  68.152.180.4
  3     8 ms     7 ms     8 ms  68.152.181.197
  4    15 ms    14 ms    14 ms  ixc01jax-ge-1-0-7.bellsouth.net [205.152.70.146]
  5    13 ms    13 ms    13 ms  ber01gnv-pos-1-0-0.bellsouth.net [65.83.239.129]
  6    14 ms    14 ms    13 ms  axr01msy-so-7-3-0.bellsouth.net [65.83.237.18]
  7    14 ms    13 ms    14 ms  axr00asm-0-3-0.bellsouth.net [65.83.236.12]
  8    13 ms    13 ms    14 ms  65.83.238.142
  9    25 ms    25 ms    25 ms  cr2.ormfl.ip.att.net [12.122.143.214]
 10    25 ms    25 ms    24 ms  cr1.attga.ip.att.net [12.122.5.142]
 11    24 ms    24 ms    25 ms  ggr7.attga.ip.att.net [12.122.87.61]
 12    25 ms    24 ms    24 ms  192.205.35.214
 13    25 ms    31 ms    36 ms  ae-62-51.ebr2.atlanta2.level3.net [4.68.103.29]
 14    43 ms    42 ms    43 ms  ae-1-6.bar1.tampa1.level3.net [4.69.137.113]
 15    44 ms    43 ms    42 ms  ae-5-5.car1.tampa1.level3.net [4.69.133.49]
 16    43 ms    43 ms    43 ms  ae-14-14.car4.tampa1.level3.net [4.69.133.58]
 17    43 ms    42 ms    43 ms  hostway-cor.car4.tampa1.level3.net [4.71.2.14]
 18    43 ms    43 ms    43 ms  te49.dr5.as30217.net [84.40.24.82]
 19    43 ms    42 ms    43 ms  bce0-ss1.sr1a.as30217.net [84.40.24.134]
 20    42 ms    43 ms    42 ms  speedguide.net [66.230.207.58]

Trace complete.
```

Specific Responsibilities of the Network Layer

- **Addressing** - Adds a header that includes the logical addresses of the sender and receiver to the packet coming from the upper layer.
- **Routing** - Provides routing mechanism to ensure that packets traveling through the Internet (LANs, WANs, etc.) gets to the destination host.

Logical Addressing

- Need for global addressing scheme in order to communicate computers anywhere in the world over the Internet.
- Today, we use the term IP address to mean a logical address in the network layer.
- Two types of addressing:
    1. **IPv4** (IP version 4) - IP address is 32 bits in length; max number of address is 2^{32}.
    2. **IPv6** (IP version 6) - Came about due to need of more addresses. IP address is 128 bits in length.

Timeline of IPv6 Transition

1974 — The IP addressing system (IPv4) is developed
4.3 billion addresses are created

1980 — IPv4 is used for the first time in a test network

1982 — World Wide Web protocol is developed

1992 — RIPE (the first RIR) is established to register and distribute addresses in Europe
IETF concludes that IPv4 will not scale

1993 — APNIC is established for Asia

1997 — ARIN is established for North America

1998 — IPv6 introduced, with 340 *trillion trillion, trillion* addresses
6bone, the first worldwide IPv6 test network, is launched

1999 — LACNIC is established for South America

2005 — AfriNIC is established for Africa

2011 — IANA's supply of IPv4 addresses exhausted
APNIC's supply of IPv4 addresses exhausted
Kalorama transacts its first IPv4 deal

2012 — World IPv6 launch
RIPE's projected IPv4 exhaustion

2013 — ARIN's projected IPv4 exhaustion

2016 — 3.4 billion Internet users
19 billion network connections
10 billion mobile devices with Internet connectivity
39% of mobile devices could be IPv6 capable

KALORAMA

IPv4 Addresses

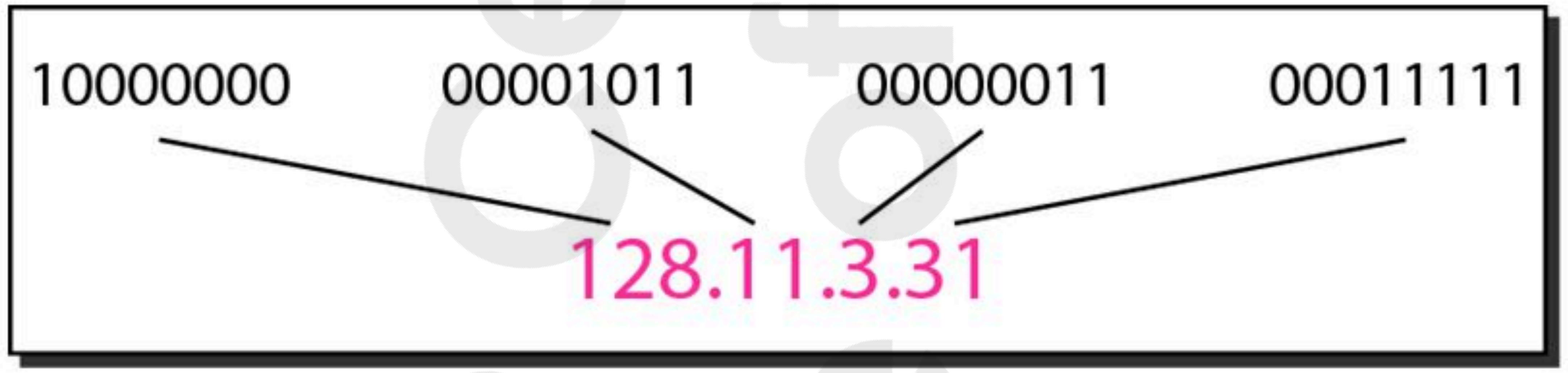
- Unique in the sense that each address defines one, and only one, connection to the Internet.
- Two devices on the Internet can never have the same address at the same time.
- Address space or total number of addresses used by the protocol is 2^{32} or 4, 294, 967, 296.
- Notations:

    1. **Binary Notation** - Displayed as 32 bits. Each octet (8 bits) is often referred to as a byte hence 4-byte address.

        Example: 01110101 10010101 00011101 00000010

    2. **Dotted-Decimal Notation** - Decimal form with a decimal point (dot) separating bytes.

        Example: 117.149.29.2


10000000
00001011
00000011
00011111
128.11.3.31

Example 19.1

Change the following IPv4 addresses from binary notation to dotted-decimal notation.

a. 10000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

Solution

We replace each group of 8 bits with its equivalent decimal number (see Appendix B) and add dots for separation.

a. 129.11.11.239

b. 193.131.27.255

Example 19.2

Change the following IPv4 addresses from dotted-decimal notation to binary notation.

a. 111.56.45.78

b. 221.34.7.82

Solution

We replace each decimal number with its binary equivalent (see Appendix B).

a. 01101111 00111000 00101101 01001110

b. 11011101 00100010 00000111 01010010

Example 19.3

Find the error, if any, in the following IPv4 addresses.

a. 111.56.045.78
b. 221.34.7.8.20
c. 75.45.301.14
d. 11100010.23.14.67

Solution

a. There must be no leading zero (045).
b. There can be no more than four numbers in an IPv4 address.
c. Each number needs to be less than or equal to 255 (301 is outside this range).
d. A mixture of binary notation and dotted-decimal notation is not allowed.

Classful Addressing

- Address space is divided into five classes: A, B, C, D and E.

	First byte	Second byte	Third byte	Fourth byte
Class A	0			
Class B	10			
Class C	110			
Class D	1110			
Class E	1111			

a. Binary notation

	First byte	Second byte	Third byte	Fourth byte
Class A	0–127			
Class B	128–191			
Class C	192–223			
Class D	224–239			
Class E	240–255			

b. Dotted-decimal notation

Class D: multicast
Class E: reserved

- Class A - For large organizations with a large number of hosts/routers.
- Class B - For midsize organizations with tens of thousands of hosts/routers.
- Class C - For small organizations with a small number of hosts/routers.

	First byte	Second byte	Third byte	Fourth byte
Class A	0			
Class B	10			
Class C	110			
Class D	1110			
Class E	1111			

a. Binary notation

	First byte	Second byte	Third byte	Fourth byte
Class A	0–127			
Class B	128–191			
Class C	192–223			
Class D	224–239			
Class E	240–255			

b. Dotted-decimal notation

Example 19.4

Find the class of each address.

a. 00000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

c. 14.23.120.8

d. 252.5.15.111

Solution

a. The first bit is 0. This is a class A address.

b. The first 2 bits are 1; the third bit is 0. This is a class C address.

c. The first byte is 14 (between 0 and 127); the class is A.

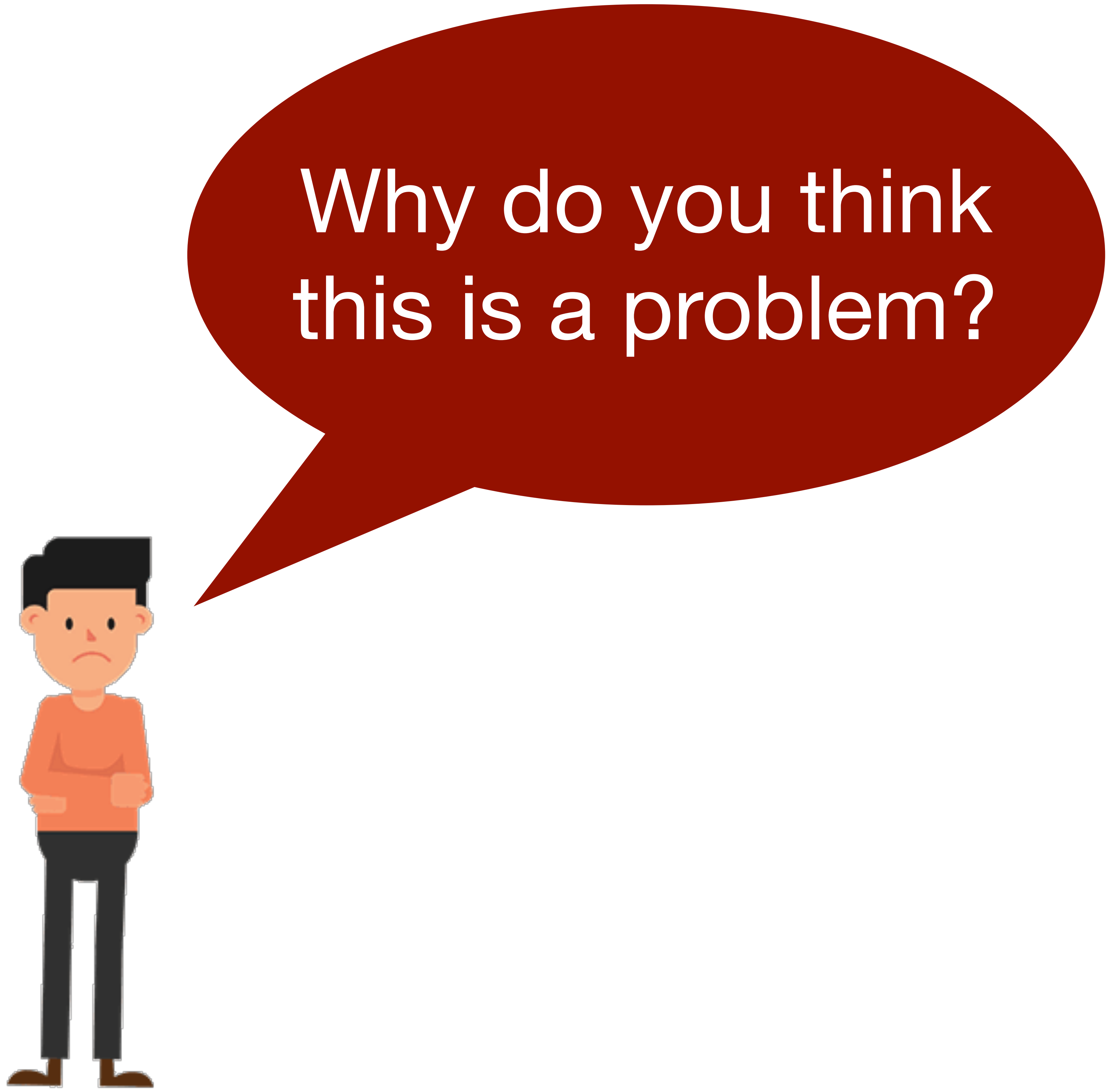
d. The first byte is 252 (between 240 and 255); the class is E.

Classes and Blocks

- In classful addressing, each class is divided into a fixed number of blocks with each block having a fixed size.

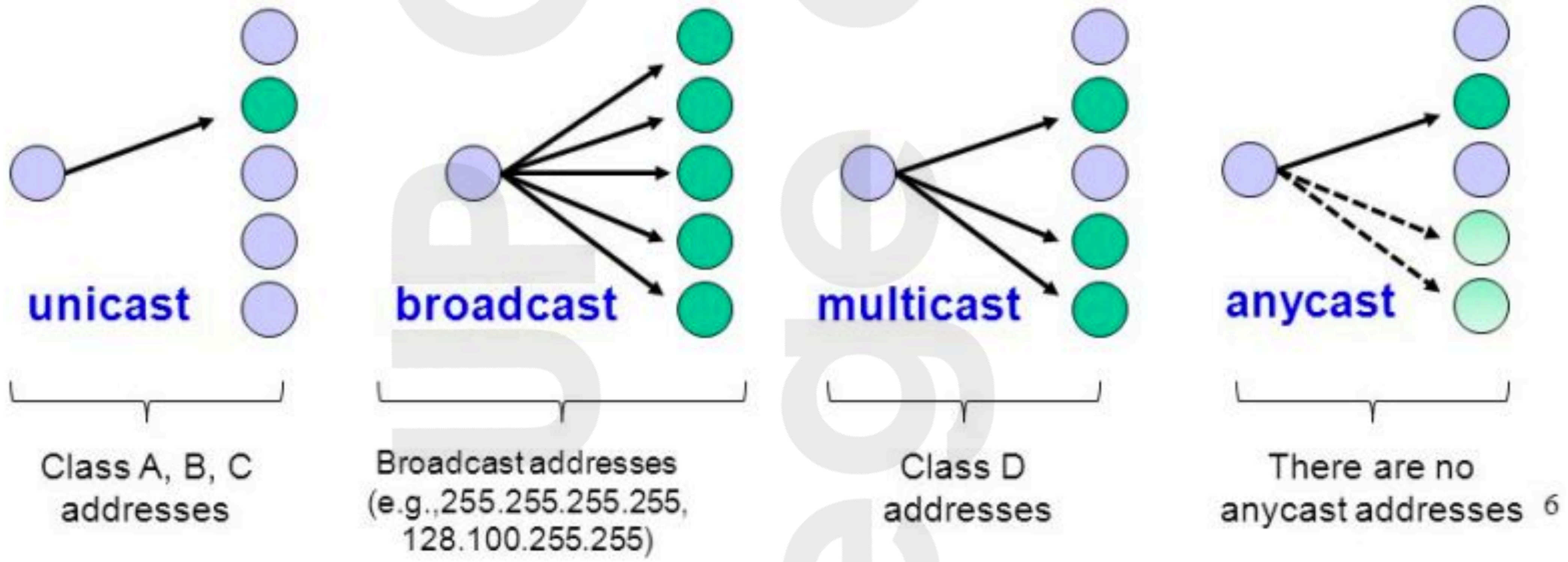
Class	*Number of Blocks*	*Block Size*	*Application*
A	128	16,777,216	Unicast
B	16,384	65,536	Unicast
C	2,097,152	256	Unicast
D	1	268,435,456	Multicast
E	1	268,435,456	Reserved

We can see the flaw in this design. A block in class A address is too large for almost any organization. This means most of the addresses in class A were wasted and were not used. A block in class B is also very large, probably too large for many of the organizations that received a class B block. A block in class C is probably too small for many organizations. Class D addresses were designed for multicasting as we will see in a later chapter. Each address in this class is used to define one group of hosts on the Internet. The Internet authorities wrongly predicted a need for 268,435,456 groups. This never happened and many addresses were wasted here too. And lastly, the class E addresses were reserved for future use; only a few were used, resulting in another waste of addresses.



Delivery modes

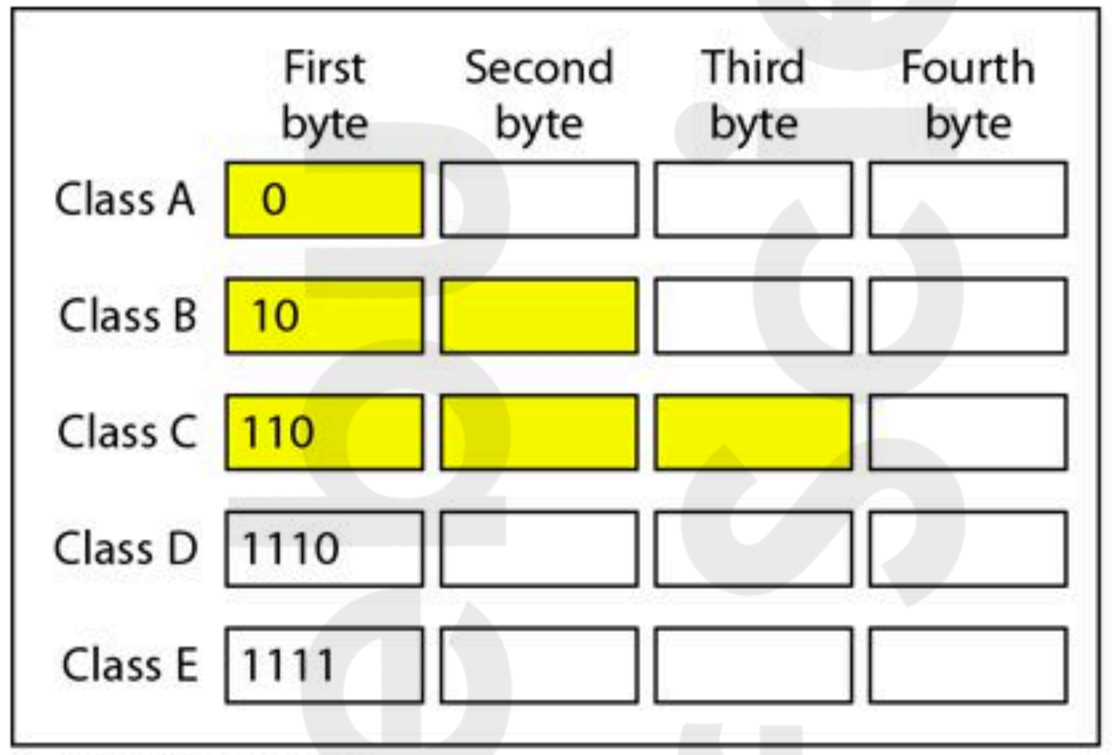
- Supported by IPv4
  - one-to-one (unicast)
  - one-to-all (broadcast)
  - one-to-many (multicast)
- Not supported by IPv4:
  - one-to-any (anycast)



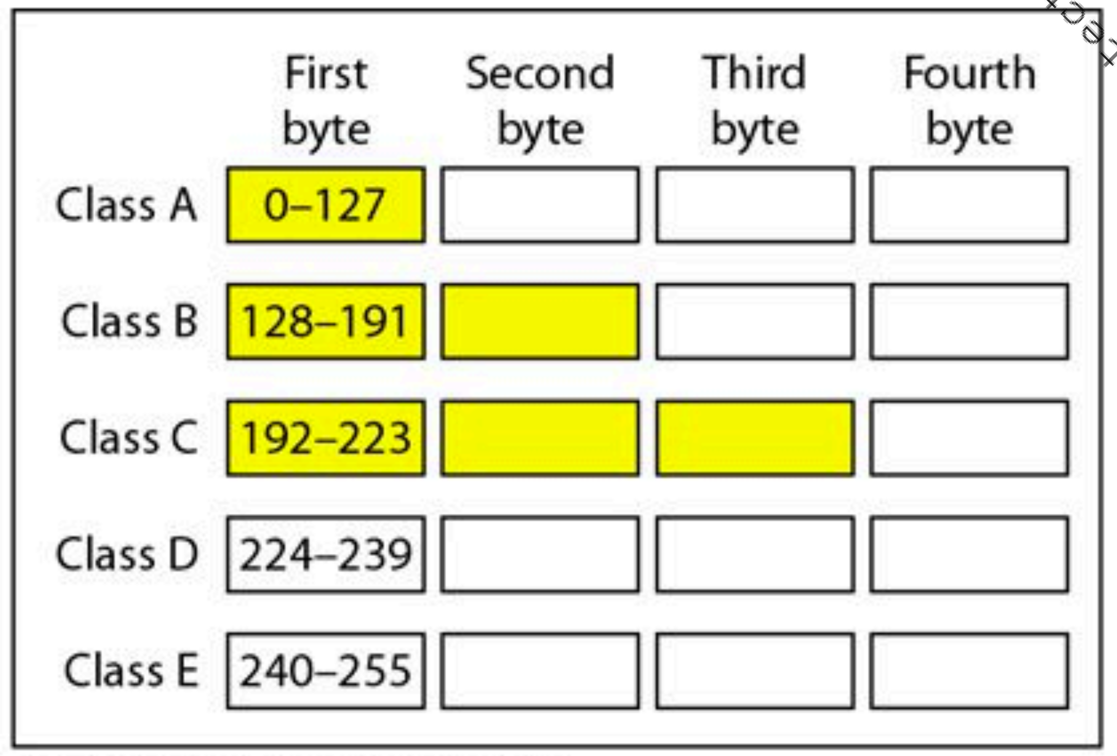
6

Netid and Hostid

- Netid - Network ID
- Hostid - Host ID



a. Binary notation



b. Dotted-decimal notation

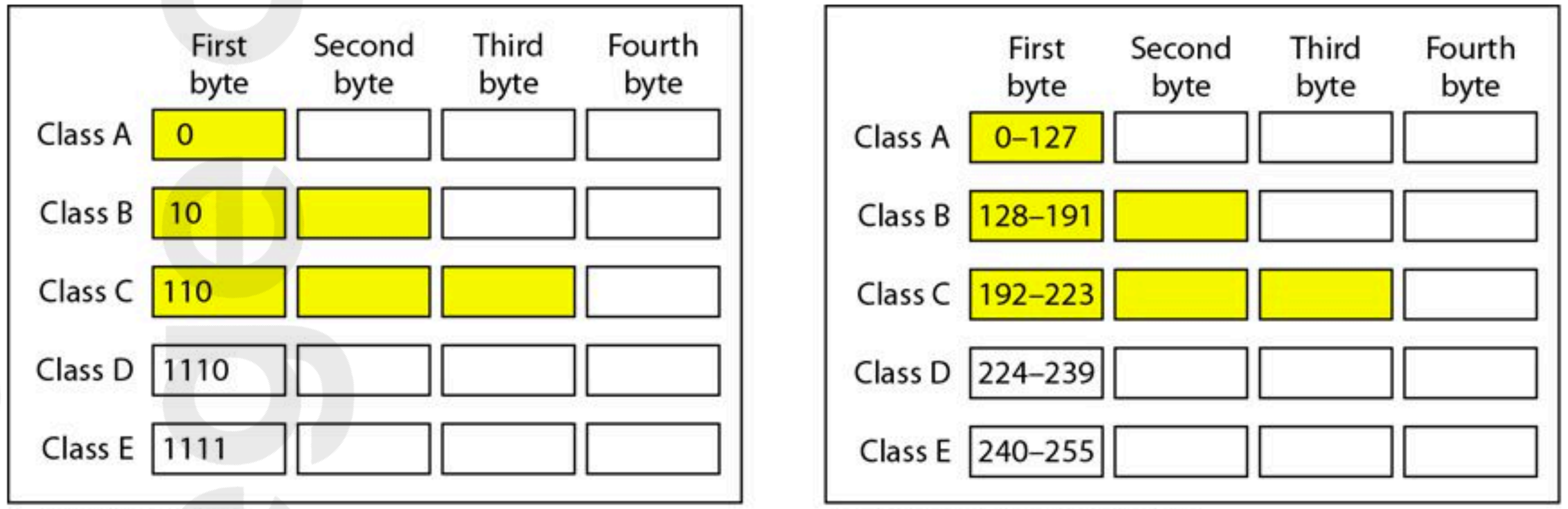
Class D: multicast
Class E: reserved

Class	*Number of Blocks*	*Block Size*	*Application*
A	128	16,777,216	Unicast
B	16,384	65,536	Unicast
C	2,097,152	256	Unicast
D	1	268,435,456	Multicast
E	1	268,435,456	Reserved

Mask

- Or default mask.
- A 32-bit number made of contiguous 1s followed by contiguous 0s.
- Help find the netid and hostid.
- Slash notation or Classless Interdomain Routing (CIDR).

Class	*Binary*	*Dotted-Decimal*	*CIDR*
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 11111111 11111111 00000000	255.255.255.0	/24



a. Binary notation

b. Dotted-decimal notation

Class D: multicast
Class E: reserved

Subnetting and Supernetting

- **Subnetting**
  - Was introduced during era of classful addressing.
  - If an organization was granted a large block in class A or B, it could divide the addresses into several contiguous groups and assign each group to smaller networks (subnets) or, in rare cases, share part of the addresses with neighbors.
- **Supernetting**
  - Combination of several networks as demand increases.
  - Example, an organization that needs 1000 addresses can be granted four contiguous class C blocks and create one super network.

Address Depletion

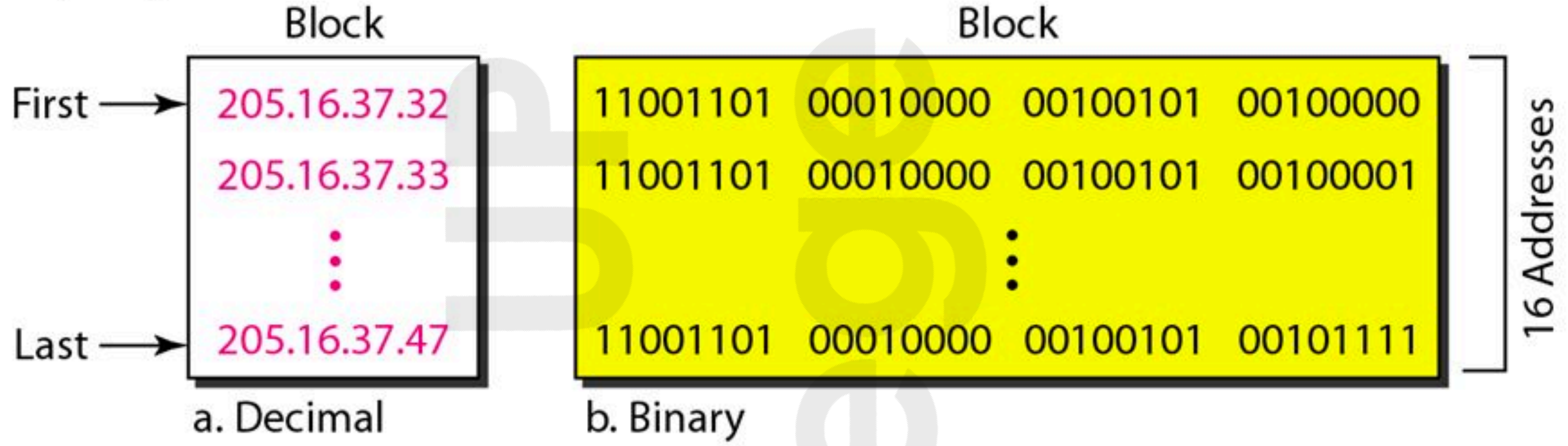
- The flaws in classful addressing scheme combined with the fast growth of the Internet led to the near depletion of available addresses.
- Yet the number of devices on the Internet is much less than 2^{32} address space.
- We have run out of class A and B addresses, and a class C block is too small for most midsize organization.
- One solution that has alleviated the problem is the idea of **classless addressing**.

Classless Addressing

- No classes but address are still granted blocks.
- Address Blocks
  - When entity, small or large, needs to be connected to the Internet, it is granted a block (range) of addresses.
  - Size of the block (number of addresses) varies based on the nature and size of the entity.
- Restrictions to simplify handling of addresses:
  1. The addresses in a block must be contiguous, one after another.
  2. The number of addresses in a block must be a power of 2 (1, 2, 4, 8, …)
  3. The first address must be evenly divisible by the number of addresses.

Figure 19.3 shows a block of addresses, in both binary and dotted-decimal notation, granted to a small business that needs 16 addresses.

We can see that the restrictions are applied to this block. The addresses are contiguous. The number of addresses is a power of 2 ($16 = 2^4$), and the first address is divisible by 16. The first address, when converted to a decimal number, is 3,440,387,360, which when divided by 16 results in 215,024,210.



Mask in Classless Addressing

- Here, a block of addresses can be defined as

  x.y.z.t /n

  - x.y.z.t defines one of the addresses
  - /n defines the mask

  The address and the /n defines the whole block (first address, last address and number of addresses).

  The *first address* in the block can be found by setting the rightmost 32 - n bits to 0.

First address: Set rightmost 32 - n bits to 0s…

A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/28. What is the first address in the block?

Solution

The binary representation of the given address is

11001101 00010000 00100101 00100111

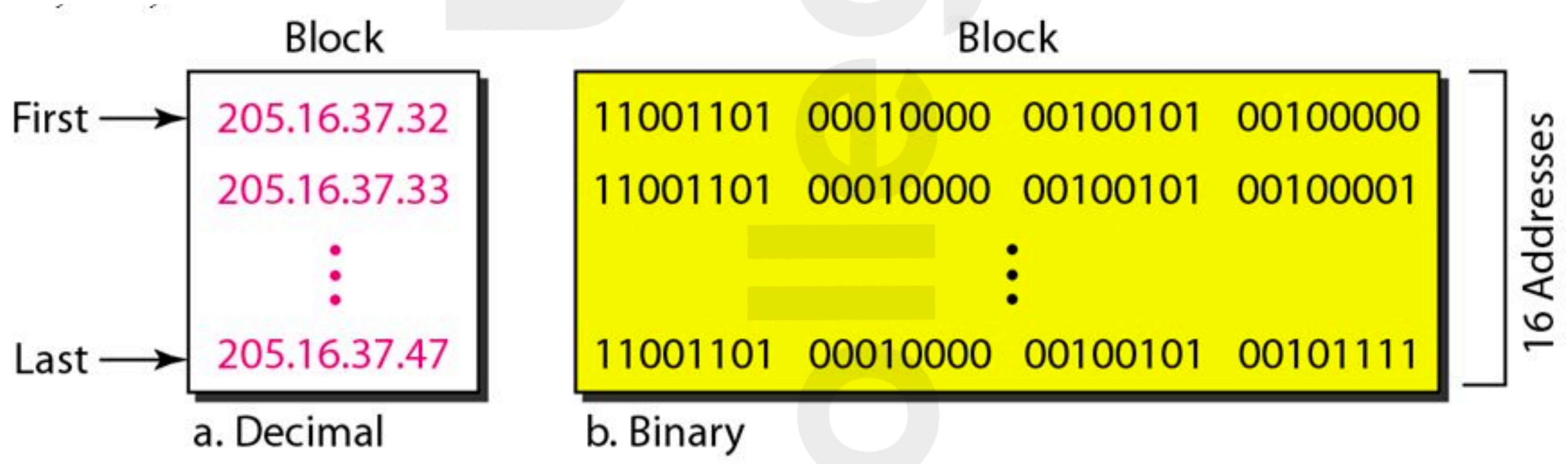
If we set 32−28 rightmost bits to 0, we get

11001101 00010000 00100101 0010000

or

205.16.37.32.

This is actually the block shown in Figure 19.3.



Last address: Set rightmost 32 - n bits to 1s…

Find the last address for the block in Example 19.6.

Solution

The binary representation of the given address is

11001101 00010000 00100101 00100111

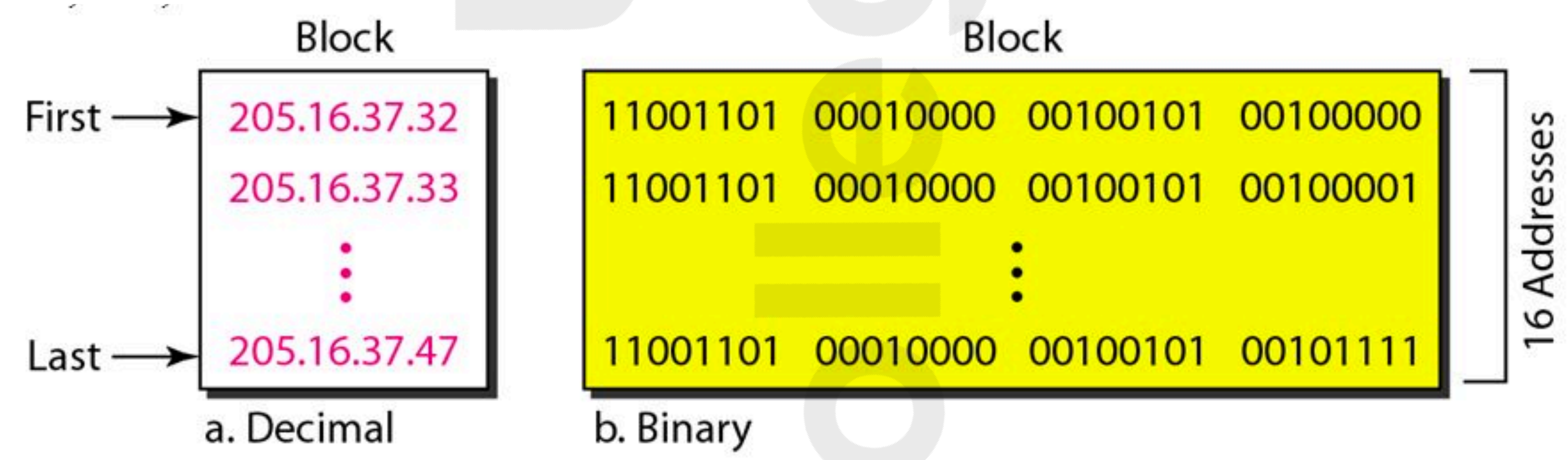
If we set 32 − 28 rightmost bits to 1, we get

11001101 00010000 00100101 00101111

or

205.16.37.47

This is actually the block shown in Figure 19.3.



Number of addresses: $2^{32 - n}$…

Find the number of addresses in Example 19.6.

Solution

The value of n is 28, which means that number of addresses is 2^{32-28} or 16.

Example 19.9

Another way to find the first address, the last address, and the number of addresses is to represent the mask as a 32-bit binary (or 8-digit hexadecimal) number. This is particularly useful when we are writing a program to find these pieces of information. In Example 19.5 the /28 can be represented as 11111111 11111111 11111111 11110000 (twenty-eight Is and four Os). Find

a. The first address
h. The last address
c. The number of addresses

Solution

a. The first address can be found by ANDing the given addresses with the mask. ANDing here is done bit by bit. The result of ANDing 2 bits is 1 if both bits are Is; the result is 0 otherwise.

Address:	11001101 00010000 00100101 00100111
Mask:	11111111 11111111 11111111 11110000
First address:	11001101 00010000 00100101 00100000

b. The last address can be found by ORing the given addresses with the complement of the mask. ORing here is done bit by bit. The result of ORing 2 bits is 0 if both bits are Os; the result is 1 otherwise. The complement of a number is found by changing each 1 to 0 and each 0 to 1.

Address:	11001101 00010000 00100101 00100111
Mask complement:	00000000 00000000 00000000 00001111
Last address:	11001101 00010000 *00100101* *00101111*

c. The number of addresses can be found by complementing the mask, interpreting it as a decimal number, and adding 1 to it.

Mask complement:	000000000 00000000 00000000 00001111
Number of addresses:	15 + 1 = 16

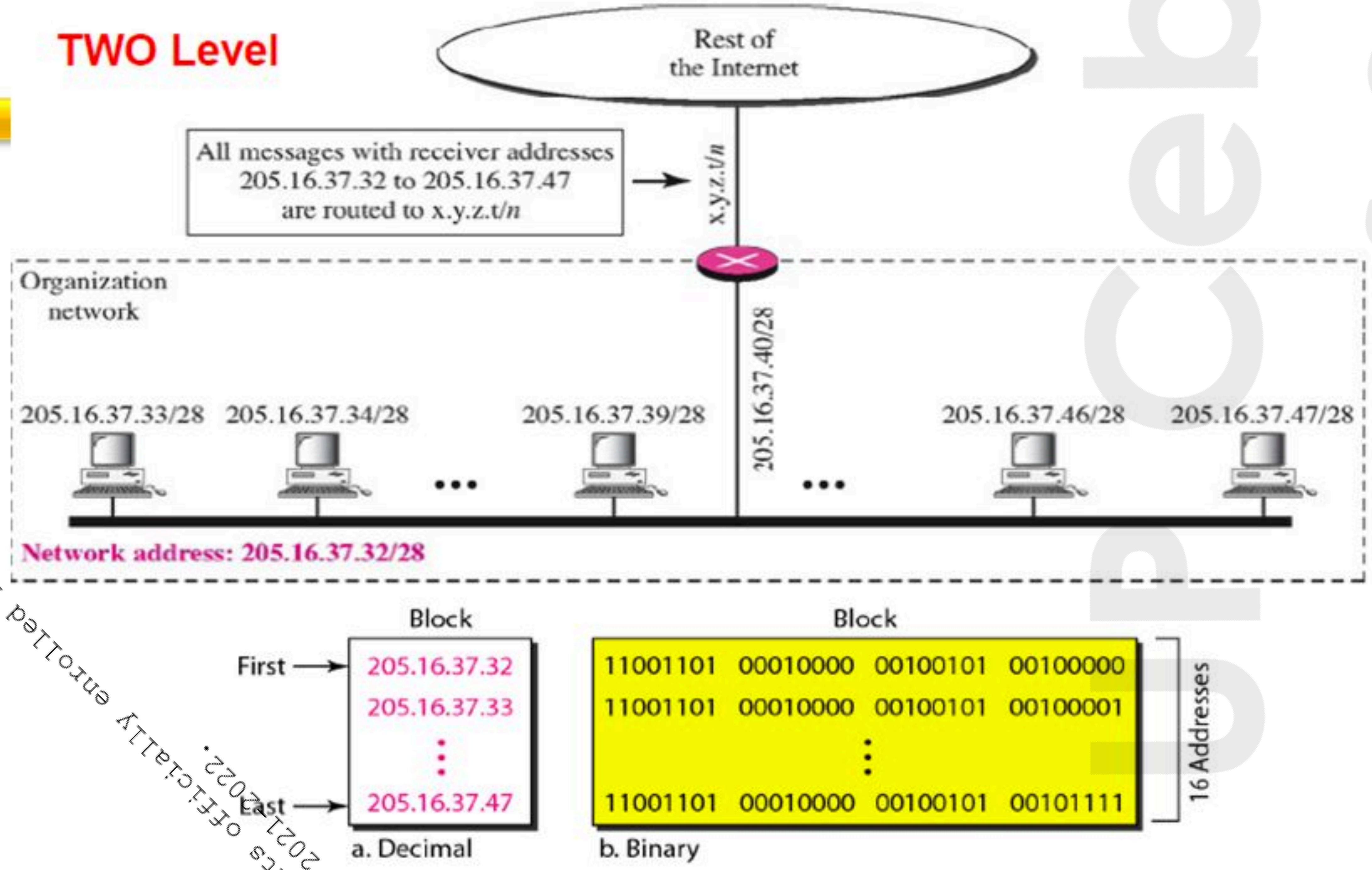
Network Addresses

- When organization is given a block of addresses, the organization is free to allocate the addresses to the devices that need to be connected to the Internet.
- First address in the class is normally (not always) treated as a special address (network address). This defines the organization network to the rest of the world.

A network configuration for the block 205.16.37.32/28

19.45

TWO Level



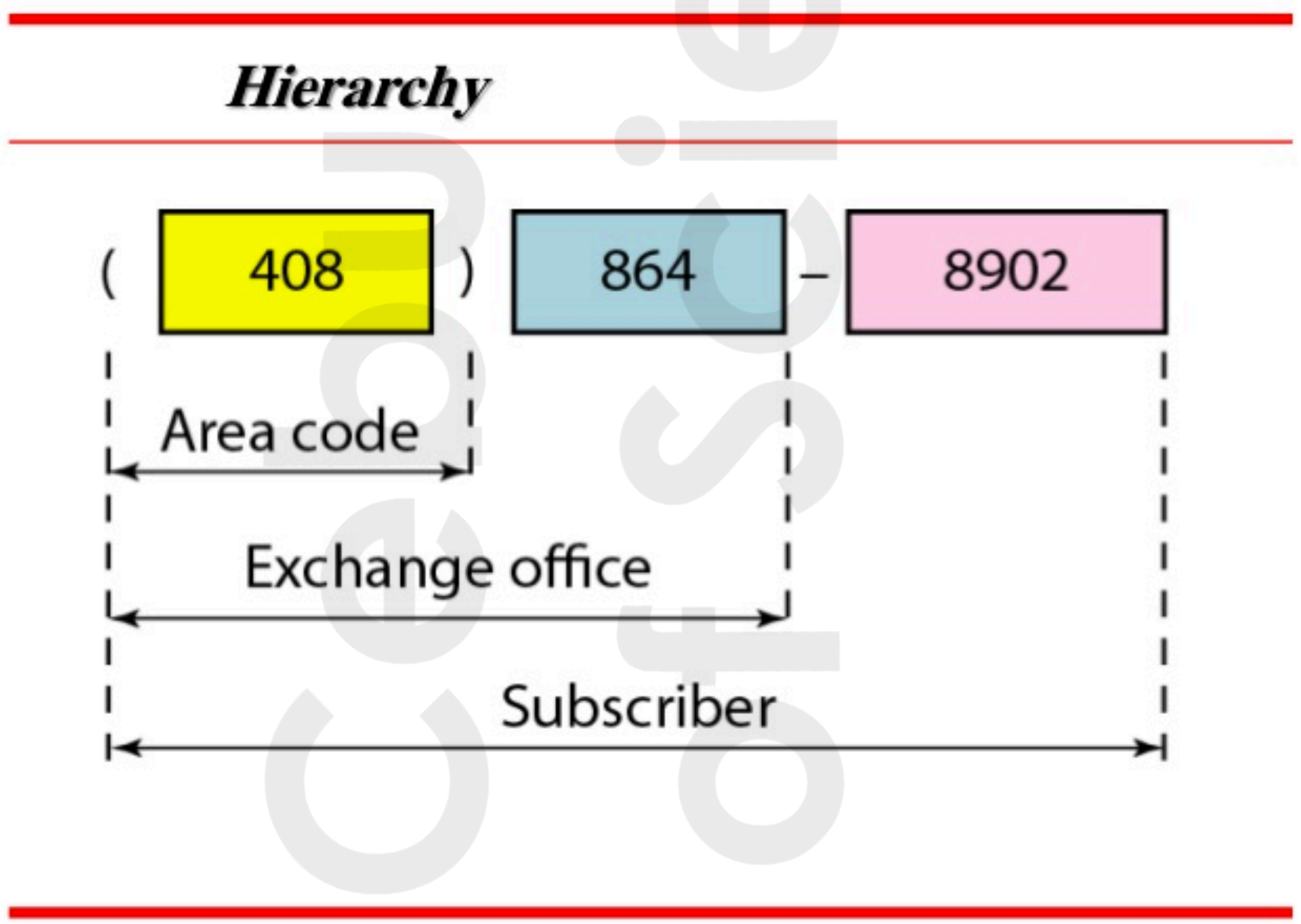
The router, at least the common home network device that we usually call a router, is the piece of network hardware that allows communication between your local home network - i.e. your personal computers and other connected devices - and the Internet.

The router used in home and small networks is more accurately called a residential gateway but you'll never see them called that.

A router is a device that analyzes the contents of data packets transmitted within a network or to another network. Routers determine whether the source and destination are on the same network or whether data must be transferred from one network type to another, which requires encapsulating the data packet with routing protocol header information for the new network type.

- Organization network is connected to the Internet via a router.
- Router has two addresses: one to the granted block; the other to the network at the other side of the router.
- The second address is x.y.z.t /n since no knowledge about the network it is connected to at the other side.

Hierarchy

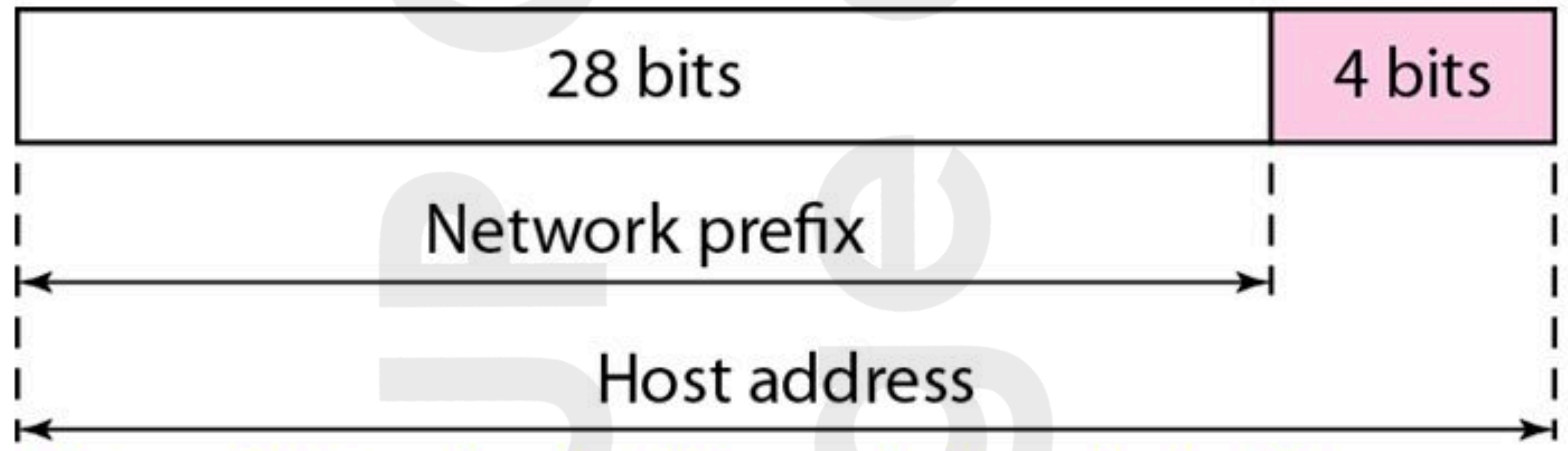


19.39 Figure 19.5 *Hierarchy in a telephone network in North America*

- Two-Level Hierarchy: No Subnetting
- Three-Level Hierarchy: Subnetting
- More Levels of Hierarchy

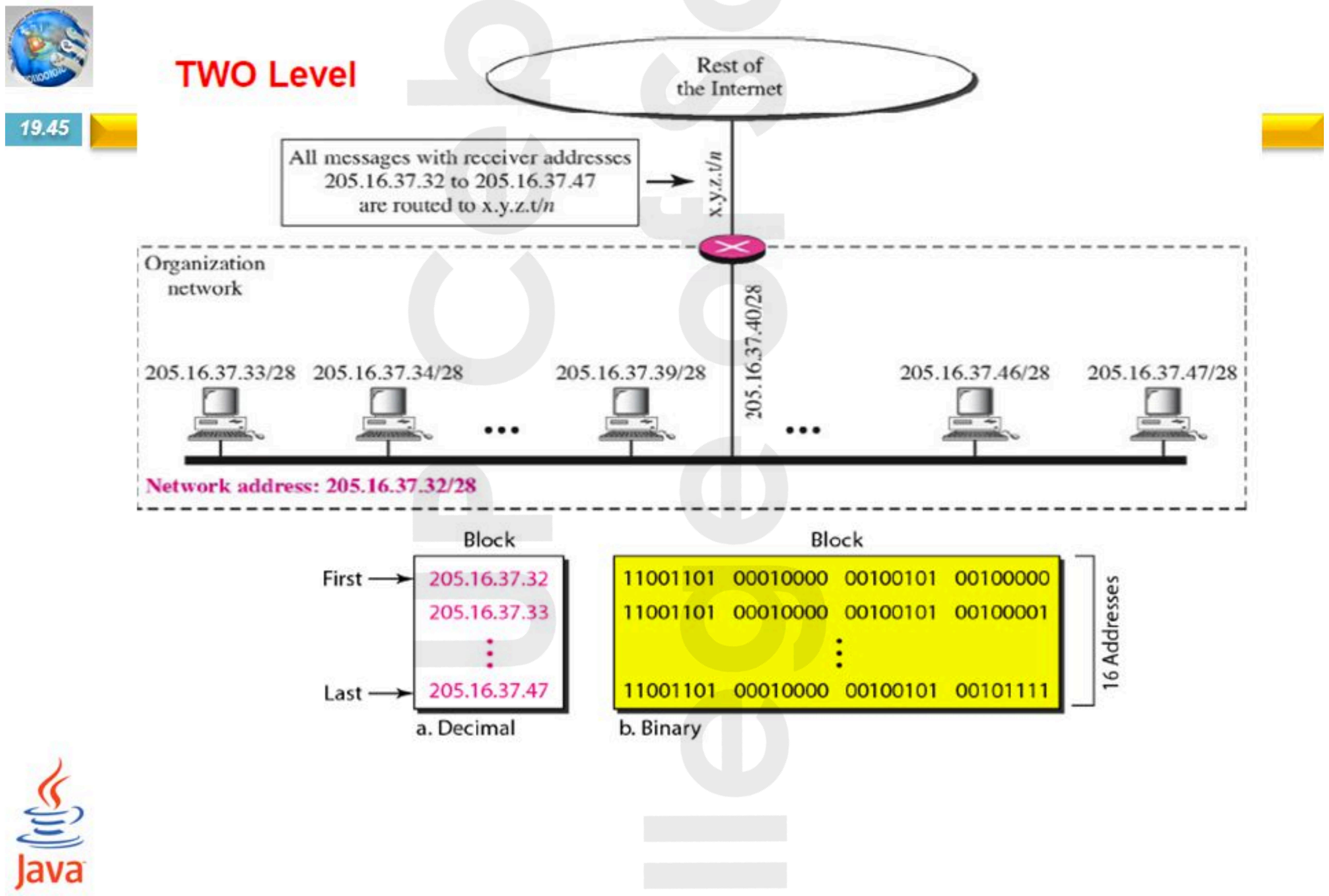
Two-Level Hierarchy: No Subnetting

- An IP address can define only two levels of hierarchy when not subnetted.
- The *n* leftmost bits (prefix) of the address x.y.z.t /n define the organization network.
- The 32 - *n* rightmost bits (suffix) define the particular host (computer router) to the network.



- The prefix is common to all addresses in the network; the suffix changes from one device to another.

A network configuration for the block 205.16.37.32/28



Three-Level Hierarchy: Subnetting

- Possible to create clusters of networks (subnets) and divide the addresses between the different subnets.
- The rest of the world still sees the organization as one entity; however, internally there are several subnets.
- The organization has its own mask; each subnet must also have its own.

Example

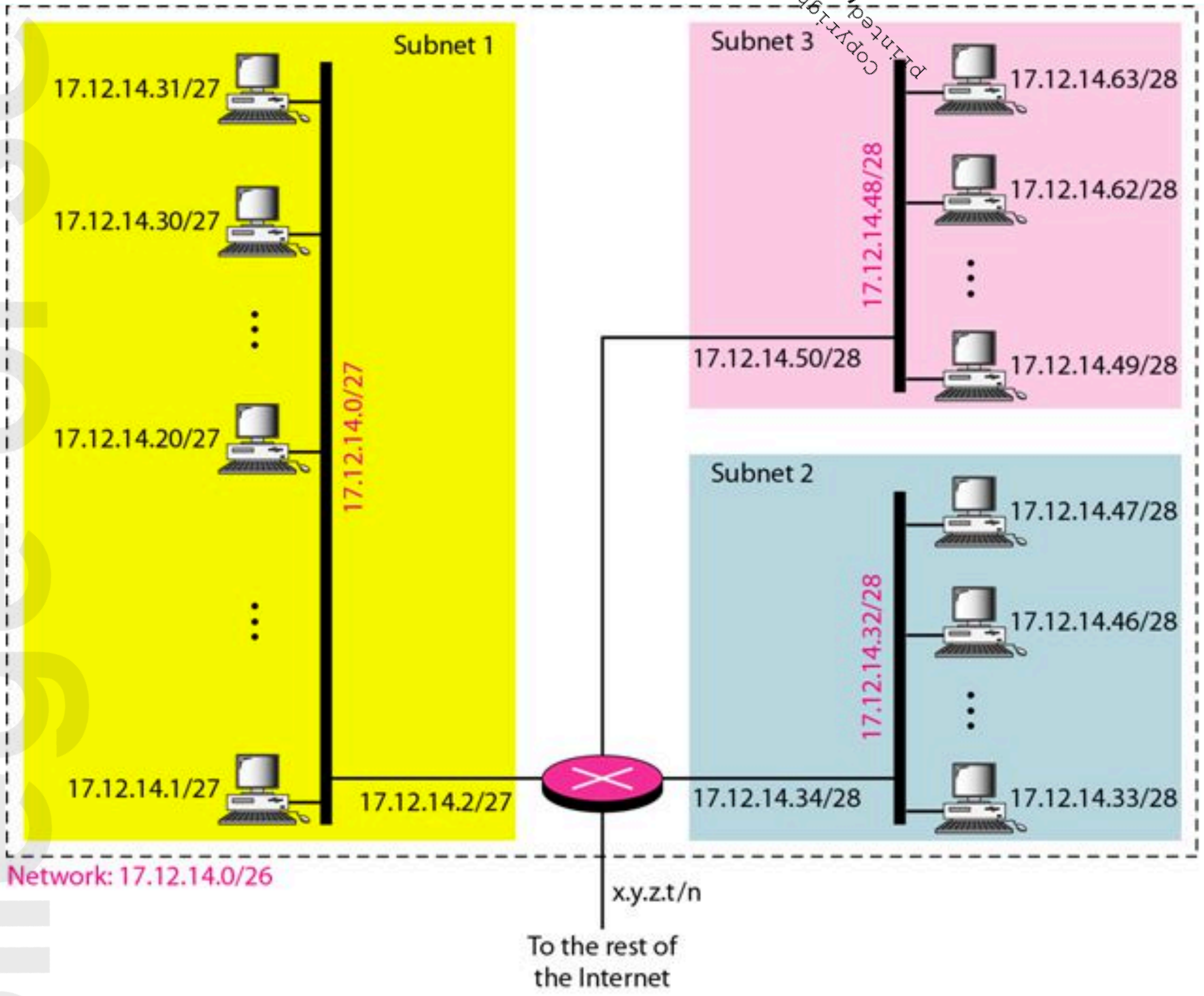
Three-Levels of Hierarchy: Subnetting

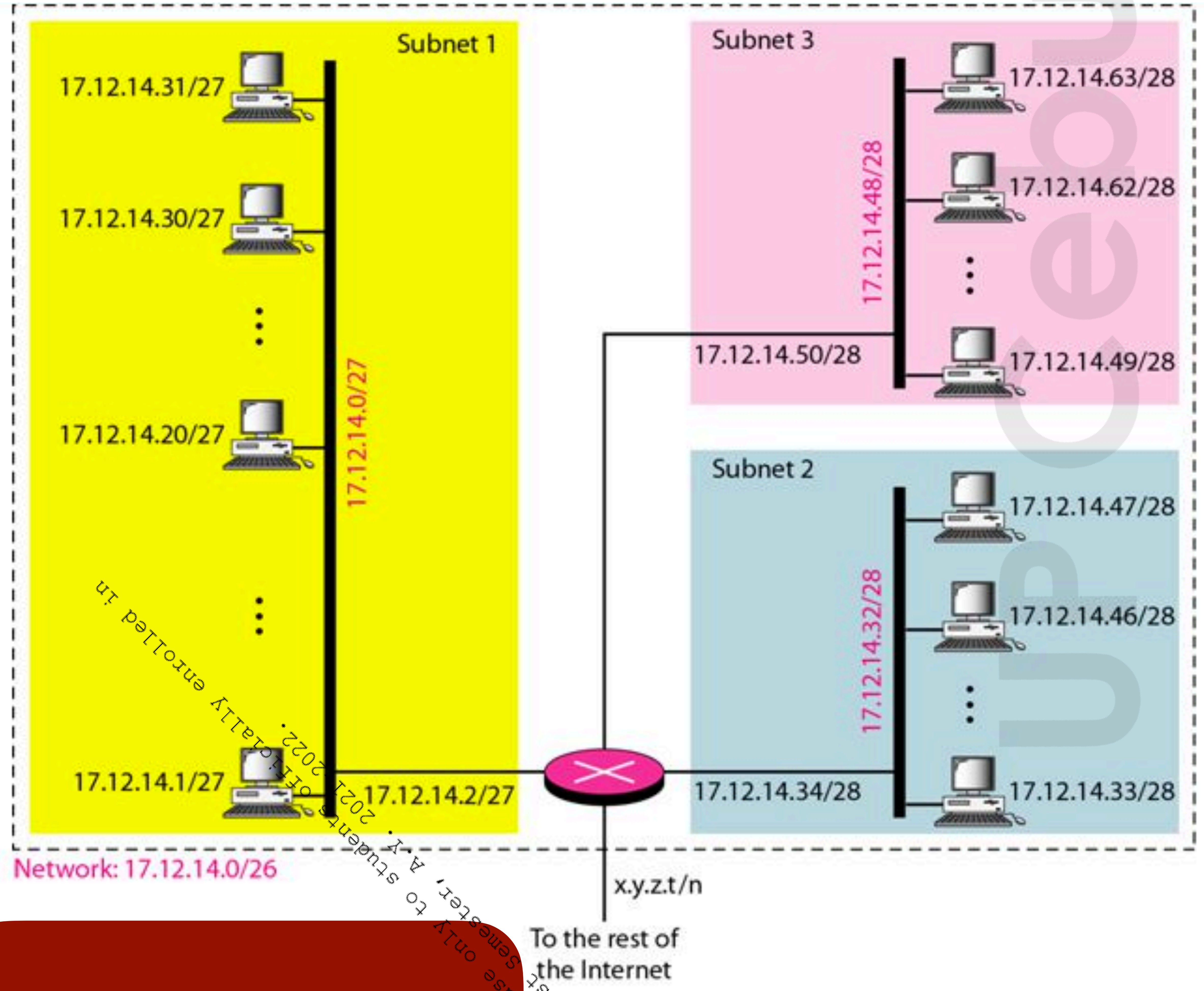
An organization that is granted a large block of addresses may want to create clusters of networks (called subnets) and divide the addresses between the different subnets.
As an example, suppose an organization is given the block 17.12.40.0/26, which contains 64 addresses. The organization has three offices and needs to divide the addresses into three subblocks of 32, 16, and 16 addresses.
We can find the new masks by using the following arguments:

1. Suppose the mask for the first subnet is n1, then 2^(32- n1) must be 32, which means that n1 =27.
2. Suppose the mask for the second subnet is n2, then 2^(32- n2) must be 16, which means that n2 = 28.
3. Suppose the mask for the third subnet is n3, then 2^(32- n3) must be 16, which means that n3 =28.

This means that we have the masks 27, 28, 28 with the organization mask being 26.

Behrouz A. Forouzan” Data communications and Networking





Let us check to see if we can find the subnet addresses from one of the addresses in the subnet.

a. In subnet 1, the address 17.12.14.29/27 can give us the subnet address if we use the mask /27 because

Host:	00010001 00001100 00001110 00011101
Mask:	/27
Subnet:	00010001 00001100 00001110 00000000 .... (17.12.14.0)

b. In subnet 2, the address 17.12.14.45/28 can give us the subnet address if we use the mask /28 because

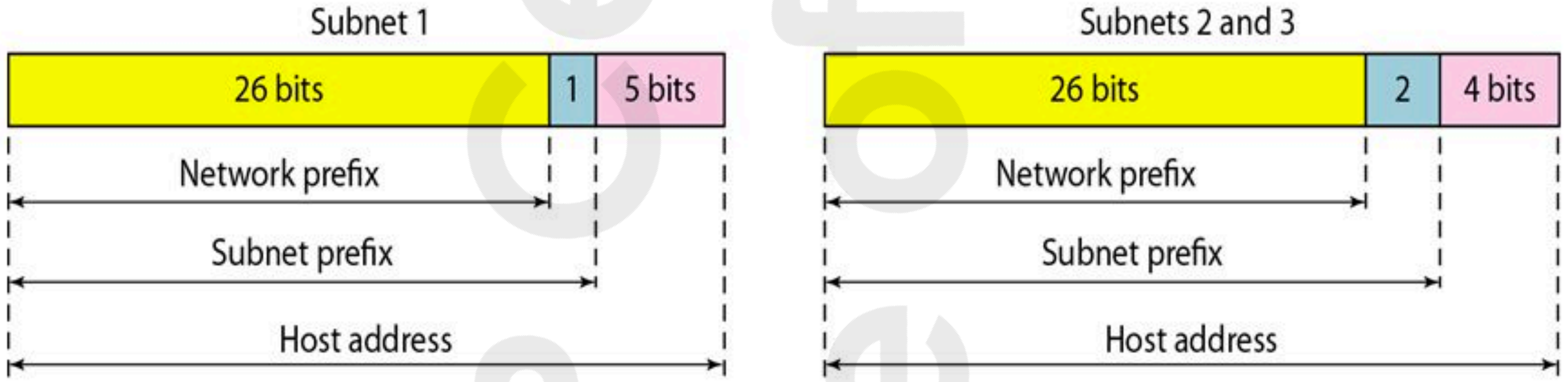
Host:	00010001 00001100 00001110 00101101
Mask:	/28
Subnet:	00010001 00001100 00001110 00100000 .... (17.12.14.32)

c. In subnet 3, the address 17.12.14.50/28 can give us the subnet address if we use the mask /28 because

Host:	00010001 00001100 00001110 00110010
Mask:	/28
Subnet:	00010001 00001100 00001110 00110000 .... (17.12.14.48)

Let's do this in the Lab!

Note that applying the mask of the network, /26, to any of the addresses gives us the network address 17.12.14.0/26.


Subnet 1
26 bits
1
5 bits
Network prefix
Subnet prefix
Host address
Subnets 2 and 3
26 bits
2
4 bits
Network prefix
Subnet prefix
Host address

More Levels of Hierarchy

- The structure of classless addressing does not restrict the number of hierarchical levels.
- An organization can divide the granted block of addresses into sub-blocks, in turn can be divided into smaller sub-blocks, and so on.
- One example of this is seen in the ISPs.
  - A *national ISP* can divide a granted large block into smaller blocks and assign each of them to a regional ISP.
  - A *regional ISP* can divide the block received from the national ISP into smaller blocks and assign each one to a local ISP.
  - A *local ISP* can divide the block received from the regional ISP into smaller blocks and assign each one to a different organization.
  - Finally, an *organization* can divide the received block and make several subnets out of it.

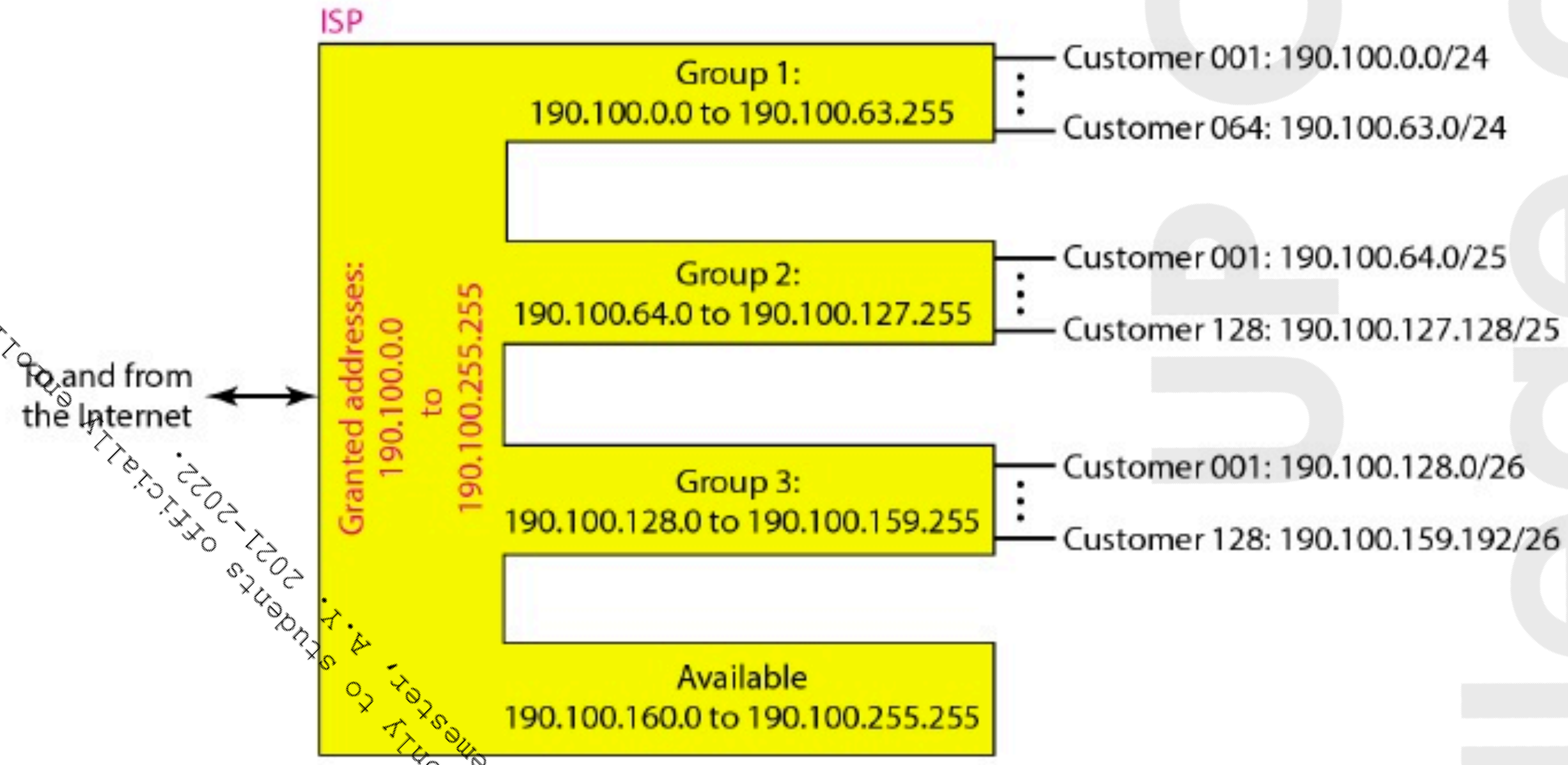
Example

Example 19.10

An ISP is granted a block of addresses starting with 190.100.0.0/16 (65,536 addresses). The IS needs to distribute these addresses to three groups of customers as follows:

a. The first group has 64 customers; each needs 256 addresses.
b. The second group has 128 customers; each needs 128 addresses.
c. The third group has 128 customers; each needs 64 addresses.

Design the subblocks and find out how many addresses are still available after these allocations



1. **Group 1**

For this group, each customer needs 256 addresses. This means that 8 (log2256) bits are needed to define each host. The prefix length is then 32 - 8 =24. The addresses are

1st Customer:	*190.100.0.0/24*	*190.100.0.255/24*
2nd Customer:	*190.100.1.0/24*	*190.100.1.255/24*
64th Customer:	*190.100.63.0/24*	*190.100.63.255/24*

Total =64 x 256 = 16,384

2. **Group2**

For this group, each customer needs 128 addresses. This means that 7 (10g2 128) bits are needed to define each host. The prefix length is then 32 – 7 =25. The addresses are

1st Customer:	*190.100.64.0/25*	*190.100.64.127/25*
2nd Customer:	*190.100.64.128/25*	*190.100.64.255/25*
128th Customer:	*190.100.127.128/25*	*190.100.127.255/25*

Total =128 x 128 = 16,384

3. **Group3**

For this group, each customer needs 64 addresses. This means that 6 (logz 64) bits are needed to each host. The prefix length is then 32 - 6 =26. The addresses are

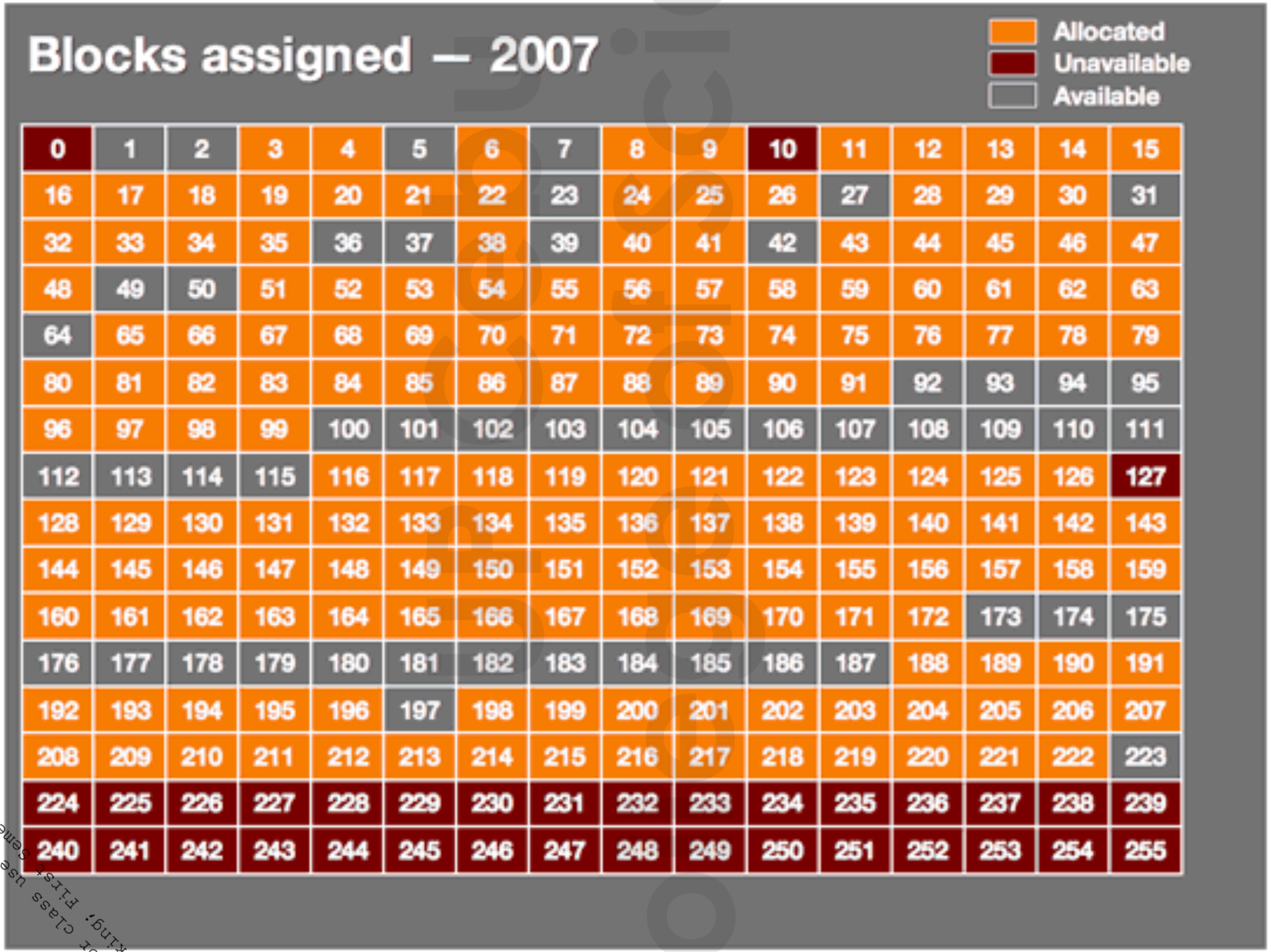
1st Customer:	*190.100.128.0/26*	*190.100.128.63/26*
2nd Customer:	*190.100.128.64/26*	*190.100.128.127/26*
128th Customer:	*190.100.159.192/26*	*190.100.159.255/26*

Total =128 x 64 =8192

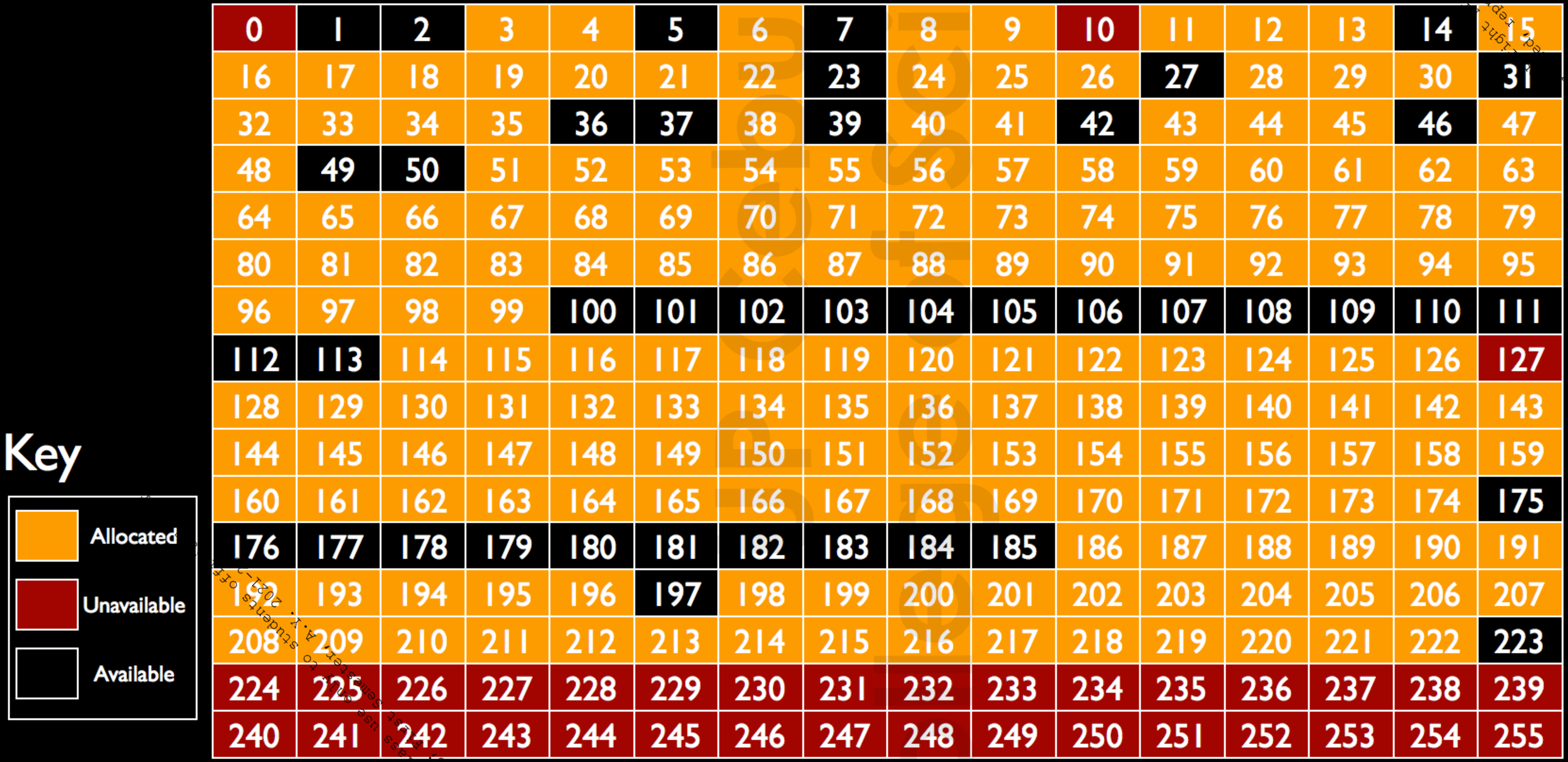
Number of granted addresses to the ISP: 65,536
Number of allocated addresses by the ISP: 40,960
Number of available addresses: 24,576

Address Allocation

- The next issue in classless addressing is address allocation (how are the blocks allocated?).
- The ultimate responsibility of address allocation is given to a global authority called the Internet Corporation for Assigned Names and Addresses (ICANN).
- ICANN does not normally allocate addresses to individual organizations but to ISPs (address aggregation).


Blocks assigned — 2007
Allocated
Unavailable
Available
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47
48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63
64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79
80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111
112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127
128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143
144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159
160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175
176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191
192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207
208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223
224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239
240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255

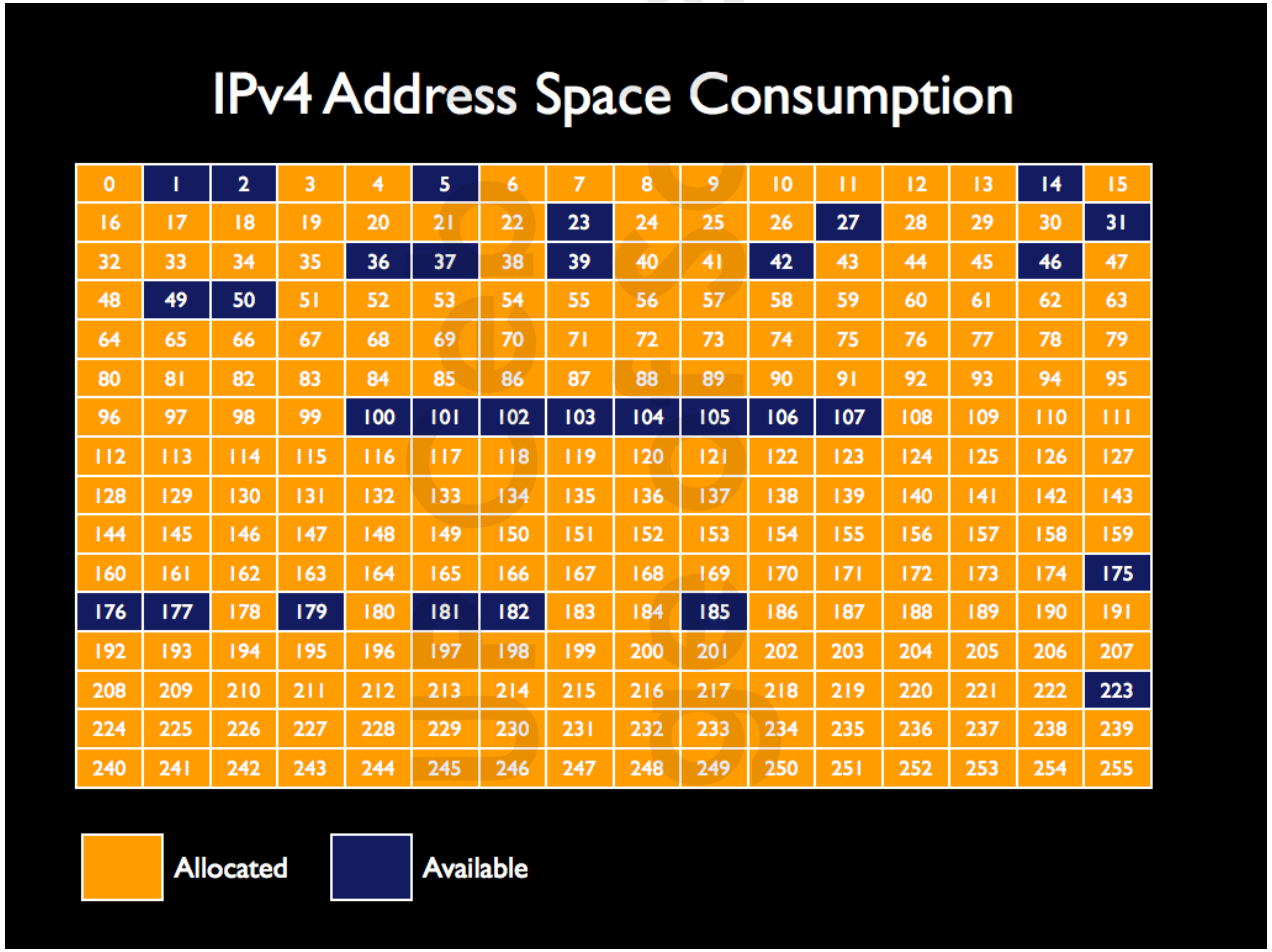
IPv4 Address Space Consumption



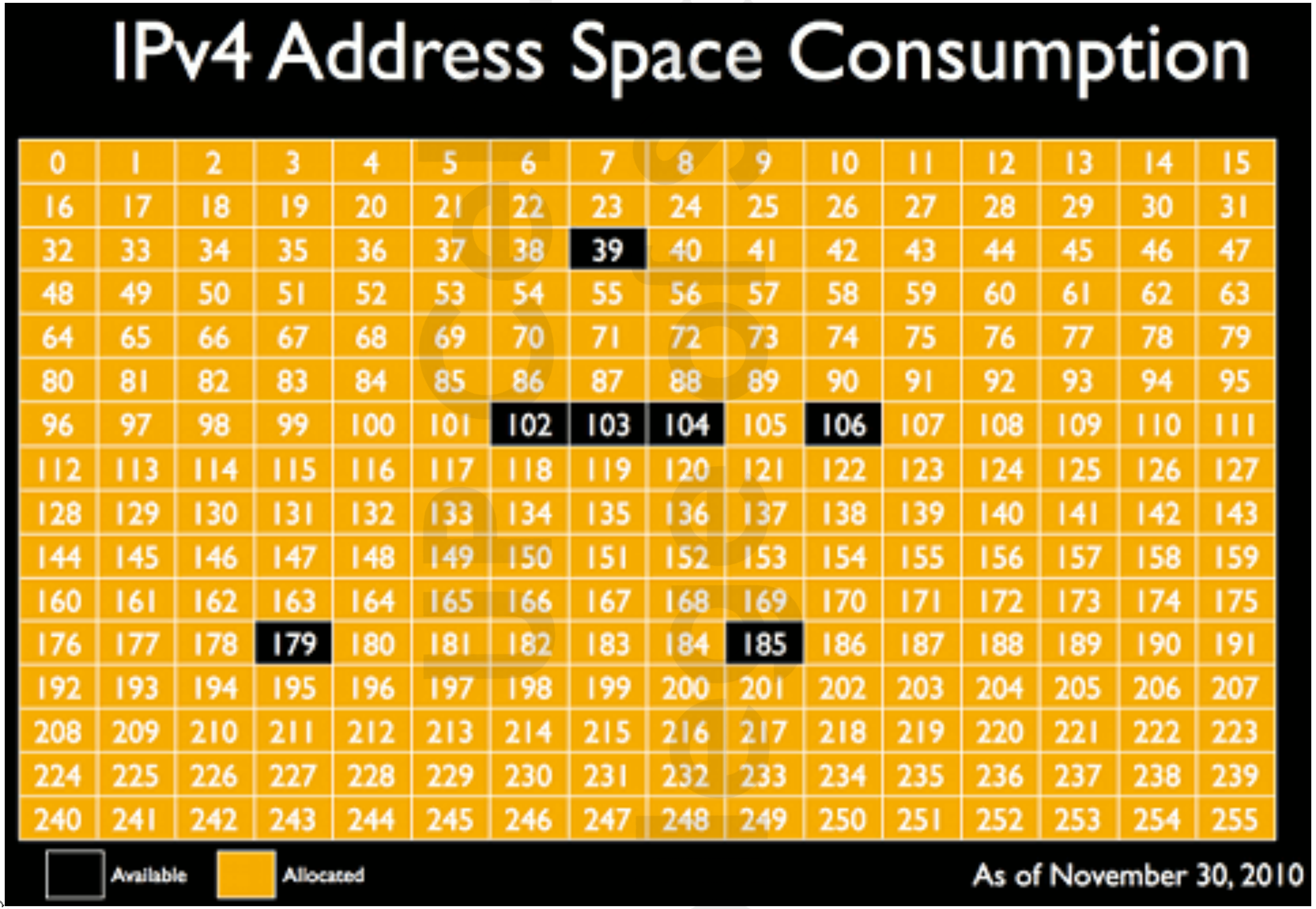
Source: http://www.iana.org/assignments/ipv4-address-space

As of February 14, 2008

2009


IPv4 Address Space Consumption
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47
48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63
64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79
80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111
112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127
128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143
144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159
160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175
176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191
192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207
208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223
224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239
240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255
Allocated
Available

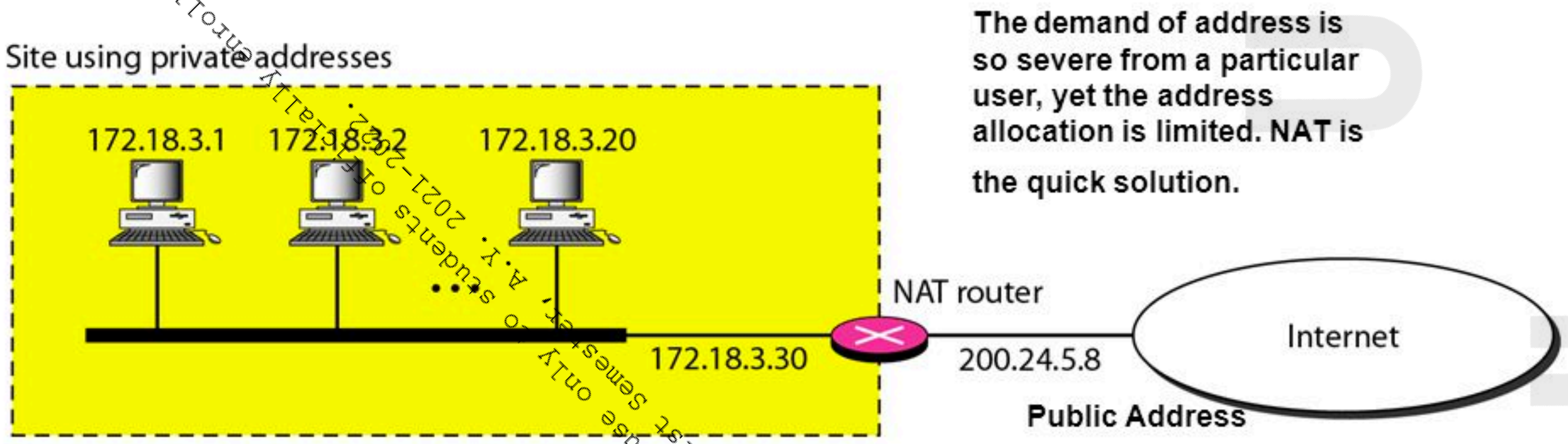
2010


IPv4 Address Space Consumption
Available
Allocated
As of November 30, 2010

Network Address Translation (NAT)

- A quick solution to the problem of shortage of addresses.
- NAT enables a user to have a large set of addresses internally and one address, or a small set of addresses, externally.
- Reserved three sets of addresses as private addresses:

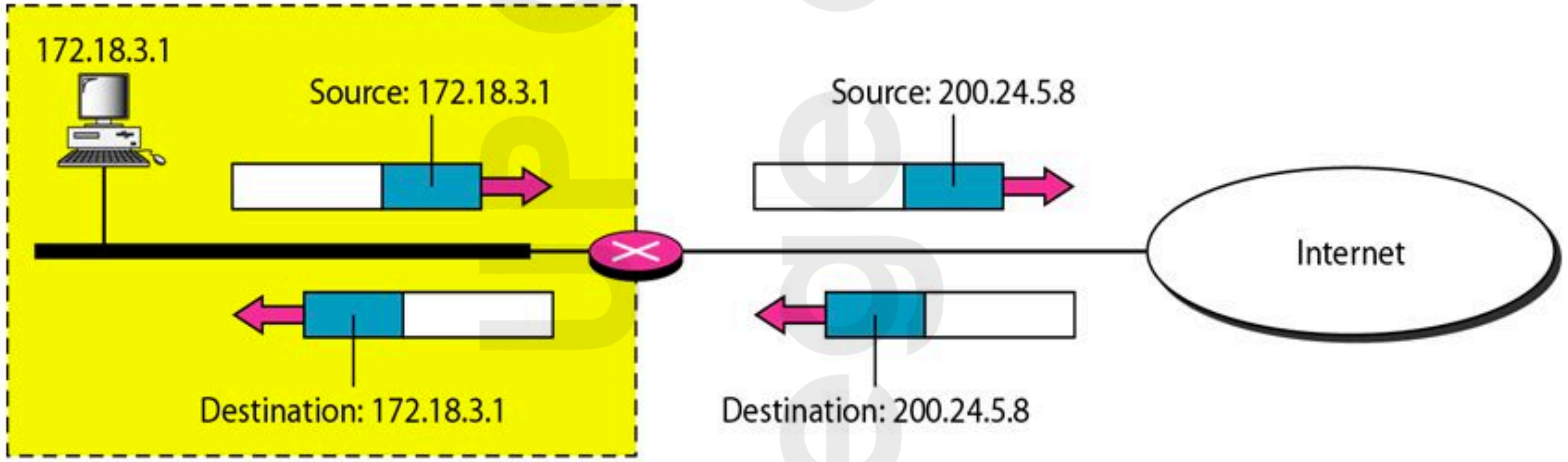
Range			Total
10.0.0.0	to	10.255.255.255	2^{24}
172.16.0.0	to	172.31.255.255	2^{20}
192.168.0.0	to	192.168.255.255	2^{16}



- Can use any address in the set without permission from the Internet authorities.
- Unique inside the organization but not globally.
- No router will forward packet with one of these addresses as destination address.
- The site must have only one single connection to the global Internet through a router that runs the NAT software.
- The router that connects the network to the global address uses one private address and one global address.

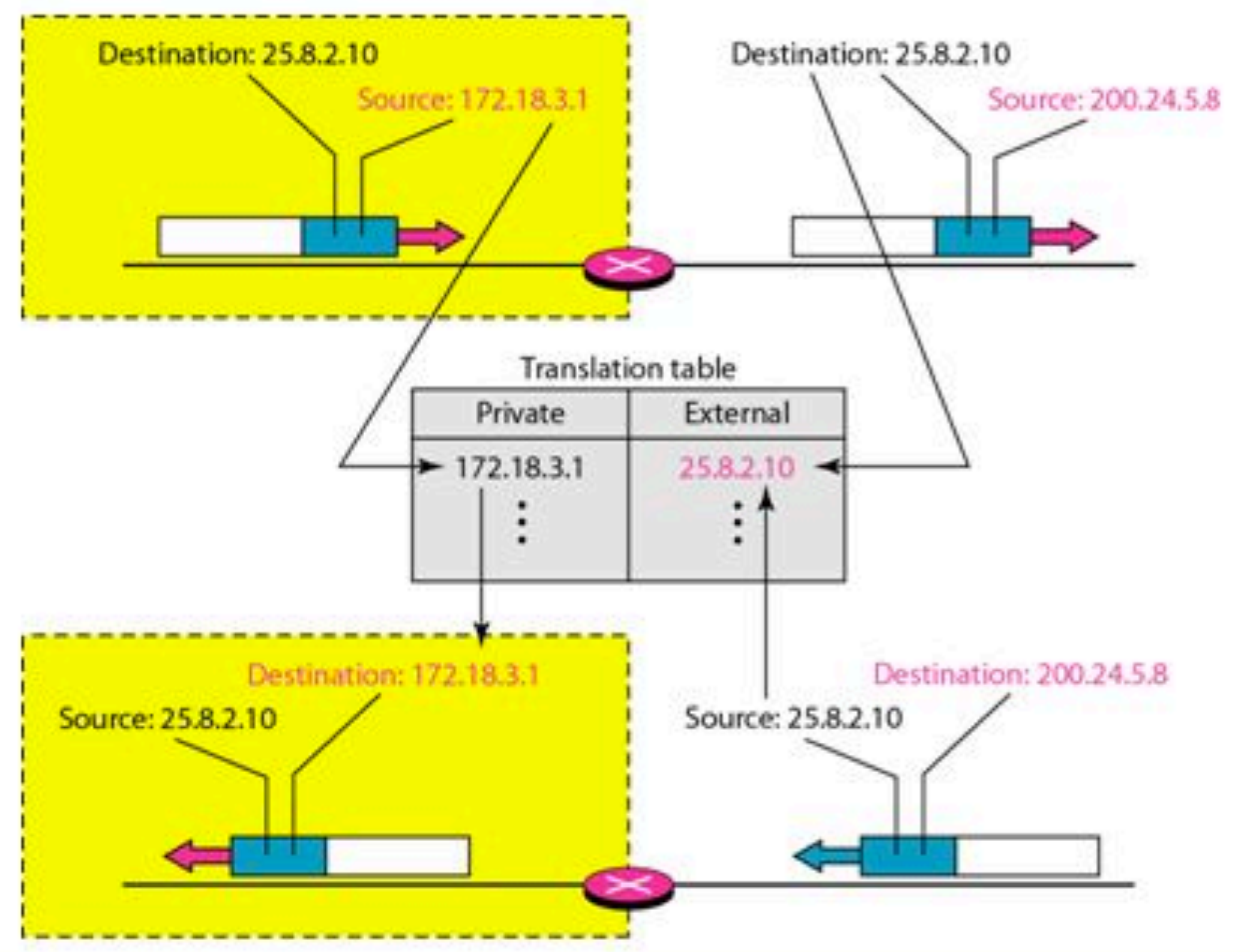
Address Translation

- All the **outgoing packets** go through the NAT router, which replaces the source address in the packet with the global NAT address.
- All **incoming packets** also pass through the NAT router, which replaces the destination address in the packet (the NAT router global address) with the appropriate private address.

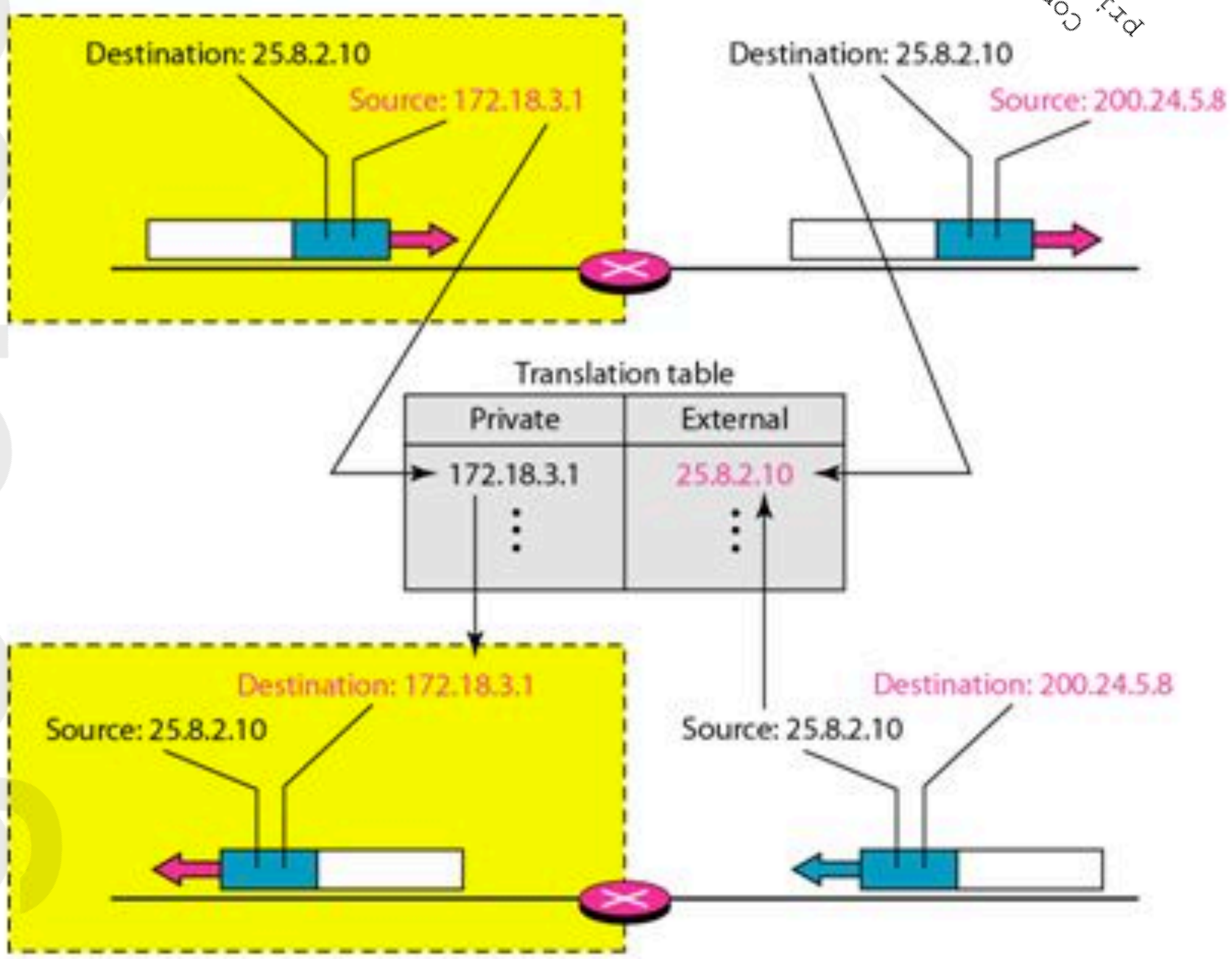


Translation Table

- Has only two columns: private address and external address (destination address of the packet).
- When the router translates the source address of the outgoing packet, it also make note of the destination address - where the packet is going.
- When response comes back from destination, the router uses the source address of the packet (as the external address) to find the private address of the packet.
- Always client-initiated.



What’s the problem with having one global address?



- NAT routers use a pool of global addresses.
- For example, instead of using only one global address (200.24.5.8), the NAT router can use four addresses (200.24.5.8, 200.24.5.9, 200.24.5.10, and 200.24.5.11).
- Here, four private network hosts can communicate with the same external host at the same time.
- Each pair of addresses defines a connection.

Problem?

- No more than 4 connections can be made to the same destination.
- No private-network host can access two external server programs (e.g. HTTP and FTP) at the same time.

Hence, Port Numbers

- Five-column translation table allowing many-to-many relationship between private-network hosts and external server programs.

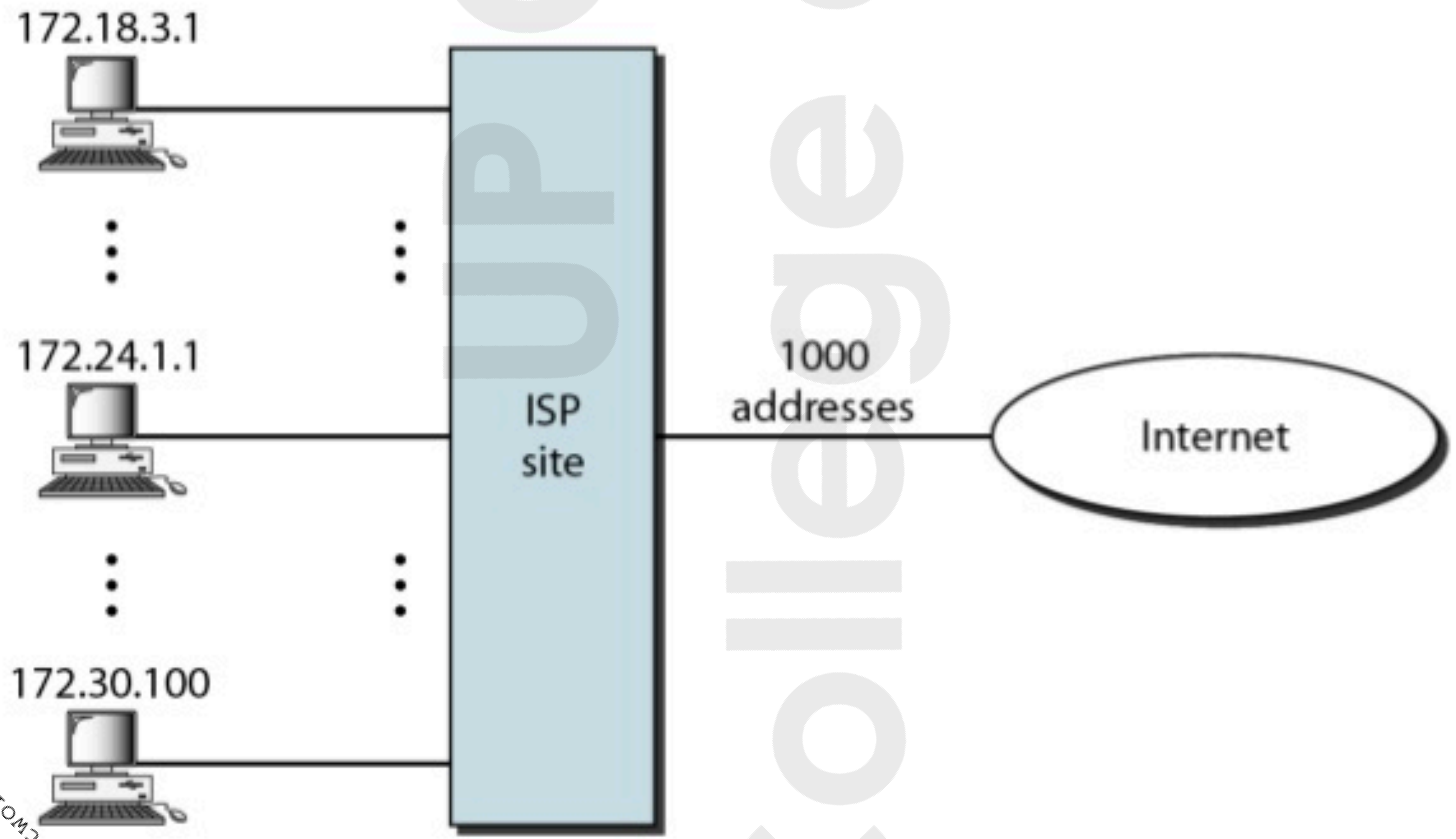
Table 19.4 *Five-column translation table*

Private Address	*Private Port*	*External Address*	*External Port*	*Transport Protocol*
172.18.3.1	1400	25.8.3.2	80	TCP
172.18.3.2	1401	25.8.3.2	80	TCP
. . .	. . .	. . .	. . .	. . .

- Private port numbers are unique in each connection.

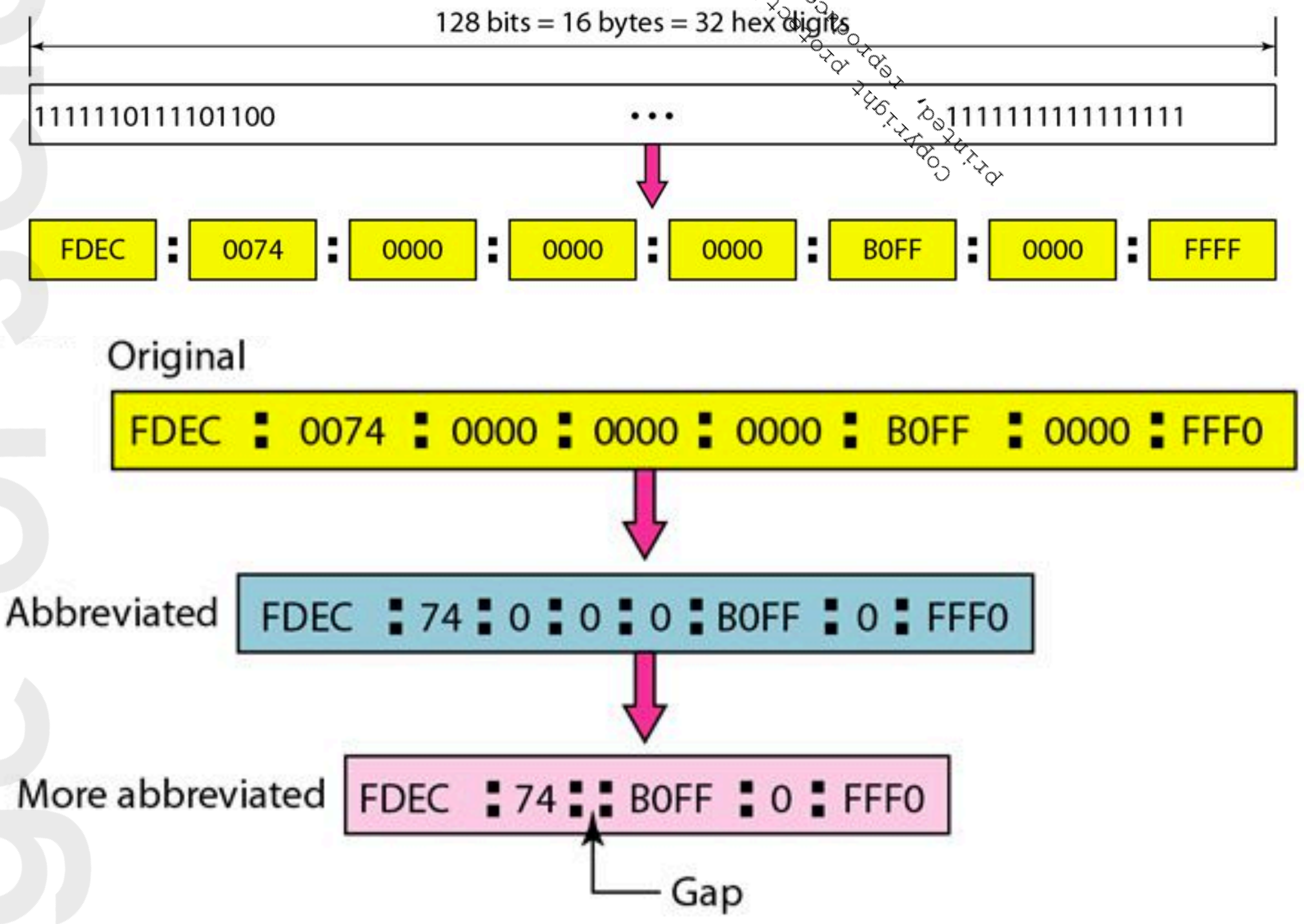
NAT and ISP

An ISP that serves dial-up customers can use NAT technology to conserve addresses. For example, suppose an ISP is granted 1000 addresses, but has 100,000 customers. Each of the customers is assigned a private network address. The ISP translates each of the 100,000 source addresses in outgoing packets to one of the 1000 global addresses; it translates the global destination address in incoming packets to the corresponding private address. Figure 19.13 shows this concept.



IPv6 Addresses

- Motivation: (a) long-term problem of address depletion of classless addressing, DHCP, and NAT and (b) lack of accommodation for real-time audio and video transmission, and encryption and authentication of data for some applications.
- Structure: Consists of 16 bytes (octets), i.e. 128 bits long.
- Hexadecimal colon notation: 128 bits divided into eight sections, each 2 bytes long (4 hex digits).
- Abbreviation: Many leading zeros can be omitted. Not applicable to trailing zeros.
- More abbreviation: Consecutive sections consisting of zeros only, remove zeros altogether and replace with double semicolon. Allowed only once per address.
- Re-expansion: Align the unabbreviated portions and insert zeros to get the original expanded address.



Dynamic Host Configuration Protocol (DHCP) is a client/server protocol that automatically provides an Internet Protocol (IP) host with its IP address and other related configuration information such as the subnet mask and default gateway.

Example 19.11

Expand the address 0:15::1:12:1213 to its original.

Solution

We first need to align the left side of the double colon to the left of the original pattern and the right side of the double colon to the right of the original pattern to find now many 0s we need to replace the double colon.

xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

0: 15: 1: 12: 1213

This means that the original address is

0000:0015:0000:0000:0000:0001 :0012: 1213

Address Space

- 2^{128} addresses
- 340,282,366,920,938,463,463,374,607,431,768,211,456 addresses

Address Space

The designers of IPv6 divided the address into several categories. A few leftmost bits, called the type prefix.

Type Prefix	Type	Fraction
0000 0000	Reserved	1/256
0000 0001	Unassigned	1/256
0000 001	ISO network addresses	1/128
0000 010	IPX (Novell) network addresses	1/128
0000 011	Unassigned	1/128
0000 1	Unassigned	1/32
0001	Reserved	1/16
001	Reserved	1/8
010	**Provider-based unicast addresses**	**1/8**

Table 19.5 *Type prefixes for IPv6 addresses*

Table 19.5 *Type prefixes for IPv6 addresses (continued)*

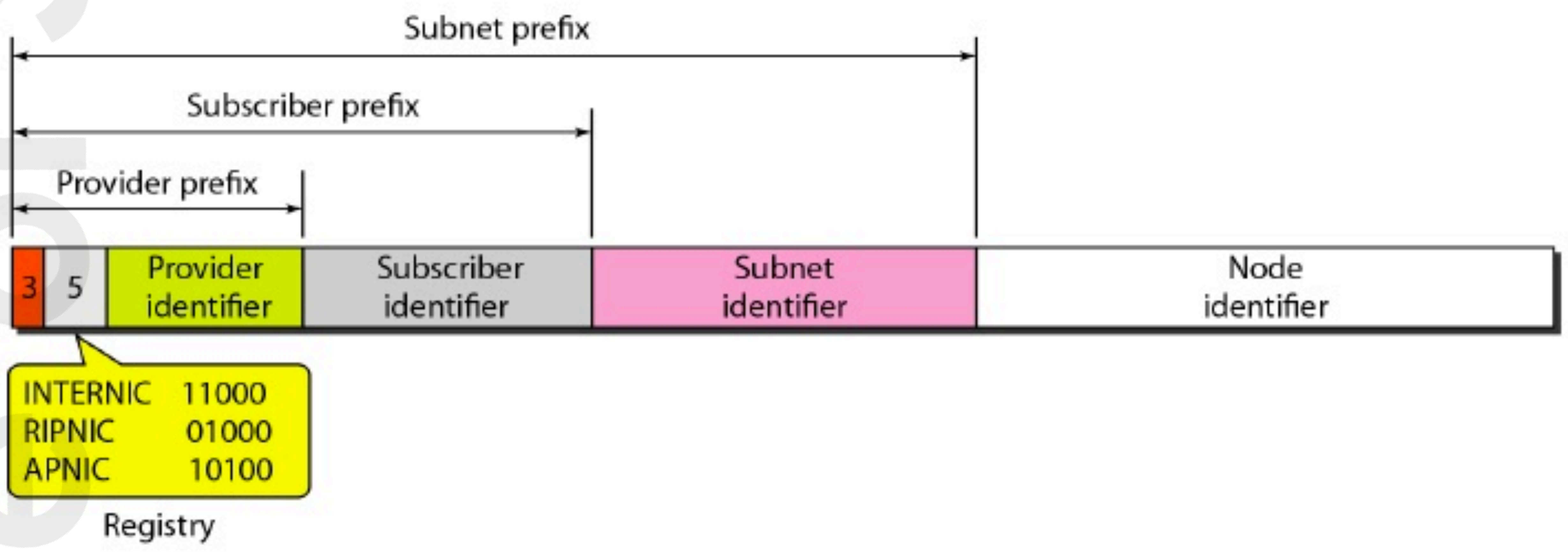
Type Prefix	Type	Fraction
011	Unassigned	1/8
100	Geographic-based unicast addresses	1/8
101	Unassigned	1/8
110	Unassigned	1/8
1110	Unassigned	1/16
1111 0	Unassigned	1/32
1111 10	Unassigned	1/64
1111 110	Unassigned	1/128
1111 1110 0	Unassigned	1/512
1111 1110 10	Link local addresses	1/1024
1111 1110 11	Site local addresses	1/1024
1111 1111	Multicast addresses	1/256

Unicast Addresses

- Defines a single computer.
- Packet sent to a unicast address must be delivered to that specific computer.
- Two types of unicast addresses:
    1. Geographically-based
    2. Provider-based

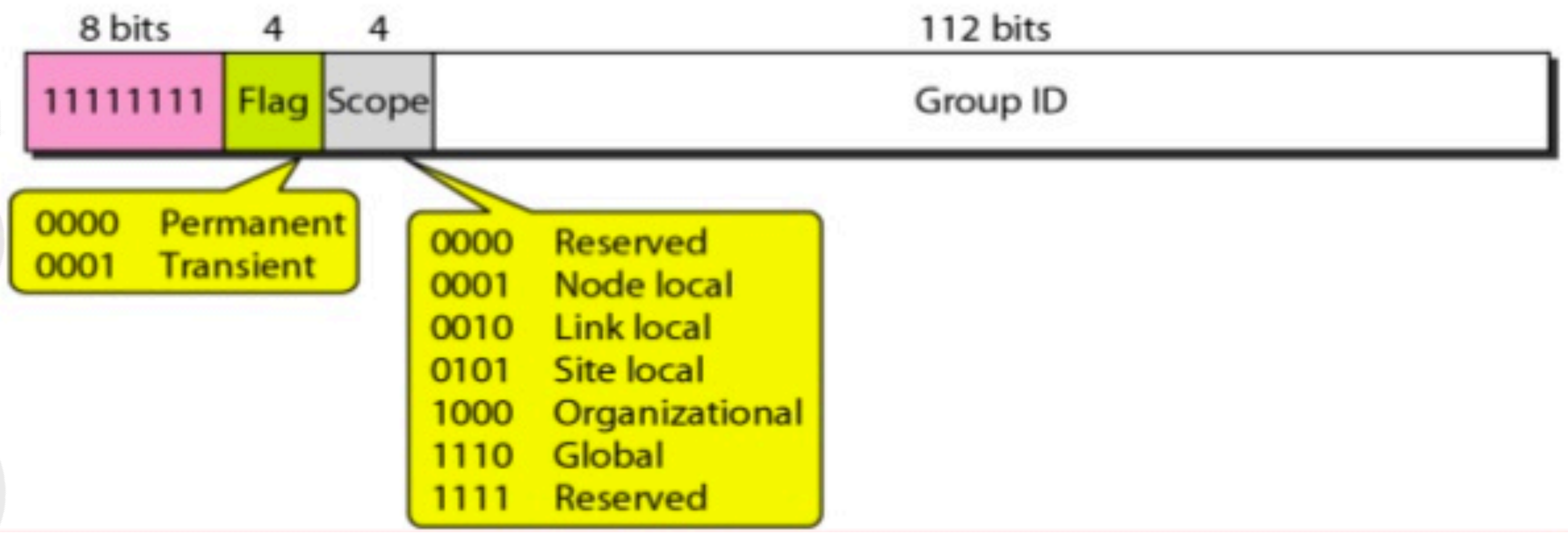
Prefixes for Provider-based Unicast Address

- Type Identifier (3 bits) - Defines the address as a provider-based address.
- Registry Identifier (5 bits) - Indicates the agency that has registered address. INTERNIC (code 11000 North America), RIPNIC (code 01000 Europe), APNIC (code 10100 Asia and Pacific).
- Provider Identifier (variable-length) - Identifies provider for Internet access (ISP). 16 bits recommended.
- Subscriber Identifier - Subscriber identification. 24 bits recommended.
- Subnet Identifier - Subscriber's subnetwork. 32 bits recommended.
- Node Identifier - Subnet's node. 48 bits recommended.



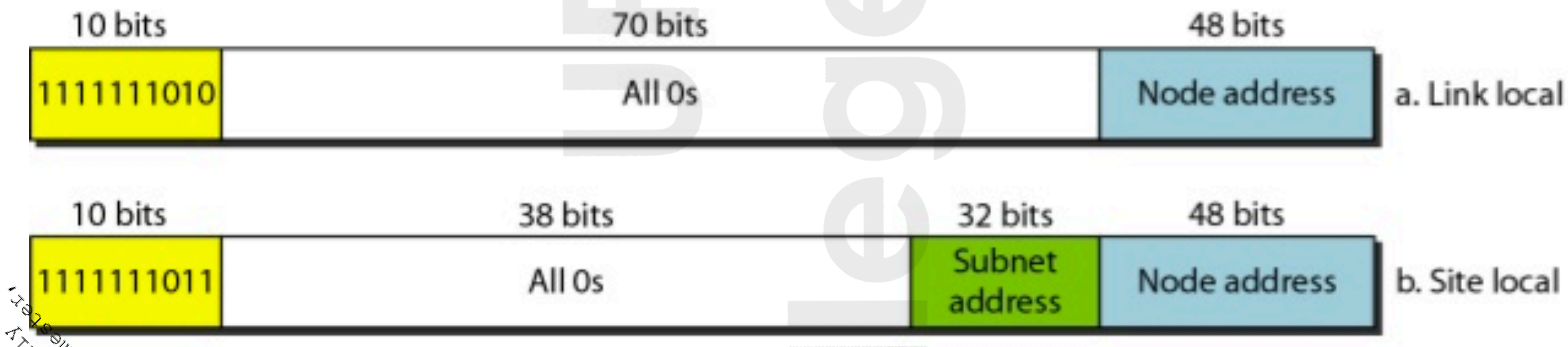
Multicast Addresses

- Used to define a group of hosts instead of just one.
- A packet sent to a multicast address must be delivered to each member of the group.
- Flag - defines the group address as either permanent or transient.
  - Permanent - Defined by Internet authorities.
  - Transient - Temporary. Teleconference use transient.
- Scope - Defines scope of group address.



Local Addresses

- Used when an organization wants to use IPv6 protocol without being connected to the global Internet.
- Provide addressing for private networks.
- Two types:
  - Link Local Address - Used in an isolated subnet.
  - Site Local Address - Use in an isolated site with several subnets.

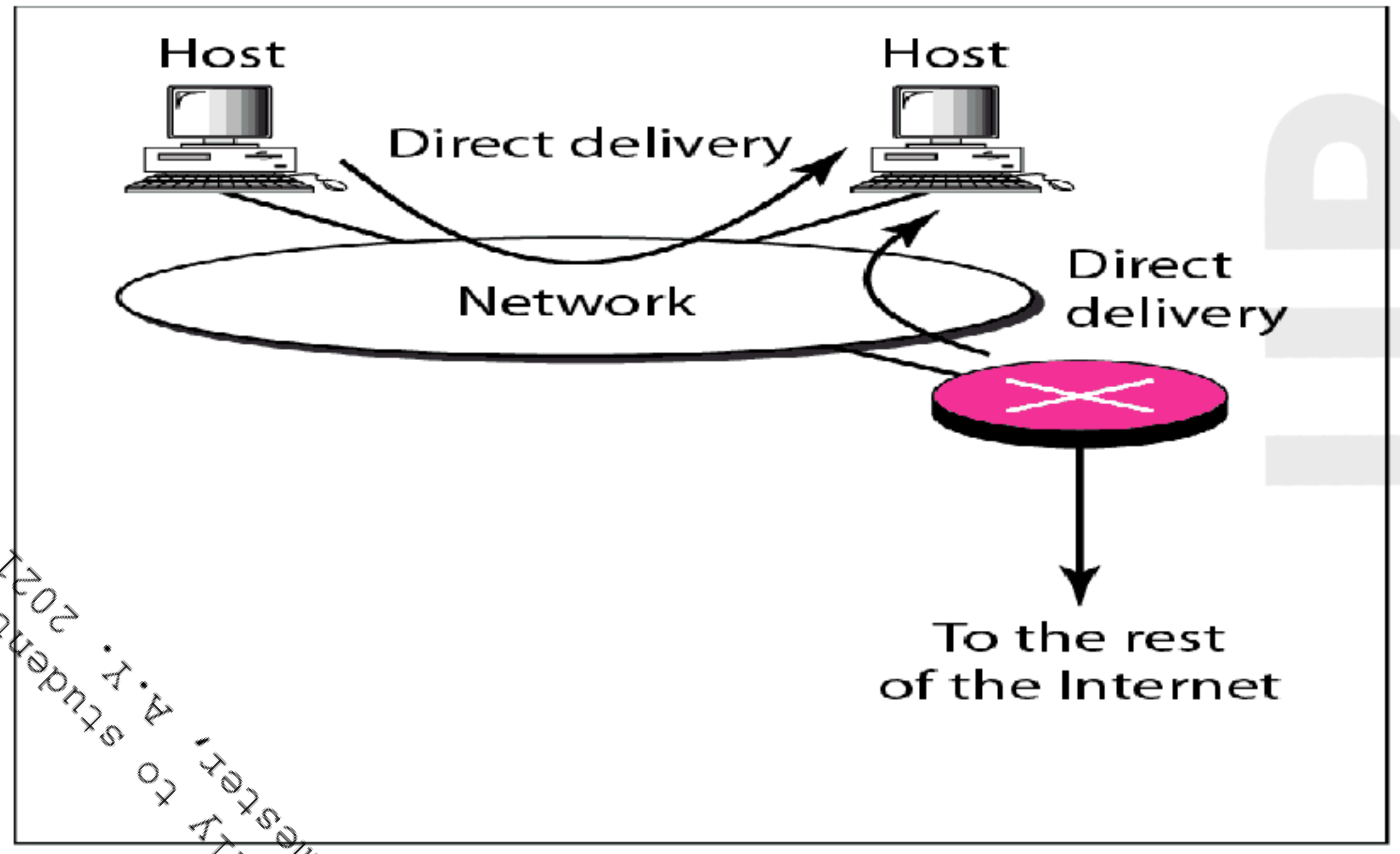


Delivery, Forwarding, and Routing of Packets

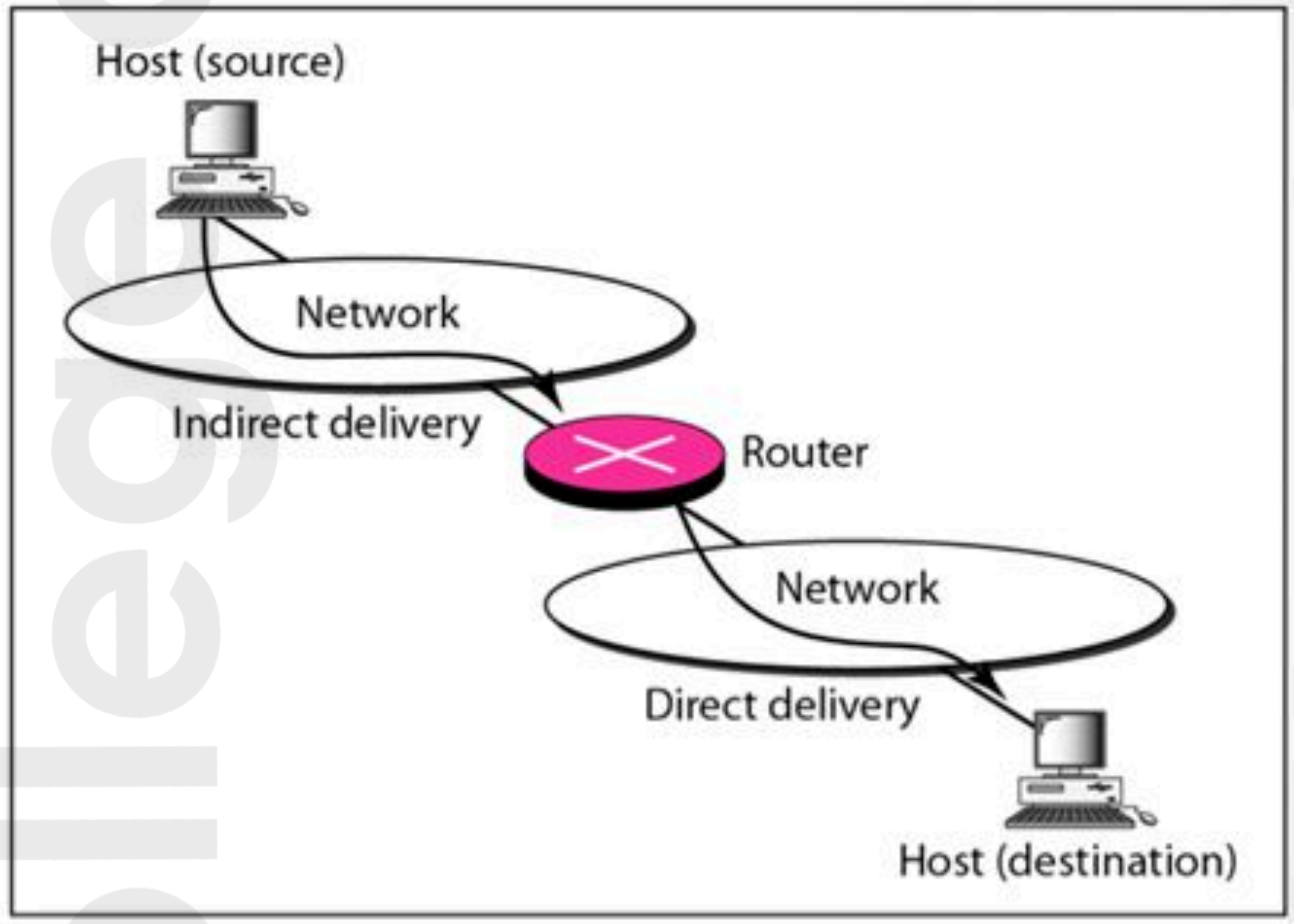
- Delivery - Refers to the way a packet is handled by the underlying networks under the control of the network layer.
- Forwarding - Refers to the way a packet is delivered to the next station.
- Routing - Refers to the way routing tables are created to help in forwarding.

Delivery

- Direct Delivery
  - The final destination of the packet is a host connected to the same physical network as the deliverer.
- Indirect Delivery
  - If the destination host is not on the same network as the deliverer.



a. Direct delivery



b. Indirect and direct delivery

Forwarding

- To place the packet in its route to its destination.
- Requires a host or a router to have a routing table.
- When a host has a packet to send or when a router has received a packet to be forwarded, it looks at the routing table to find the route to the final destination.
- However, this simple solution is impossible today in an internetwork such as the Internet since the number of entries needed in the routing table would make table lookups inefficient.
- Forwarding techniques:
  1. Next-Hop Method versus Route Method
  2. Network-Specific Method versus Host-Specific Method
  3. Default Method

1. Next-Hop Method versus Route Method

- Next-Hop Method is one technique to reduce contents of a routing table which holds only the address of the next hop instead of information about the complete route (Route Method).

a. Routing tables based on route

b. Routing tables based on next hop

Routing table for host A

Destination	Route
Host B	R1, R2, host B

Destination	Next hop
Host B	R1

Routing table for R1

Destination	Route
Host B	R2, host B

Destination	Next hop
Host B	R2

Routing table for R2

Destination	Route
Host B	Host B

Destination	Next hop
Host B	---

2. Network-Specific Method versus Host-Specific Method

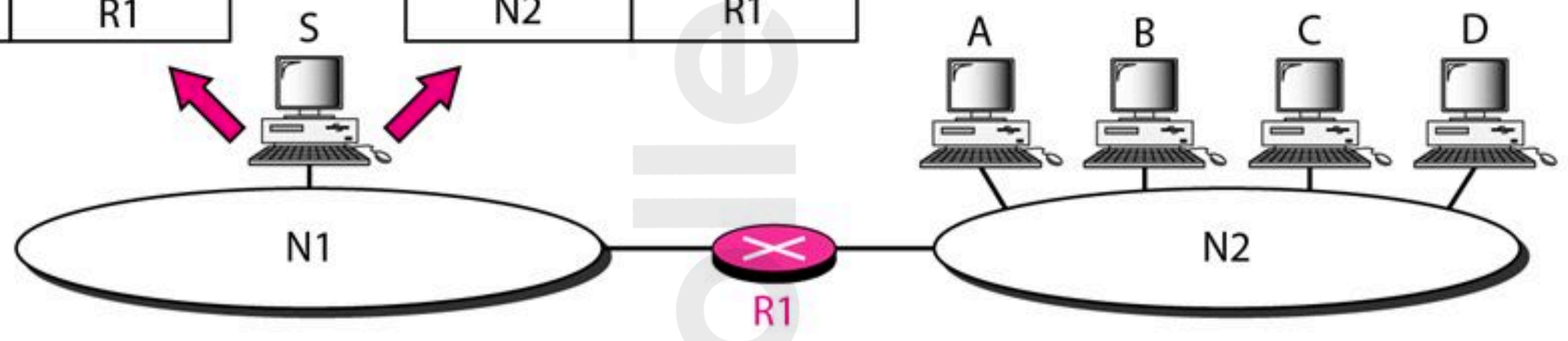
- Network-Specific Method is a second technique to reduce the routing table and simplify the searching process. Here, instead of having an entry for every destination host connected to the same physical network (Host-Specific Method), there is only one entry that defines the address of the destination network itself. In other words, all hosts connected to the same network are considered as one entity.

Routing table for host S based on host-specific method

Destination	Next hop
A	R1
B	R1
C	R1
D	R1

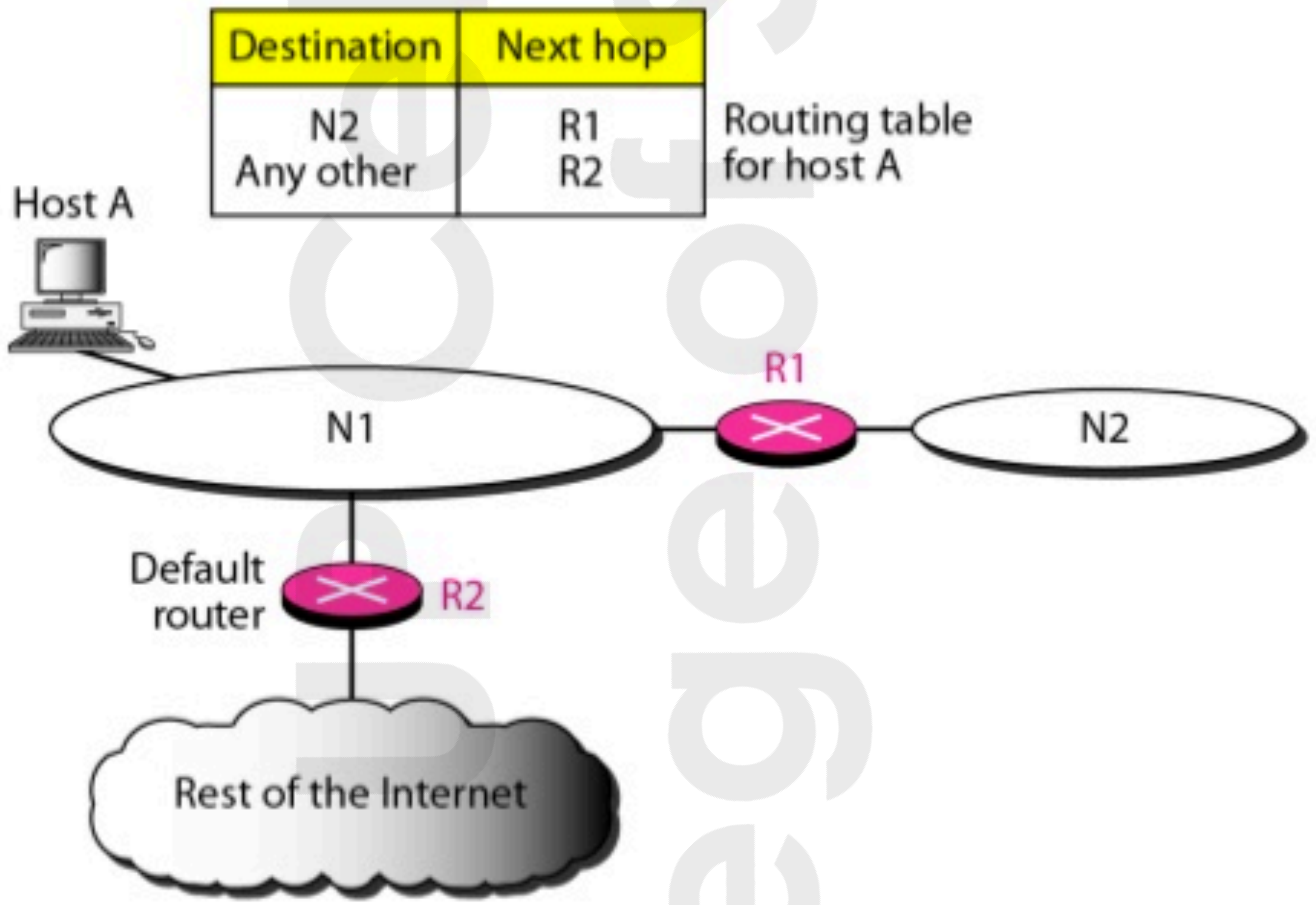
Routing table for host S based on network-specific method

Destination	Next hop
N2	R1



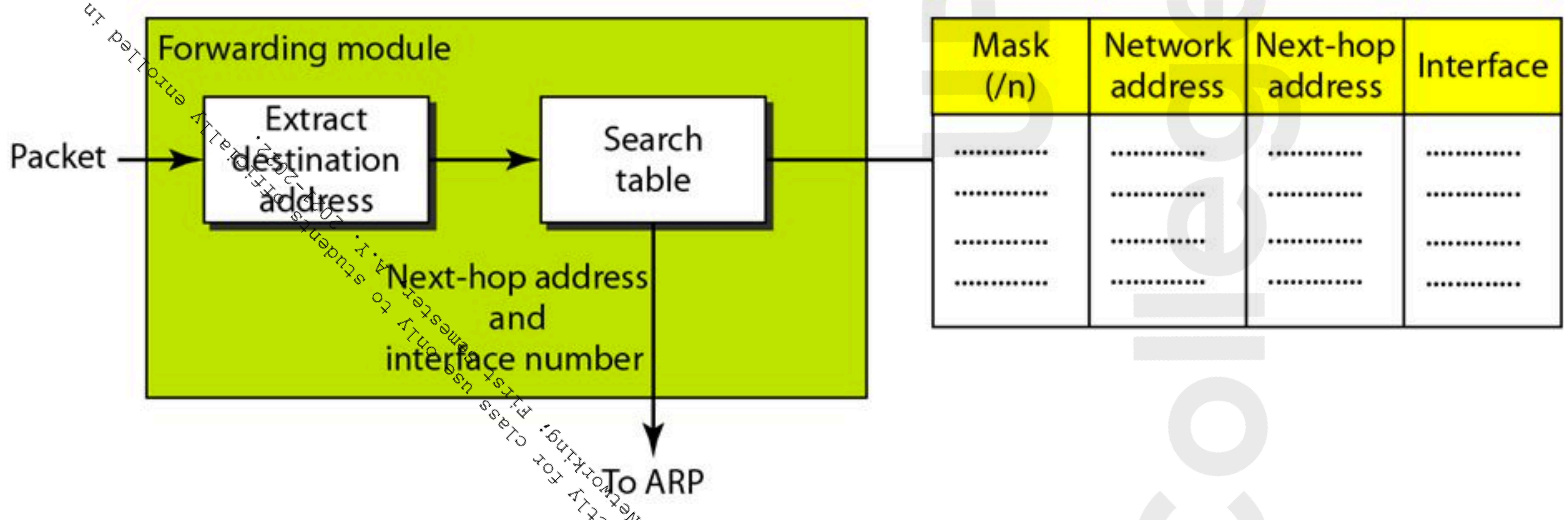
3. Default Method

- Here, one router is considered as default gateway to the Internet.



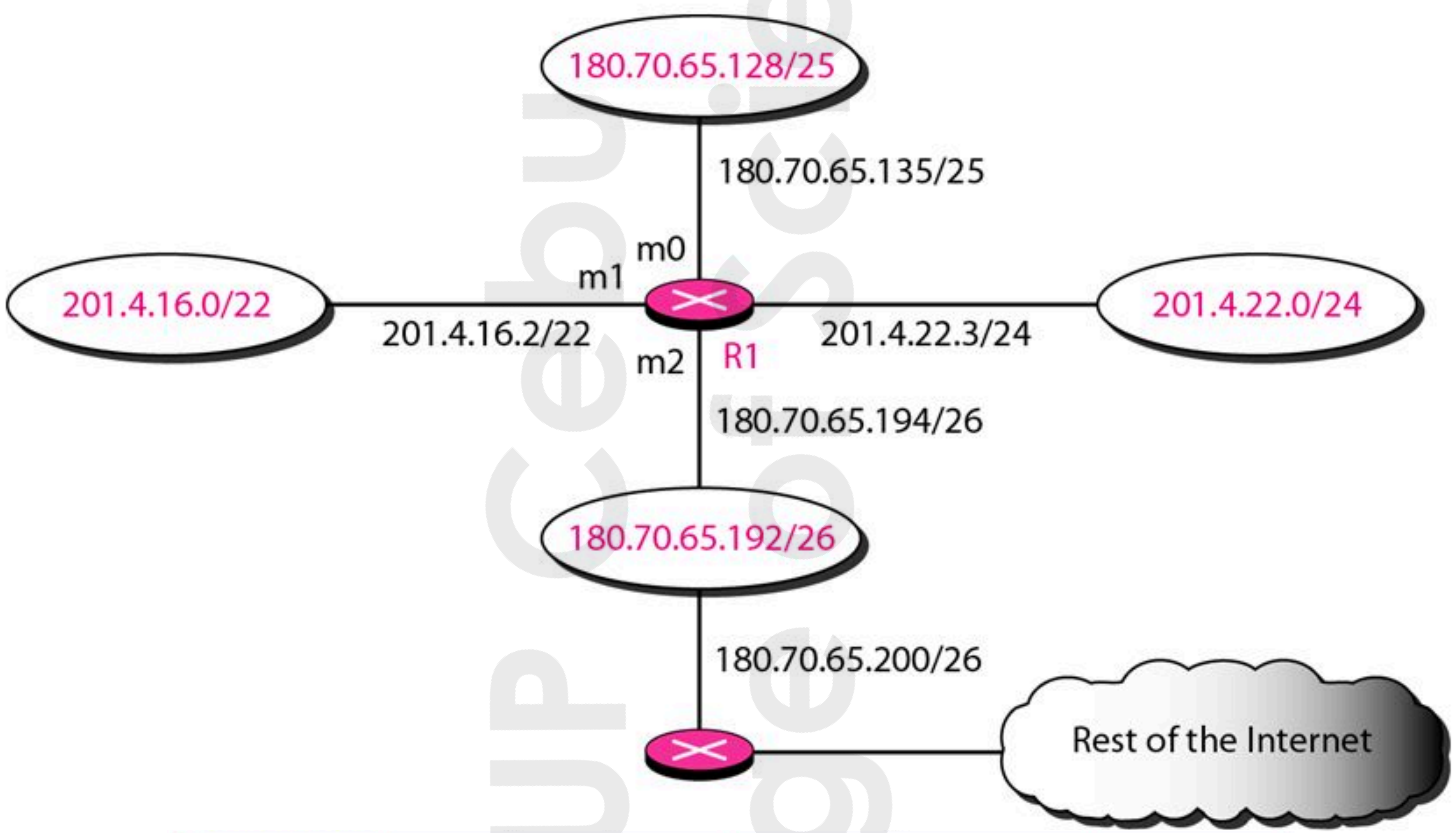
Forwarding Process

- Classless addressing.
- Routing table needs to have one row of information for each block involved.
- The table needs to be searched based on the network address (first address in the block.)
- Need to include mask (/n) in the table to give clue about the network address.



- **Address Resolution Protocol (ARP)** is a protocol for mapping an Internet Protocol address (IP address) to a physical machine address that is recognized in the local network.
- In an Ethernet local area network, addresses for attached devices are 48 bits long. (The physical machine address is also known as a Media Access Control or MAC address.)
- A table, usually called the ARP cache, is used to maintain a correlation between each MAC address and its corresponding IP address.

Example: Make a routing table for Router R1 below.



Mask	*Network Address*	*Next Hop*	*Interface*
/26	180.70.65.192	—	m2
/25	180.70.65.128	—	m0
/24	201.4.22.0	—	m3
/22	201.4.16.0	....	m1
Any	Any	180.70.65.200	m2

- Example 22.2 - Show the forwarding process if a packet arrives at R1 with the destination address of 180.70.65.140.

1. The first mask (/26) is applied to the destination address. The result is 180.70.65.128, which does not match the corresponding network address.
2. The second mask (/25) is applied to the destination address. The result is 180.70.65.128, which matches the corresponding network address. The next-hop address (the destination address of the packet in this case) and the interface number m0 are passed to ARP for further processing.

Mask	*Network Address*	*Next Hop*	*Interface*
/26	180.70.65.192	—	m2
/25	180.70.65.128	—	m0
/24	201.4.22.0	—	m3
/22	201.4.16.0	....	m1
Any	Any	180.70.65.200	m2

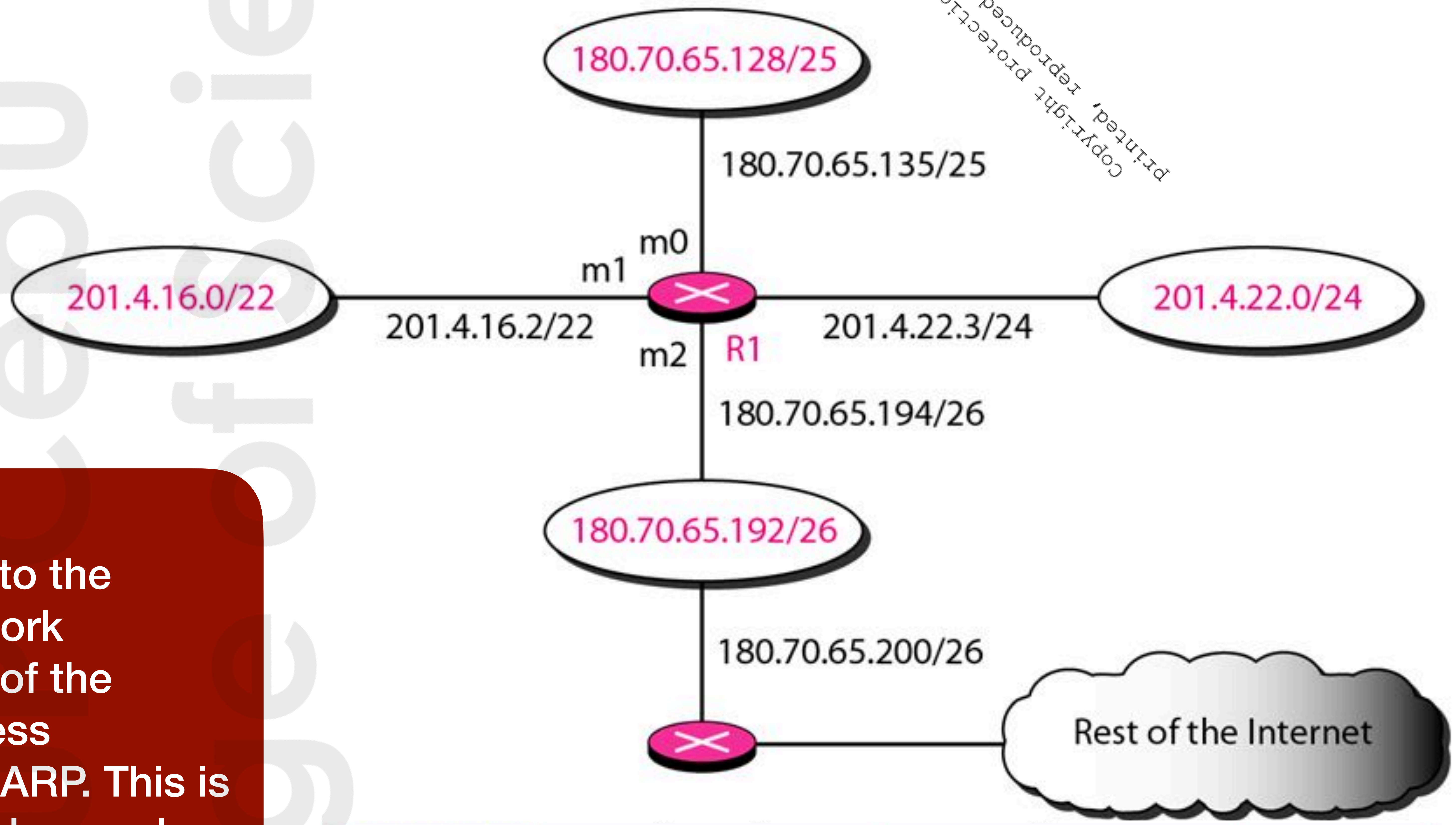
- Example 22.3 - Show the forwarding process if a packet arrives at R1 with the destination address 201.4.22.35.

1. The first mask (/26) is applied to the destination address. The result is 201.4.22.0, which does not match the corresponding network address.
2. The second mask (/25) is applied to the destination address. The result is 201.4.22.0, which does not match the corresponding network address.
3. The third mask (/24) is applied to the destination address. The result is 201.4.22.0, which matches the corresponding network address. The destination address of the packet and the interface number m3 are passed to ARP.

Mask	Network Address	Next Hop	Interface
/26	180.70.65.192	—	m2
/25	180.70.65.128	—	m0
/24	201.4.22.0	—	m3
/22	201.4.16.0	....	m1
Any	Any	180.70.65.200	m2

- Example 22.4 - Show the forwarding process if a packet arrives at R1 with the destination address 18.24.32.78.

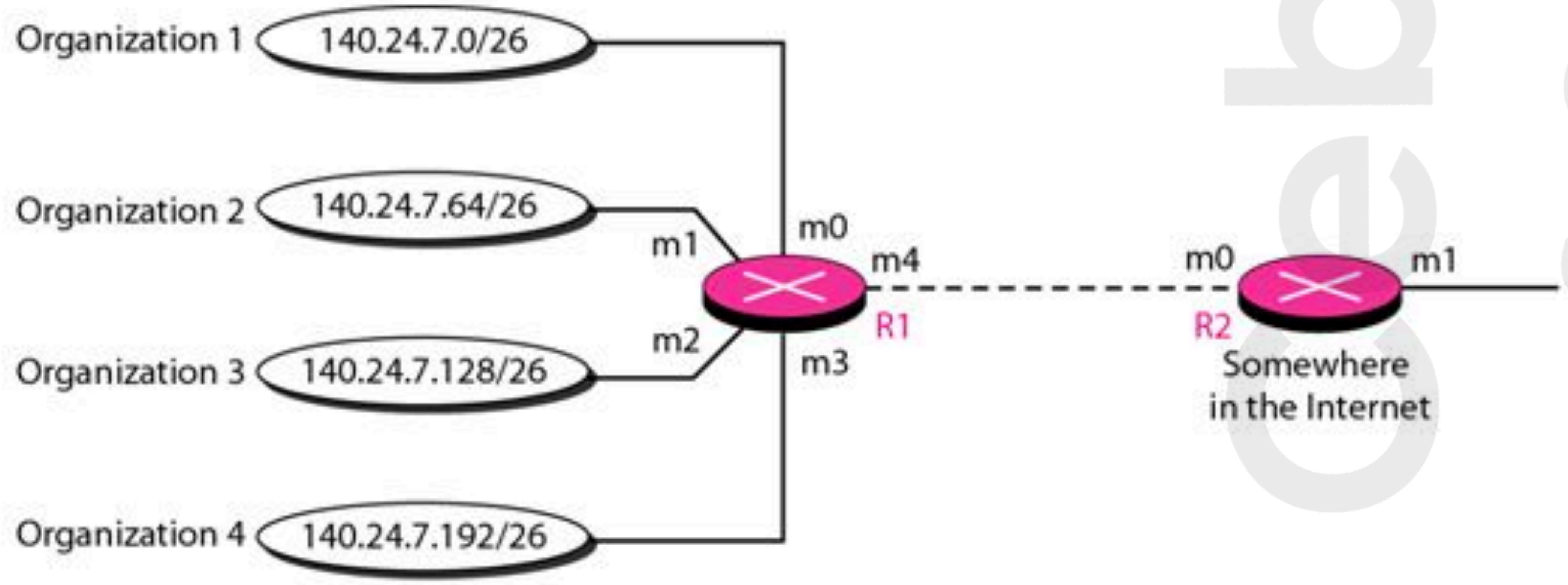
This time all mask are applied one by one, to the destination address, but no matching network address is found. When it reaches the end of the table, the module gives the next-hop address 180.70.65.200 and interface number m2 to ARP. This is probably an outgoing packet that needs to be send, via the default router, to someplace in the Internet.



Mask	*Network Address*	*Next Hop*	*Interface*
/26	180.70.65.192	—	m2
/25	180.70.65.128	—	m0
/24	201.4.22.0	—	m3
/22	201.4.16.0	....	m1
Any	Any	180.70.65.200	m2

Address Aggregation

- Problem: Classless addressing increases the number of routing table entries.
- Solution: Address aggregation is designed in order to alleviate the problem.



Mask	Network address	Next-hop address	Interface
/26	140.24.7.0	----------	m0
/26	140.24.7.64	----------	m1
/26	140.24.7.128	----------	m2
/26	140.24.7.192	----------	m3
/0	0.0.0.0	Default	m4

Routing table for R1

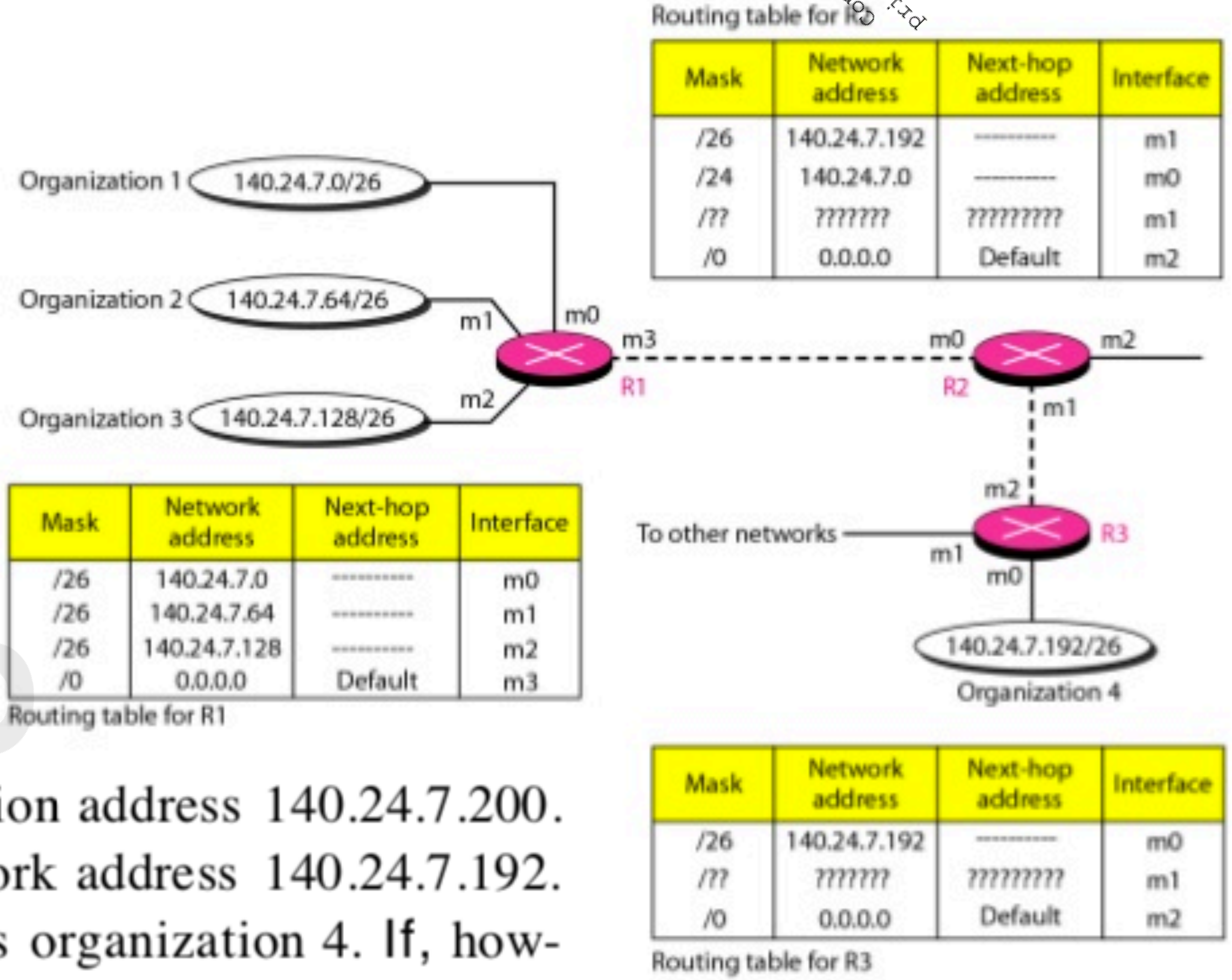
Mask	Network address	Next-hop address	Interface
/24	140.24.7.0	----------	m0
/0	0.0.0.0	Default	m1

Routing table for R2

Router R1 is connected to networks of four organizations that each use 64 addresses. Router R2 is somewhere far from R1. Router R1 has a longer routing table because each packet must be correctly routed to the appropriate organization. Router R2, on the other hand, can have a very small routing table. For R2, any packet with destination 140.24.7.0 to 140.24.7.255 is sent out from interface m0 regardless of the organization number. This is called address aggregation because the blocks of addresses for four organizations are aggregated into one larger block. Router R2 would have a longer routing table if each organization had addresses that could not be aggregated into one block.

Longest Mask Matching

- Principle that states that the routing table is sorted from the longest mask to the shortest mask.
- Allows routing of organizations not geographically close to the others hence cannot be connected to the same router for some reason.



Routing table for R2

Mask	Network address	Next-hop address	Interface
/26	140.24.7.192	----------	m1
/24	140.24.7.0	----------	m0
/??	???????	?????????	m1
/0	0.0.0.0	Default	m2

Mask	Network address	Next-hop address	Interface
/26	140.24.7.0	----------	m0
/26	140.24.7.64	----------	m1
/26	140.24.7.128	----------	m2
/0	0.0.0.0	Default	m3

Routing table for R1

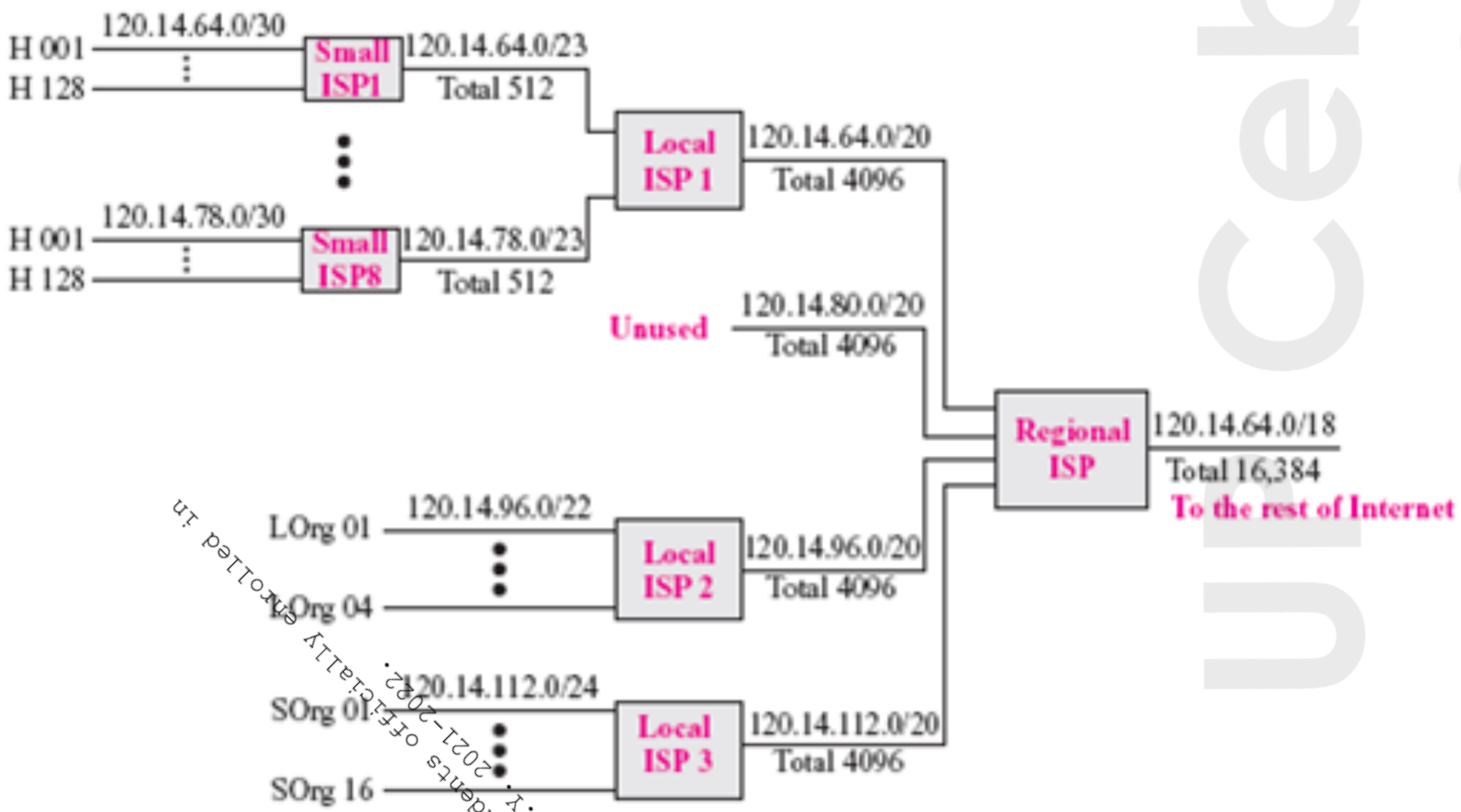
Mask	Network address	Next-hop address	Interface
/26	140.24.7.192	----------	m0
/??	???????	?????????	m1
/0	0.0.0.0	Default	m2

Routing table for R3

Suppose a packet arrives for organization 4 with destination address 140.24.7.200. The first mask at router R2 is applied, which gives the network address 140.24.7.192. The packet is routed correctly from interface ml and reaches organization 4. If, however, the routing table was not stored with the longest prefix first, applying the /24 mask would result in the incorrect routing of the packet to router R1.

Hierarchical Routing

- Create a sense of hierarchy to routing tables.



As an example of hierarchical routing, let us consider Figure 22.9. A regional ISP is granted 16,384 addresses starting from 120.14.64.0. The regional ISP has decided to divide this block into four subblocks, each with 4096 addresses. Three of these subblocks are assigned to three local ISPs; the second subblock is reserved for future use. Note that the mask for each block is *120* because the original block with mask /18 is divided into 4 blocks.

The first local ISP has divided its assigned subblock into 8 smaller blocks and assigned each to a small ISP. Each small ISP provides services to 128 households (HOOI to HI28), each using four addresses. Note that the mask for each small ISP is now *123* because the block is further divided into 8 blocks. Each household has a mask of *130*, because a household has only four addresses (2^{32-30} is 4).

The second local ISP has divided its block into 4 blocks and has assigned the addresses to four large organizations (LOrg01 to LOrg04). Note that each large organization has 1024 addresses, and the mask is /22.

The third local ISP has divided its block into 16 blocks and assigned each block to a small organization (SOrg01 to SOrg16). Each small organization has 256 addresses, and the mask is /24.

There is a sense of hierarchy in this configuration. All routers in the Internet send a packet with destination address 120.14.64.0 to 120.14.127.255 to the regional ISP.

The regional ISP sends every packet with destination address 120.14.64.0 to 120.14.79.255 to local ISP1. Local ISP1 sends every packet with destination address 120.14.64.0 to 120.14.64.3 to H001.

Geographical Routing

- Divide the entire address space into a few large blocks.
- Assign a block each to North America, Europe, Asia, Africa, and so on.
- Routers of ISPs outside Europe will have only one entry of packets to Europe in their routing tables and similarly with others.

How do you think the routing table is created?

Routing Table

- Every host or a router has a routing table with an entry for each destination, or a combination of destinations, to route IP packets.
- The routing table can either be:
  1. **Static Routing Table** - Contains information entered manually by the administrator. Applicable to small internet but not in big internet such as the Internet.
  2. **Dynamic Routing Table** - Updated periodically by using one of the dynamic routing protocols such as RIP (Routing Information Protocol), OSPF (Open Shortest Path First), or BGP (Border Gateway Protocol). When there's change in Internet (router shutdown or link break), protocols update all tables in the routers automatically.

Format

- Minimum of four columns.
- Vendor-dependent.

Mask	Network address	Next-hop address	Interface	Flags	Reference count	Use
....	....	....	....	....	....	....

Mask. This field defines the mask applied for the entry.

Network address. This field defines the network address to which the packet is finally delivered. In the case of host-specific routing, this field defines the address of the destination host.

Next-hop address. This field defines the address of the next-hop router to which the packet is delivered.

Interface. This field shows the name of the interface.

Flags. This field defines up to five flags. Flags are on/off switches that signify either presence or absence. The five flags are U (up), G (gateway), H (host-specific), D (added by redirection), and M (modified by redirection).

a. U (up). The U flag indicates the router is up and running. If this flag is not present, it means that the router is down. The packet cannot be forwarded and is discarded.

b. G (gateway). The G flag means that the destination is in another network. The packet is delivered to the next-hop router for delivery (indirect delivery). When this flag is missing, it means the destination is in this network (direct delivery).

c. H (host-specific). The H flag indicates that the entry in the network address field is a host-specific address. When it is missing, it means that the address is only the network address of the destination.

d. D (added by redirection). The D flag indicates that routing information for this destination has been added to the host routing table by a redirection message from ICMP. We discussed redirection and the ICMP protocol in Chapter 21.

e. M (modified by redirection). The M flag indicates that the routing information for this destination has been modified by a redirection message from ICMP. We discussed redirection and the ICMP protocol in Chapter 21.

Reference count. This field gives the number of users of this route at the moment. For example, if five people at the same time are connecting to the same host from this router, the value of this column is 5.

Use. This field shows the number of packets transmitted through this router for the corresponding destination.

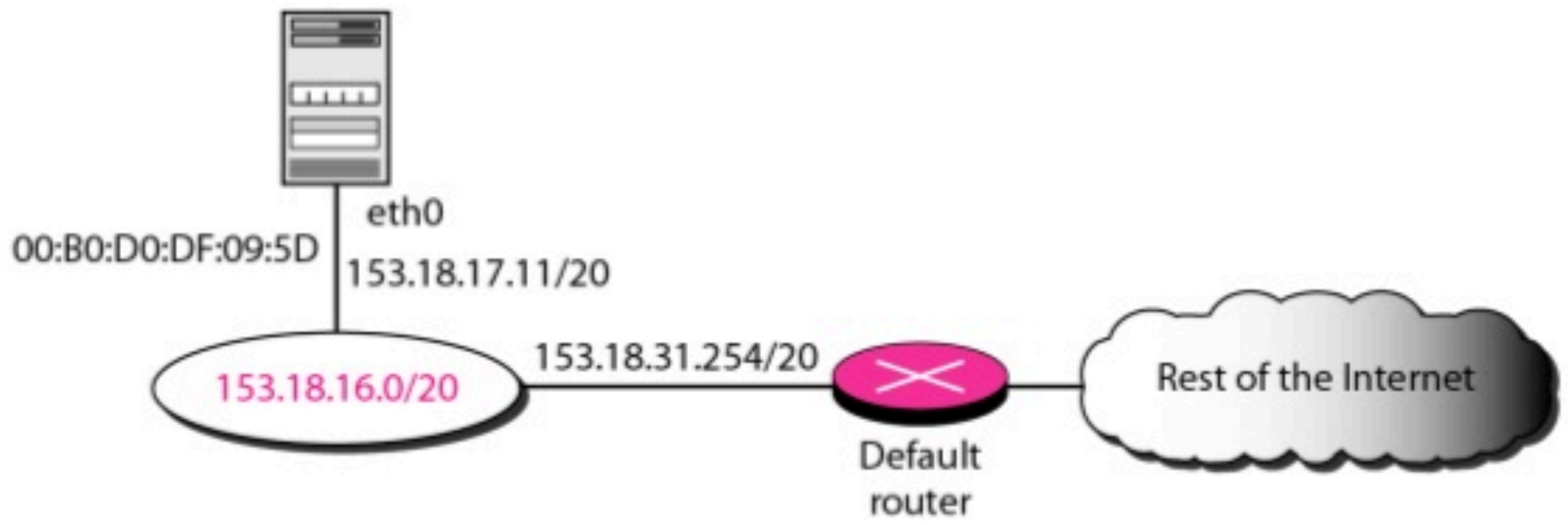
Utilities

One utility that can be used to find the contents of a routing table for a host or router is *netstat* in UNIX or LINUX. The following shows the list of the contents of a default server. We have used two options, r and n. The option r indicates that we are interested in the routing table, and the option n indicates that we are looking for numeric addresses. Note that this is a routing table for a host, not a router. Although we discussed the routing table for a router throughout the chapter, a host also needs a routing table.

$ netstat-rn

Kernel IP routing table

Destination	Gateway	Mask	Flags	Iface
153.18.16.0	0.0.0.0	255.255.240.0	U	ethO
127.0.0.0	0.0.0.0	255.0.0.0	U	10
0.0.0.0	153.18.31.254	0.0.0.0	UO	ethO



Note also that the order of columns is different from what we showed. The destination column here defines the network address. The term *gateway* used by UNIX is synonymous with *router*. This column acmally defines the address of the next hop. The value 0.0.0.0 shows that the delivery is direct. The last entry has a flag of G, which means that the destination can be reached through a router (default router). The *Iface* defines the interface. The host has only one real interface, ethO, which means interface 0 connected to an Ethernet network. The second interface, la, is actually a virmalloopback interface, indicating that the host accepts packets with loopback address 127.0.0.0.

More information about the IP address and physical address of the server can be found by using the *ifconfig* command on the given interface (ethO).

$ ifconfig ethO

ethO Link encap:Ethemet HWaddr 00:BO:DO:DF:09:5D

inet addr:153.18.17.11 Bcast: 153.18.31.255 Mask:255.255.240.0