IP Addresses: Classful Addressing

Objectives

Upon completion you will be able to:

- *Understand IPv4 addresses and classes*
- *Identify the class of an IP address*
- *Find the network address given an IP address*
- *Understand masks and how to use them*
- *Understand subnets and supernets*

4.1 INTRODUCTION

The identifier used in the IP layer of the TCP/IP protocol suite to identify each device connected to the Internet is called the Internet address or IP address. An IP address is a 32-bit address that uniquely and universally defines the connection of a host or a router to the Internet. IP addresses are unique. They are 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.

The topics discussed in this section include:

Address Space
Notation

Note:

An IP address is a 32-bit address.

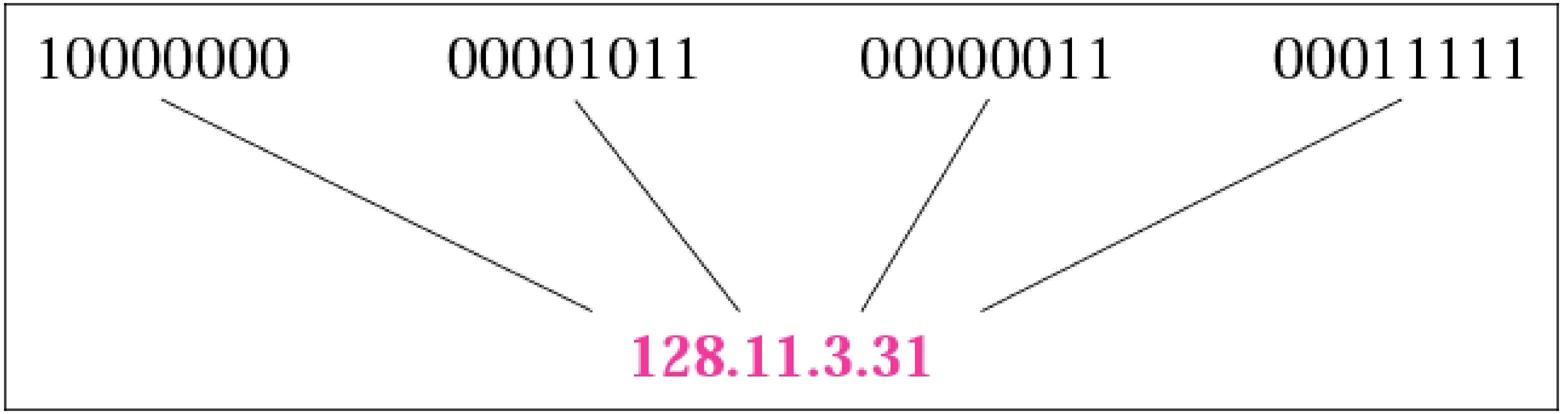
Note:

The IP addresses are unique.

Note:

The address space of IPv4 is 2^{32} or 4,294,967,296.

Figure 4.1 ***Dotted-decimal notation***



Note:

The binary, decimal, and hexadecimal number systems are reviewed in Appendix B.

Example 1

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

a. 10000001 00001011 00001011 11101111
b. 11000001 10000011 00011011 11111111
c. 11100111 11011011 10001011 01101111
d. 11111001 10011011 11111011 00001111

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***
c. 231.219.139.111 ***d. 249.155.251.15***

Example 2

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

***a.* 111.56.45.78**

***b.* 221.34.7.82**

***c.* 241.8.56.12**

***d.* 75.45.34.78**

Solution

We replace each decimal number with its binary equivalent:

***a.* 01101111 00111000 00101101 01001110**

***b.* 11011101 00100010 00000111 01010010**

***c.* 11110001 00001000 00111000 00001100**

***d.* 01001011 00101101 00100010 01001110**

Example 3

Find the error, if any, in the following IP 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 are no leading zeroes in dotted-decimal notation (045).

b. We may not have more than four numbers in an IP address.

c. In dotted-decimal notation, each number is less than or equal to 255; 301 is outside this range.

d. A mixture of binary notation and dotted-decimal notation is not allowed.

Example 4

Change the following IP addresses from binary notation to hexadecimal notation.

a.* *10000001 00001011 00001011 11101111

b.* *11000001 10000011 00011011 11111111

Solution

We replace each group of 4 bits with its hexadecimal equivalent (see Appendix B). Note that hexadecimal notation normally has no added spaces or dots; however, 0X (or 0x) is added at the beginning or the subscript 16 at the end to show that the number is in hexadecimal.

a. ***0X810B0BEF or*** $810B0BEF_{16}$

b. ***0XC1831BFF or*** $C1831BFF_{16}$

4.2 CLASSFUL ADDRESSING

IP addresses, when started a few decades ago, used the concept of classes. This architecture is called classful addressing. In the mid-1990s, a new architecture, called classless addressing, was introduced and will eventually supersede the original architecture. However, part of the Internet is still using classful addressing, but the migration is very fast.

The topics discussed in this section include:

Recognizing Classes
Netid and Hostid
Classes and Blocks
Network Addresses
Sufficient Information
Mask
CIDR Notation
Address Depletion

Figure 4.2 *Occupation of the address space*

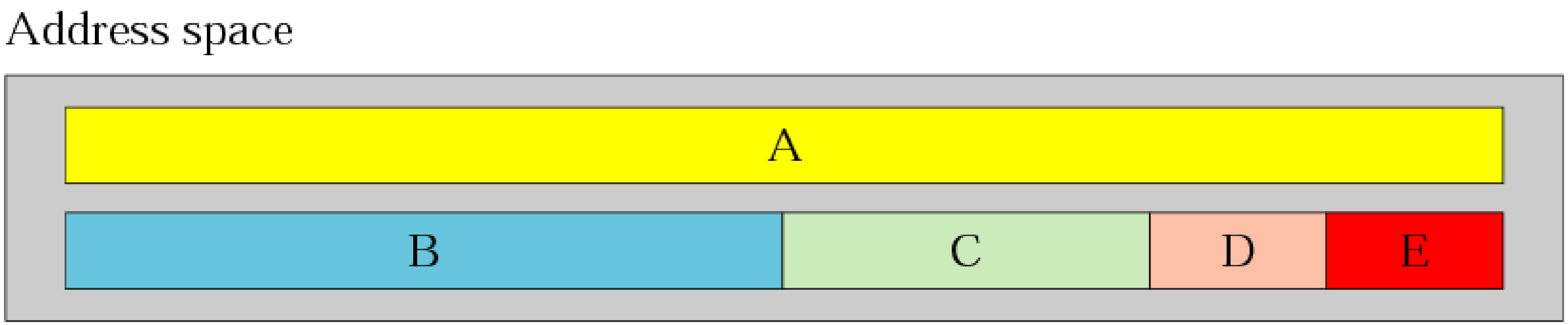


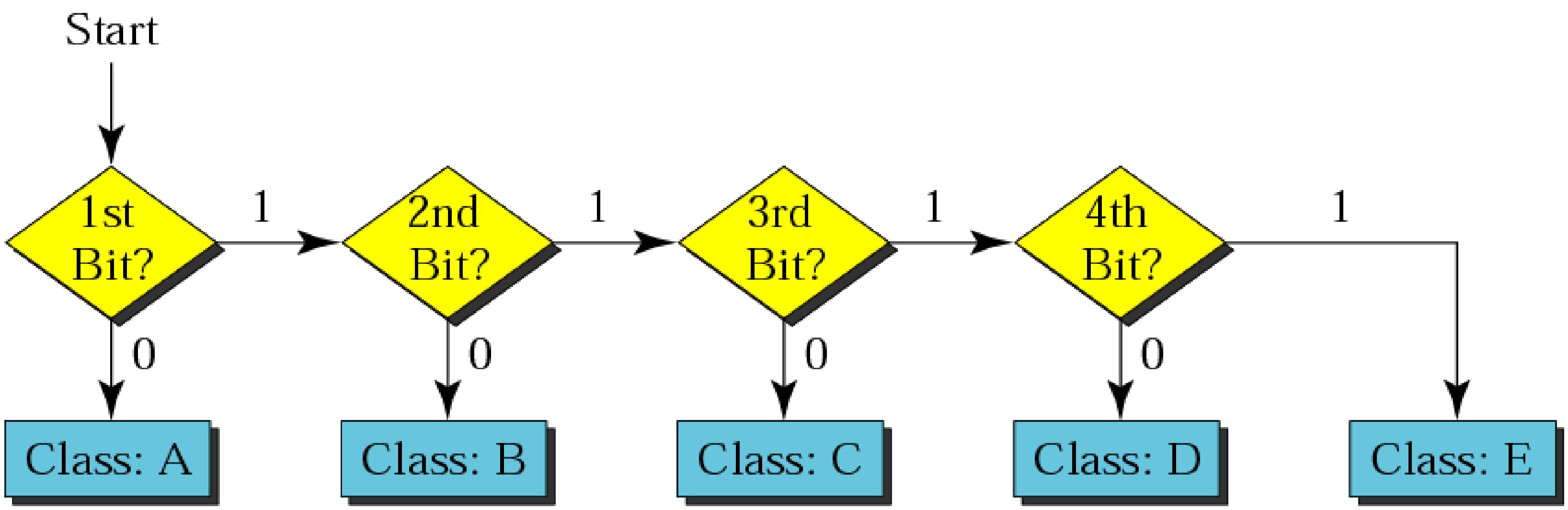
Table 4.1 Addresses per class

Class	*Number of Addresses*	*Percentage*
A	$2^{31} = 2,147,483,648$	50%
B	$2^{30} = 1,073,741,824$	25%
C	$2^{29} = 536,870,912$	12.5%
D	$2^{28} = 268,435,456$	6.25%
E	$2^{28} = 268,435,456$	6.25%

Figure 4.3 *Finding the class in binary notation*

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

Figure 4.4 *Finding the address class*



Example 5

How can we prove that we have 2,147,483,648 addresses in class A?

Solution

In class A, only 1 bit defines the class. The remaining 31 bits are available for the address. With 31 bits, we can have 2^{31} or 2,147,483,648 addresses.

Example 6

Find the class of each address:

a. 00000001 00001011 00001011 11101111
b. 11000001 10000011 00011011 11111111
c. 10100111 11011011 10001011 01101111
d. 11110011 10011011 11111011 00001111

Solution

See the procedure in Figure 4.4.
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 bit is 0; the second bit is 1. This is a class B address.
d. The first 4 bits are 1s. This is a class E address..

Figure 4.5 ***Finding the class in decimal notation***

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

Example 7

Find the class of each address:

a. ***227.12.14.87*** ***b.193.14.56.22*** ***c.14.23.120.8***
d. 252.5.15.111 ***e.134.11.78.56***

Solution

a. The first byte is 227 (between 224 and 239); the class is D.
b. The first byte is 193 (between 192 and 223); the class is C.
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.
e. The first byte is 134 (between 128 and 191); the class is B.

Example 8

In Example 5 we showed that class A has 2^{31} (2,147,483,648) addresses. How can we prove this same fact using dotted-decimal notation?

Solution

The addresses in class A range from 0.0.0.0 to 127.255.255.255. We need to show that the difference between these two numbers is 2,147,483,648. This is a good exercise because it shows us how to define the range of addresses between two addresses. We notice that we are dealing with base 256 numbers here. Each byte in the notation has a weight. The weights are as follows (see Appendix B):

See Next Slide

Example 8 *(continued)*

$$256^3, 256^2, 256^1, 256^0$$

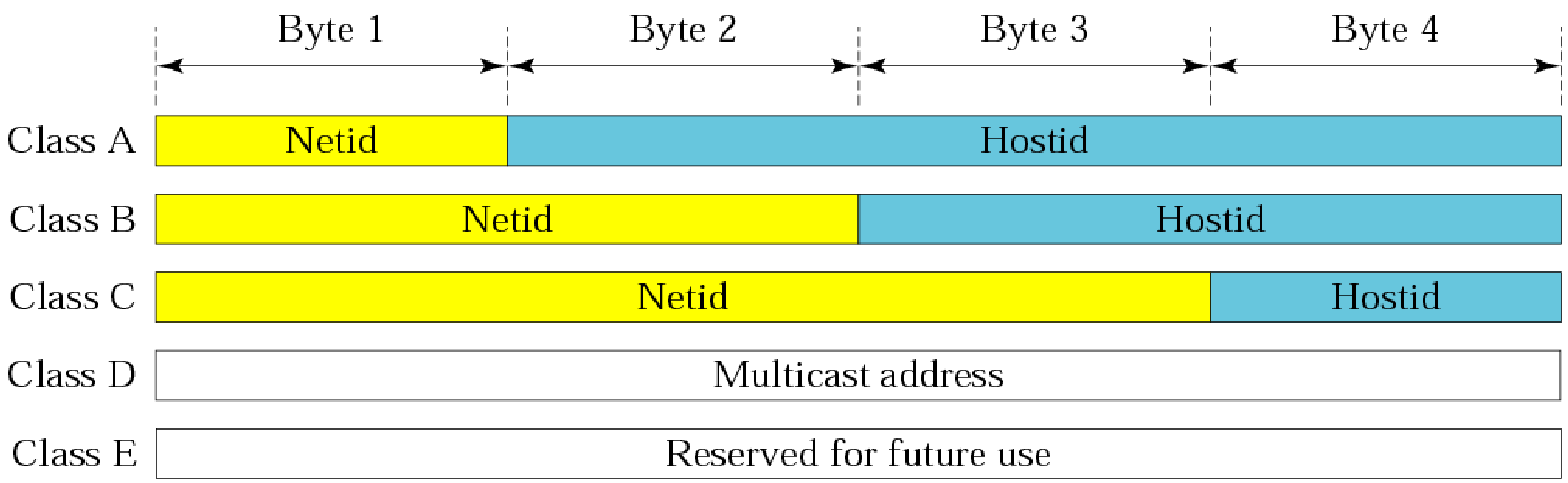
Now to find the integer value of each number, we multiply each byte by its weight:

Last address: $127 \times 256^3 + 255 \times 256^2 + 255 \times 256^1 + 255 \times 256^0 = 2{,}147{,}483{,}647$

First address: $= 0$

If we subtract the first from the last and add 1 to the result (remember we always add 1 to get the range), we get 2,147,483,648 or 2^{31}.

Figure 4.6 ***Netid and hostid***



Note:

Millions of class A addresses are wasted.

Figure 4.7 *Blocks in class A*

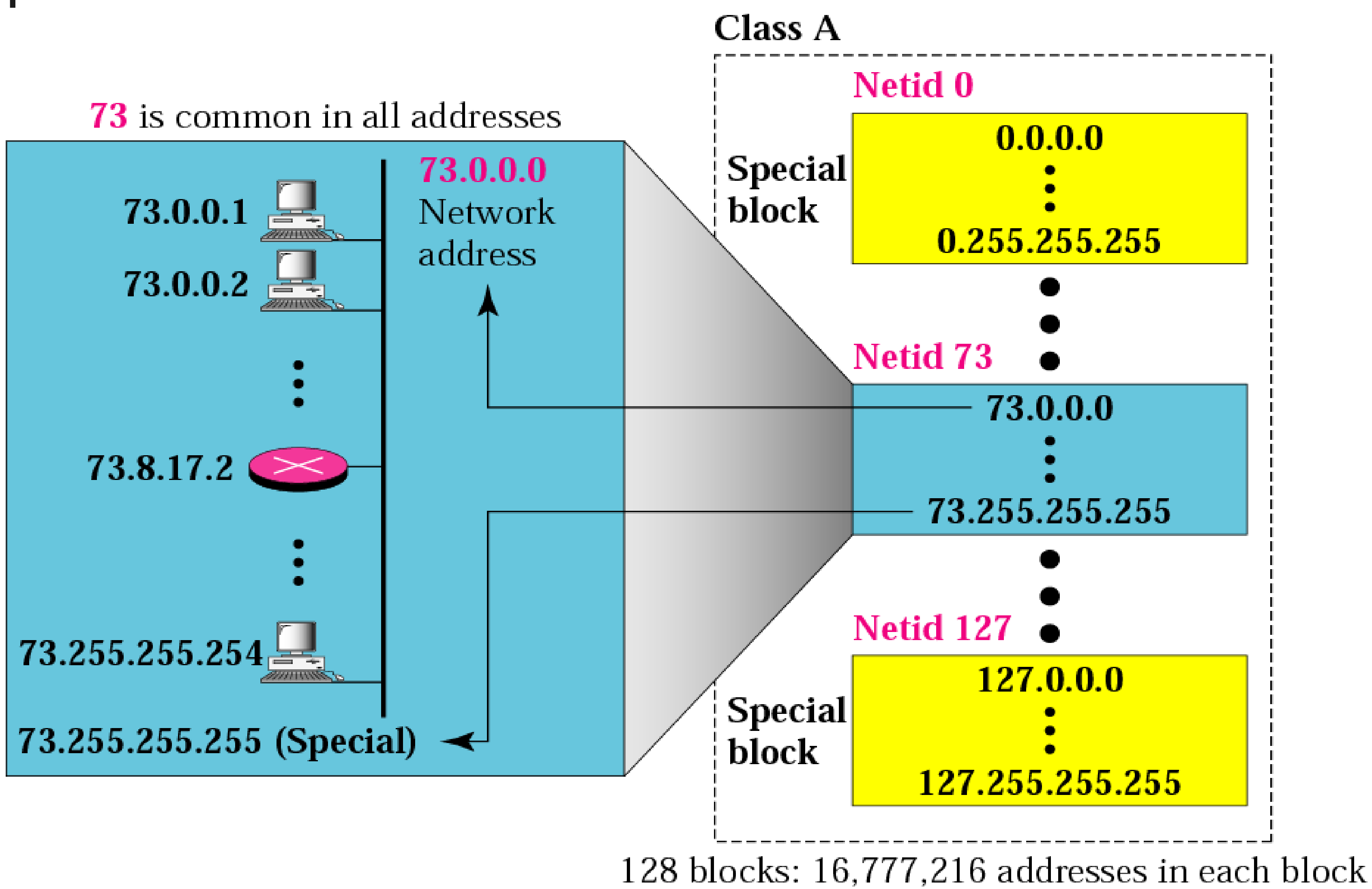
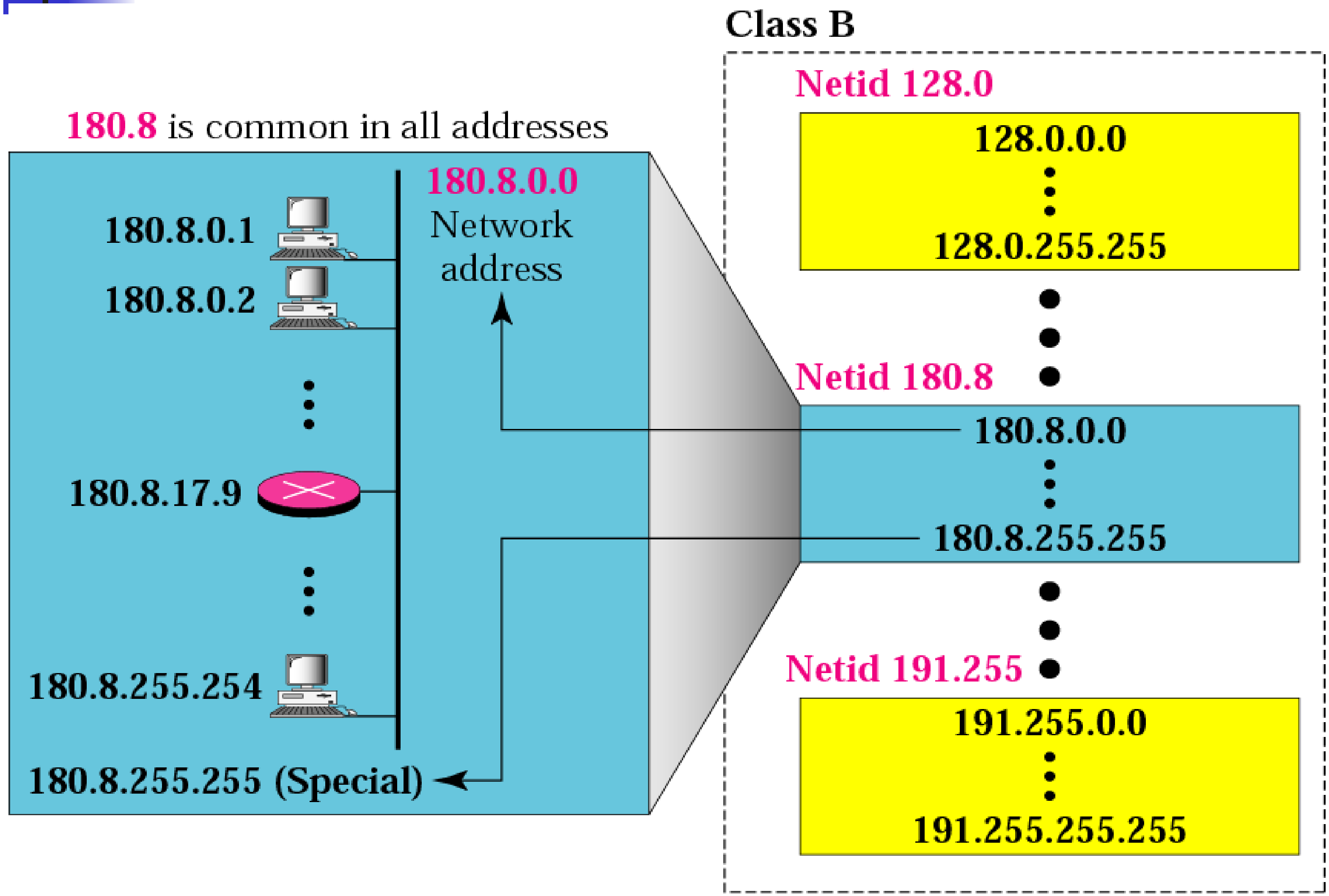


Figure 4.8 *Blocks in class B*

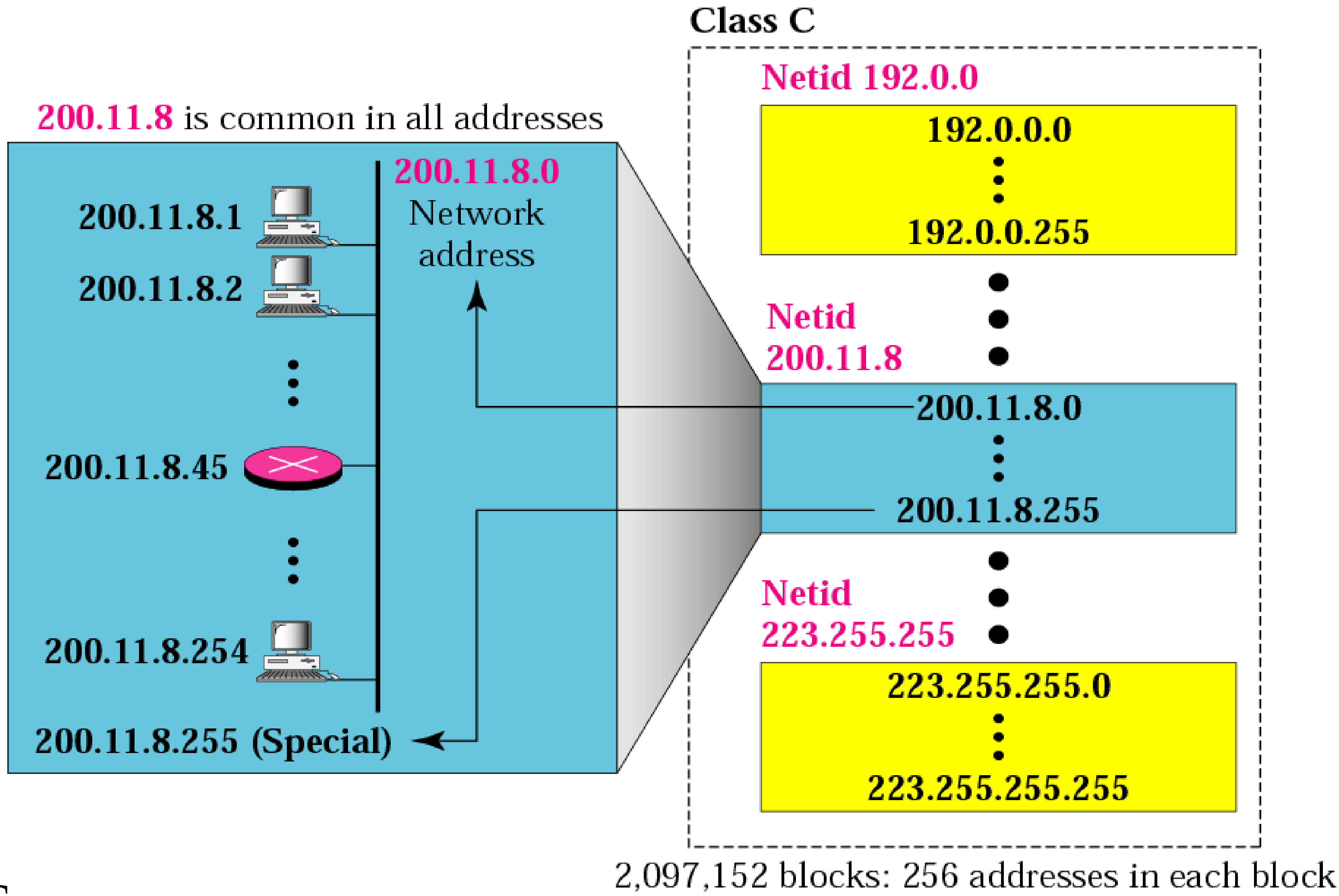


16,384 blocks: 65,536 addresses in each block

Note:

Many class B addresses are wasted.

Figure 4.9 *Blocks in class C*



Note:

The number of addresses in class C is smaller than the needs of most organizations.

Note:

Class D addresses are used for multicasting; there is only one block in this class.

Note:

Class E addresses are reserved for future purposes; most of the block is wasted.

Note:

In classful addressing, the network address (the first address in the block) is the one that is assigned to the organization. The range of addresses can automatically be inferred from the network address.

Example 9

Given the network address 17.0.0.0, find the class, the block, and the range of the addresses.

Solution

The class is A because the first byte is between 0 and 127. The block has a netid of 17. The addresses range from 17.0.0.0 to 17.255.255.255.

Example 10

Given the network address 132.21.0.0, find the class, the block, and the range of the addresses.

Solution

The class is B because the first byte is between 128 and 191. The block has a netid of 132.21. The addresses range from 132.21.0.0 to 132.21.255.255.

Example 11

Given the network address 220.34.76.0, find the class, the block, and the range of the addresses.

Solution

The class is C because the first byte is between 192 and 223. The block has a netid of 220.34.76. The addresses range from 220.34.76.0 to 220.34.76.255.

Figure 4.10 ***Masking concept***

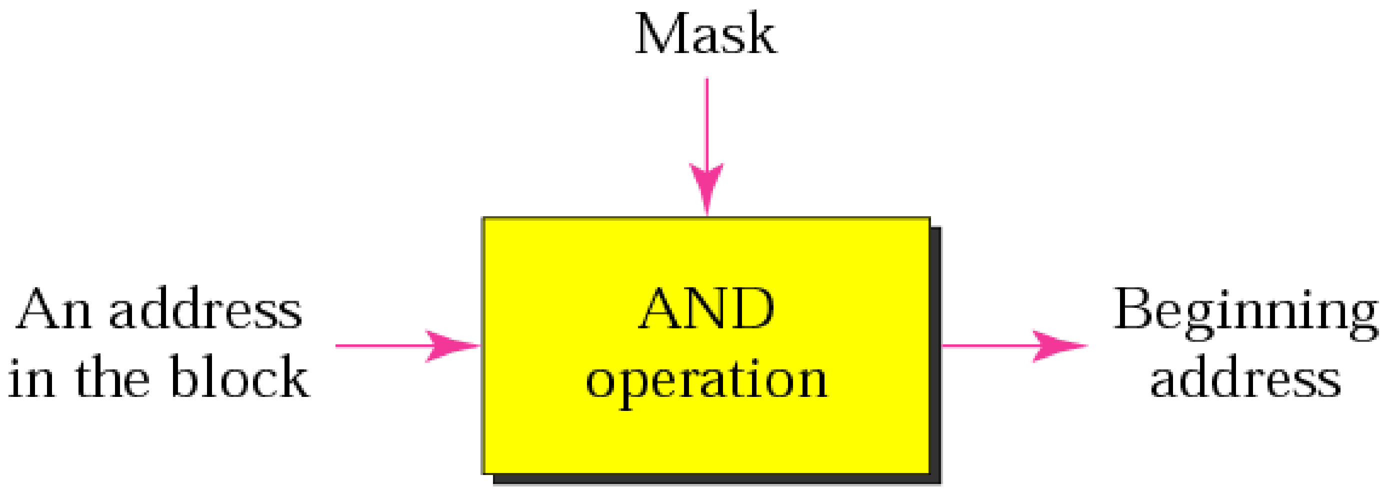


Figure 4.11 *AND operation*

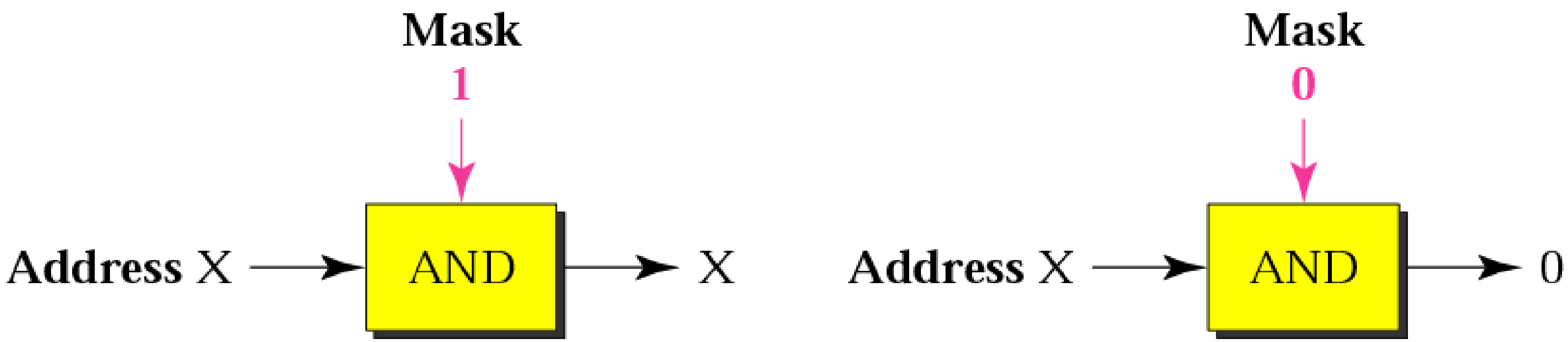


Table 4.2 Default masks

Class	*Mask in binary*	*Mask in dotted-decimal*
A	**11111111** 00000000 00000000 00000000	**255.**0.0.0
B	**11111111 11111111** 00000000 00000000	**255.255.**0.0
C	**11111111 111111111 11111111** 00000000	**255.255.255.**0

Note:

The network address is the beginning address of each block. It can be found by applying the default mask to any of the addresses in the block (including itself). It retains the netid of the block and sets the hostid to zero.

Example 12

Given the address 23.56.7.91, find the beginning address (network address).

Solution

The default mask is 255.0.0.0, which means that only the first byte is preserved and the other 3 bytes are set to 0s. The network address is 23.0.0.0.

Example 13

Given the address 132.6.17.85, find the beginning address (network address).

Solution

The default mask is 255.255.0.0, which means that the first 2 bytes are preserved and the other 2 bytes are set to 0s. The network address is 132.6.0.0.

Example 14

Given the address 201.180.56.5, find the beginning address (network address).

Solution

The default mask is 255.255.255.0, which means that the first 3 bytes are preserved and the last byte is set to 0. The network address is 201.180.56.0.

Note:

Note that we must not apply the default mask of one class to an address belonging to another class.