



## **Module 5**

# **IP Routing Protocol Enabling**

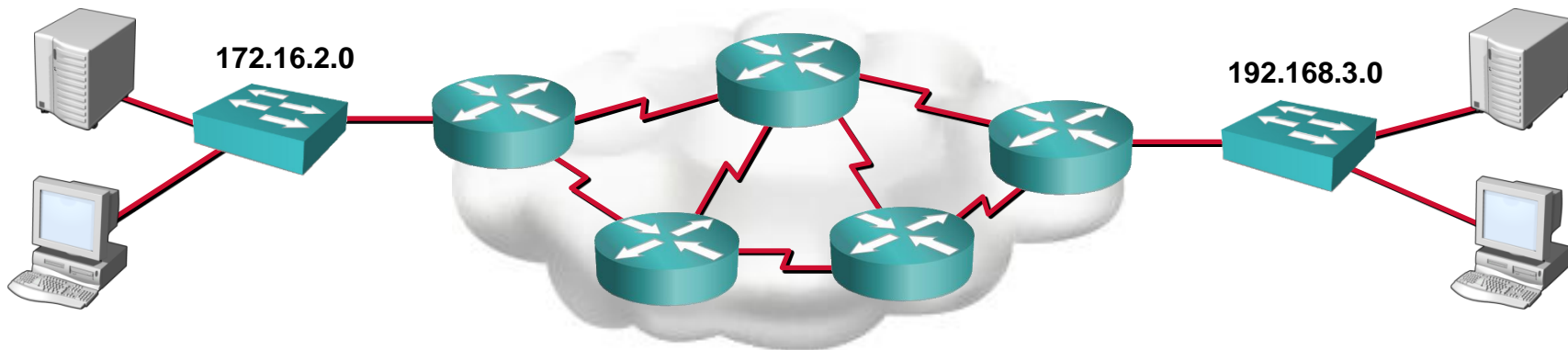
# Survey

- **Routing의 개념 소개**
- **Distance Vector Routing Protocol**
- **Link State & Hybrid Routing Protocol**
- **RIP 구성하기**
- **VLSM 개요**

# Routing의 개념 소개

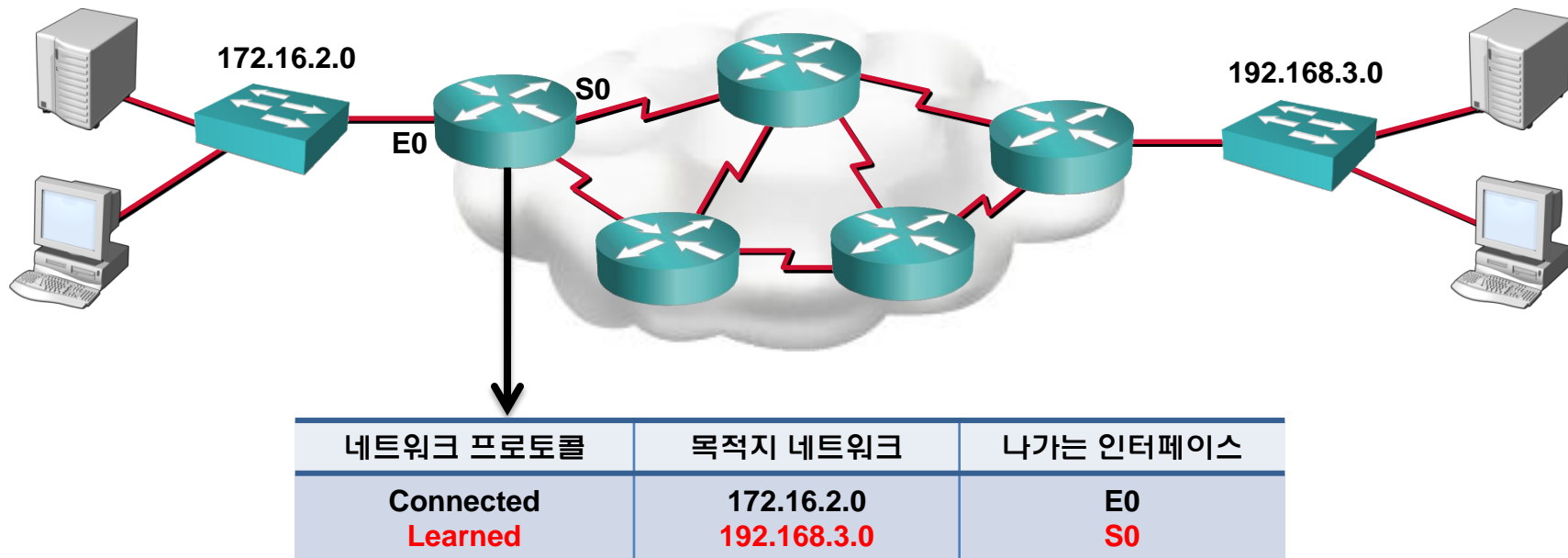
- Routing 개요
- Static & Dynamic Routing
- Static Route 설정하기
- Default Route 설정하기
- Dynamic Routing 개요
- IP Classless 소개
- Inter-VLSN Routing

# Routing Survey



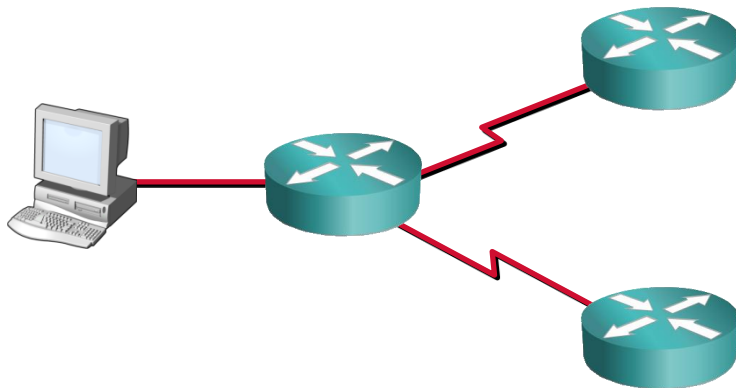
- 데이터를 최적의 경로를 선택하여 목적지까지 이송하는 모든 절차
- IP Protocol이 올라가 있는 Router, Computer or Host, L-3 Ethernet Switch들이 이러한 작업이 가능
- 라우터가 데이터를 Routing하기 위하여 알아야 하는 것
  - 소스와 목적지 주소
  - 입/출력 인터페이스 형태
  - 가능성 있는 모든 경로(route)들에 대한 정보 수집
  - 가능성 있는 모든 경로 중에서 최적의 경로 선택
  - 지속적인 네트워크 상태를 확인하고 유지하는 것

# Routing Survey



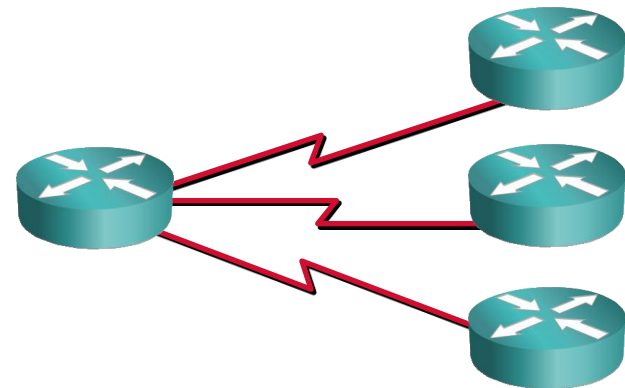
- 라우터는 직접 연결된 곳 이외의 다른 목적지에 대해서는 반드시 Connected, Static, Dynamic Routing Protocol, Redistribution과 같은 다양한 방법으로 학습 할 수 있다

# Static & Dynamic Routing



## Static Routing

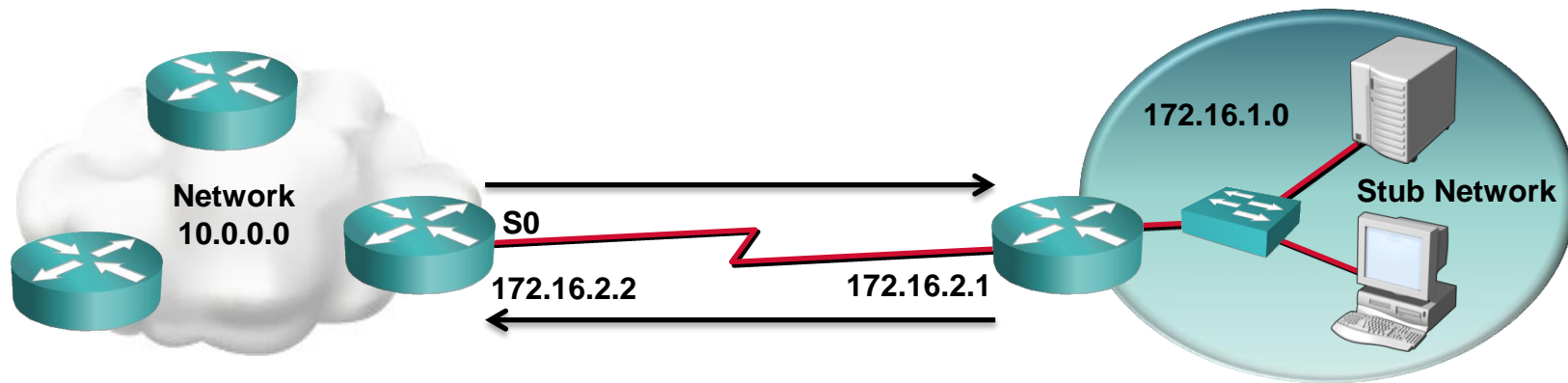
- 관리자가 직접 수동으로 Router에게 필요한 Route 정보들을 입력한다
- Network의 변화에 대해 Router가 자동으로 반응하지 못하며 관리자가 직접 Network의 변화를 Router에 설정해야 한다



## Dynamic Routing

- Routing Protocol을 이용하여 자동으로 Route 정보를 수집한다
- Network 변화에 대해 자동으로 반응한다

# Static Route 설정하기

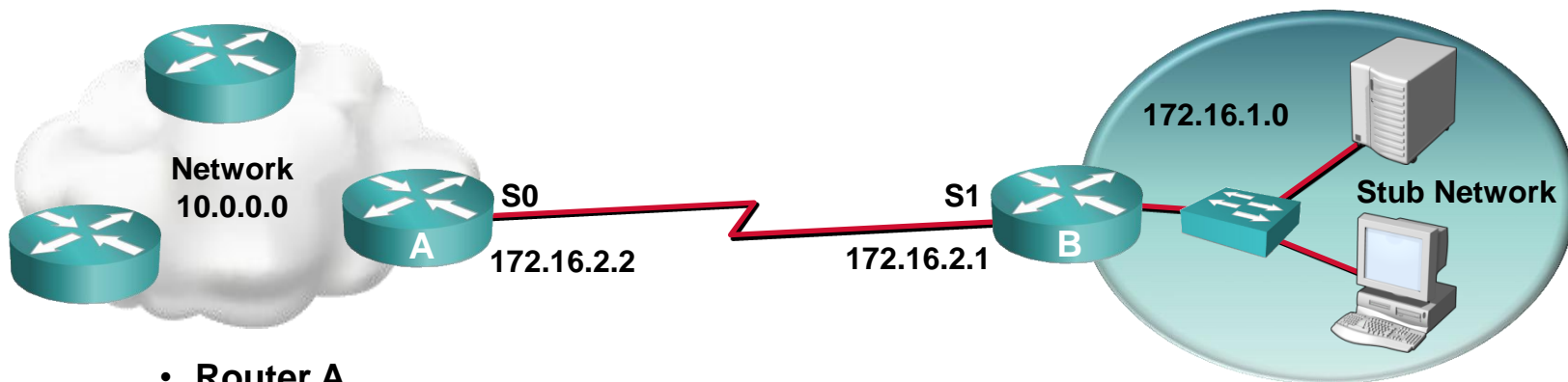


```
Router(config)#ip route network mask {address|interface} [distance] [permanent]
```

<b>Network</b>	도착지 Network
<b>Mask</b>	도착지 Network의 Subnet mask
<b>Address</b>	도착지 network로 도달하기 위한 Next-hop address
<b>Interface</b>	도착지 network로 도달하기 위한 Next-hop Router와 연결된 local interface
<b>Distance</b>	정의된 Route의 Administrator Distance 값
<b>Permanent</b>	정의된 Static Route가 Routing table에서 제거되지 않도록 한다.

# Static Route 설정하기

- Static Route 설정 예제 #1



- Router A

```
Router(config)#ip route 172.16.1.0 255.255.255.0 172.16.2.1
```

- Router B

```
Router(config)#ip route 10.0.0.0 255.0.0.0 172.16.2.2
```

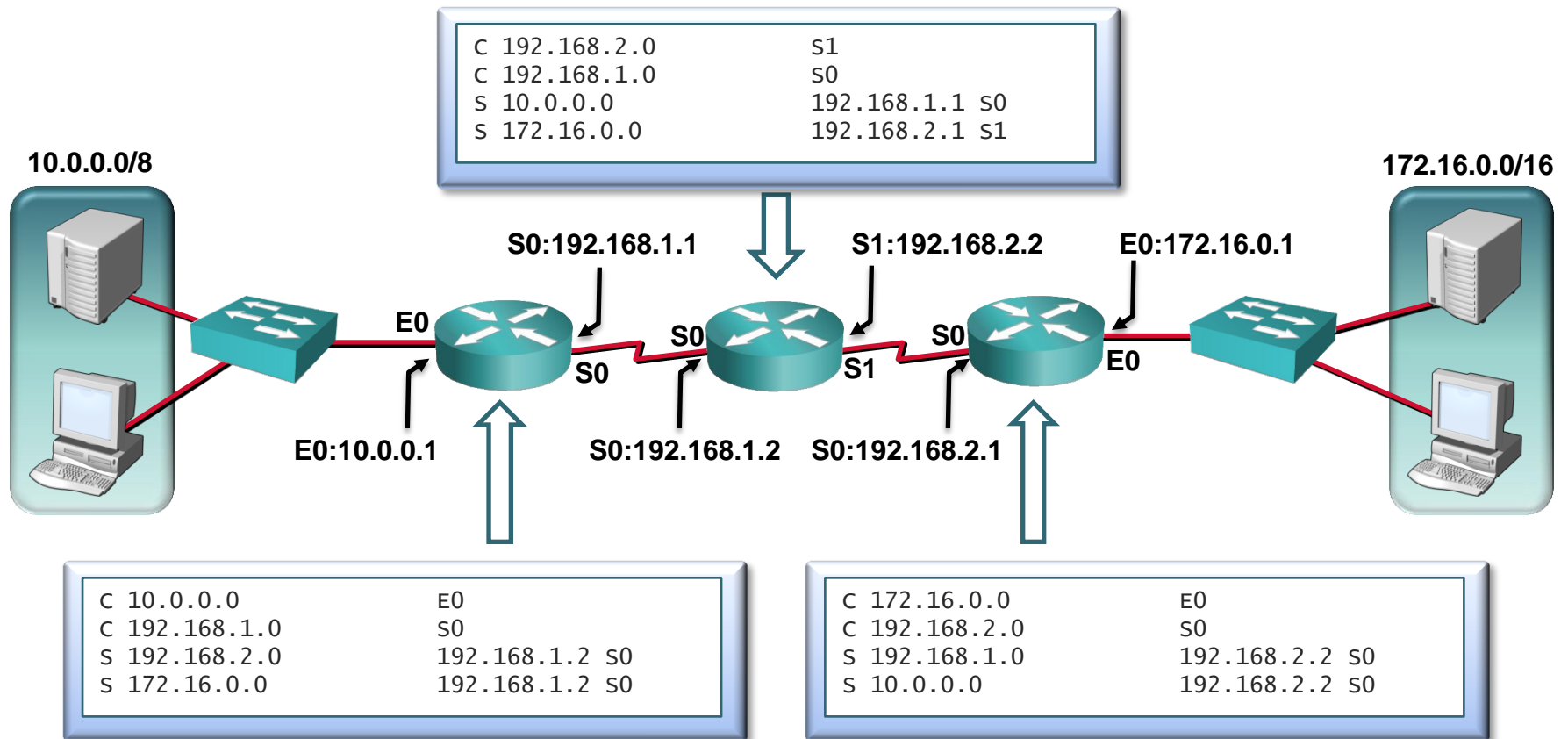
*or*

```
Router(config)#ip route 10.0.0.0 255.0.0.0 serial 1
```



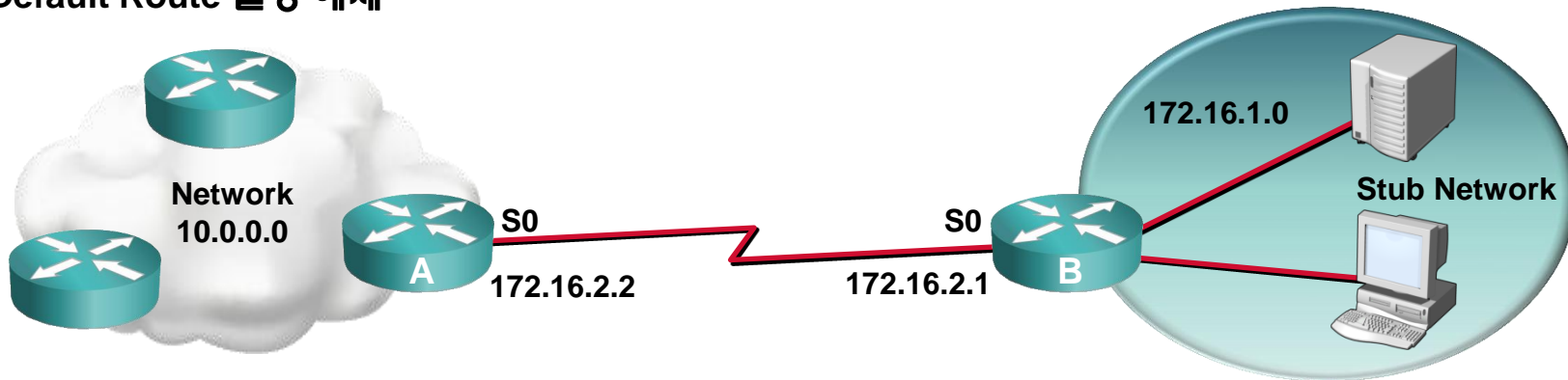
# Static Route 설정하기

## • Static Route 설정 예제 #2



# Default Route 설정하기

- Default Route 설정 예제



- RouterA

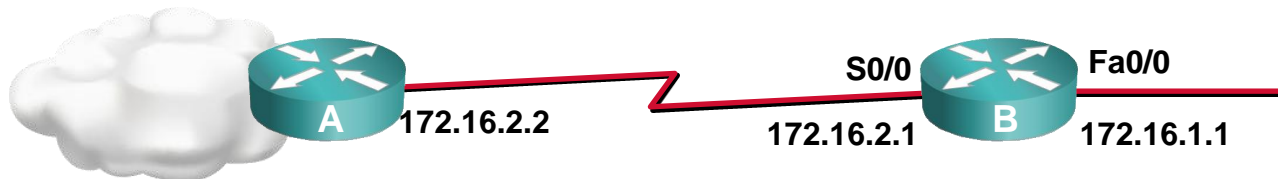
```
Router(config)#ip route 172.16.1.0 255.255.255.0 172.16.2.1
```

- RouterB

```
Router(config)#ip route 0.0.0.0 0.0.0.0 172.16.2.2
```

# Default Route 설정하기

- Default Route 설정 확인

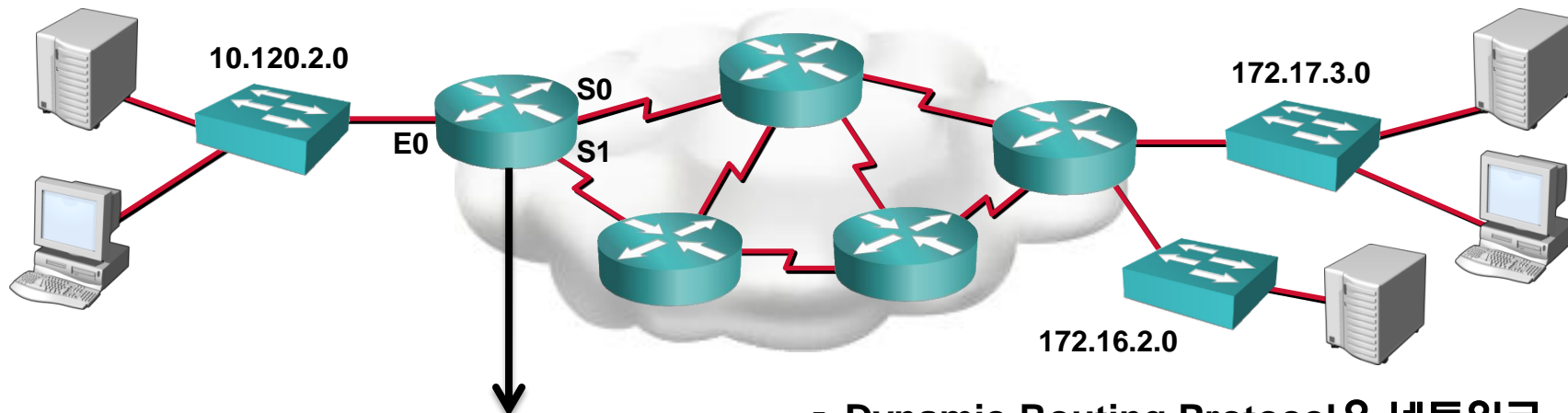


```
Router-B#config t
Router-B(config)#ip route 0.0.0.0 0.0.0.0 172.16.2.2
Router-B(config)#exit
Router-B#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

    172.16.0.0/24 is subnetted, 3 subnets
C      172.16.1.0 is directly connected, FastEthernet0/0
C      172.16.2.0 is directly connected, Serial0/0
S*     0.0.0.0/0 [ 1/0 ] via 172.16.2.2
Router-B#
```

# Dynamic Routing 개요



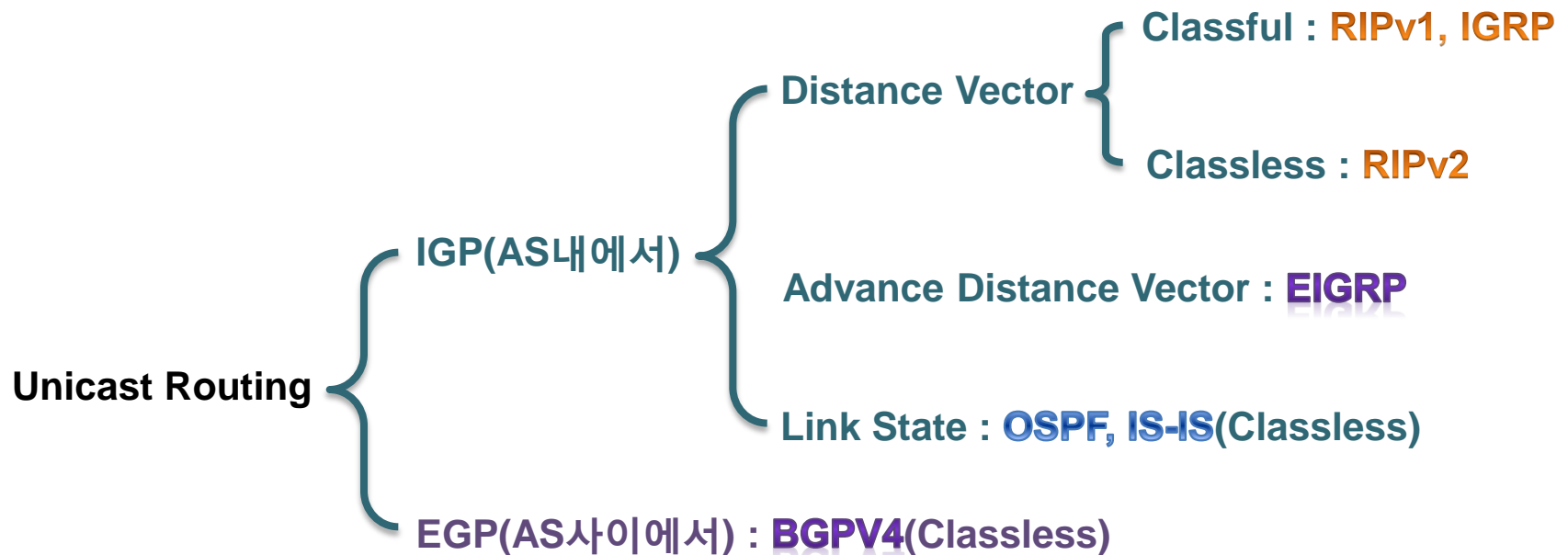
라우팅 프로토콜	목적지 네트워크	나가는 인터페이스
Connected <b>RIP</b>	10.120.2.0 <b>172.16.2.0</b>	E0 <b>S0</b>

- Dynamic Routing Protocol은 네트워크 정보를 교환하여 최적의 경로를 결정하고, 라우팅 테이블을 지속적으로 유지한다
- 하나의 경로가 결정되면 라우터는 Routed Protocol들을 라우트 할 수 있다

- Routed Protocol** : IP, IPX, Apple Talk
- Routing Protocol** : RIP, IGRP, EIGRP, OSPF, IS-IS, BGP, DVMRP, MOSPF, PIM Dense & Sparse

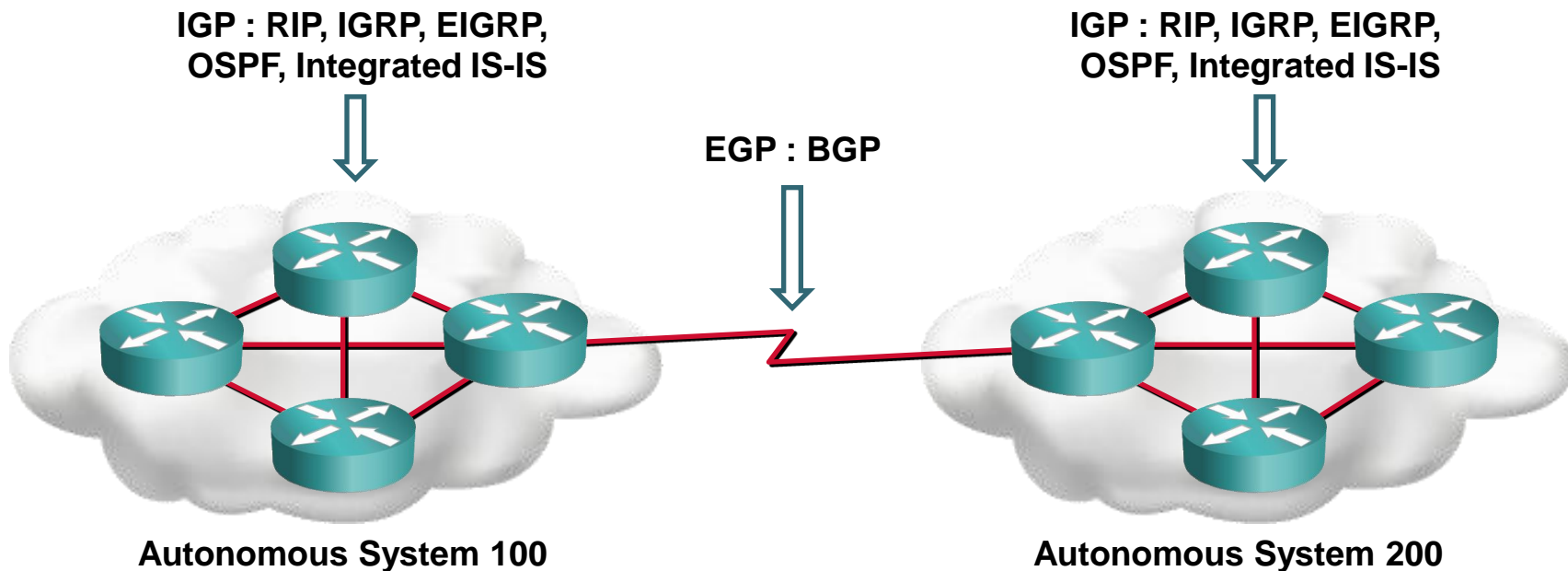
# Dynamic Routing 개요

- IP Routing Protocol의 종류



Multicast Routing : **DVMRP, MOSPF, PIM Dense & Sparse**

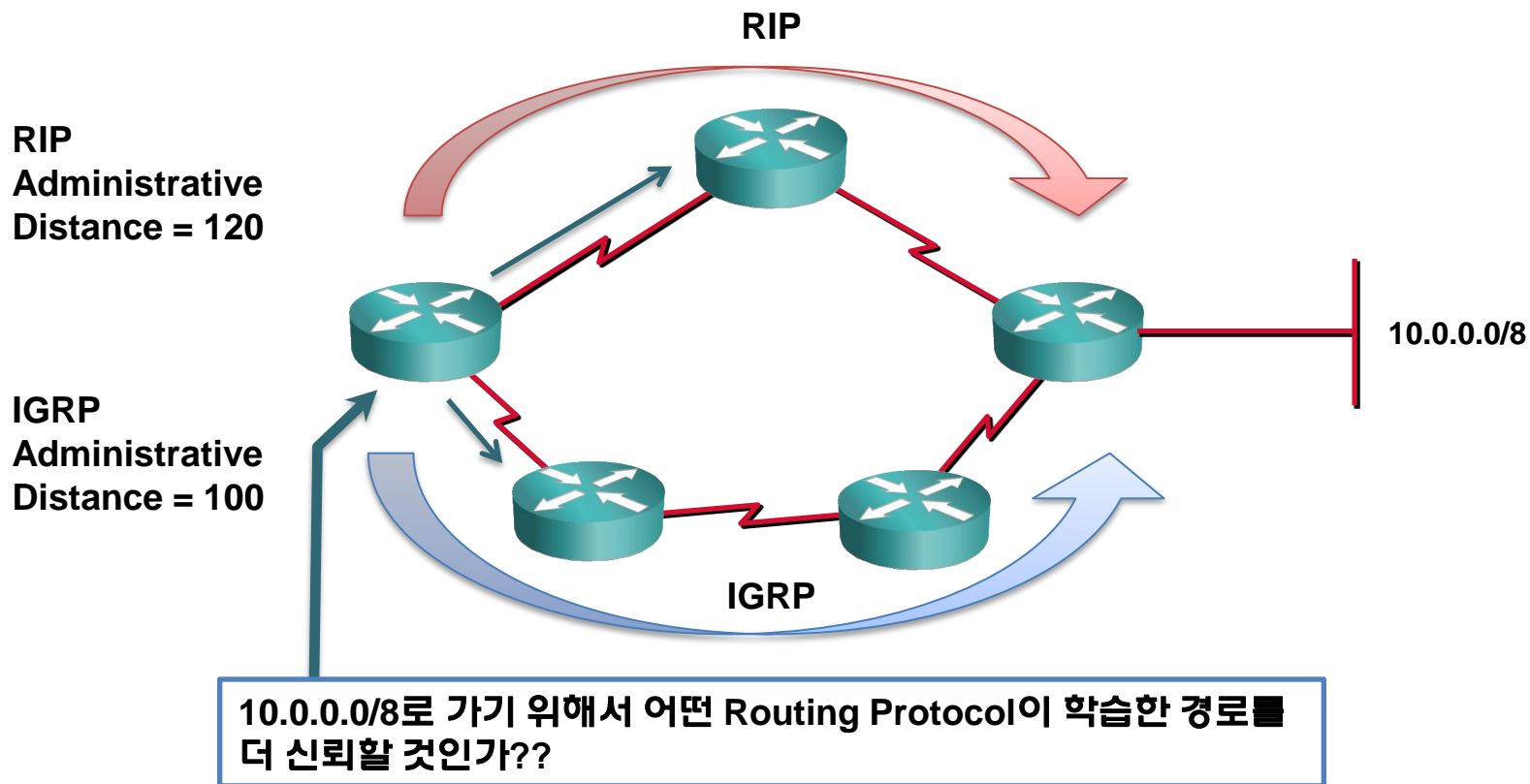
# Dynamic Routing 개요



- Autonomous System은 일반적인 관리 영역하에 있는 네트워크들의 집합
- IGP들은 Autonomous System안에서 운영
- EGP들은 다른 Autonomous System간의 운영

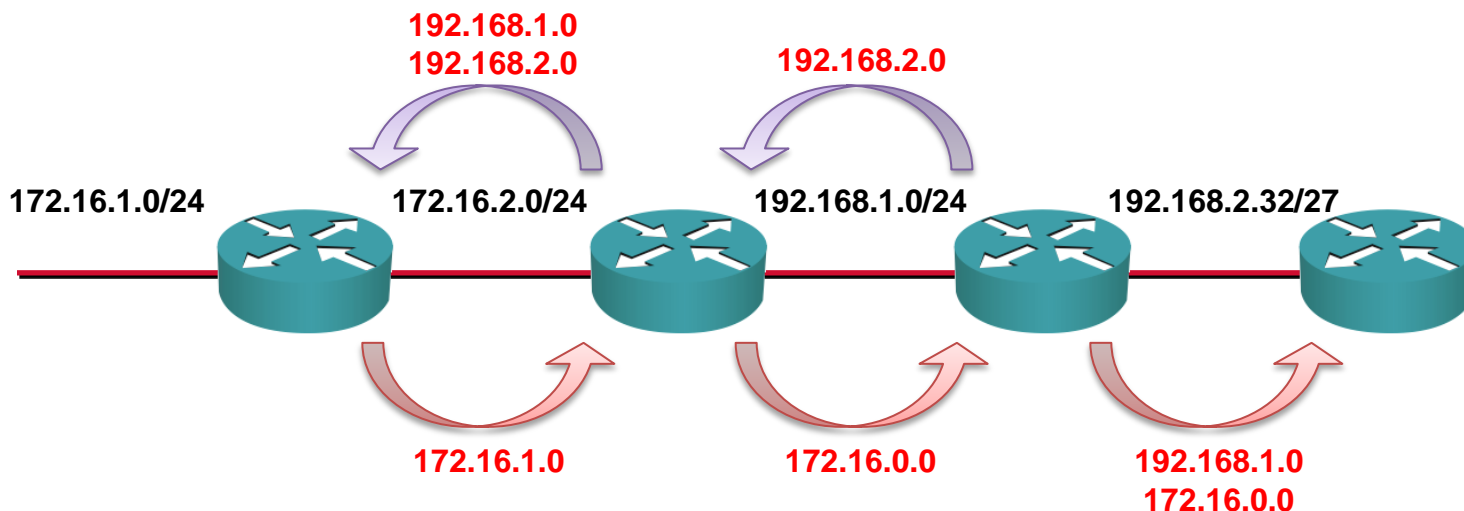
# Dynamic Routing 개요

- Administrative Distance



# Dynamic Routing 개요

- Classful Routing

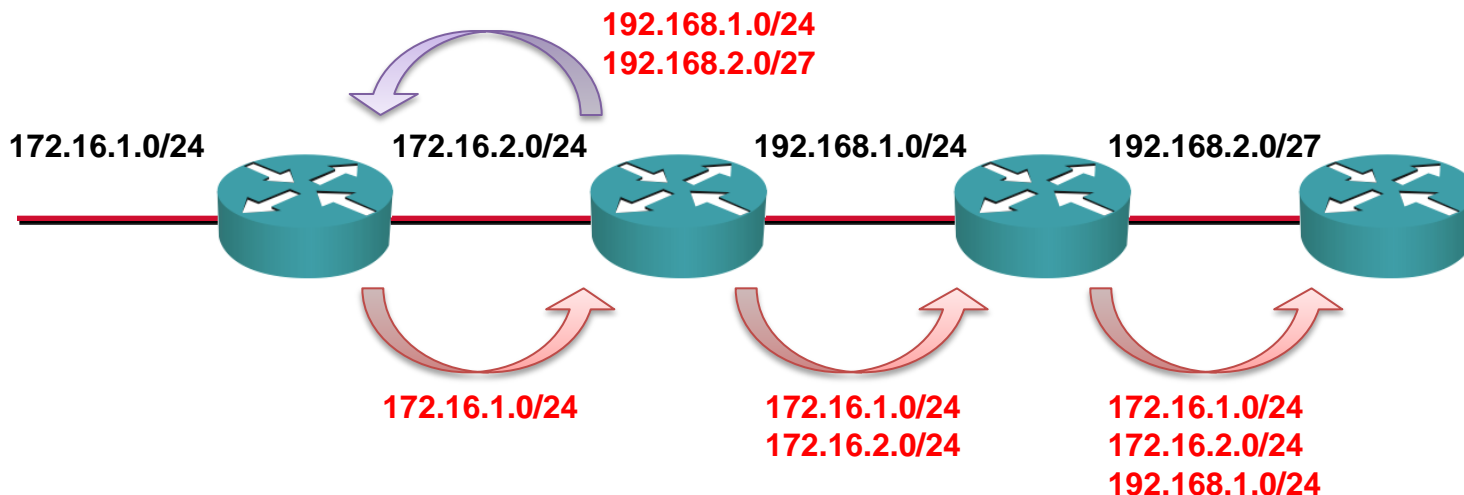


- Routing 정보 전달 시에 Subnet mask 정보를 전달하지 않는다
- 같은 network에 연결된 Router들은 같은 Subnet mask로 설정되어 있다고 가정한다
- Network이 다른 router와 Routing 정보 교환 시에는 자동으로 Classful 경계를 기반으로 Summary된 정보를 전달한다
- RIP Version 1과 IGRP가 여기 속한다



# Dynamic Routing 개요

- Classless Routing



- Routing 정보 전달 시에 Subnet mask 정보를 함께 전달한다
- Network에 연결된 Router들은 다양한 Subnet mask로 설정되어 있을 수 있다 (VLSM 지원)
- Network이 다른 router와 Routing정보 교환 시는 수동적으로 Summary된 정보를 전달 할 수도 있다
- RIP Version 1과 IGRP를 제외한 모든 Routing Protocol이 이를 지원한다

# Dynamic Routing 개요

- Routing Protocol들의 비교 #1

특징	RIPv1	RIPv2	IGRP	EIGRP**	OSPF
Distance vector	O	O	O	O	
Link-State					O
Classful (auto route summ.)	O	O	O	O	
Classless(VLSM support)		O		O	O
Proprietary			O	O	
Scalability	Small	Small	Med.	Large	Large
Convergence time	Slow	Slow	Slow	Fast	Fast

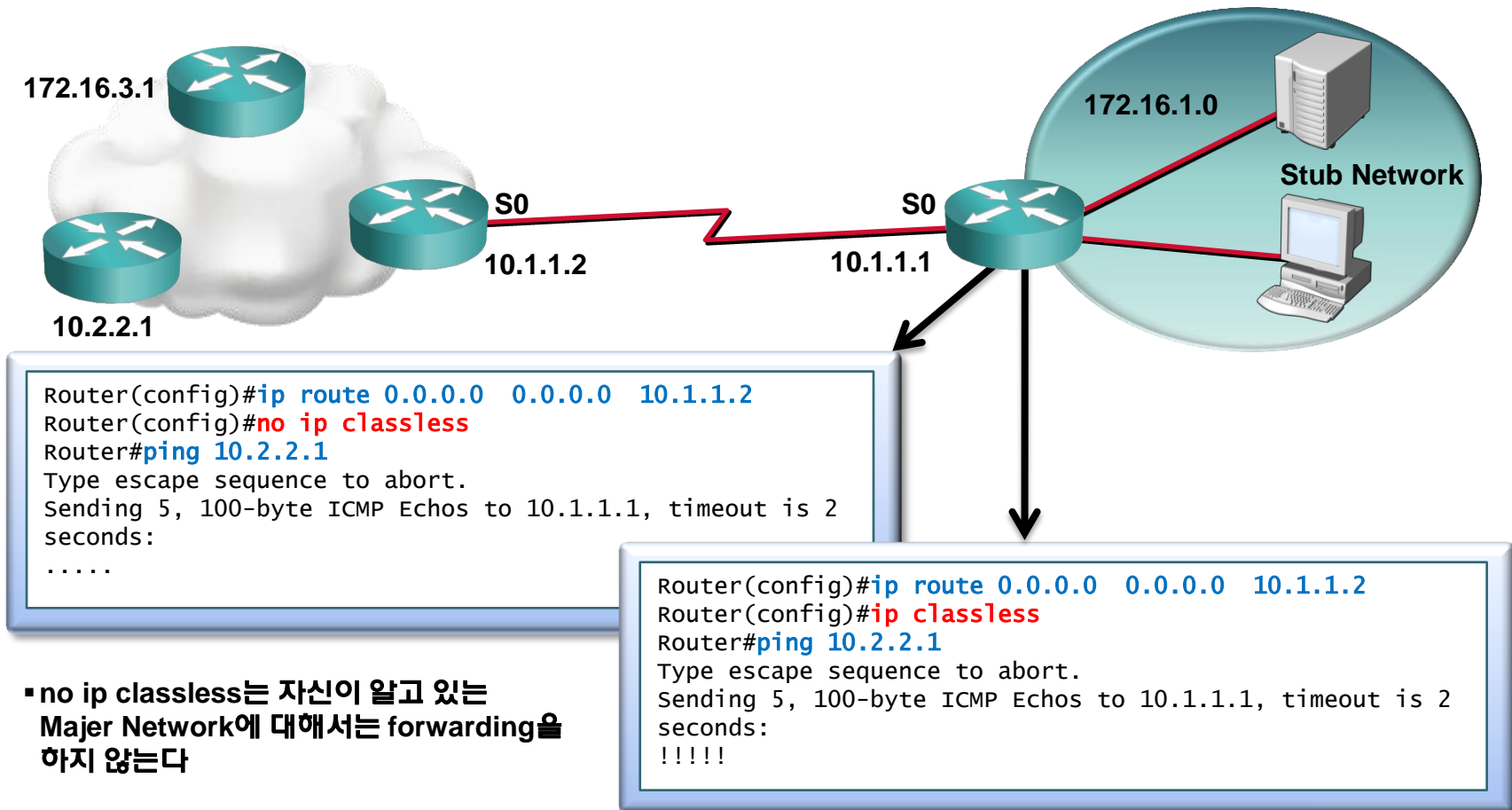
# Dynamic Routing 개요

- Routing Protocol들의 비교 #2

특    정	RIPv1	RIPv2	IGRP	EIGRP**
Count to infinity	O	O	O	
Split horizon	O	O	O	O
Hold-down timer	O	O	O	
Triggered updates with route poisoning	O	O	O	O
Load balancing-Equal paths	O	O	O	O
Load balancing-Unequal paths			O	O
VLSM support		O		O
Routing algorithm	B-F	B-F	B-F	Dual
Metric	Hops	Hops	Comp	Comp
Hop count limit	16	16	100	100
Scalability	Med	Med	Large	Large

\*\* EIGRP는 Advanced Distance Vector Protocol (Hybrid)

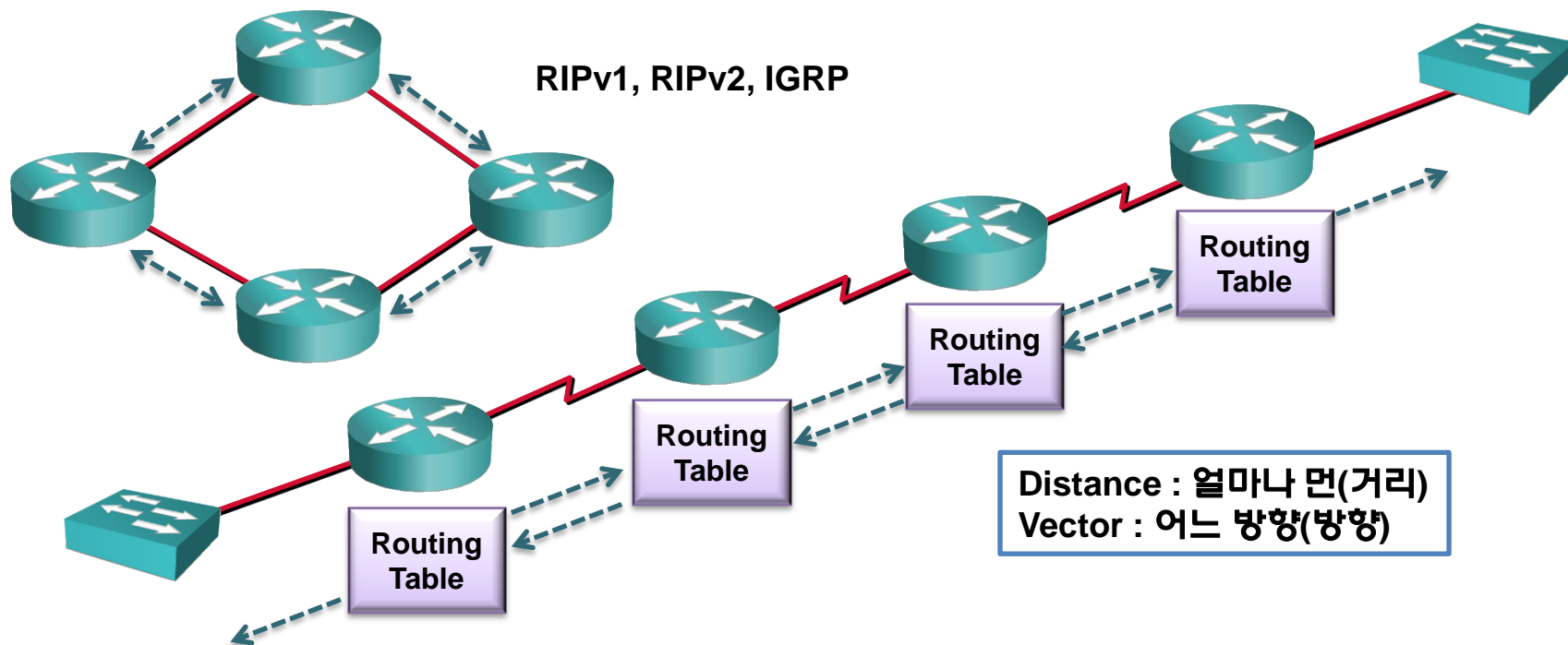
# IP Classless 개요



# Distance Vector Routing

- Distance Vector Routing 개요
- Distance Vector의 경로 선택
- Routing 정보 관리
- Distance Vector의 Routing Loop
- Routing Loop 문제 해결
- Distance Vector의 Operation

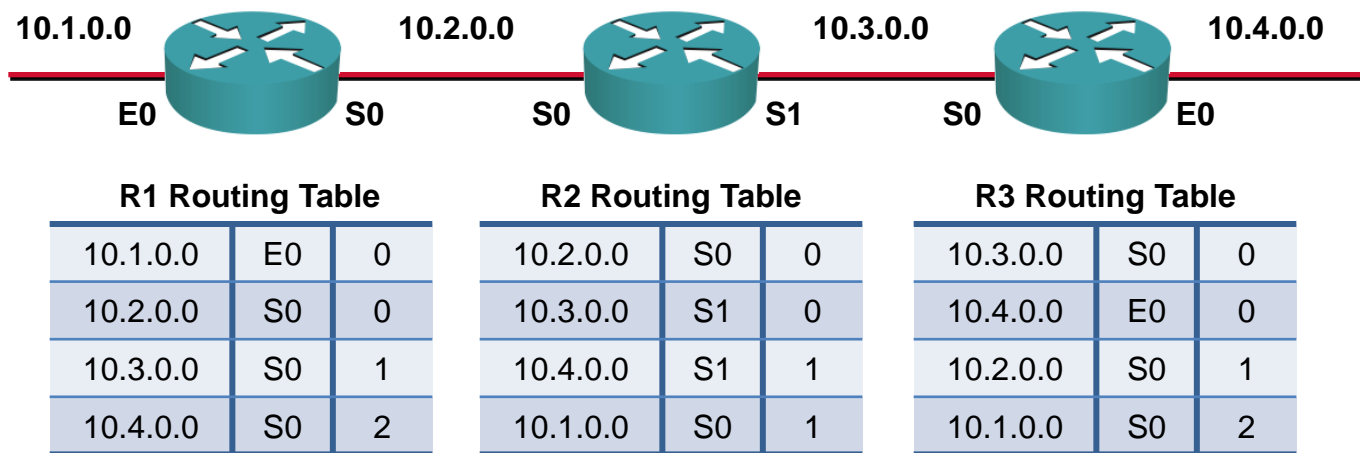
# Distance Vector Routing의 개요



- 최적의 라우팅 리스트를 만든 후, 주기적으로 라우팅 테이블을 인접관계에 있는 라우터에게 전달

# Distance Vector Routing의 개요

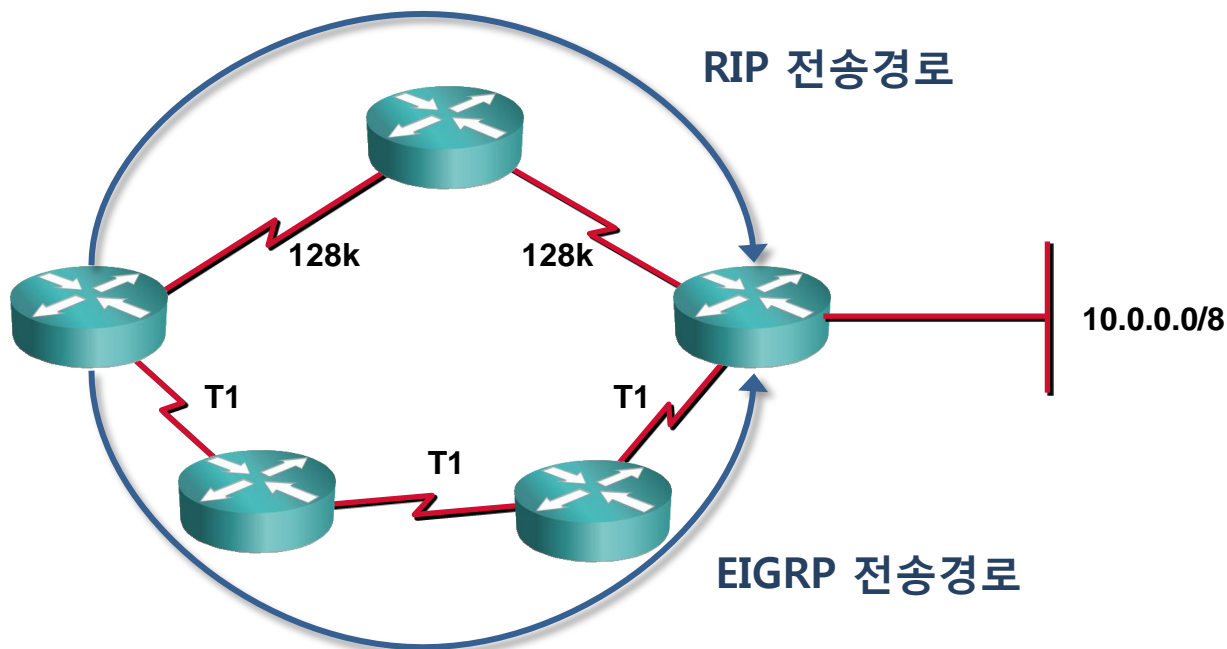
- Distance Vector의 경로 정보 수집



- Router들은 Network상에 각 Destination에 대해 최적의 경로를 선택 후 이를 관리 및 유지 한다

# Distance Vector의 경로 선택

- Distance Vector의 Metric

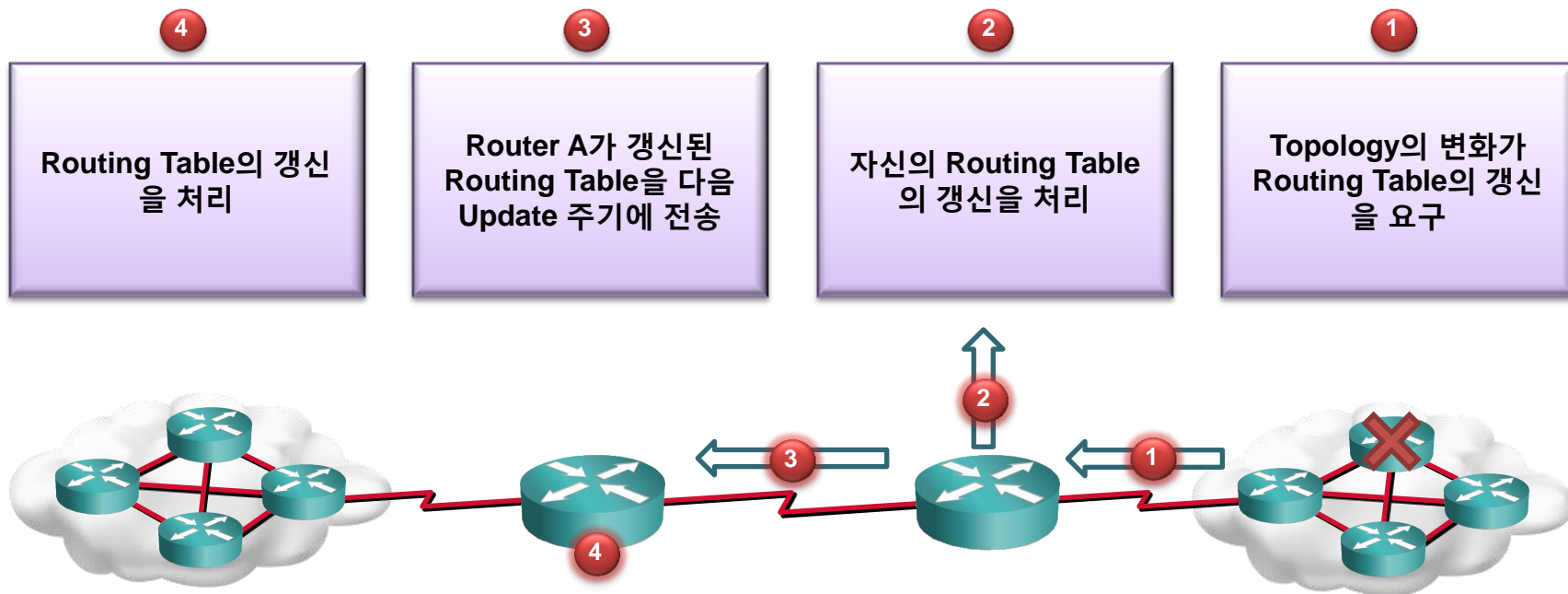


- RIP Metric
  - Hop Count

- IGRP Metric
  - Bandwidth, Delay
  - Load, Reliability, MTU



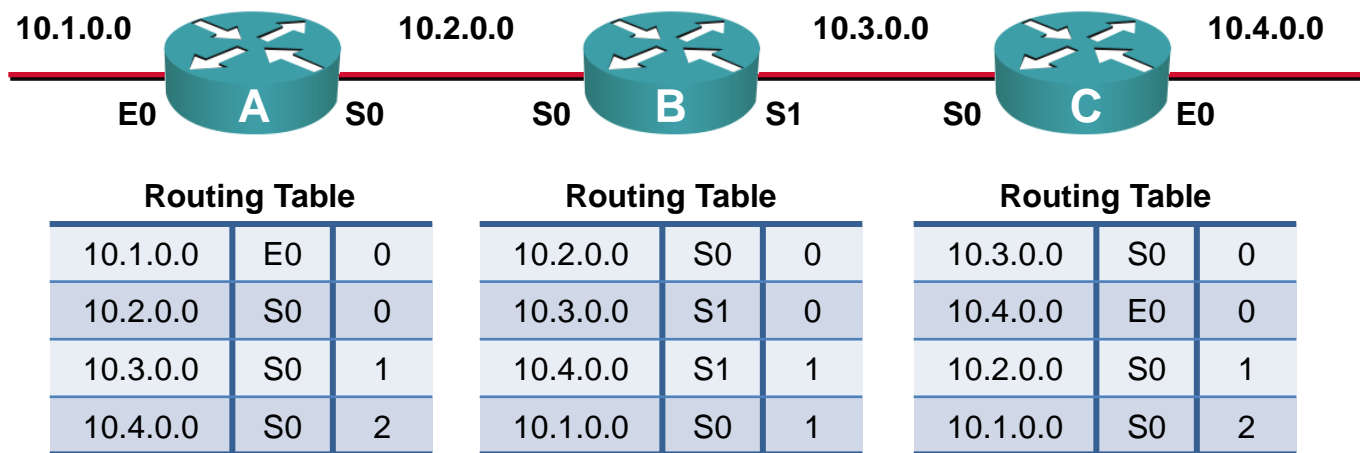
# Routing 정보 관리



- Update 절차는 라우터에서 다른 라우터로 Step-by-Step으로 진행

# Distance Vector의 Routing Loop

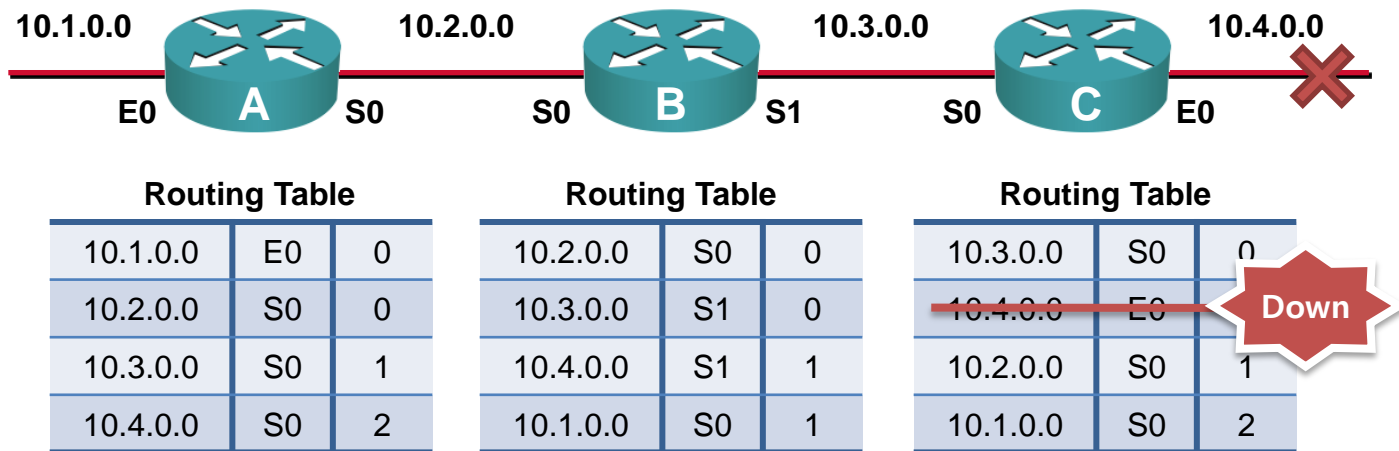
- Routing Loop 예제



일반적인 상황에서의 Routing Table

# Distance Vector의 Routing Loop

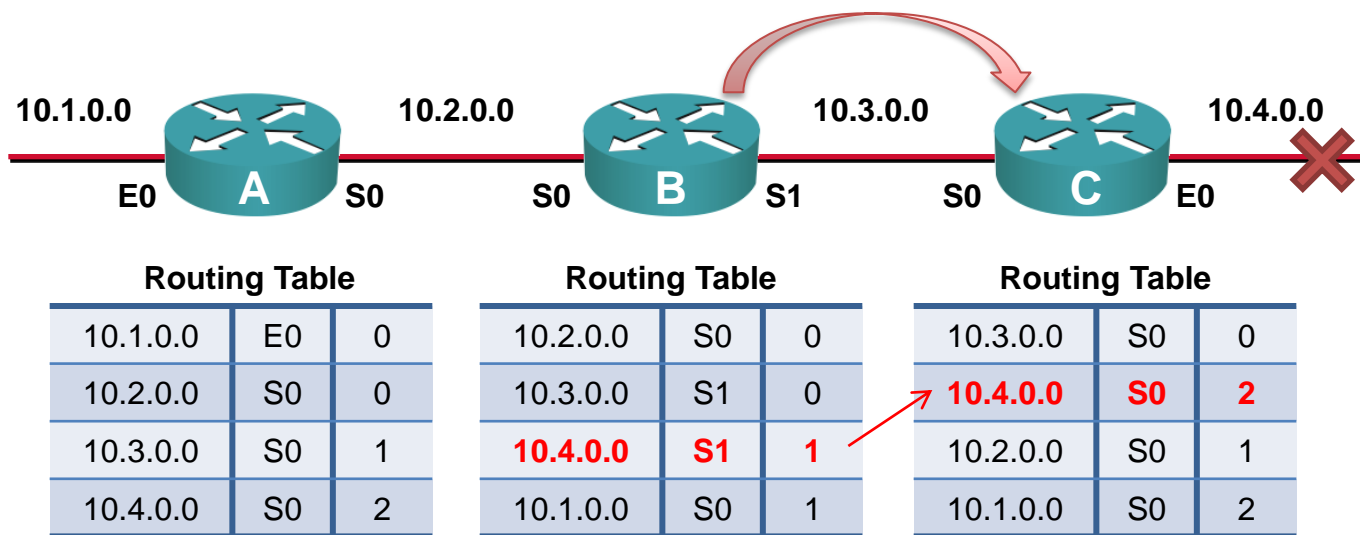
- Routing Loop 예제 (계속)



- Router-C에서 Down된 Ethernet 구간은 Routing Table에서 경로 제거
- 이러한 Topology의 변화가 다른 Router에게 얼마나 빨리 전달되는가 ?

# Distance Vector의 Routing Loop

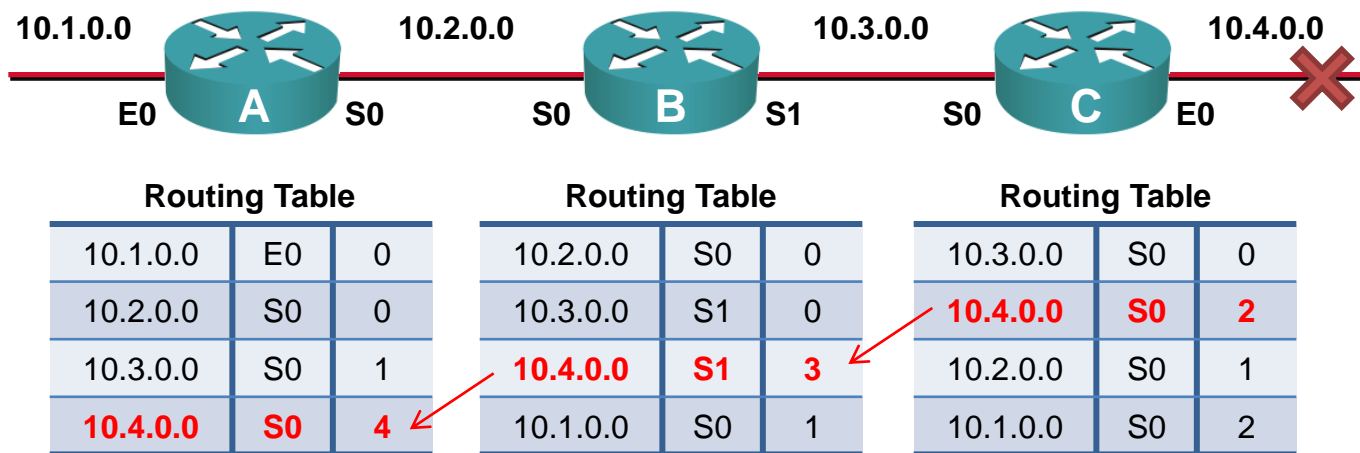
- Routing Loop 예제 (계속)



- Distance Vector는 Network Topology를 이해하지 못한다
- Router-C는 Router-B에서 받은 Routing 정보를 기반으로 10.4.0.0에 도달할 수 있는 또 다른 경로가 있다고 판단한다

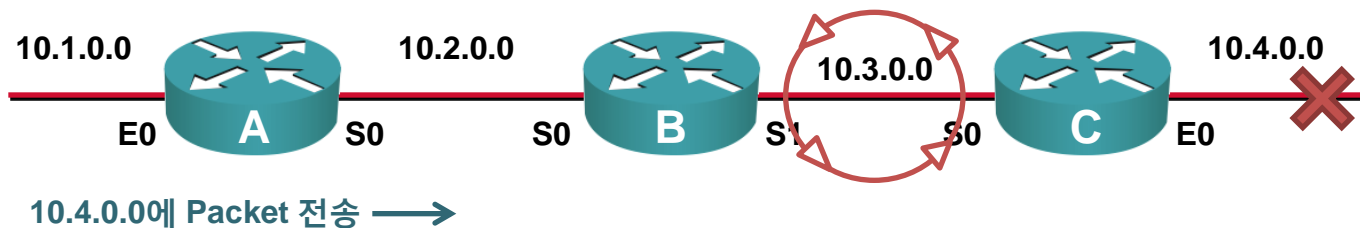
# Distance Vector의 Routing Loop

- Routing Loop 예제 (계속)



- 결국 Hop Count가 Maximum=16까지 증가되어서야 10.4.0.0의 Network이 도달할 수 없음을 모든 Router가 인지하게 된다
- Hop Count가 Maximum까지 증가되면 Routing Table에서 해당 경로를 제거한다

# Routing Loop 문제 해결



Routing Table		
10.1.0.0	E0	0
10.2.0.0	S0	0
10.3.0.0	S0	1
<b>10.4.0.0</b>	<b>S0</b>	<b>4</b>

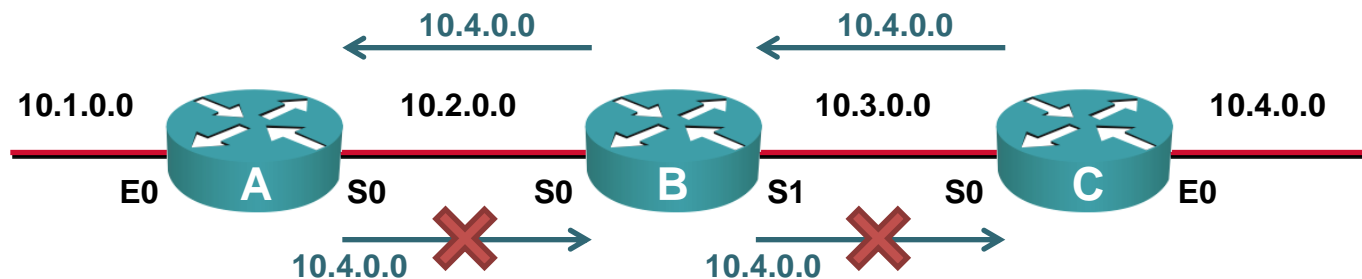
Routing Table		
10.2.0.0	S0	0
10.3.0.0	S1	0
<b>10.4.0.0</b>	<b>S1</b>	<b>3</b>
10.1.0.0	S0	1

Routing Table		
10.3.0.0	S0	0
<b>10.4.0.0</b>	<b>S0</b>	<b>2</b>
10.2.0.0	S0	1
10.1.0.0	S0	2

- Router-B와 Router-C 사이에서 Routing Loop 발생

# Routing Loop 문제 해결

- Split Horizon



Routing Table		
10.1.0.0	E0	0
10.2.0.0	S0	0
10.3.0.0	S0	1
10.4.0.0	S0	2

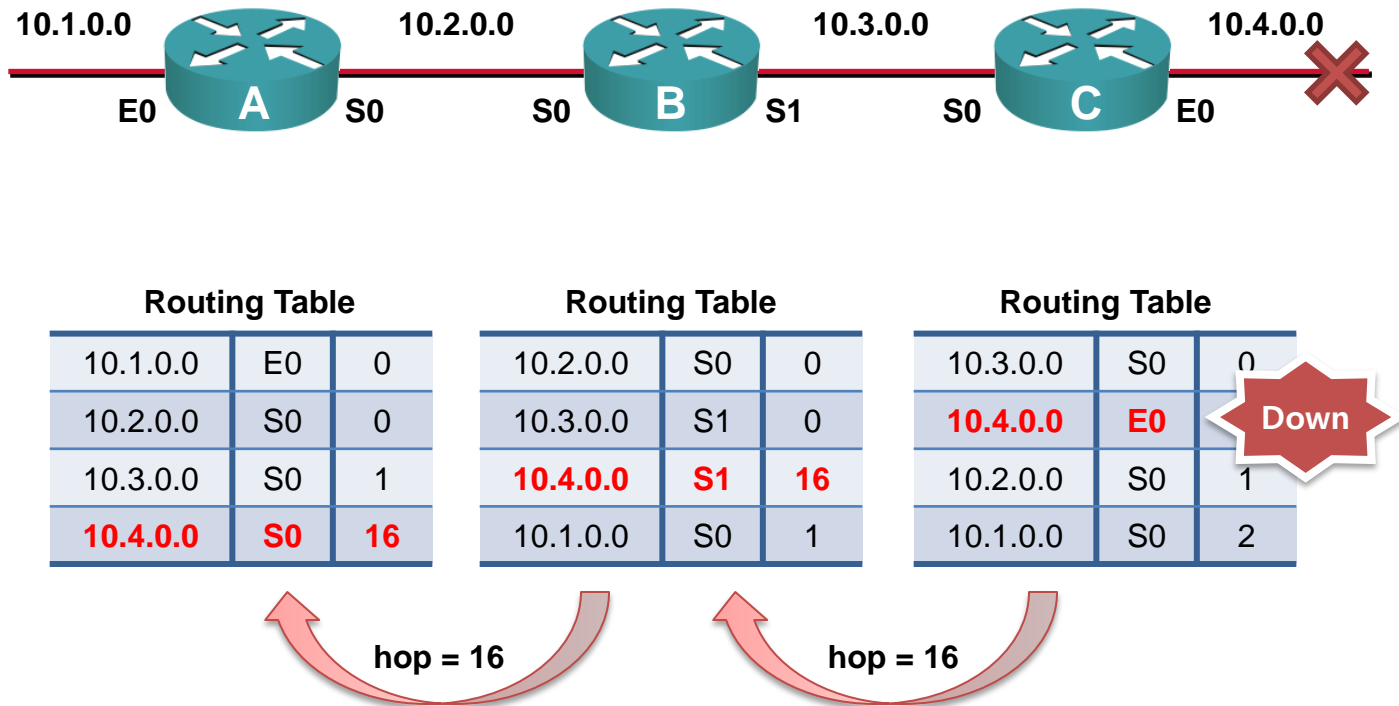
Routing Table		
10.2.0.0	S0	0
10.3.0.0	S1	0
10.4.0.0	S1	1
10.1.0.0	S0	1

Routing Table		
10.3.0.0	S0	0
10.4.0.0	E0	0
10.2.0.0	S0	1
10.1.0.0	S0	2

- 특정 interface에서 받아온 Route정보는 차후에 그 interface를 통해 다시 전달되지 않는다

# Routing Loop 문제 해결

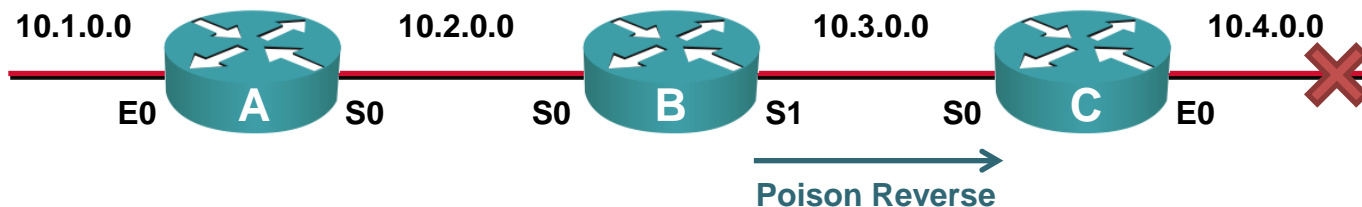
- Route Poisoning





# Routing Loop 문제 해결

- Poison Reverse



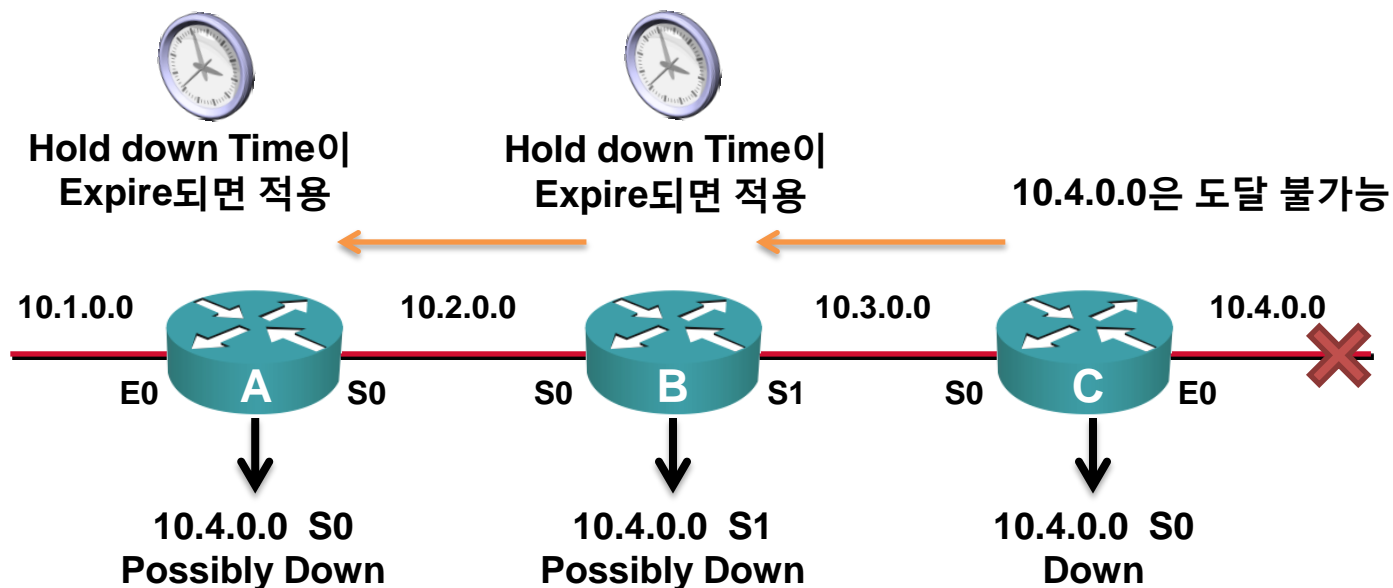
Routing Table		
10.1.0.0	E0	0
10.2.0.0	S0	0
10.3.0.0	S0	1
10.4.0.0	S0	16

Routing Table		
10.2.0.0	S0	0
10.3.0.0	S1	0
10.4.0.0	S1	0
10.1.0.0	S0	1

Routing Table		
10.3.0.0	S0	0
10.4.0.0	E0	Infinity
10.2.0.0	S0	1
10.1.0.0	S0	2

# Routing Loop 문제 해결

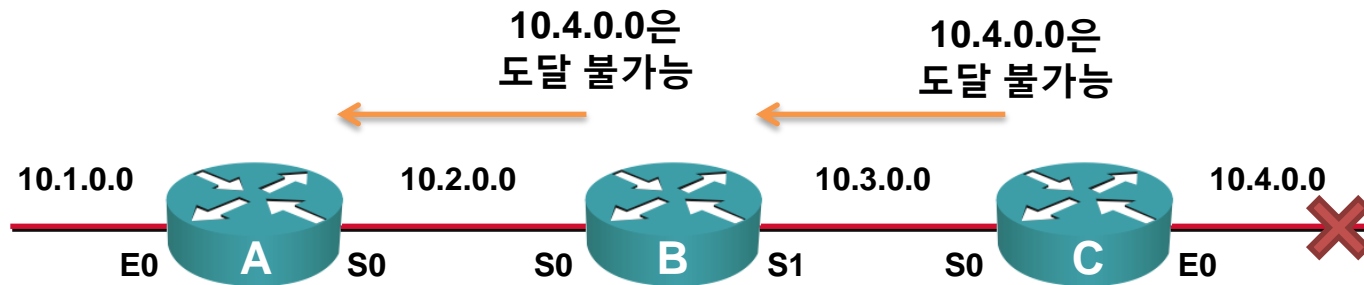
- Hold down Timer



- Router가 특정 Link의 Fail을 전달 받은 후에 해당 경로를 Routing Table에서 바로 제거하지 않고 특정 시간 동안 그 정보의 사실을 확인하기 위해 기다린다
- 이는 Topology의 변화 정보를 검증하는 용도이다

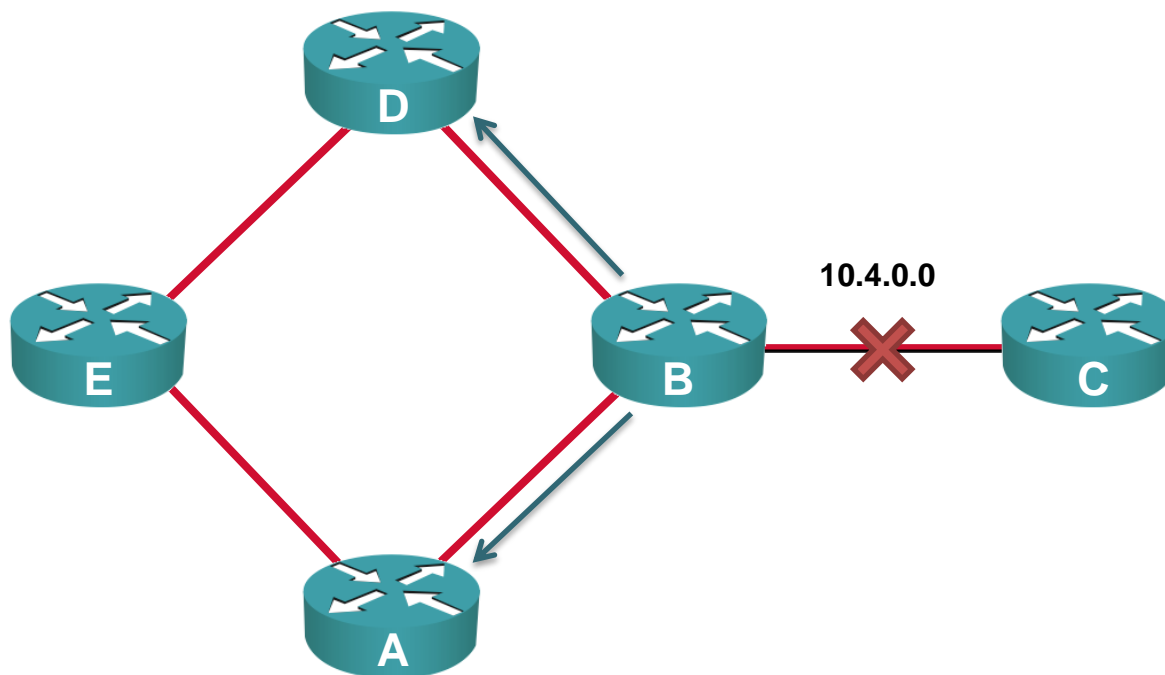
# Routing Loop 문제 해결

- Triggered Update

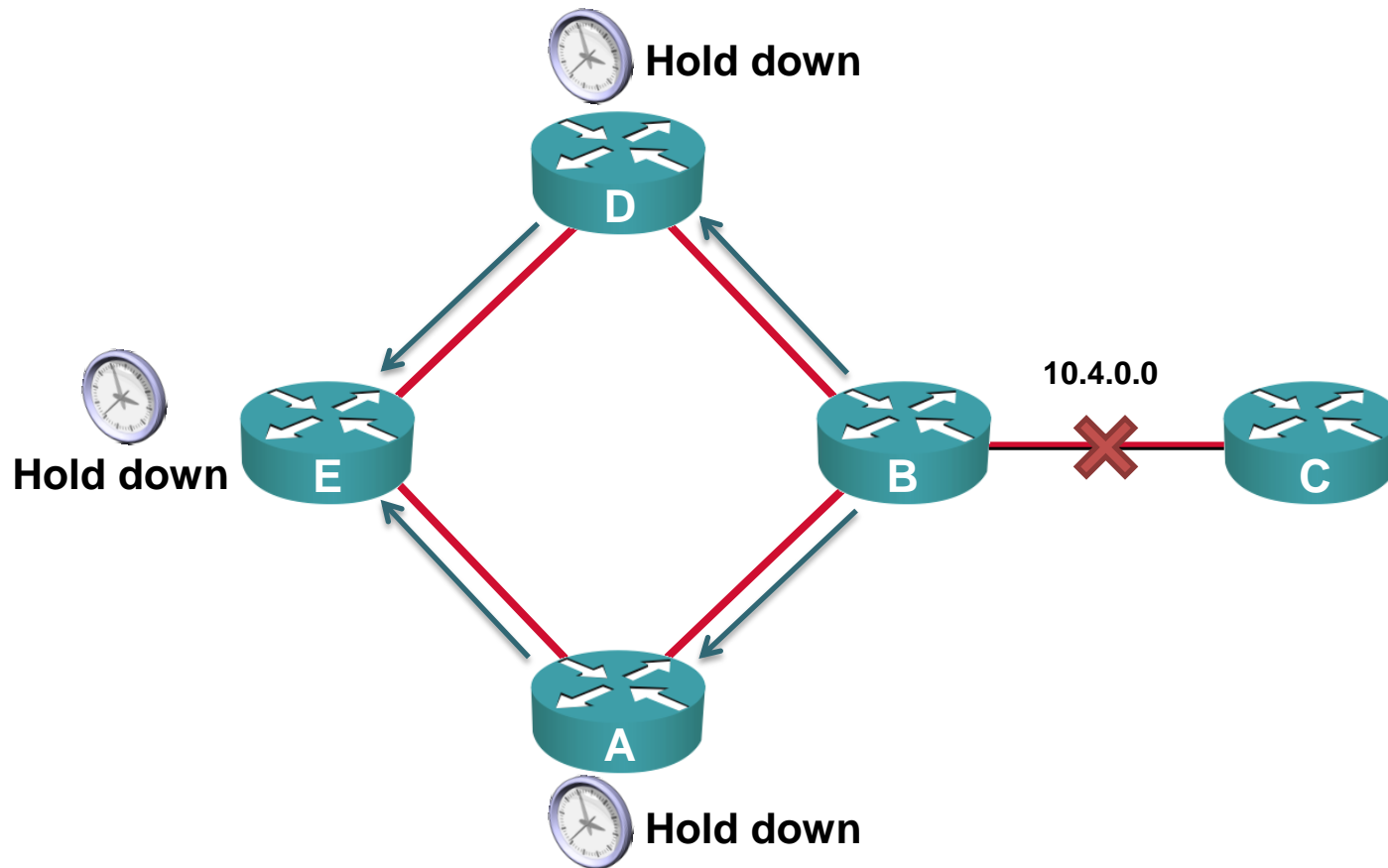


Topology의 변화를 즉시 이웃한 Router에게 알려준다

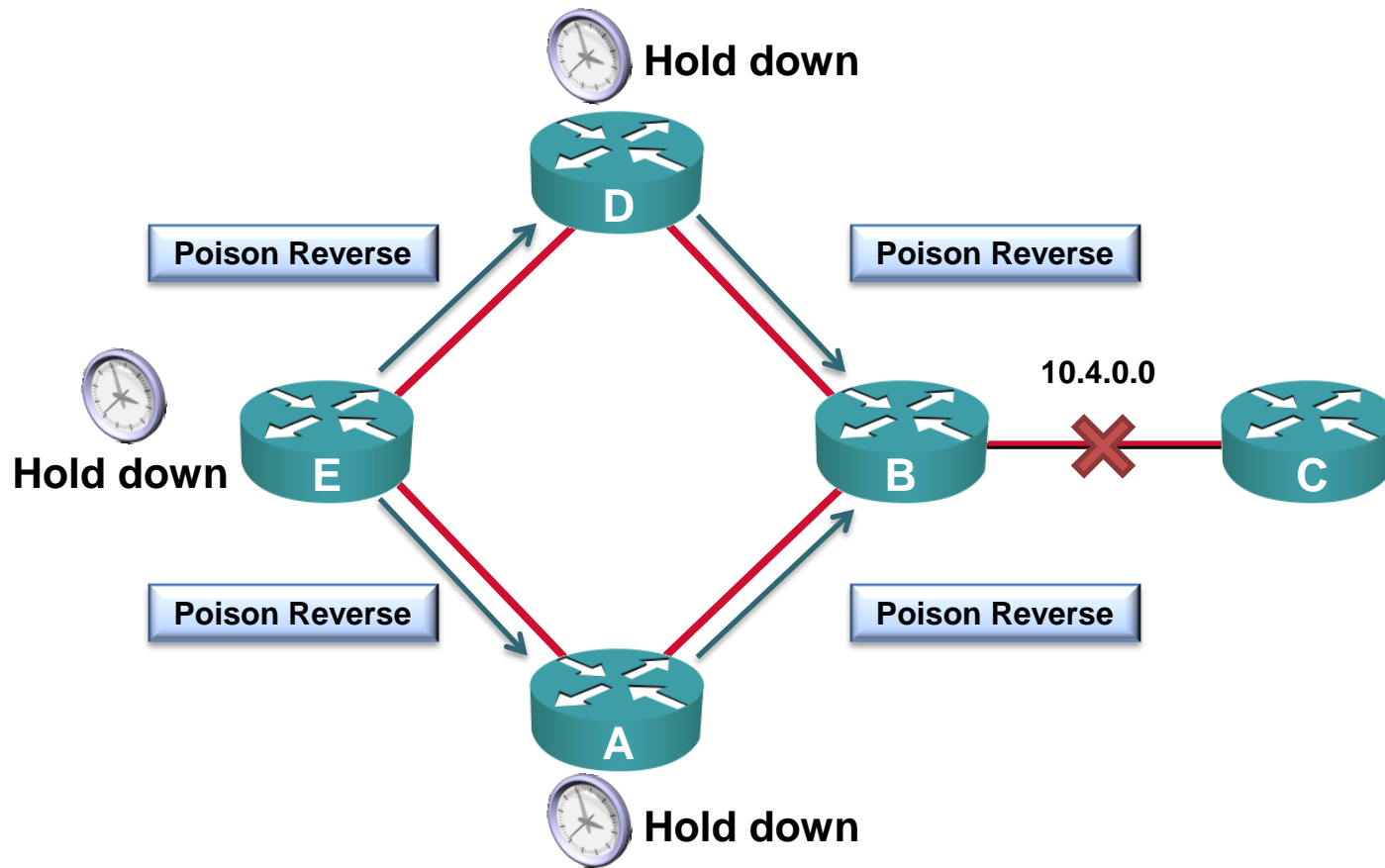
# Distance Vector Operation



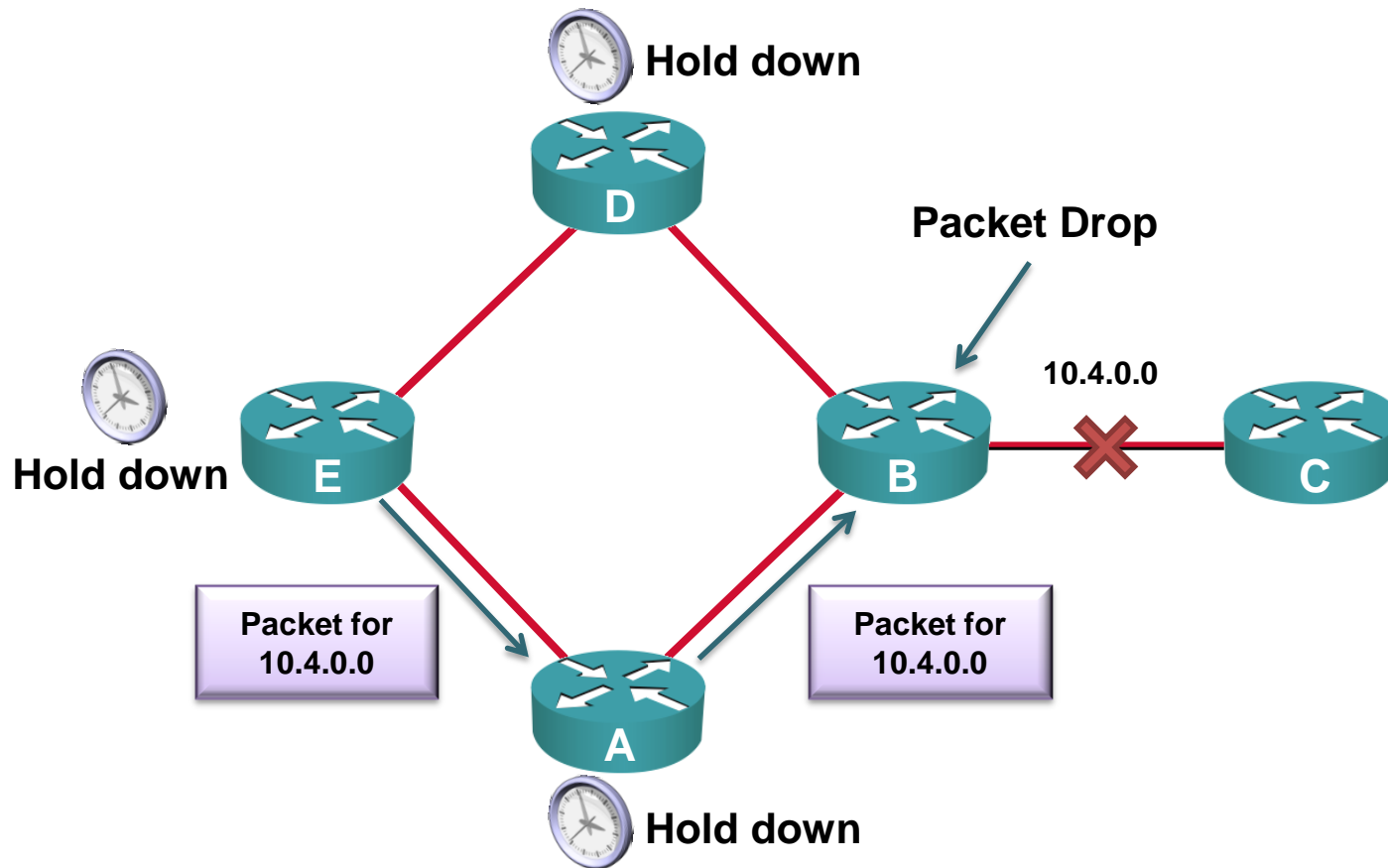
# Distance Vector Operation



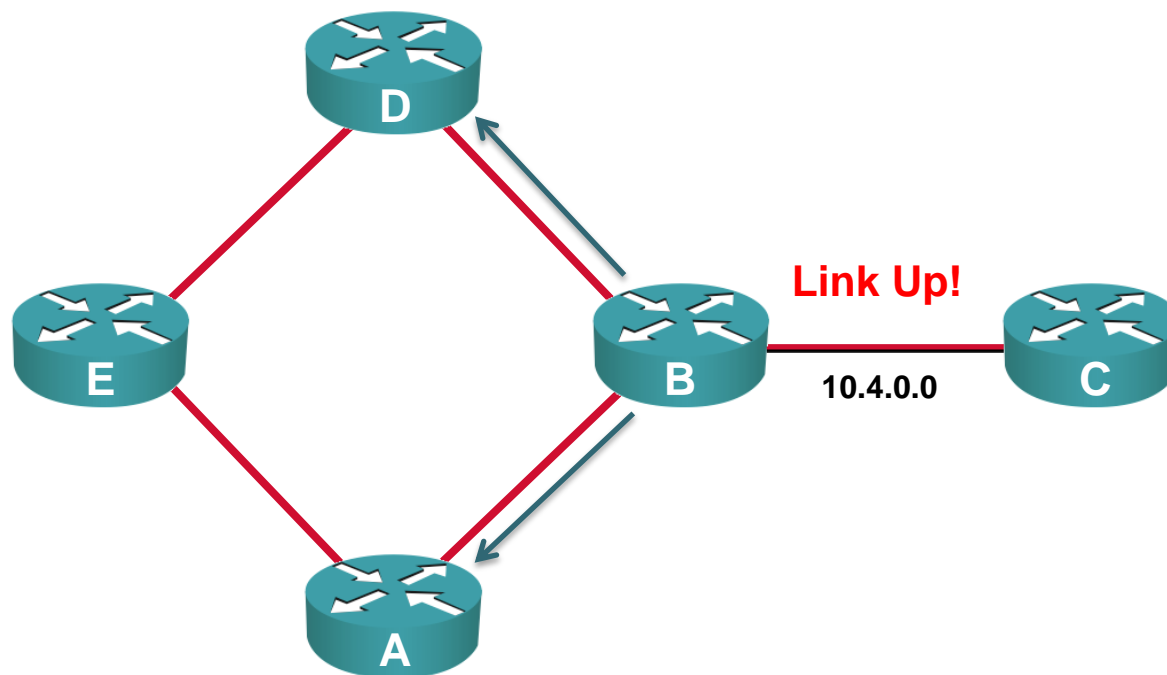
# Distance Vector Operation



# Distance Vector Operation



# Distance Vector Operation

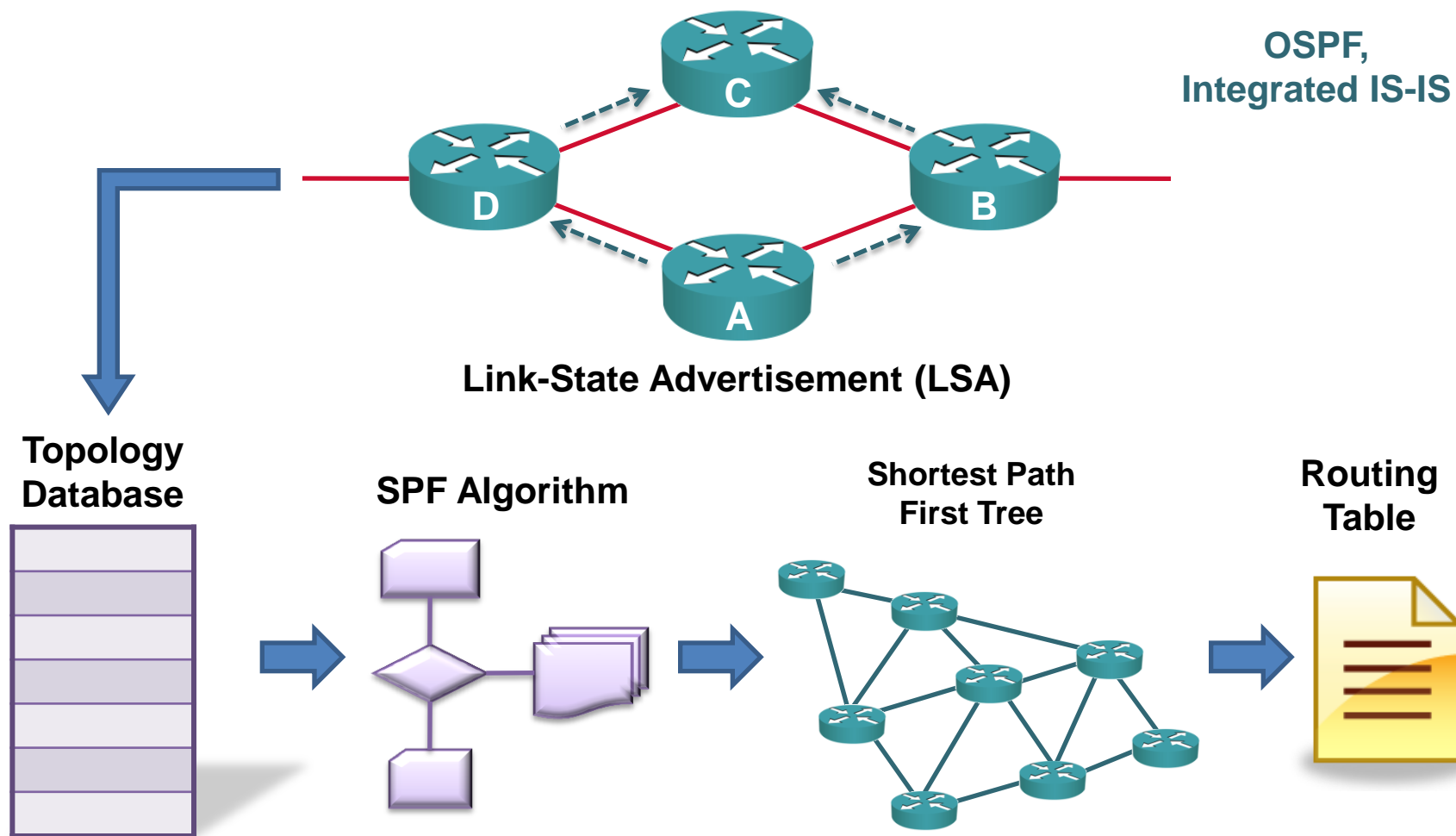




# Link-State & Hybrid Routing

- Link-State Routing Protocol 개요
- Link-State의 SPF 알고리즘
- Link-State의 계층적 구조
- Hybrid Routing Protocol

# Link-State Routing Protocol



# Link-State Routing Protocol

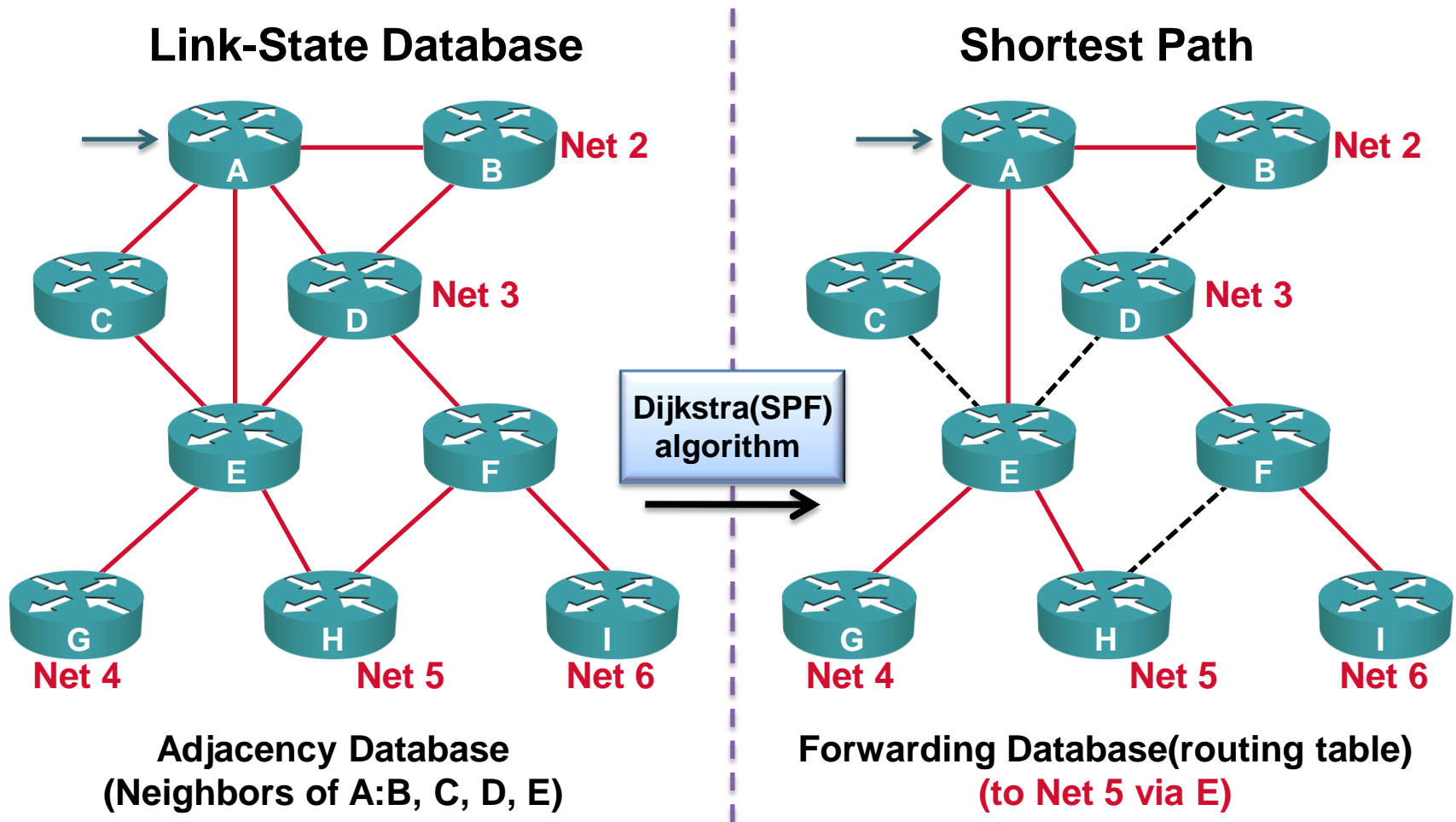
## Link-State Routing Protocol의 장점

- Fast Convergence
  - Topology의 변화에 빠른 반응을 수행한다
- Routing Loop
  - Topology를 이해하므로 SPF 알고리즘에서 Routing Loop를 방지한다
- 계층적 Design에 따라 Network 확장성이 보장된다

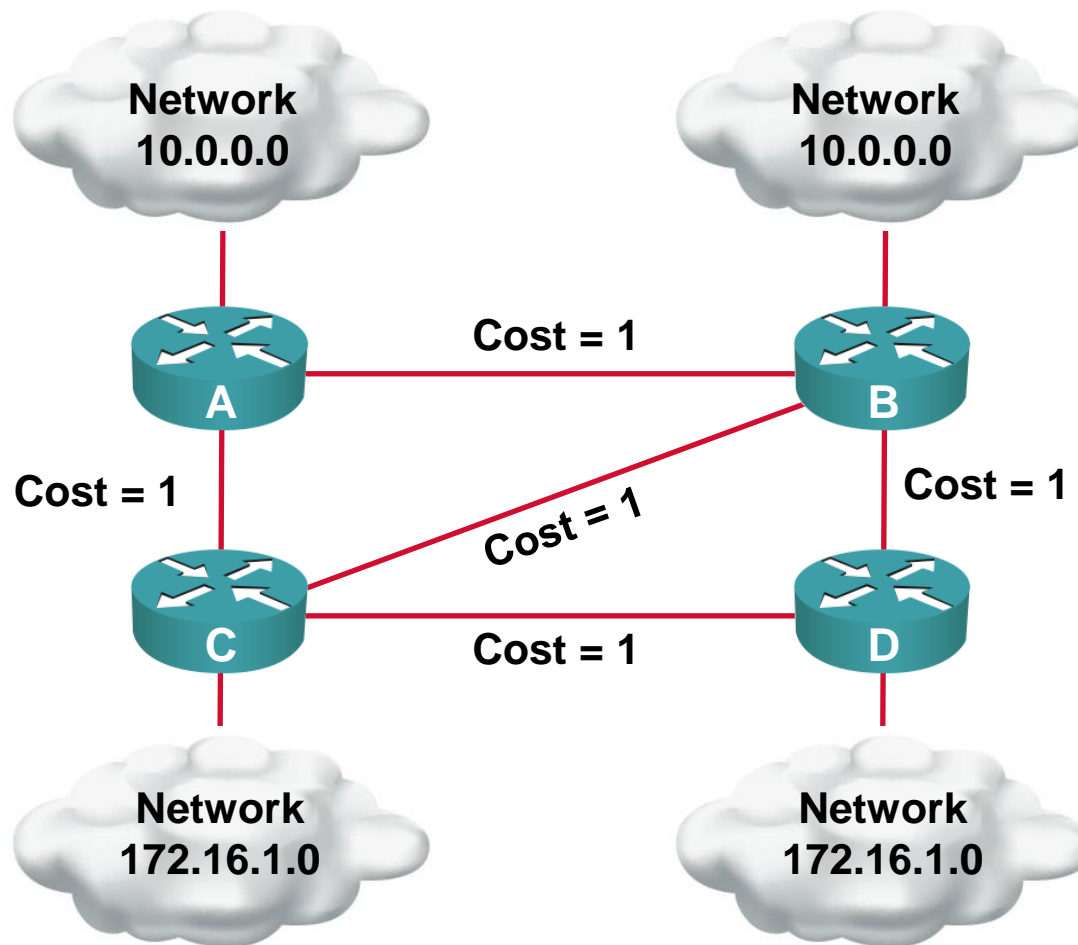
## Link-State Routing Protocol의 단점

- Router의 내부 Resource 소모가 많다
  - CPU → 잦은 SPF 알고리즘 수행
  - Memory → Network Topology 정보 관리
- 반드시 계층적 Design Rule을 따라야 한다
- 경우에 따라서는 많은 Tuning Option을 이해해야 한다

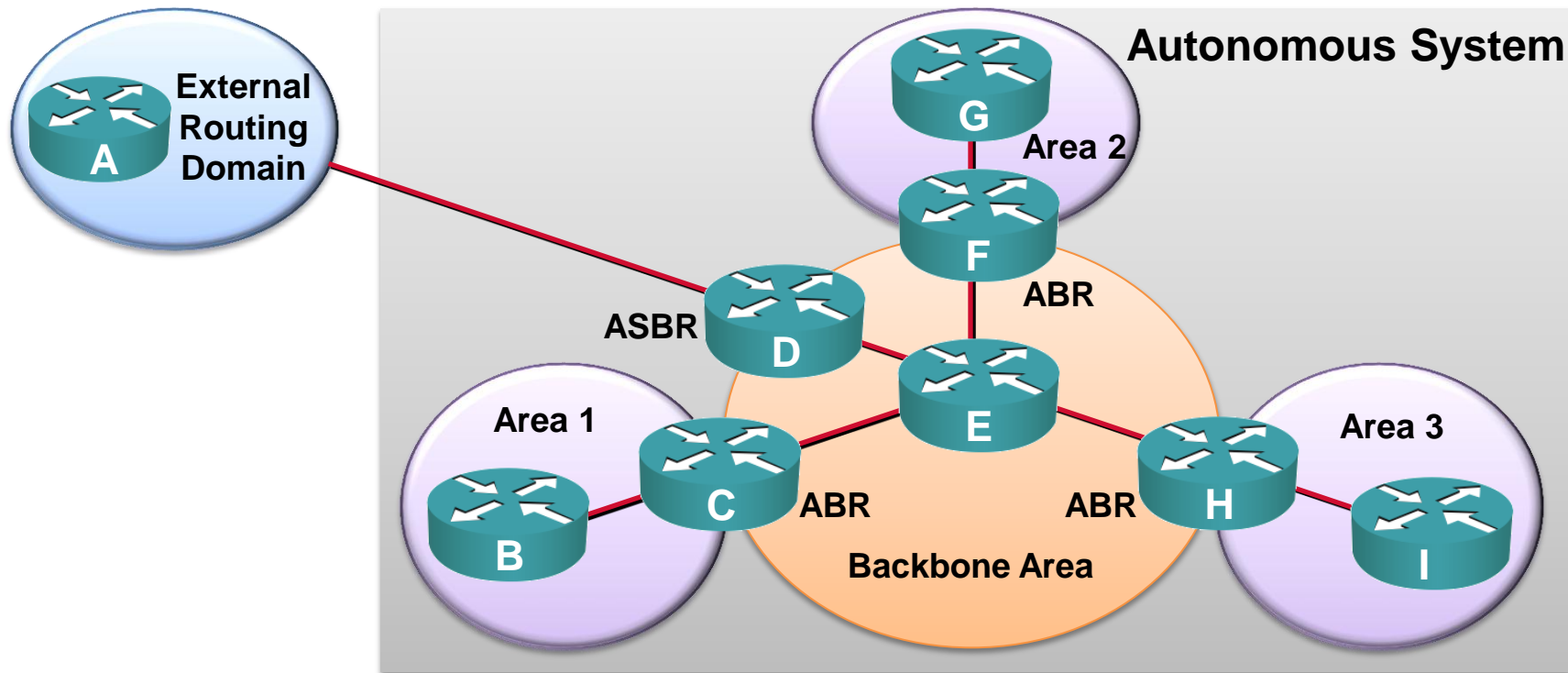
# Line-State의 SPF 알고리즘



# Link-State의 SPF 알고리즘

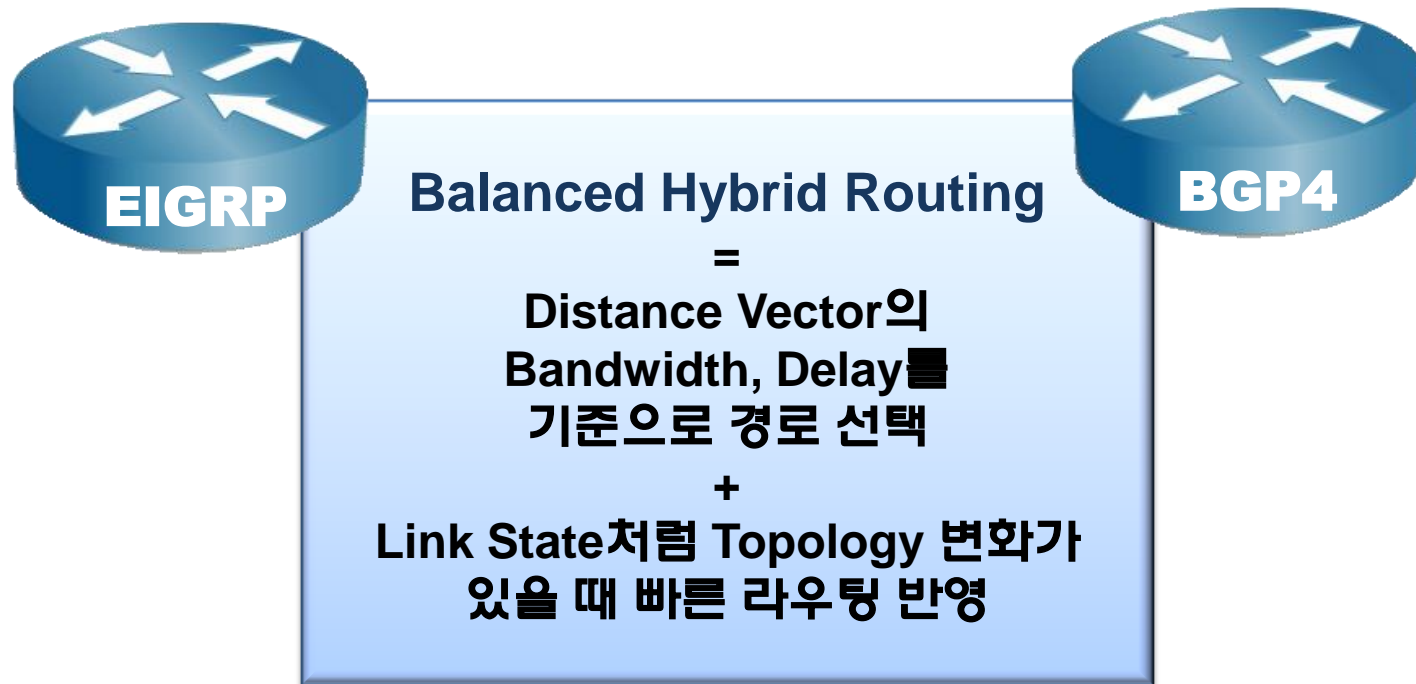


# Link-State의 계층적 구조



- 효과적인 Route Summarization을 통해 Routing Table Size 절감
- Area안에서 Topology 변화와 관련된 Traffic을 지역적으로 제한
- Router의 Processor와 Memory 자원 절감
- Routing Update Traffic 줄임

# Hybrid Routing Protocol



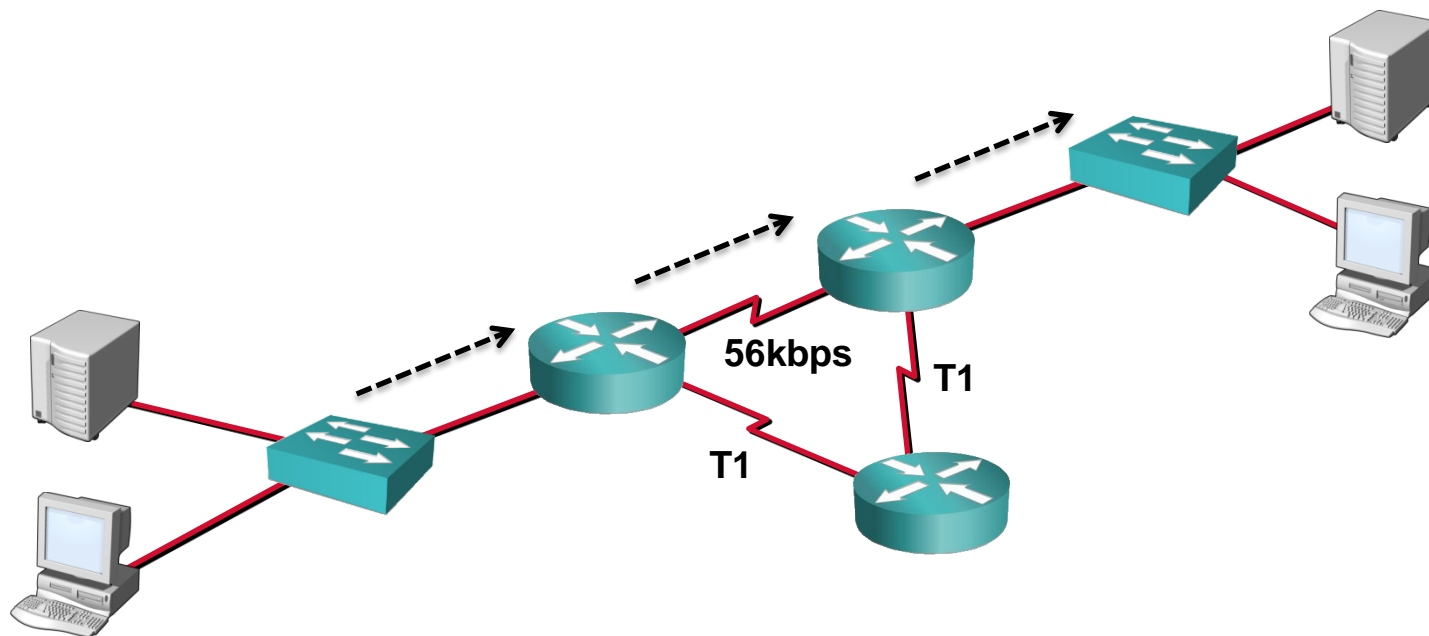
**Distance Vector와 Link-State 특성을 공유**

# RIP 구성하기

- RIP의 개요
- Dynamic Routing 설정하기
- RIP 설정하기
- RIP의 구성 검증

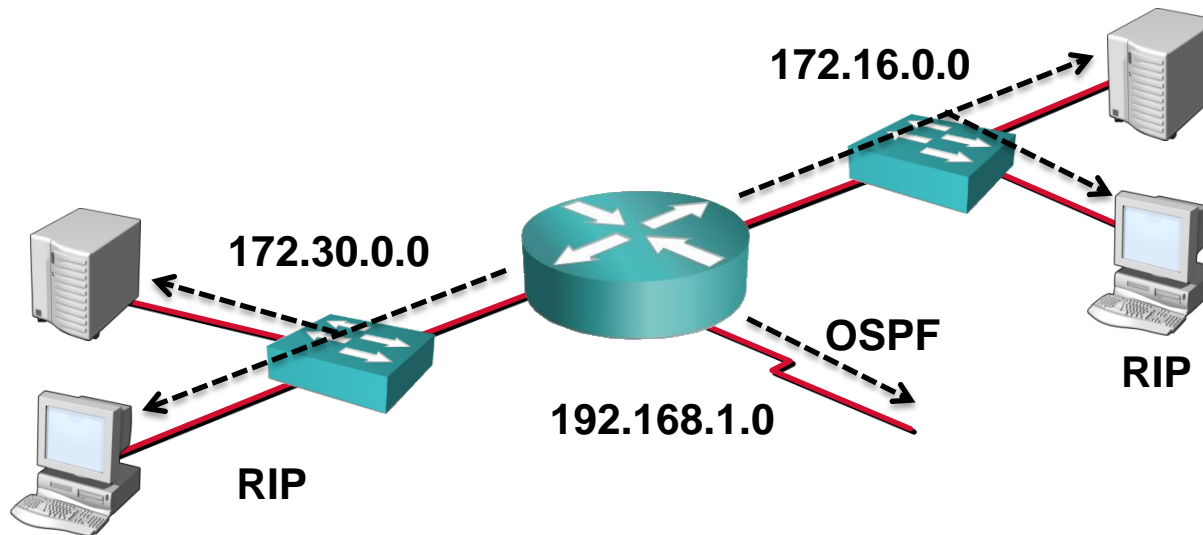


# RIP의 개요



- Hop Count를 Metric으로 사용하여 최적 경로 선택
- 매 30초마다 Routing Update를 내보냄
- 최대 6E개의 Cost qual Path 지원

# Dynamic Routing 설정하기



- Router 설정
  - Routing Protocol 선택
  - Network와 Interface 지정
  - Summarization과 같은 기타 옵션을 지정

# Dynamic Routing 설정하기

- IP Routing Protocol을 정의

```
Router(config)#router protocol [keyword]  
Router(config)#  
Router(config)#
```

- 각각의 IP Routing Process에 반드시 설정해야 하는 명령어
- 자신이 가진 Network를 알리고, 이 Network에서 파생된 IP Address가 할당된 Interface로 Routing 정보를 전송

```
Router(config-router)#network network-number [options]  
Router(config-router)#  
Router(config-router)#
```

# RIP 설정하기

- RIP Routing Process 시작

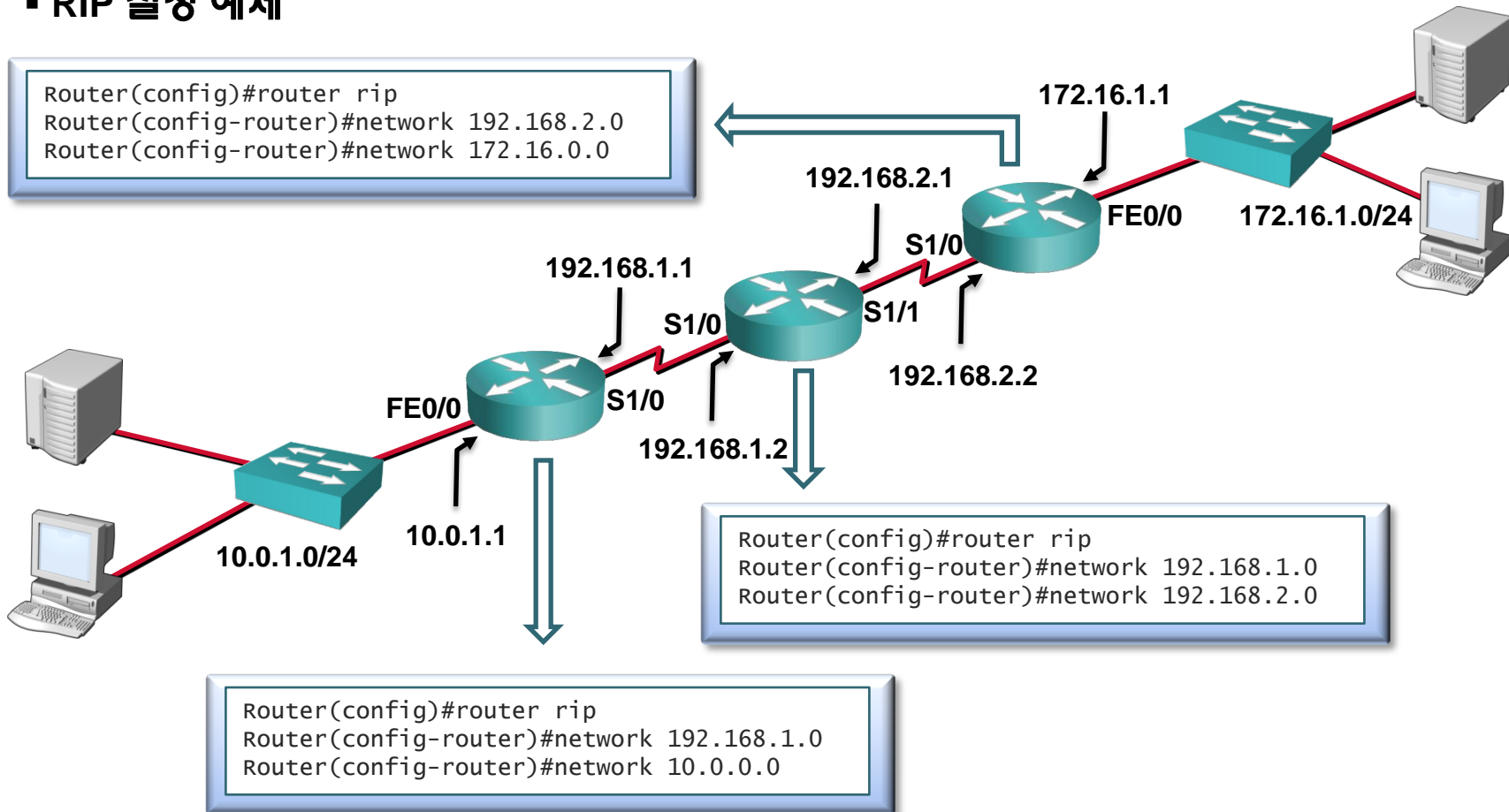
```
Router(config)#router rip  
Router(config-router)#  
Router(config-router)#
```

- 참여할 연결된 네트워크들을 선택
- Network Number는 반드시 Major Classful Network Number로 설정해야 한다

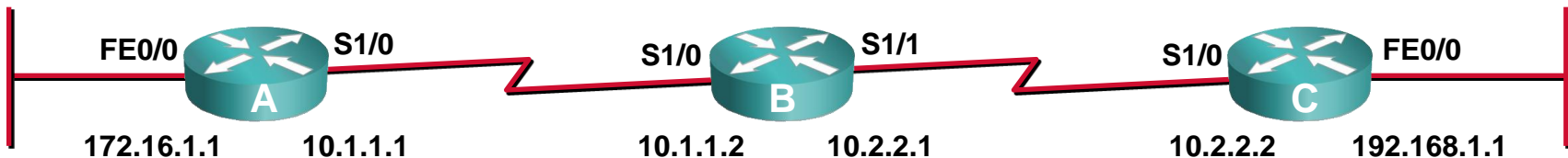
```
Router(config-router)#network network-number  
Router(config-router)#  
Router(config-router)#
```

# RIP 설정하기

## ▪ RIP 설정 예제

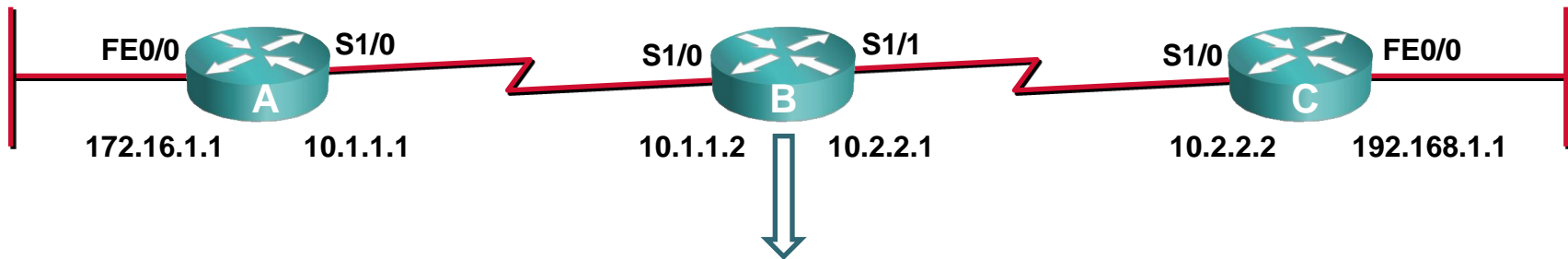


# RIP의 구성 검증



```
Router-A#show ip protocols
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 5 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 1, receive any version
    Interface        Send  Recv  Triggered RIP  Key-chain
    FastEthernet0/0    1     1 2
    Serial1/0         1     1 2
  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    10.0.0.0
    172.16.0.0
  Routing Information Sources:
    Gateway         Distance    Last Update
    10.1.1.2        120        00:00:18
  Distance: (default is 120)
```

# RIP의 구성 검증



```
Router-B#show ip route
```

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
```

```
Gateway of last resort is not set
```

```
10.0.0.0/24 is subnetted, 2 subnets
```

```
C    10.1.1.0/24 is directly connected, Serial1/0
```

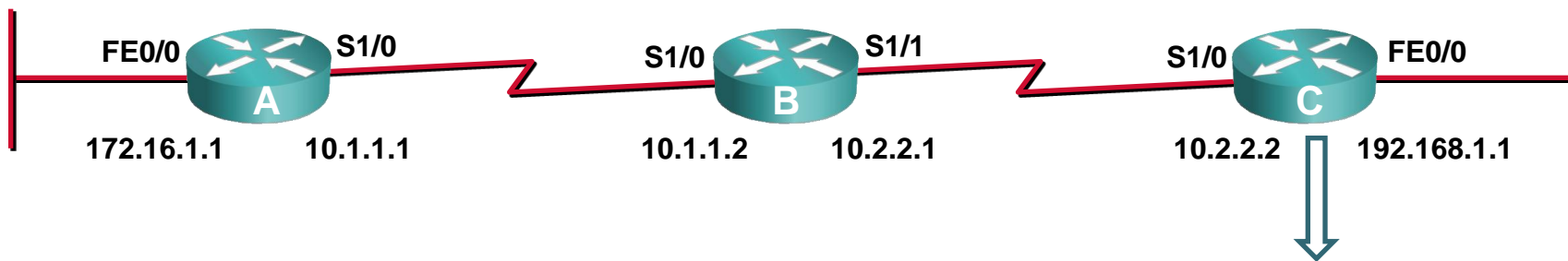
```
C    10.2.2.0/24 is directly connected, Serial1/1
```

```
R    172.16.0.0/16 [120/1] via 10.1.1.1, 00:00:18, serial1/0
```

```
R    192.168.1.0/24 [120/1] via 10.2.2.2, 00:00:20, serial1/1
```

```
Router-B#clear ip route *
```

# RIP의 구성 검증



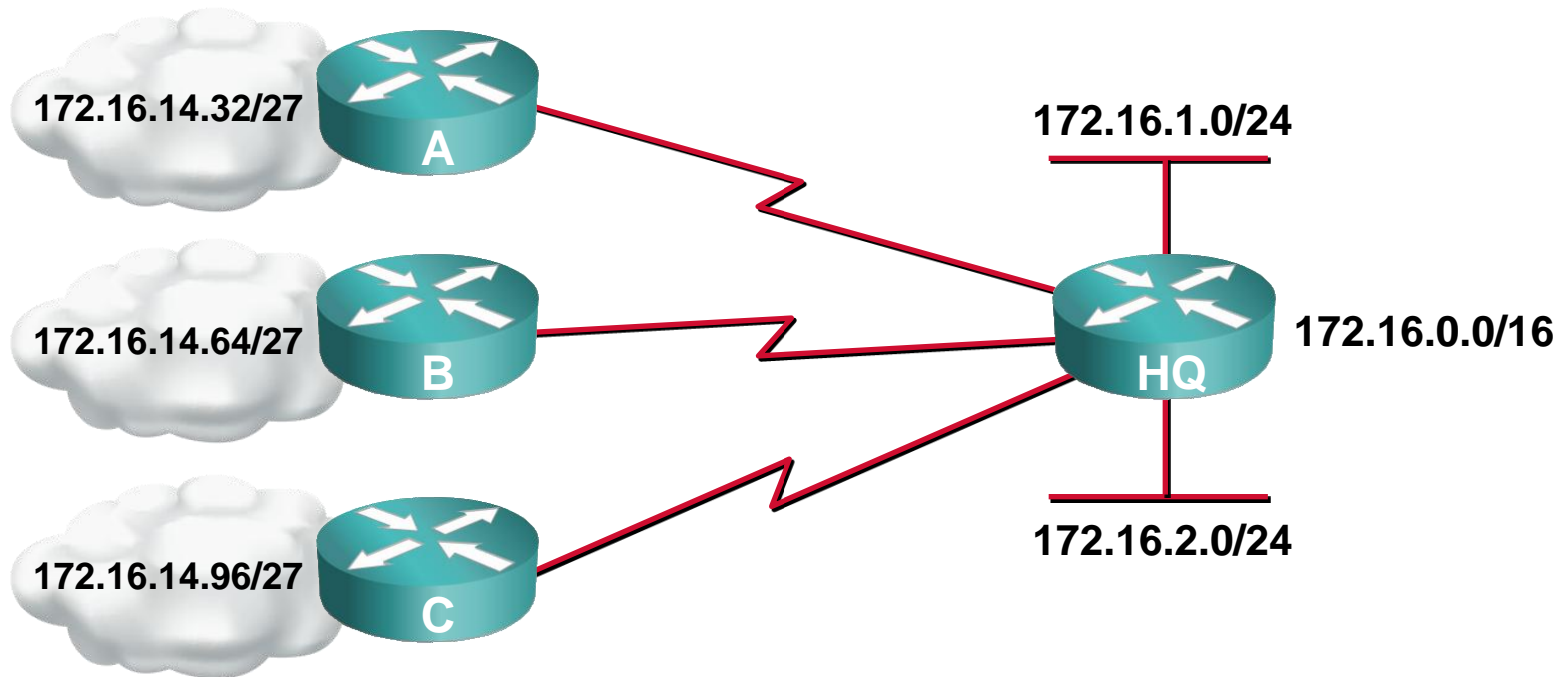
```
Router-C#debug ip rip
RIP protocol debugging is on
Router-C#
00:27:58: RIP: sending v1 update to 255.255.255.255 via FastEthernet0/0 (192.168.1.1)
00:27:58: RIP: build update entries
00:27:58:         network 172.16.0.0 metric 3
00:27:58:         network 10.0.0.0 metric 1
00:27:58: RIP: sending v1 update to 255.255.255.255 via Serial1/0 (10.2.2.2)
00:27:58: RIP: build update entries
00:27:58:         network 192.168.1.0 metric 1
00:28:00: RIP: received v1 update from 10.2.2.2 on Serial1/0
00:28:00:         10.0.0.0 in 1 hops
00:28:00:         172.16.0.0 in 2 hops
.
.
<중간 생략>
```



# VLSM & Routing Summarization

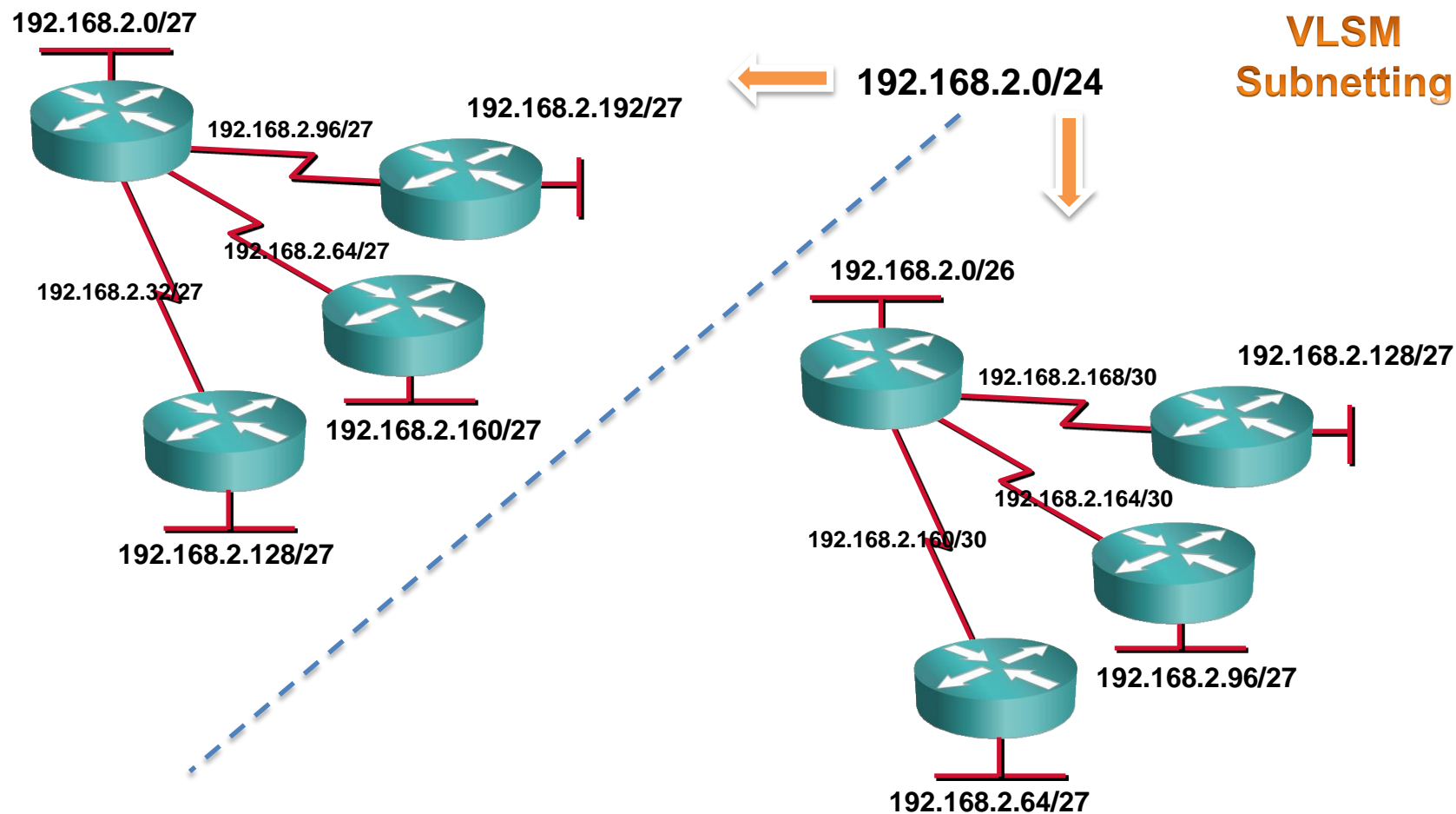
- **VLSM 개요**
- **Standard와 VLSM Subnetting 방법**
- **Route Summarization**

# VLSM 개요

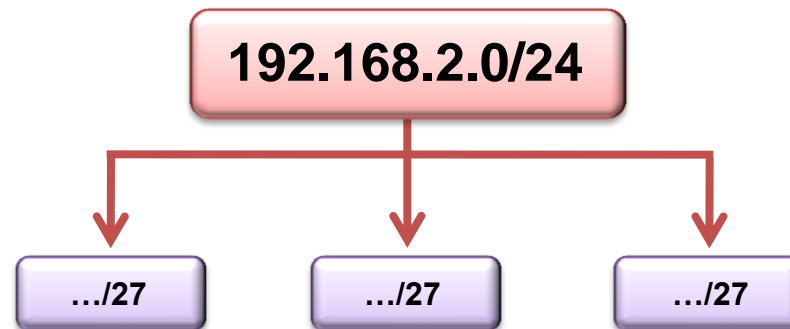
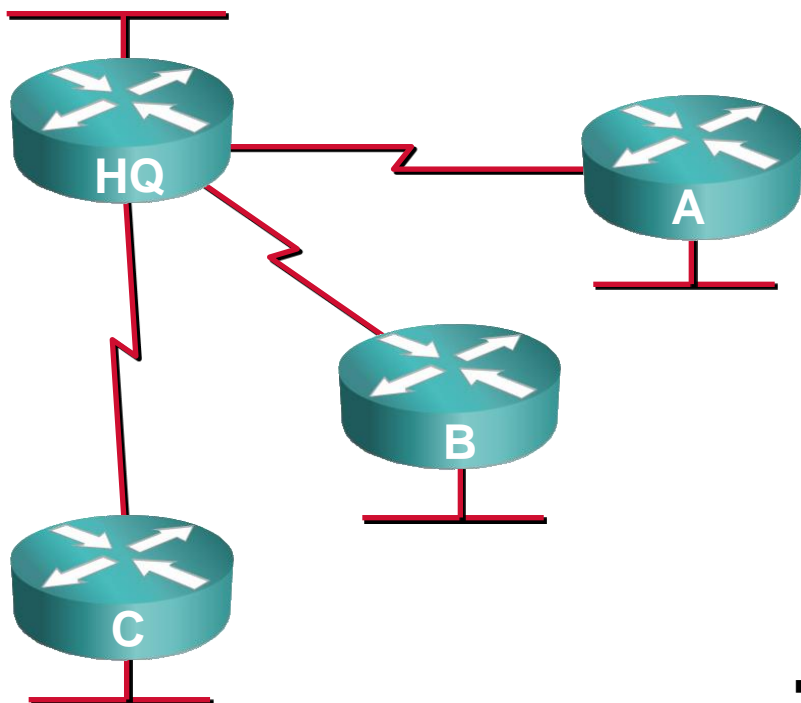


- IP Address 배분을 효율적으로 사용 가능하다
- Route Summarization의 효율성을 극대화 할 수 있다

# Standard / VLSM Subnetting

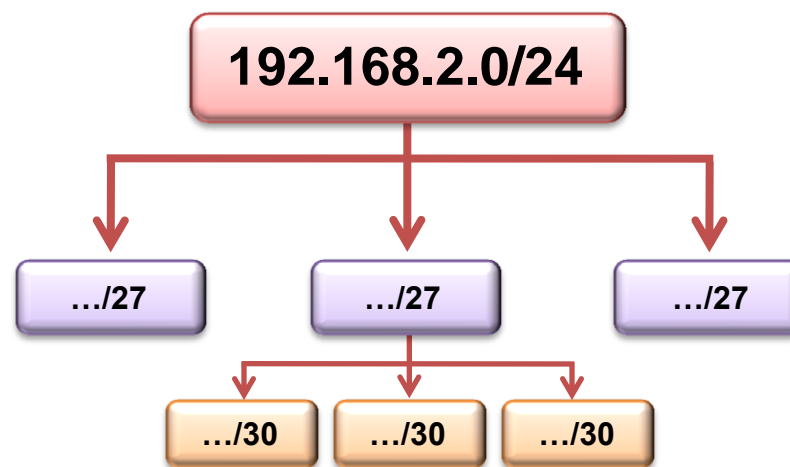
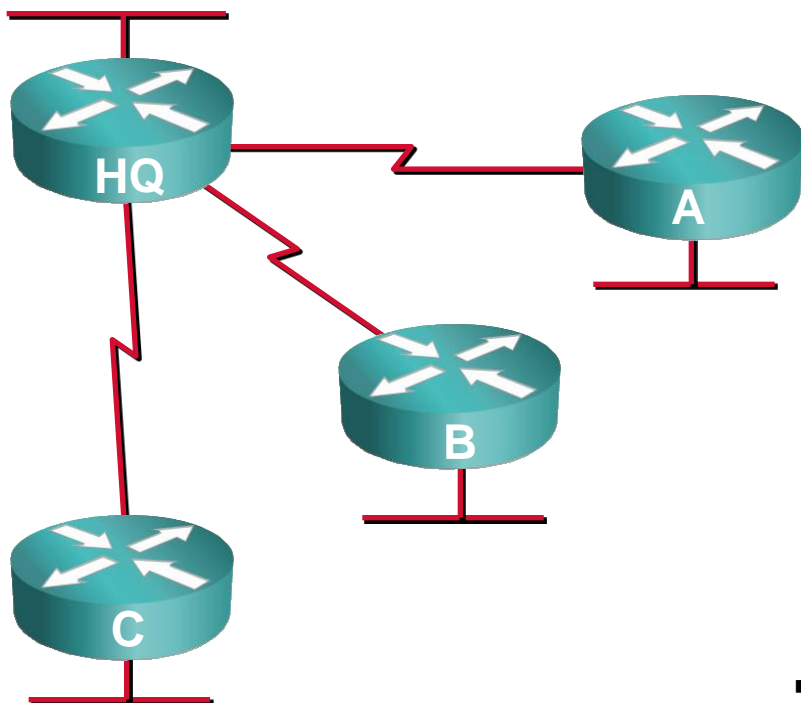


# Standard / VLSM Subnetting



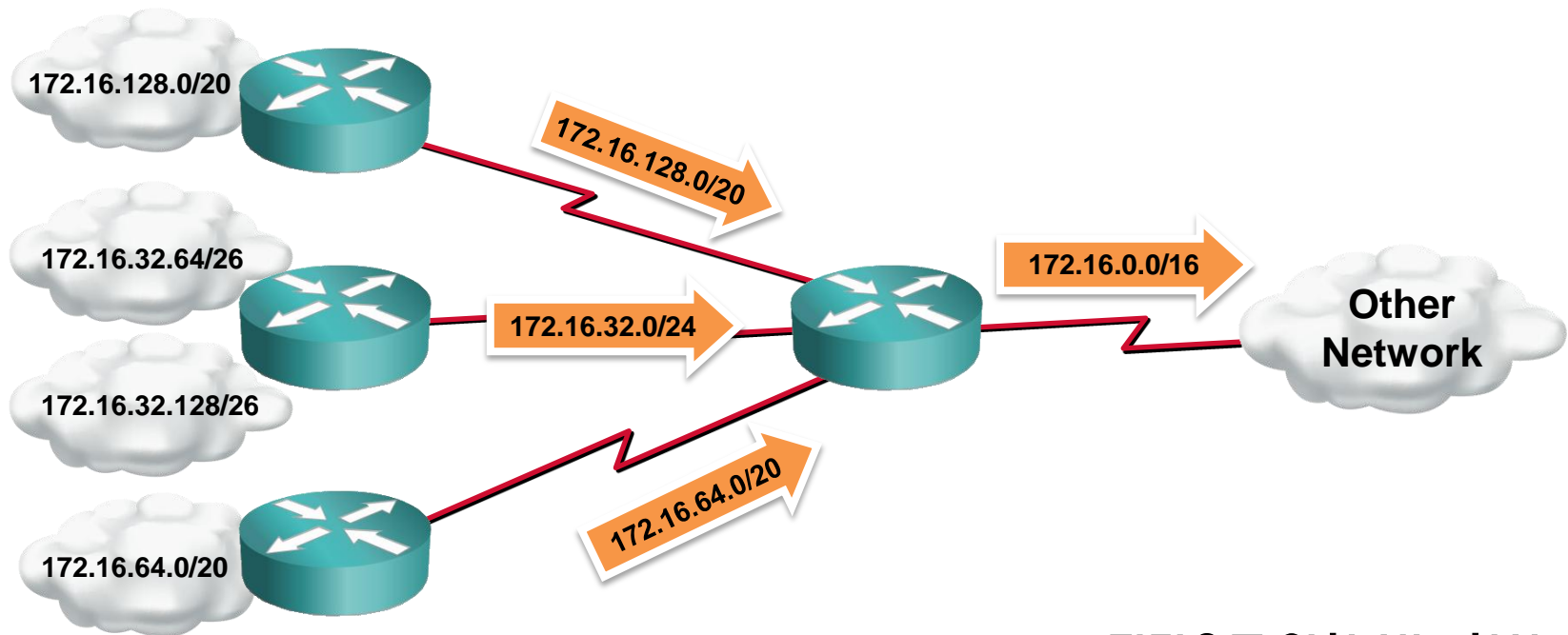
- 필요한 Subnet의 수를 결정
- Subnetting을 위한 Subnet bit의 수 결정
- IP Subnet의 범위 확인

# Standard / VLSM Subnetting



- Host의 수가 가장 많은 Subnet과 Host수가 가장 적은 Subnet들을 계층적으로 구분
- 각 Subnet을 위한 Subnet Bit의 수를 결정
- IP Subnet의 범위 확인

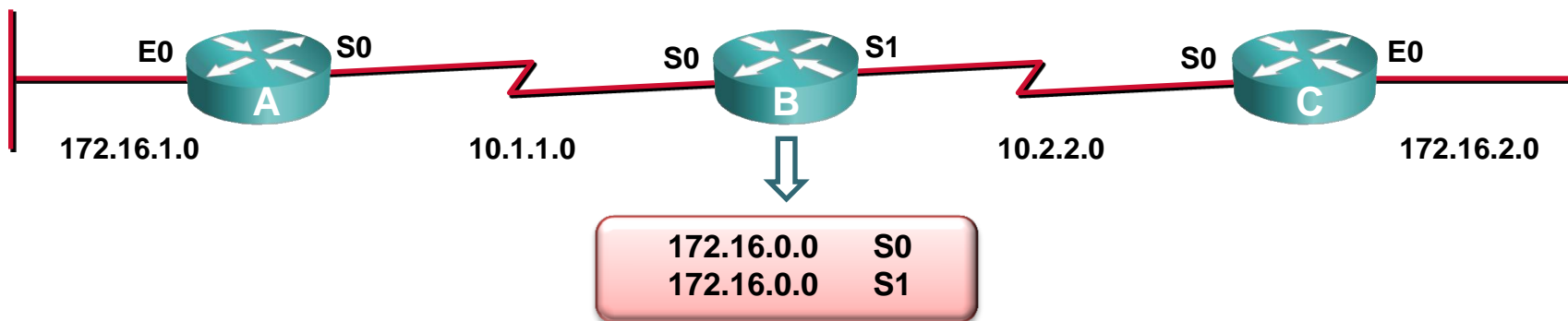
# Route Summarization



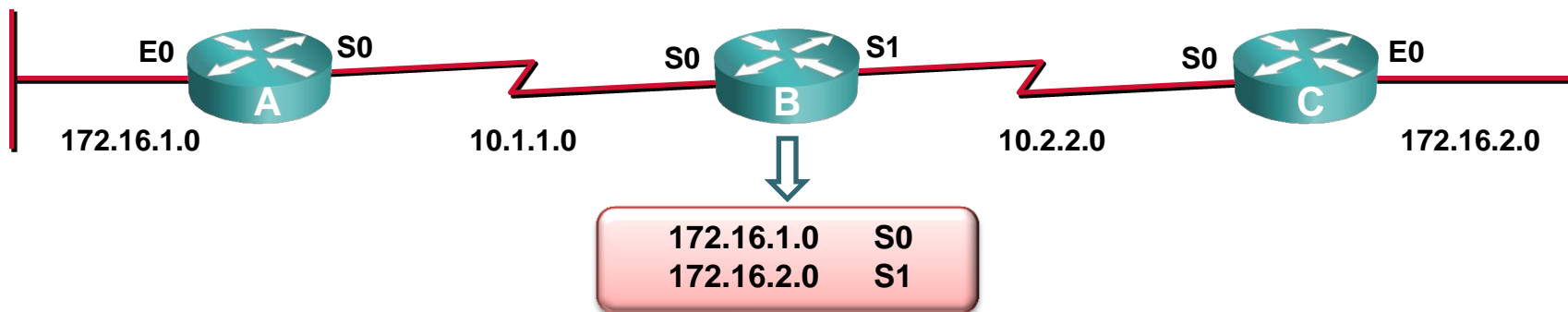
- Routing Table Size 절감으로 인한 성능 향상
- Troubleshooting 대상 영역 축소
- 계층적 Network Design 요구

# Route Summarization

## ▪ RIP Version 1과 IGRP의 경우



## ▪ RIPv2, EIGRP, OSPF, IS-IS의 경우



# Distance Vector Routing LAB

- **RIPv1 구성하기**
- **RIPv2 구성 하기 & Summary**