

7장. IP 프로토콜의 이해(2)

컴퓨터 네트워크



목차

1. 네트워크의 기초
2. 네트워크 모델
3. 네트워크 기술
4. 데이터 전송의 기초

■ 데이터 통신

5. MAC 계층
6. 데이터 링크 계층
7. **IP 프로토콜의 이해**
 1. 네트워크 계층의 기능
 2. **라우팅 프로토콜**
 3. IP 프로토콜



학습 목표

- 네트워크 계층의 필요성과 역할을 이해한다.
- 혼잡 제어 기능을 이해한다.
- 라우팅 기능을 이해하고 관련 프로토콜을 알아본다.
- IP 프로토콜 헤더의 역할을 이해한다.



간단한 라우팅 프로토콜

■ 최단 경로 라우팅

- 거리 기준은 다양하지만 중간에 거쳐가는 홉(hop) 수로 판단
- 패킷이 목적지로 가는 동안 거치는 라우터 수가 최소가 되도록 경로를 선택



[그림 7-6] 최단 경로 라우팅

- 기타 거리 기준
 - 패킷의 전송 지연, 전송 대역폭, 통신 비용 등



간단한 라우팅 프로토콜 (계속)

■ 플러딩(Flooding)

- 라우터가 입력된 패킷을 출력 가능한 모든 경로로 중개하는 방식
- 네트워크에 패킷이 무한 개 만들어질 위험
 - 홉 수를 일정 범위로 제한하고, 제거
- 중요한 데이터를 모든 호스트에게 동시에 전달하는 환경에서 제한적으로 사용

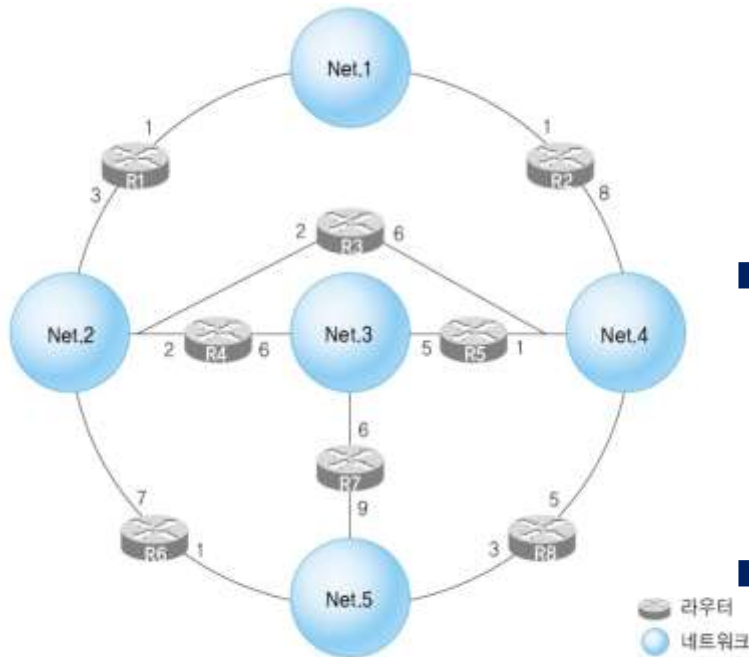


거리-벡터 프로토콜

- 라우터가 자신과 직접 연결된 주변 라우터에게 라우팅 정보를 교환하는 방식
 - 전체 네트워크에 대한 지식
 - 이웃 라우터에게만 전달
 - 일정한 주기로 정보 공유
- 교환 정보는 전체 네트워크에 속하는 개별 네트워크까지 걸리는 거리 정보
- 개별 라우터에서 유지하는 필수 정보
 - 링크 벡터: 직접 연결된 네트워크에 대한 연결 정보
 - 거리 벡터: 전체 개별 네트워크에 대한 거리 정보
 - 다음 홉 벡터: 개별 네트워크로 가기 위한 다음 홉 정보



거리-벡터 프로토콜 (계속)



[그림 7-7] 네트워크 구성의 예

■ 링크 벡터

- 링크 벡터 $L(x) = [\text{포트}(1), \text{포트}(2), \dots, \text{포트}(M)]$

- $L(R1) = [1, 3, -, -, -]$
- $L(R2) = [1, -, -, 8, -]$
- $L(R3) = [-, 2, -, 6, -]$

■ ...

■ 거리 벡터

- 거리 벡터 $D(x) = [\text{거리}(1), \text{거리}(2), \dots, \text{거리}(N)]$

- $D(R1) = [1, 1, 2, 2, 2]$
- $D(R2) = [1, 2, 2, 1, 2]$
- $D(R3) = [2, 1, 2, 1, 2]$

■ ...

■ 다음 홉 벡터

- 다음 홉 벡터 $H(x) = [\text{홉}(1), \text{홉}(2), \dots, \text{홉}(N)]$

- $H(R1) = [-, -, R4, R3, R6]$
- $H(R2) = [-, R1, R5, -, R8]$

■ ...



거리-벡터 프로토콜 (계속)

■ RIP(Routing Information Protocol)

- 거리 벡터 방식
- 소규모 네트워크 환경에 적합
- 주변 라우터가 제공하는 거리 벡터 정보가 임의의 짧은 시간 내에 모두 도착해야
 - 현실적으로 구현이 어려움 (UDP 사용 - 패킷 손실 가능성)
- 라우팅 정보 수정하는 경우
 - 거리 벡터 정보가 새로운 네트워크 주소면 적용
 - 목적지까지의 지연이 더 적으면 기존 경로를 대체
 - 거리 벡터 정보가 입력되면 등록 정보를 수정



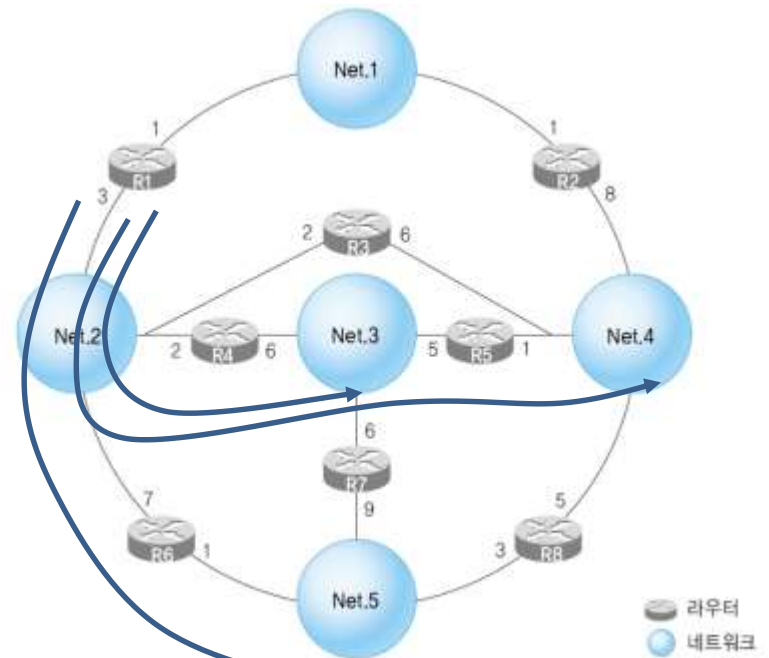
거리-벡터 프로토콜 (계속)

■ RIP(계속)

● R1의 라우팅 테이블

[표 7-1] 수정 전 라우팅 테이블

목적지 네트워크	다음 홉	거리
Net.1	-	1
Net.2	-	1
Net.3	R4	2
Net.4	R4	3
Net.5	R6	2



[그림 7-7] 네트워크 구성의 예

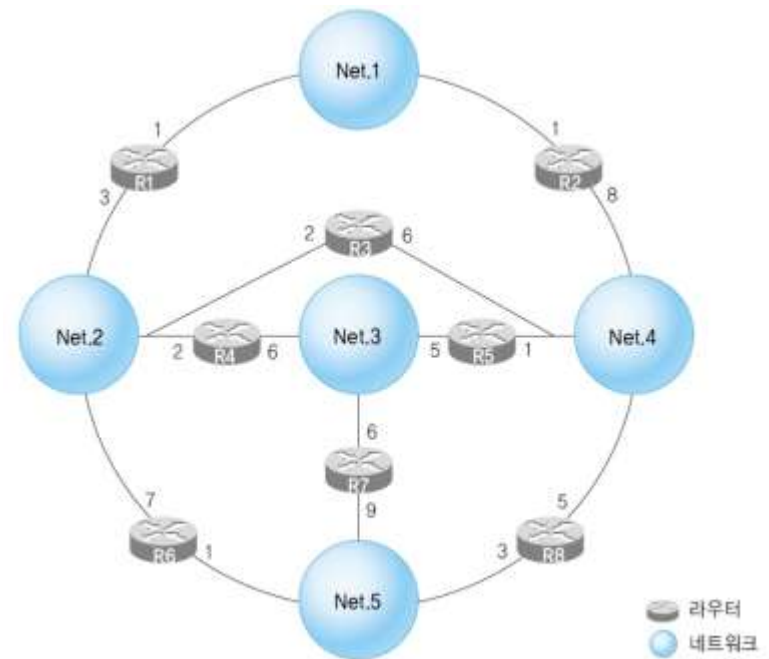


거리-벡터 프로토콜 (계속)

■ RIP(계속)

- 임의의 시점에 다음 정보 도착
 - $R2 = [1, 2, 2, 1, 2]$
- R2의 라우팅 테이블

목적지 네트워크	다음 홉	거리
Net. 1		1
Net. 2		2
Net. 3		2
Net. 4		1
Net. 5		2



[그림 7-7] 네트워크 구성의 예

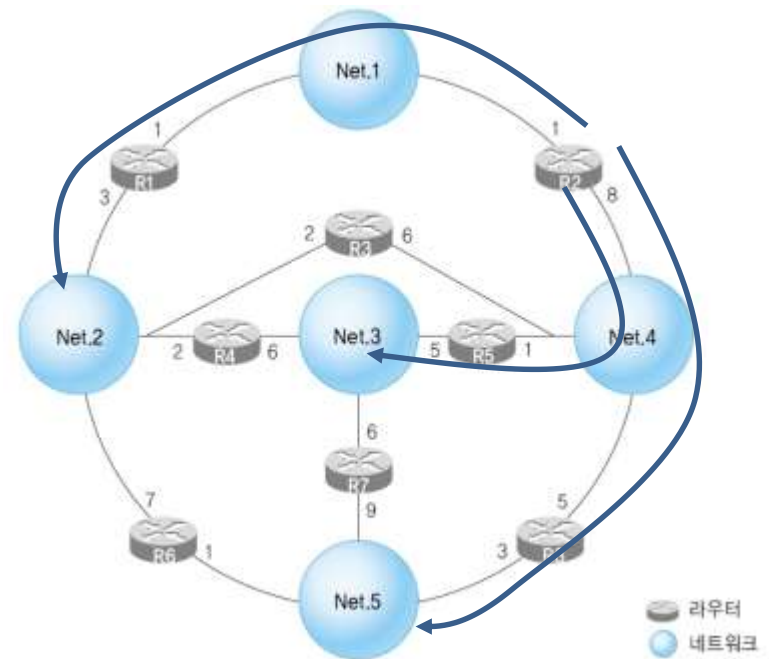


거리-벡터 프로토콜 (계속)

■ RIP(계속)

- 임의의 시점에 다음 정보 도착
 - $R2 = [1, 2, 2, 1, 2]$
- R2의 라우팅 테이블

목적지 네트워크	다음 홉	거리
Net. 1	-	1
Net. 2	R1	2
Net. 3	R5	2
Net. 4	-	1
Net. 5	R8	2



[그림 7-7] 네트워크 구성의 예

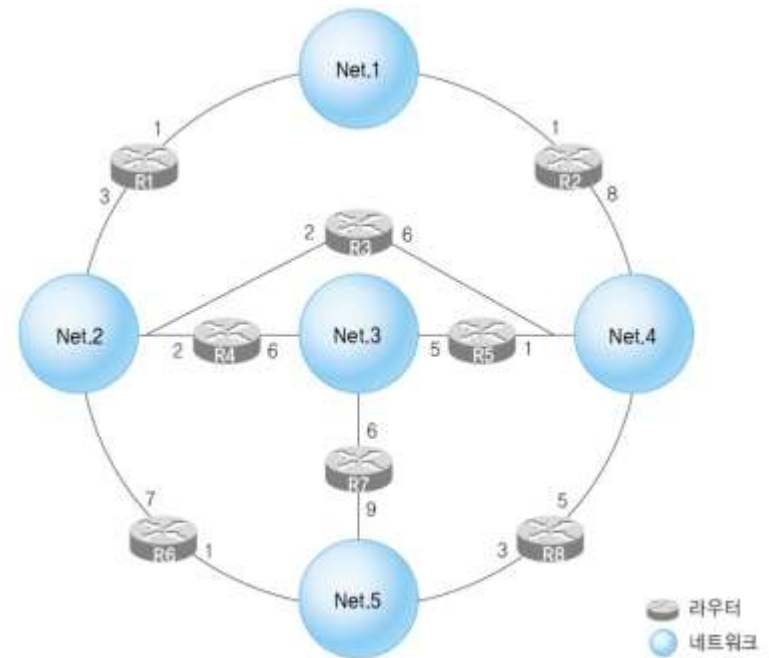


거리-벡터 프로토콜 (계속)

■ RIP(계속)

- 임의의 시점에 다음 정보 도착
 - $R3 = [2, 1, 2, 1, 2]$
- R3의 라우팅 테이블

목적지 네트워크	다음 홉	거리
Net. 1		2
Net. 2		1
Net. 3		2
Net. 4		1
Net. 5		2



[그림 7-7] 네트워크 구성의 예

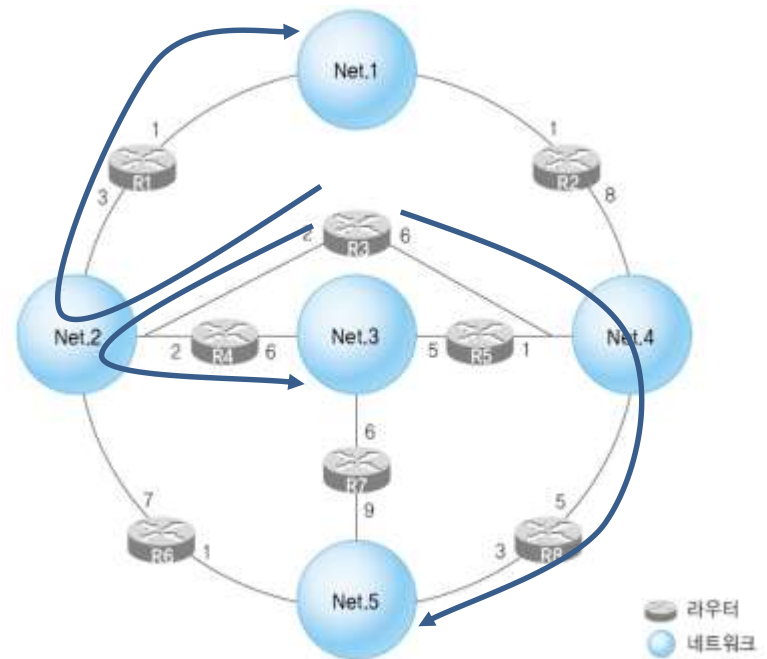


거리-벡터 프로토콜 (계속)

■ RIP(계속)

- 임의의 시점에 다음 정보 도착
 - $R3 = [2, 1, 2, 1, 2]$
- R3의 라우팅 테이블

목적지 네트워크	다음 홉	거리
Net. 1	R1	2
Net. 2	-	1
Net. 3	R4	2
Net. 4	-	1
Net. 5	R8	2



[그림 7-7] 네트워크 구성의 예

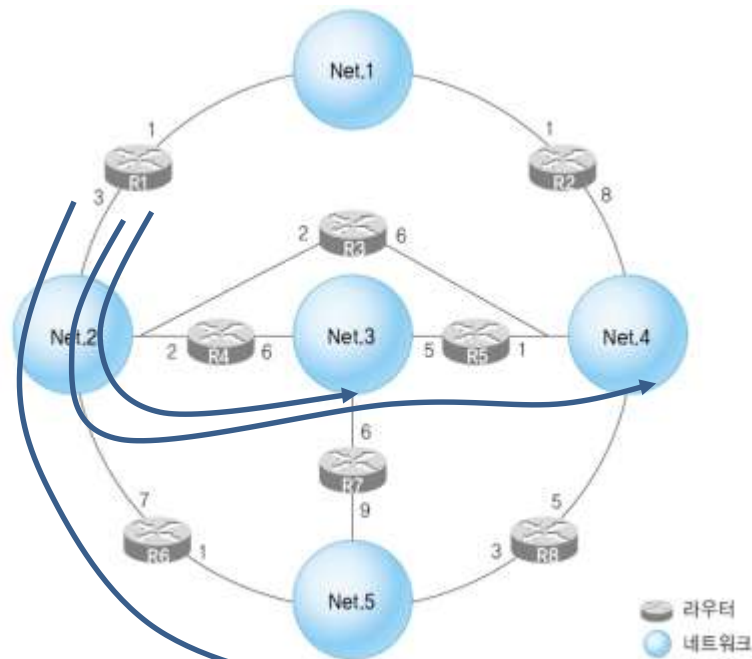


거리-벡터 프로토콜 (계속)

■ RIP (계속)

[표 7-1] 수정 전 라우팅 테이블

목적지 네트워크	다음 홉	거리
Net.1	-	1
Net.2	-	1
Net.3	R4	2
Net.4	R4	3
Net.5	R6	2



[그림 7-7] 네트워크 구성의 예

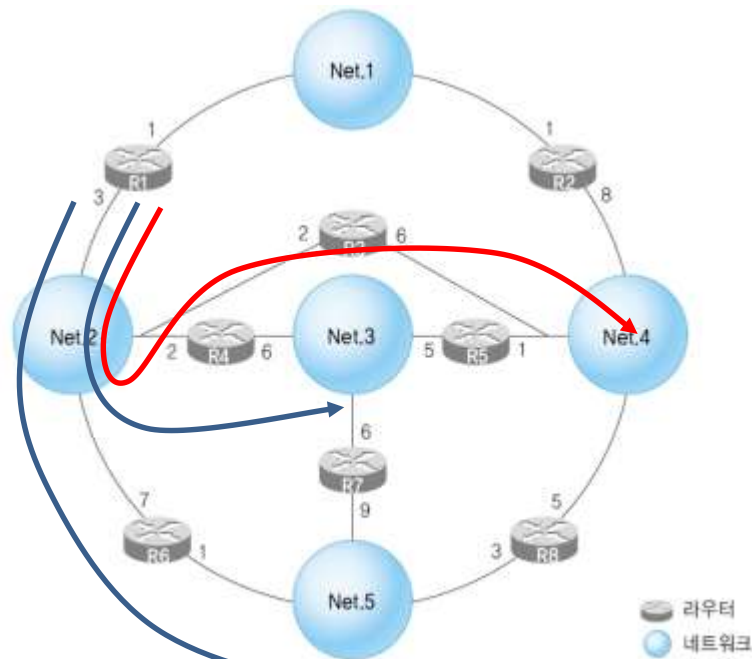
- 임의의 시점에 다음 정보 도착
 - R2 = [1, 2, 2, 1, 2]
 - R3 = [2, 1, 2, 1, 2]
 - R4 = [2, 1, 1, 2, 2]
 - R6 = [2, 1, 2, 2, 1]
- 각 라우터까지 거리 + 도착한 값과 기존 값 비교
 - R2까지 거리 1 + 도착한 값 1 < 기존 값 3
 - R3까지 거리 1 + 도착한 값 1 < 기존 값 3



거리-벡터 프로토콜 (계속)

■ RIP(계속)

- 수정 후 R1 테이블



[그림 7-7] 네트워크 구성의 예

[표 7-2] 수정 후 라우팅 테이블

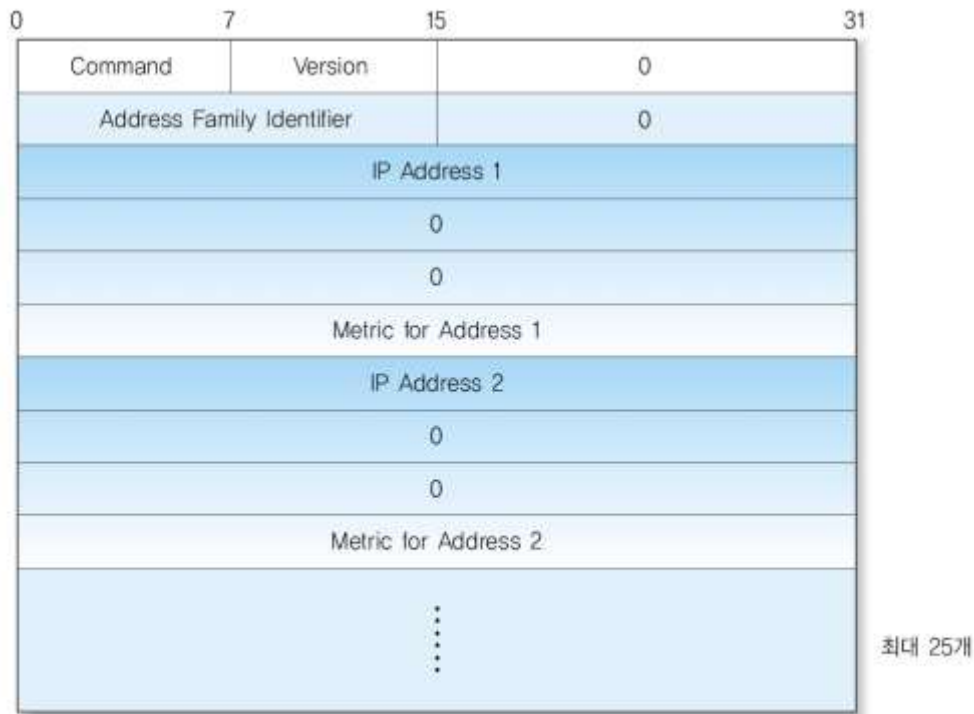
목적지 네트워크	다음 홉	거리
Net.1	-	1
Net.2	-	1
Net.3	R4	2
Net.4	R3	2
Net.5	R6	2



거리-벡터 프로토콜 (계속)

■ RIP(계속)

● 패킷 구조



[그림 7-8] RIP 패킷

- Command
 - 1이면 요구
 - 0이면 응답
 - 초기에 요구를 받으면 즉시 응답해야 함
- IP Address
 - 네트워크 지칭
- Metric
 - 목적지까지의 거리

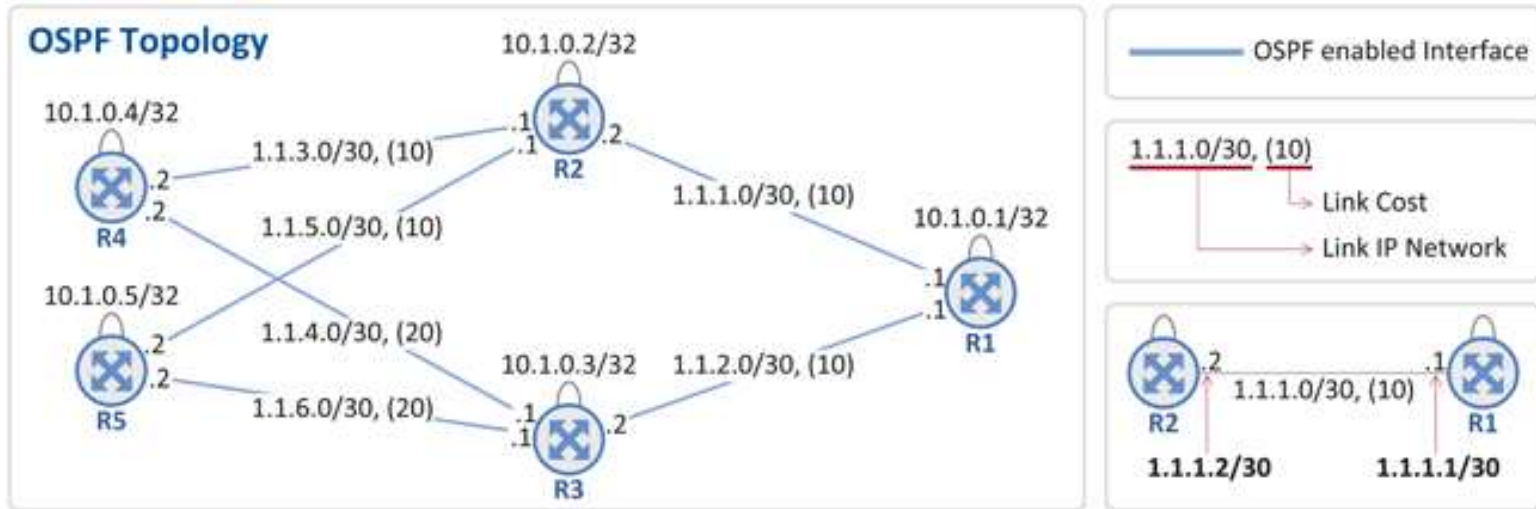


링크 상태 프로토콜

- 거리-벡터 프로토콜의 단점 개선
 - 주변 상황에 변화가 있을 때
 - 주변 라우터까지의 정보를
 - 모든 라우터에게 전달
- 플러딩(Flooding) 방식을 사용해서 정보 전달
- OSPF(Open Shortest Path First) 프로토콜



OSPF 토폴로지

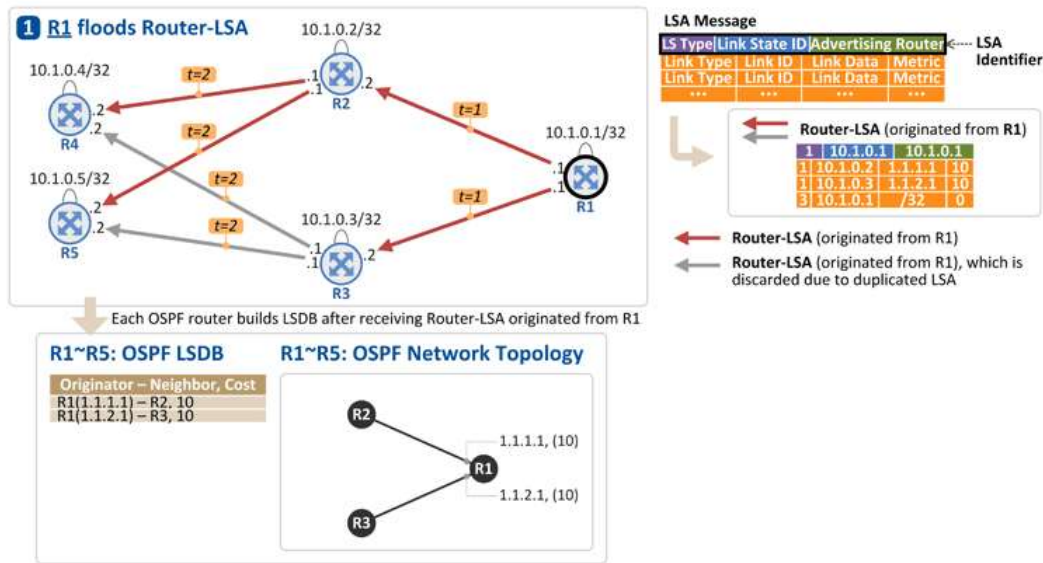


- 5개의 OSPF 라우터(R1 ~ R5)가 존재
- 라우터간 연결 링크에 대한 IP 주소와 OSPF Cost 및 각 라우터의 Loopback 주소(예. R1의 경우 10.1.0.1)가 표시
- 파란색 줄(링크)가 OSPF가 enable되어 있음을 표시



OSPF 토폴로지 (계속)

- R1이 자신의 링크 정보를 다른 OSPF 라우터들로 전파

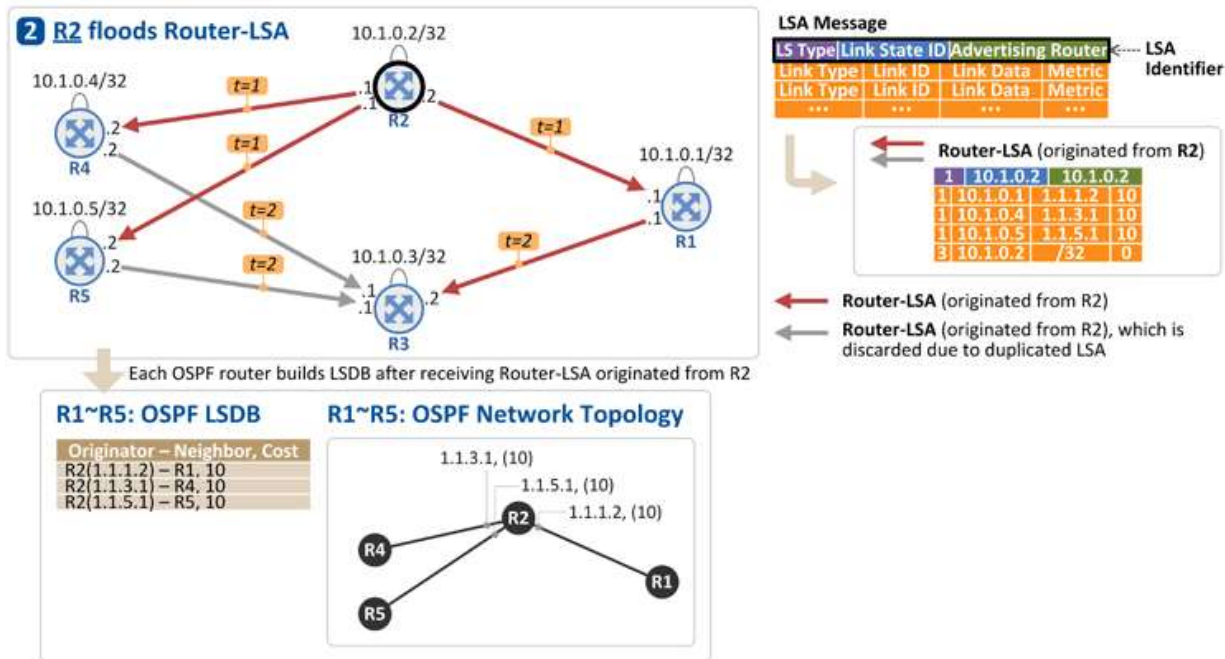


- 이를 수신한 R2, R3는 Router-LSA가 수신된 링크를 제외한 나머지 링크로 Router-LSA를 flooding
- R4와 R5는 동일한 Router-LSA를 2개씩 수신
 - LSA 식별자에 해당하는 {LS Type, Link State ID, Advertising Router}가 동일한 경우 두번째 수신한 LSA를 폐기



OSPF 토폴로지 (계속)

- R2가 자신의 링크 정보를 다른 OSPF 라우터들로 전파

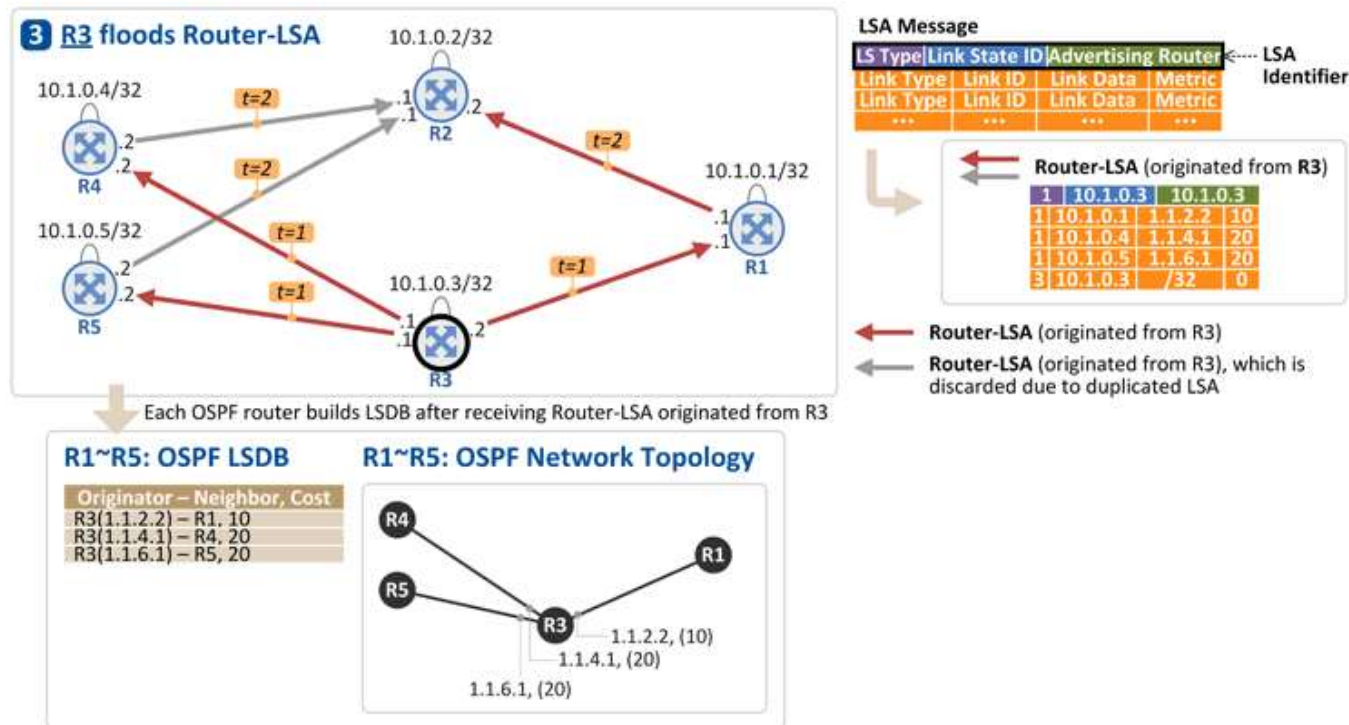


- 이를 수신한 R1, R4, R5는 Router-LSA가 수신된 링크를 제외한 나머지 링크로 Router-LSA를 flooding
- R3는 동일한 Router-LSA를 3개 수신



OSPF 토폴로지 (계속)

- R3가 자신의 링크 정보를 다른 OSPF 라우터들로 전파

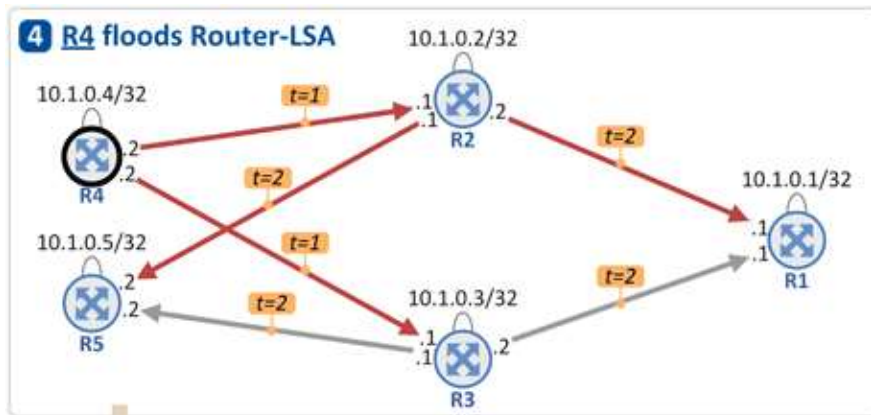


- 이를 수신한 R1, R4, R5는 Router-LSA가 수신된 링크를 제외한 나머지 링크로 Router-LSA를 flooding



OSPF 토폴로지 (계속)

- R4가 자신의 링크 정보를 다른 OSPF 라우터들로 전파



LSA Message

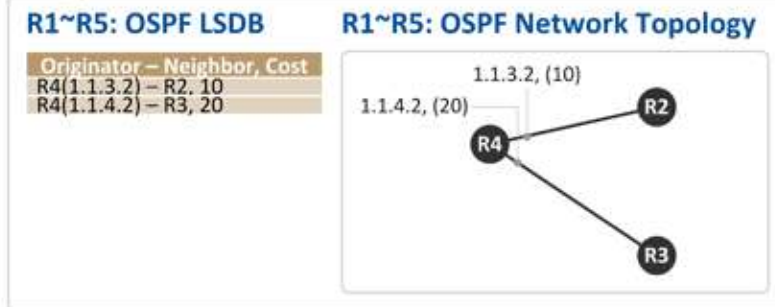
LS Type	Link State ID	Advertising Router	LSA Identifier
Link Type	Link ID	Link Data	Metric
...

Router-LSA (originated from R4)

1	10.1.0.4	10.1.0.4	
1	10.1.0.2	1.1.3.2	10
1	10.1.0.3	1.1.4.2	20
3	10.1.0.4	/32	0

- Router-LSA (originated from R4)
- Router-LSA (originated from R4), which is discarded due to duplicated LSA

Each OSPF router builds LSDB after receiving Router-LSA originated from R4

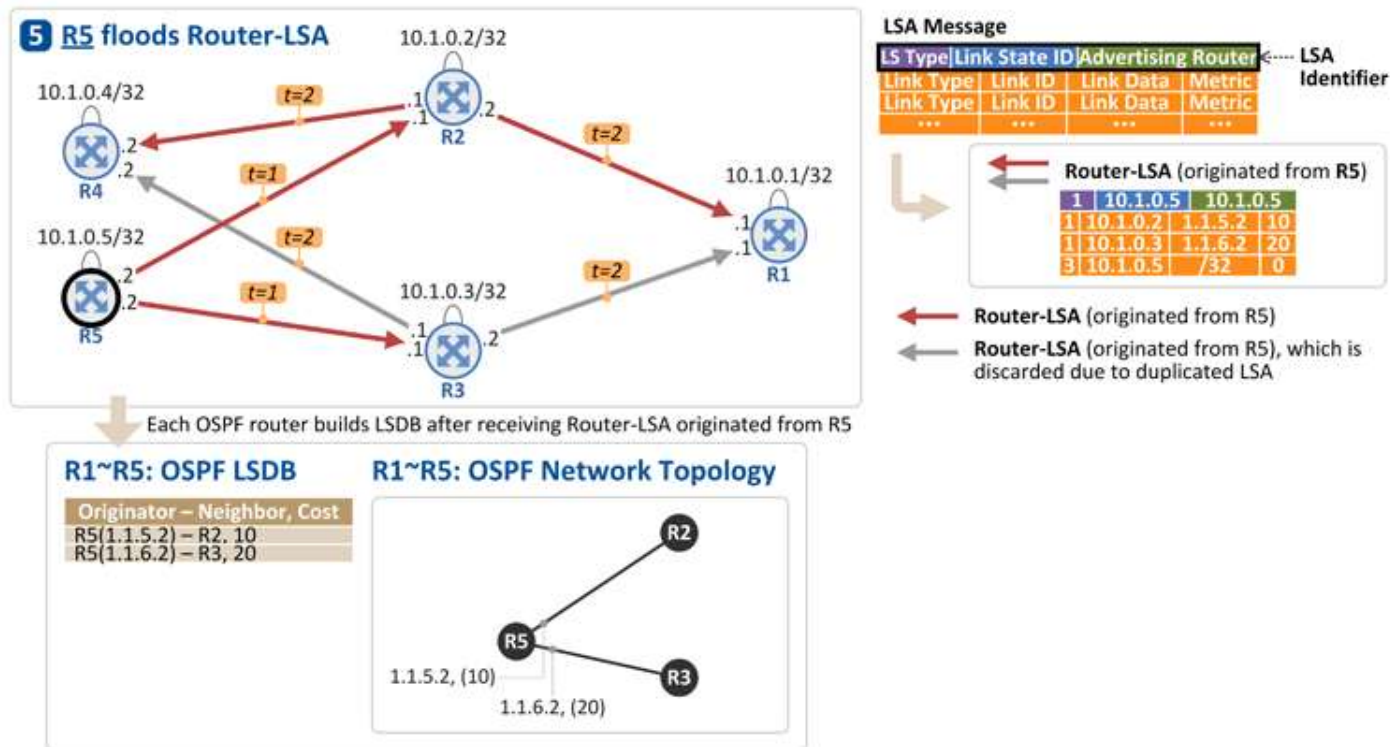


- 이를 수신한 R2, R3는 Router-LSA가 수신된 링크를 제외한 나머지 링크로 Router-LSA를 flooding



OSPF 토폴로지 (계속)

- R5가 자신의 링크 정보를 다른 OSPF 라우터들로 전파

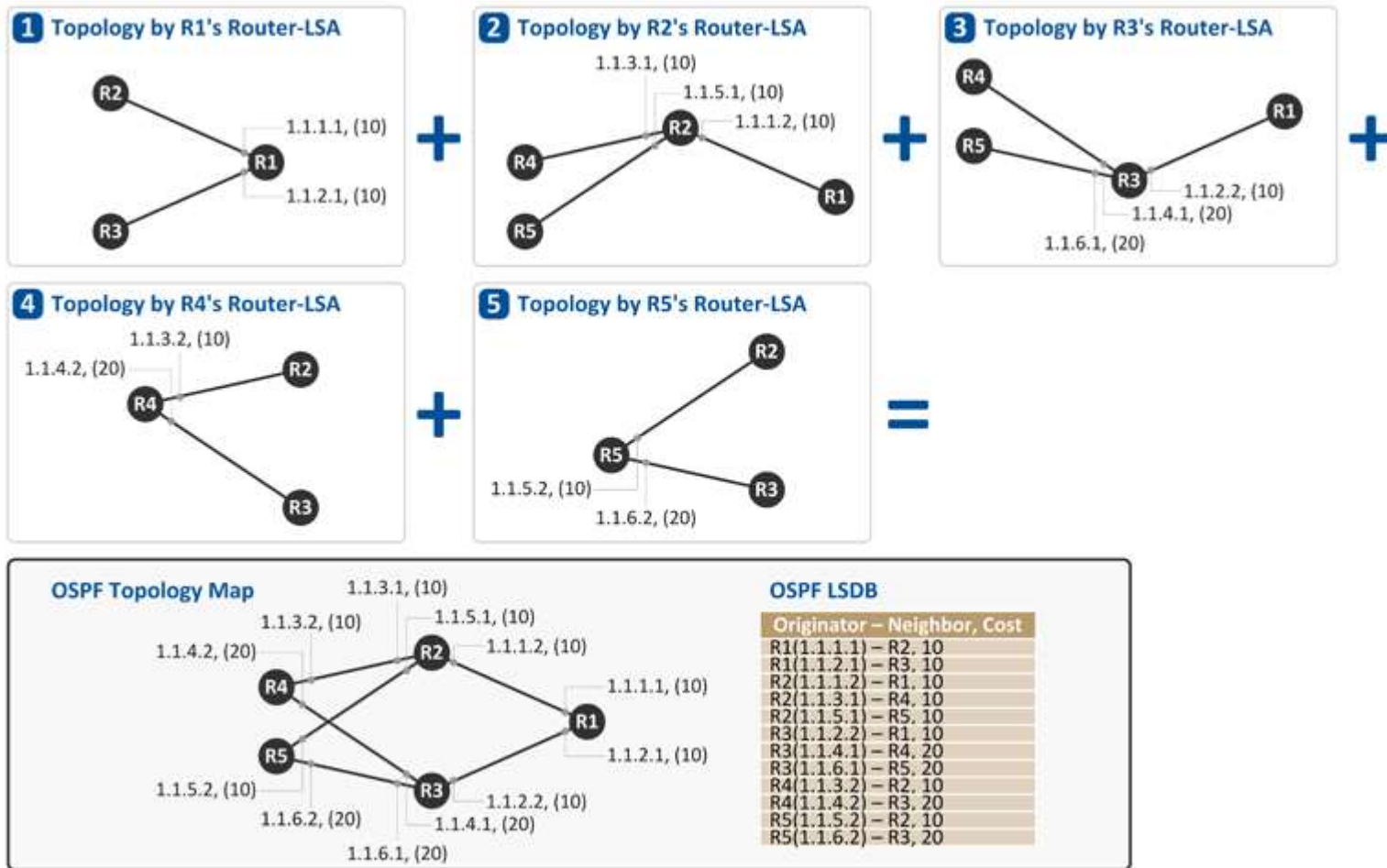


- 이를 수신한 R2, R3는 Router-LSA가 수신된 링크를 제외한 나머지 링크로 Router-LSA를 flooding



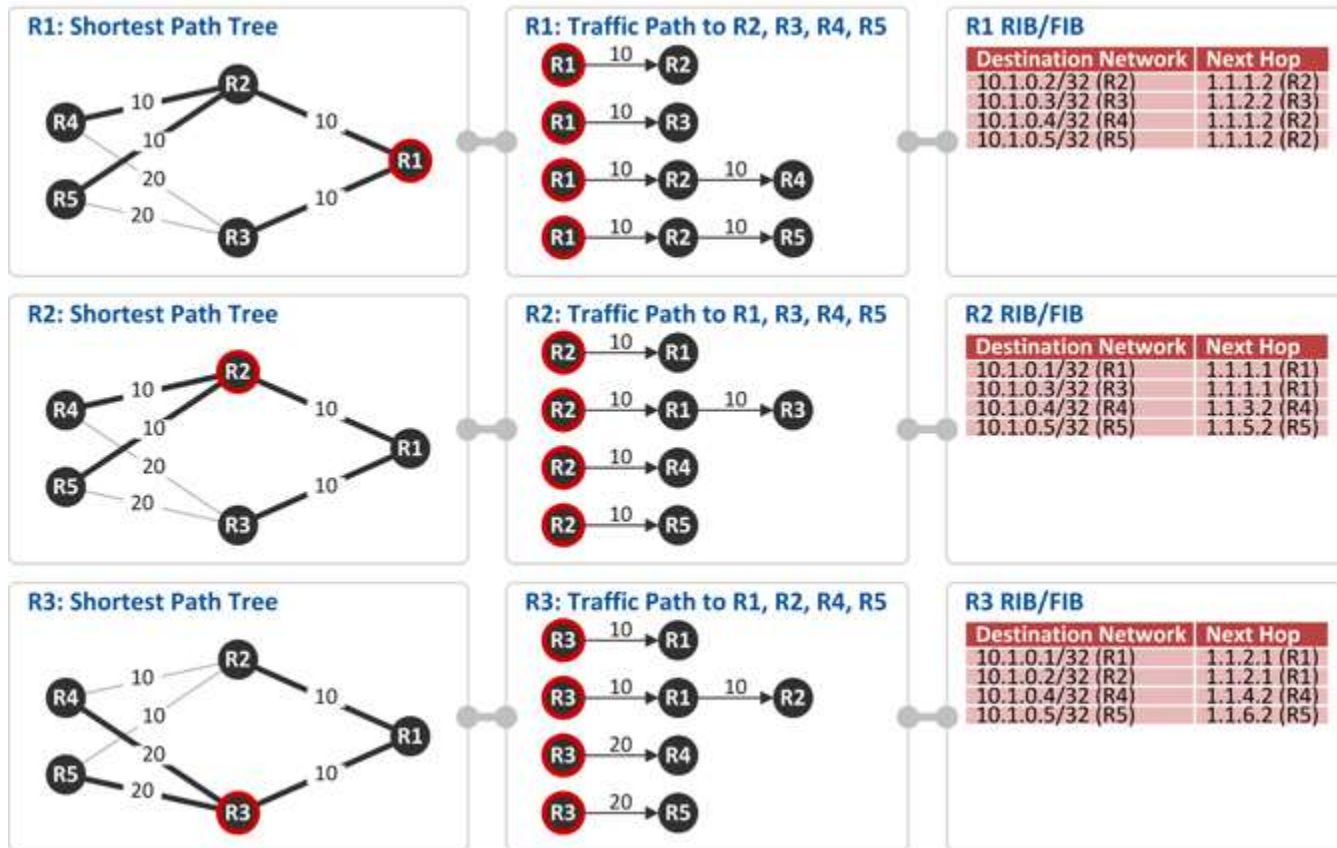
OSPF 토폴로지 (계속)

■ 각 라우터에서 OSPF Topology 그림 그리기



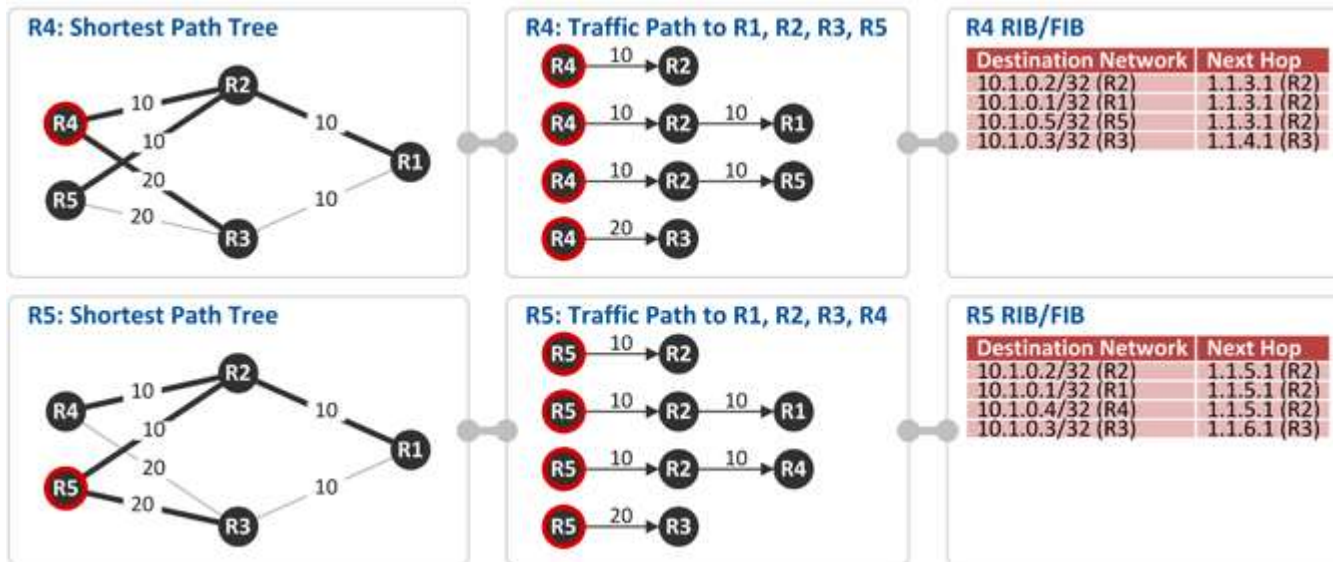
OSPF 토폴로지 (계속)

Shortest Path Tree 구성하기



OSPF 토폴로지 (계속)

■ Shortest Path Tree 구성하기 (계속)



- 그림 출처 : OSPF 쉽게 이해하기 (1편: Shortest Path Tree 토폴로지 생성 절차)

<http://www.netmanias.com/ko/?m=view&id=blog&no=5476>



외부 라우팅 프로토콜

- 외부 라우팅 프로토콜에서 사용하는 경로 벡터(Path Vector)는 경로에 관한 거리 정보 값이 필요 없는 방식
- 내부 라우팅 프로토콜과의 차이
 - 거리(비용)에 대한 처리 과정이 없음
 - 목적지 네트워크에 도착하기 위한 자율시스템에 대한 내용만 포함
- BGP(Border Gateway Protocol)
 - 인터넷에서 많이 사용
 - 서로 다른 종류의 자율시스템 간 정보 교환 가능
 - TCP를 이용하여 정보 교환
 - 메시지 종류
 - Open
 - 연관(Relationship) 생성
 - Update
 - 경로 관련 정보 전달
 - KeepAlive
 - Open에 대한 응답 기능과 주기적인 연관 확인 기능
 - Notification
 - 오류 상태 통보



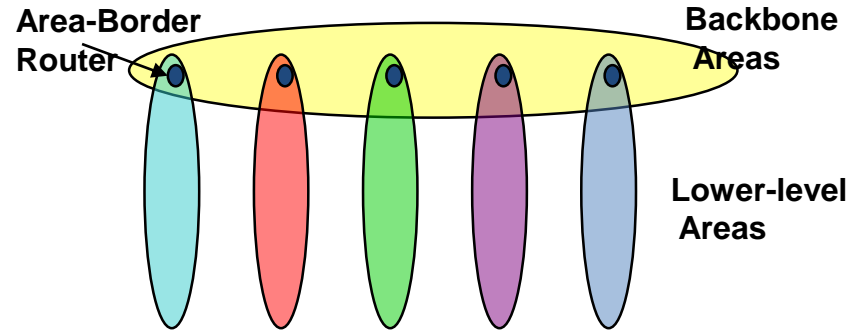
BGP

■ 계층적 라우팅 필요성

- 평면적 라우팅은 확장이 어려움
 - 저장 공간, 합의에 도달하는 시간, 통신
- 목적지에 이르는 거리가 멀수록 더 적은 정보를 이용하는 것이 타당
- 해결 방안 : 영역 계층구조 (Area Hierarchy)

■ 영역(area)

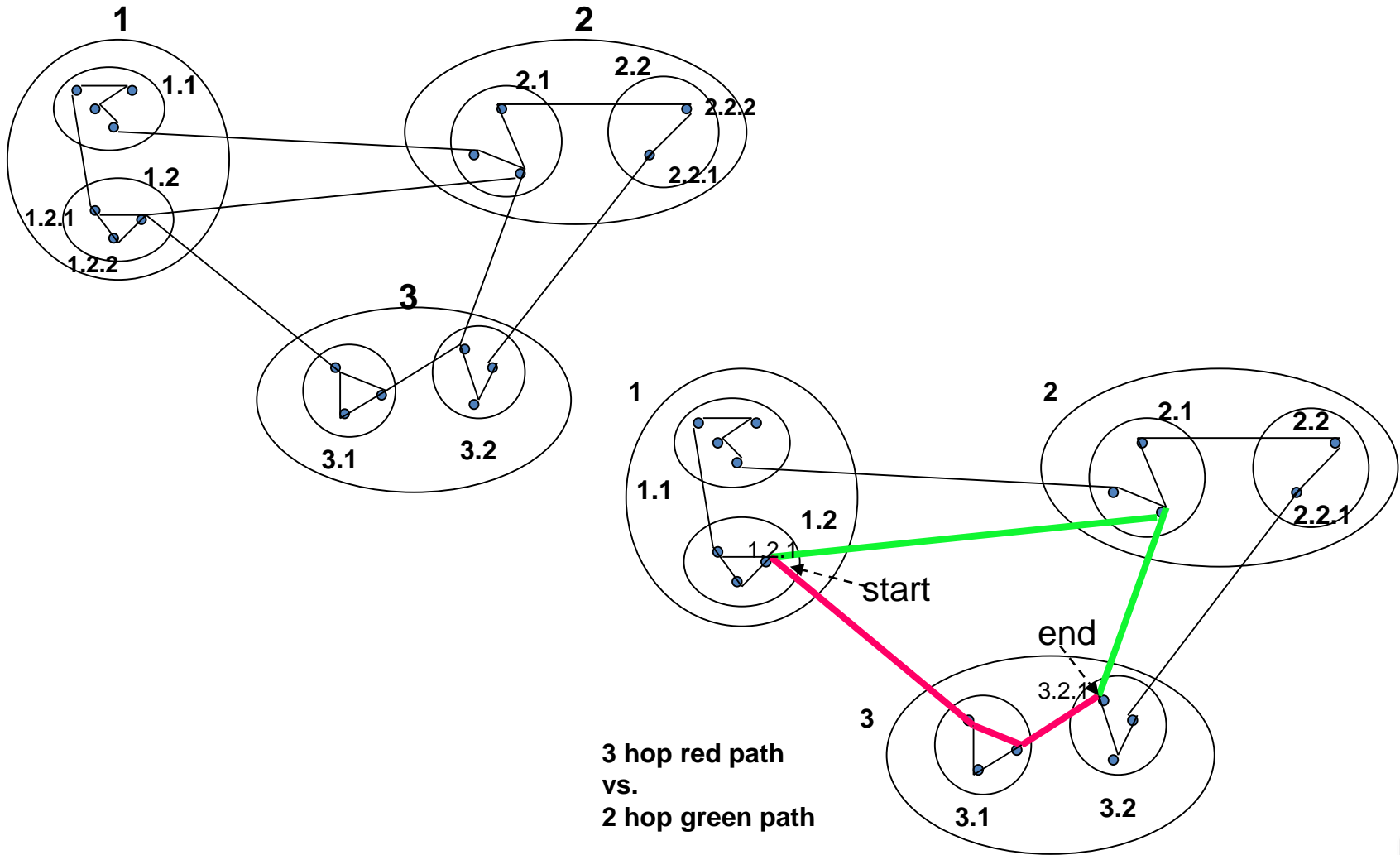
- 망을 영역들로 분할
 - 각 영역에 서브-영역 존재 가능
- 망 노드는 계층적 주소 부여



- 영역 내부
 - 각 노드는 다른 노드로 가는 경로 보유
- 영역 외부
 - 각 노드는 다른 top-level 영역으로 가는 경로만 보유
 - 영역 간 패킷은 적절한 border router에게 전달



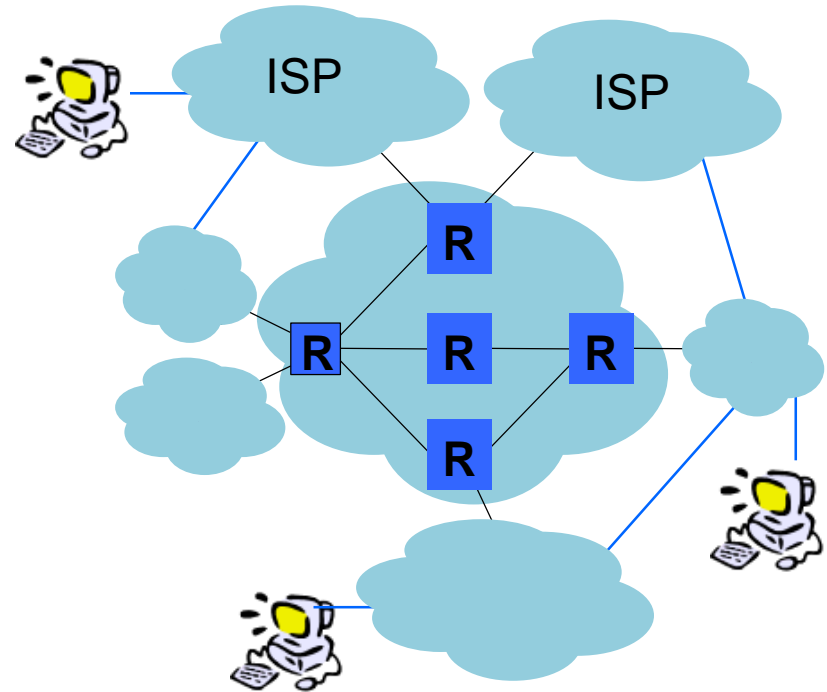
BGP (계속)



BGP (계속)

■ Logical View of the Internet

- After looking a RIP/OSPF descriptions
 - End-hosts connected to routers
 - Routers exchange messages to determine connectivity
- NOT TRUE!



BGP (계속)

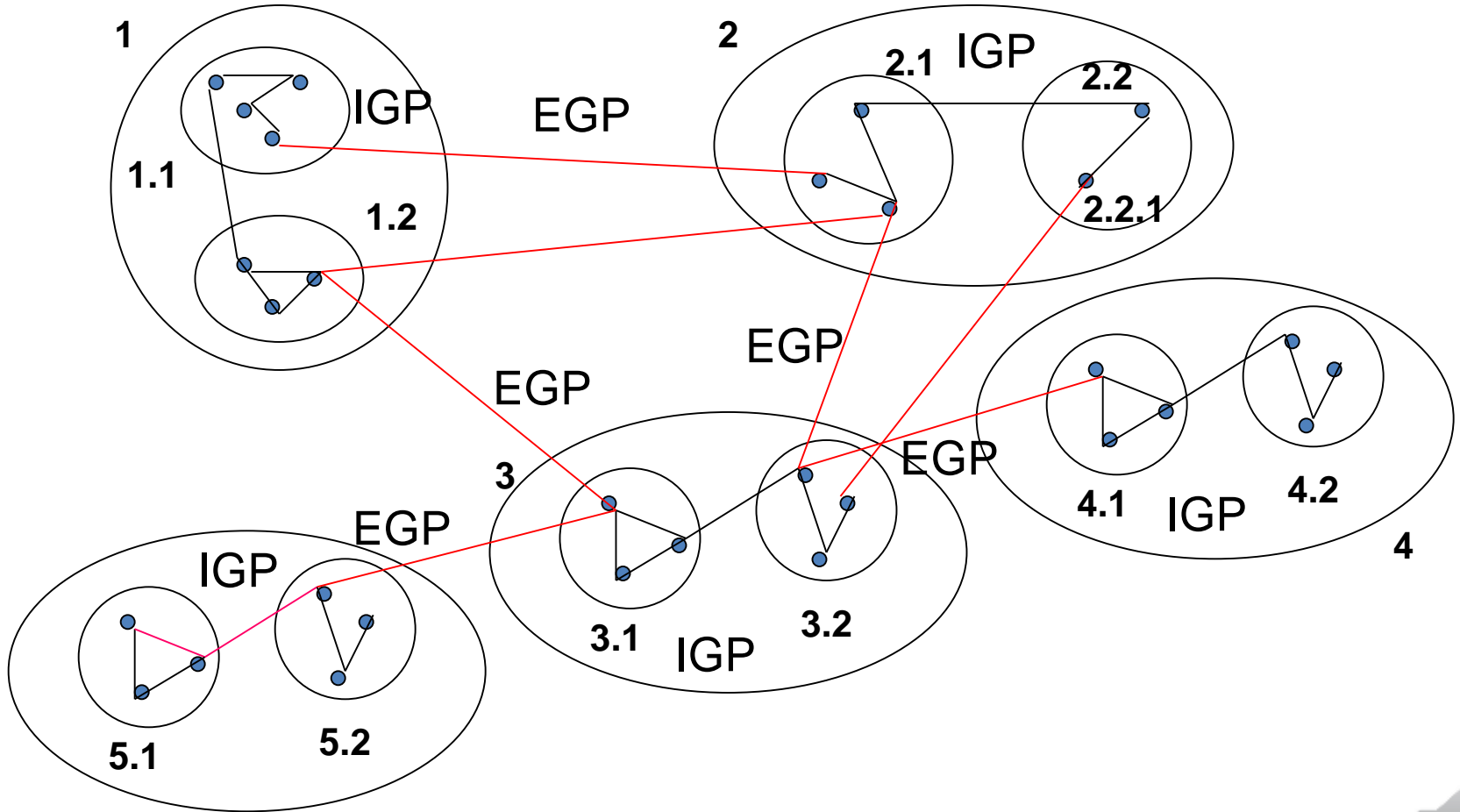
■ 인터넷의 영역 계층구조

- 자율시스템 (AS: Autonomous System)
 - 하나의 기술적 관리 하에 있는 라우터들의 집합
 - AS 안에서는 IGP(interior gateway protocol)와 패킷을 전달하기 위한 공통 메트릭 이용
 - 다른 AS로 전달하기 위해서는 EGP(exterior gateway protocol) 사용
 - 가끔 하나의 AS안에서 여러 IGP를 이용할 수도 있지만, 외부에는 하나의 AS로만 인식됨
- 각 AS는 유일한 ID 할당
 - 16 bit values
 - 64512 through 65535 are "private"
 - 예 :
 - MIT: 3
 - JANET: 786
 - AT&T: 7018, 6341, 5074, ...
 - Sprint: 1239, 1240, 6211, 6242, ...
- AS는 피어(peer) 역할



BGP (계속)

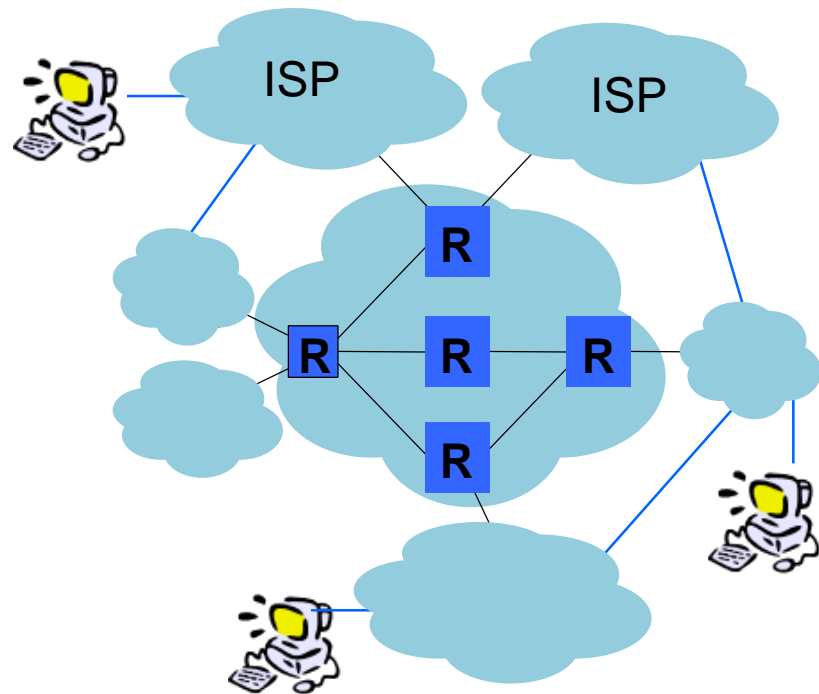
■ 예



BGP (계속)

■ Logical View of the Internet (계속)

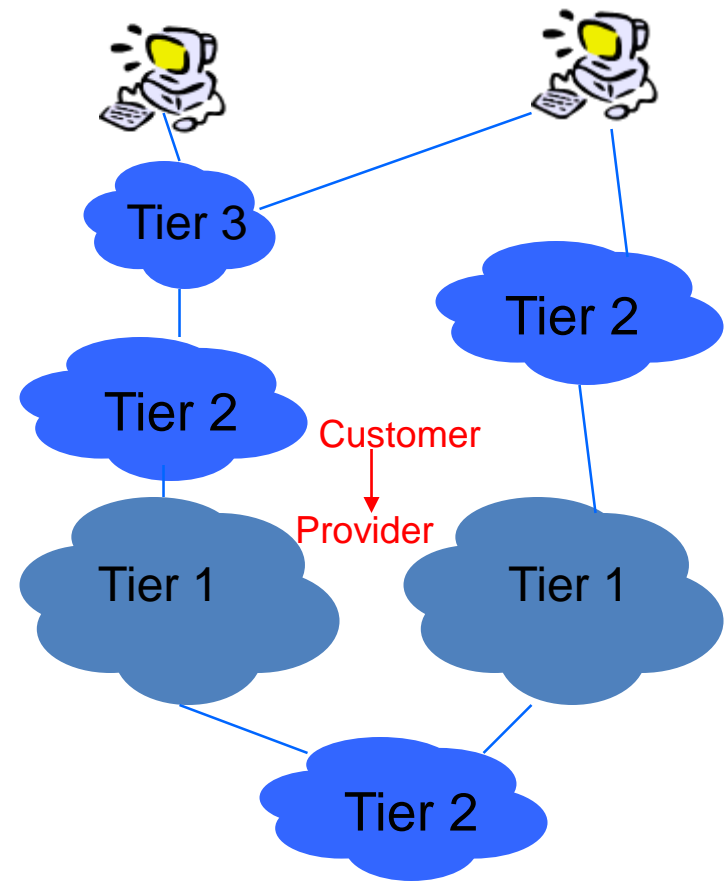
- RIP/OSPF not very scalable → area hierarchies
- NOT TRUE EITHER!
- ISP's aren't equal
 - Size
 - Connectivity



BGP (계속)

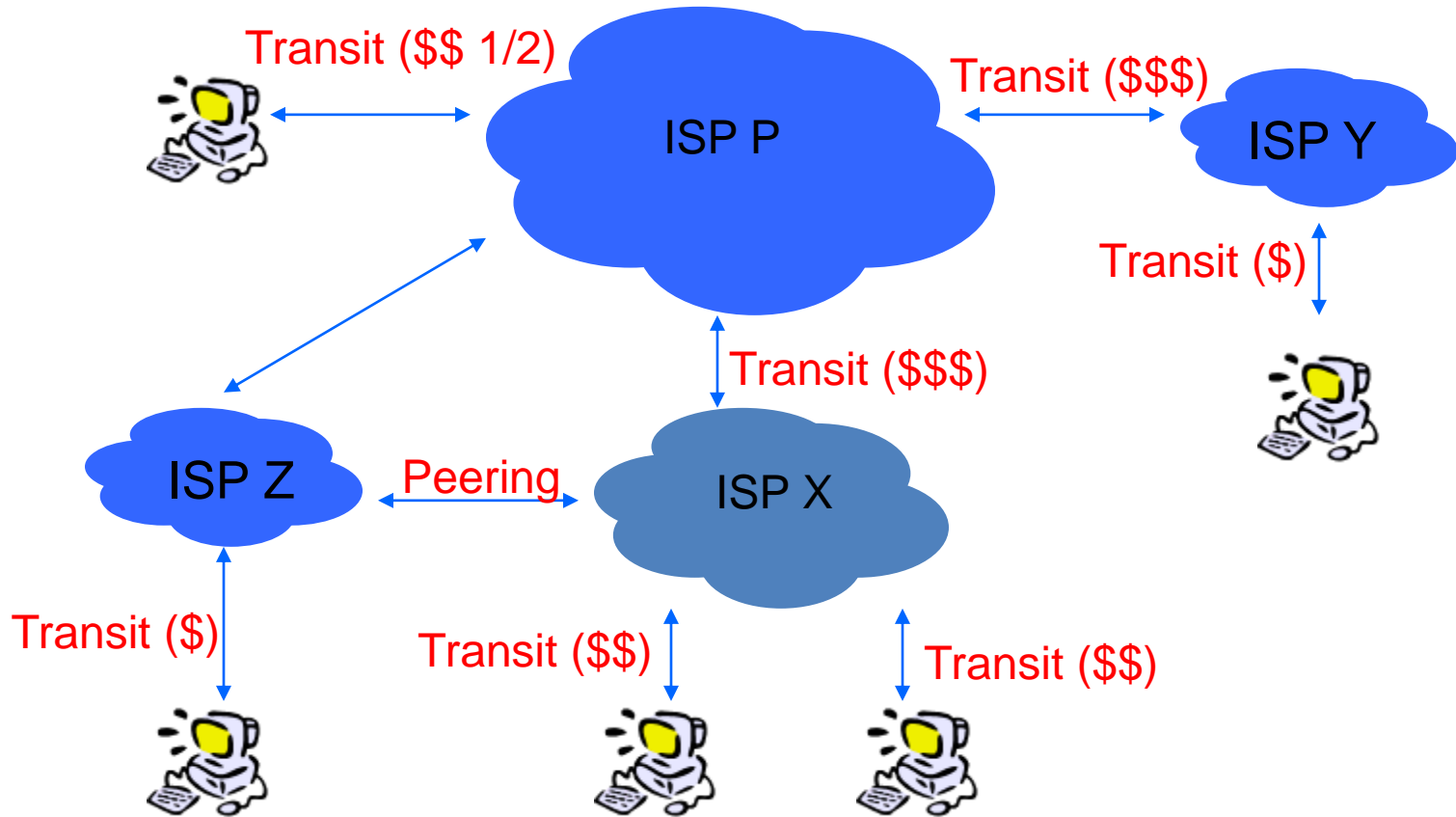
■ Logical View of the Internet (계속)

- Tier 1 ISP
 - "Default-free" with global reachability info
- Tier 2 ISP
 - Regional or country-wide
- Tier 3 ISP
 - Local



BGP (계속)

■ Transit vs. Peering



BGP (계속)

■ Choices

- Link state or distance vector?
 - No universal metric – policy decisions
- Problems with distance-vector:
 - May not converge
- Problems with link state:
 - Metric used by routers not the same – loops
 - LS database too large – entire Internet
 - May expose policies to other AS's

■ Solution: Distance Vector with Path

- Each routing update carries the entire path
- Loops are detected as follows:
 - When AS gets route check if AS already in path
 - If yes, reject route
 - If no, add self and (possibly) advertise route further
- Advantage:
 - Metrics are local - AS chooses path, protocol ensures no loops



BGP (계속)

■ Interconnecting BGP Peers

- BGP uses TCP to connect peers
- Advantages:
 - Simplifies BGP
 - No need for periodic refresh - routes are valid until withdrawn, or the connection is lost
 - Incremental updates
- Disadvantages
 - Congestion control on a routing protocol?
 - Poor interaction during high load

■ Hop-by-hop Model

- BGP advertises to neighbors only those routes that it uses
 - Consistent with the hop-by-hop Internet paradigm
 - e.g., AS1 cannot tell AS2 to route to other AS's in a manner different than what AS2 has chosen (need source routing for that)



학습한 내용

- 라우팅 기능과 관련 프로토콜
 - 간단한 라우팅 프로토콜
 - 거리-벡터 프로토콜
 - RIP(Routing Information Protocol)
 - 링크상태 프로토콜
 - OSPF(Open Shortest path First)
 - 외부 라우팅 프로토콜
 - BGP(Border Gateway Protocol)

