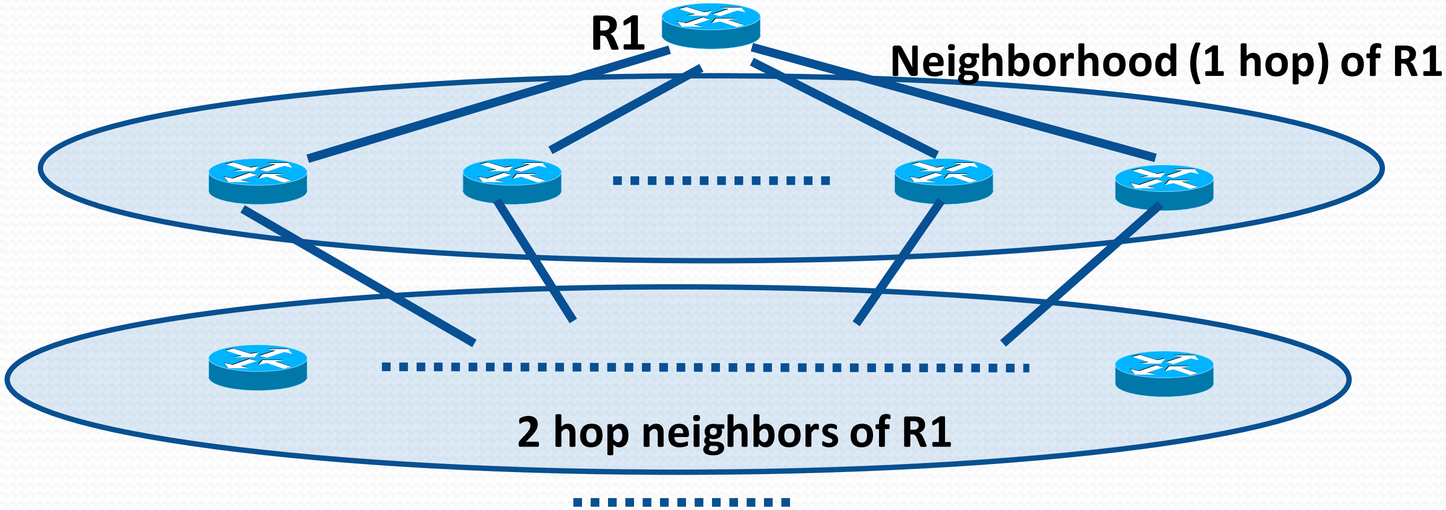
PI-Grau
(Internet Protocols)

José M. Barceló Ordinas
Departamento de Arquitectura de Computadores
(UPC)

- **Topic 3: Intra-domain Routing.**
  - Objectives
    - Introduce basic **intra-domain routing** concepts: shortest path routing, path determination, Patricia tree, convergence, etc
    - Introduce how **Link State protocols** works: flooding techniques, Dijkstra (minimum cost) algorithm, Hello's,
    - Learn the basic components of **OSPF** as intra-domain routing protocol: single area and multi-area OSPF architectures

Routing strategies (assume N nodes):

- **Routing:** Process of selecting **paths** (≡ **routes**) along a network of nodes
- **Shortest Path Routing** (distance-vector or link state)
  - source finds a 1-hop neighbor towards each destination
  - Limitation: scalability. Needs routing table entries of the order of O(N), where N is number of links in the net.
- **Delivery semantic**: defines the set of nodes that will receive the packet. Routing protocols aim to the following delivery semantics:
  - Unicast, multicast, broadcast, anycast, geocast



Topic 3: Intra-domain Routing

- **Current Internet**:
  - N~millions → is not practical either Source Routing or Shortest Path Routing
  - Solution → hierarchical routing.

- **Hierarchical Routing:**
  - Organizations designs networks to provide services
  - **Intra-domain routing**: those routing mechanisms that are internally run by any organization (inside an AS or Corporative Networks)
    - Allow to construct internal paths from any to any internal node
  - **Inter-domain routing**: those routing mechanisms that connects different organizations (AS's)
    - Allow to construct routes from any-to-any organization

Path determination

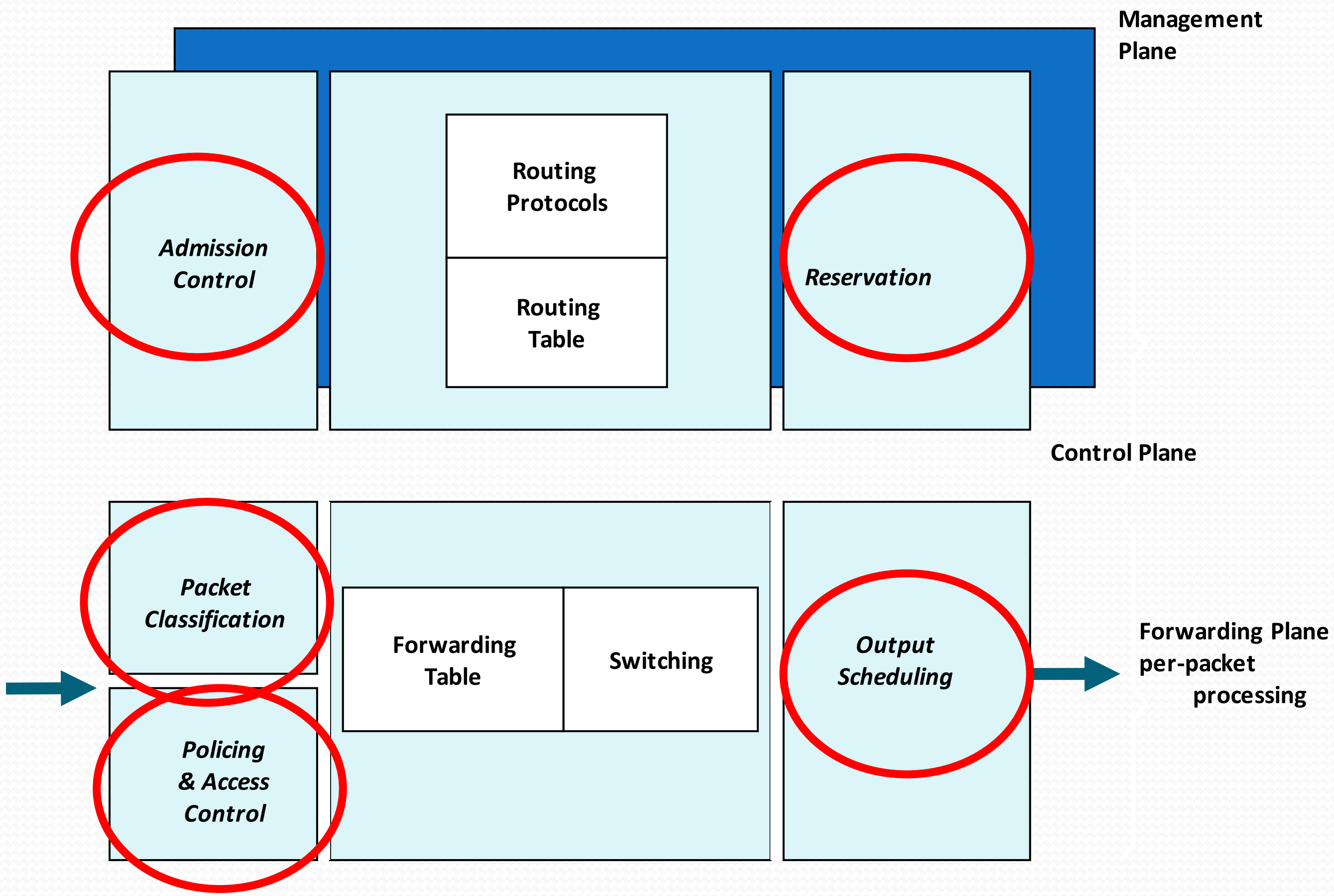
- Process in which a router determines the possible paths (neighbors in case of SPR) in which may forward a datagram towards destination
- The information exchanged by the routers to discover paths is particular to any protocol. **In general a routing protocol is characterized by**:
  - The **format and content** of routing packets exchanged between routers and how is done this exchange (e.g.; unicast, broadcast, multicast, ...)
  - The **periodicity** with which the packets are exchanged
  - **Associated algorithms** that allow to calculate the best path towards destination and thus allows to decide the leaving interface (e.g., minimum cost algorithms and associated metrics)
- The path may be chosen from information introduced by network administrators (static) or from automatic information (metrics) received by the routers (dynamic)
  - The **metrics** can be any of the following: hops, delay, load, bandwidth, link reliability, ...

Convergence of routing protocols

- **convergence time:** when all routers reach a common knowledge of the network**.** Under topological changes, routers should re-calculate routes and update routing tables
- Large convergence times imply that the routers will have more difficulties to sent information in the correct interface and these packets will be dropped
- Convergence depends on:
  - Distance in hops from the failure point
  - Number of routers in the network
  - Bandwidth and traffic load in the network
  - Router load (CPU)
  - The routing protocol chosen (minimum cost algorithm)
  - Network administrator configuration and topology (e.g.; bad designed network, loops, …)
- RIP is in the order of seconds (or even some minute), OSPF is in the order of miliseconds, and BGP can be on orders larger than minutes

- **Router Architecture** -Three planes:
  - **Management plane:** offers an API (Application Programming Interface) that allows to configure any feature offered by the router.
  - **Control plane:** any protocol or algorithm that draws network maps. For example, the routing protocols (RIP, BGPv4, OSPF, …), signaling protocols, admission control functions, etc
  - **Forwarding Plane:** is that part of the router that decides what to do with the incoming data packets. Most of the time, these actions imply packet manipulation. Functions like forwarding, scheduling (FIFO, priorities, WFQ), policing (Leaky bucket), packet classification, etc

- Router Architecture



- **What does a router when receives a datagram?**
  - Performs all the functions related to data forwarding.
  - The main function is extracting the destination @IP and **look up at the routing table** in order to decide the output interface.

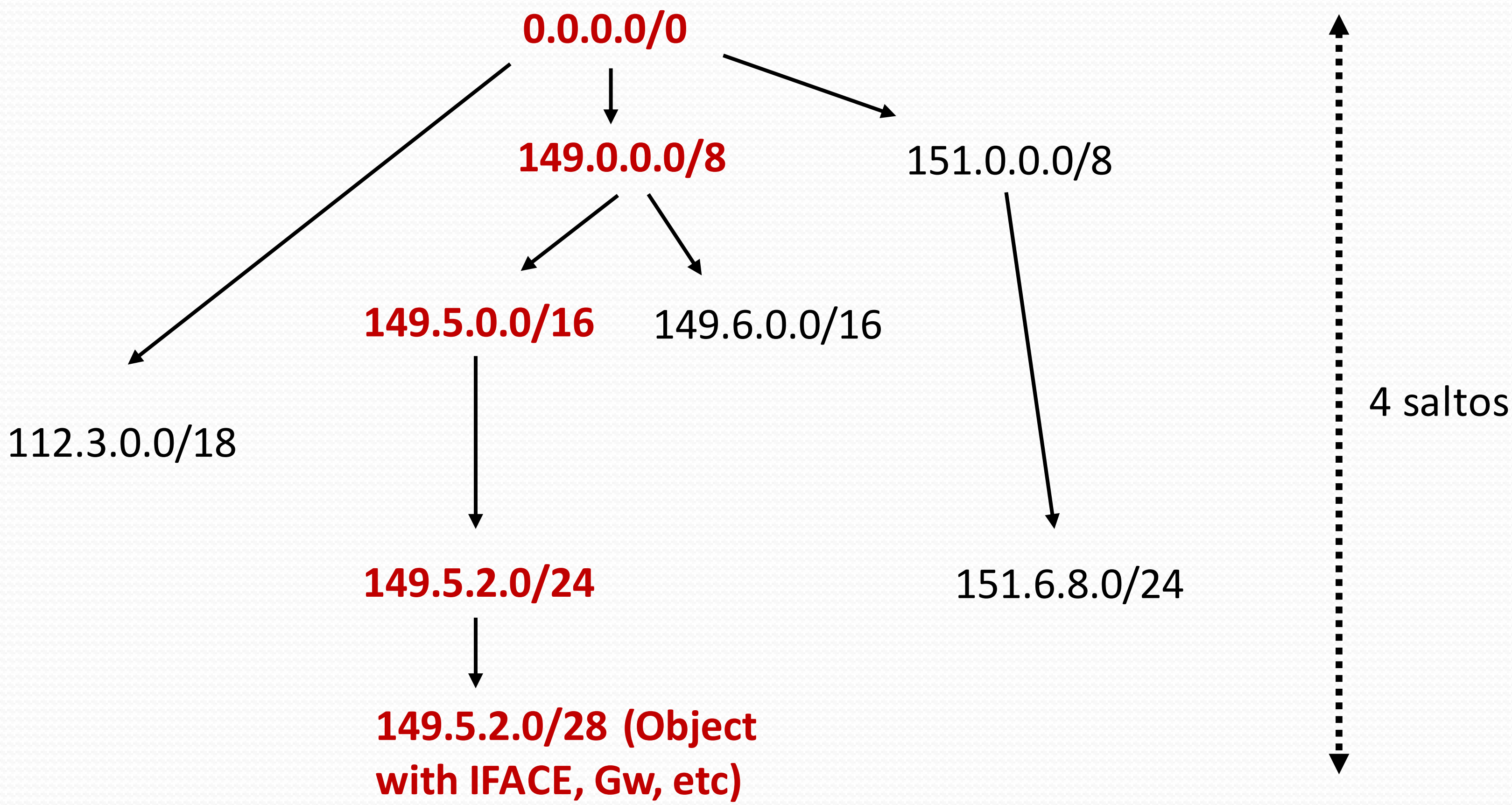
Routing table in a Linux router

@IPdestino/Másk	Gateway	Interficie
19.4.0.0/24	0.0.0.0	fe0
19.5.0.0/16	19.5.2.1	fe0
159.4.3.0/24	0.0.0.0	fe0
145.7.1.0/28	145.7.1.1	fe0

CISCO IOS routing table

C 19.4.0.0/24, is directly connected, fe0
O 19.5.0.0/16 [110/100], via 19.5.2.1, fe0
C 159.4.3.0/24, is directly connected, fe0
O 145.7.1.0/28 [110/100], via 145.7.1.1, fe0

- **What does a router when receives a datagram?**
  - **Patricia tree algorithm**: looks for the Longest Prefix Match in the routing table → that entry that has the longest mask, e.g. @IP_{dst} = **149.5.2.37**



- **Routing Protocols**
  - **Static**
    - Those ones that network administrators manually set the routing entries
    - Useful if the network is small or when the network is a stub network (only reachable from one point)
  - **Dynamic protocols**
    - Those ones that automatically set the routing table
      - Useful in medium to large networks
    - Classified in three groups
      - **Vector-distance protocols:** determine direction and distance towards any subnet in the network (e.g. RIP, IGRP, BGP, ...)
      - **Link-state protocols:** use the network topology (e.g.; OSPF, IS-IS)
      - **Hybrids:** combination of the other two

- **Class-full/Class-less routing:**
  - **Class-full routing**: those protocols that do not advertise masks (e.g.; RIPv1, IGRP)
    - Subneting not allowed
    - Take care of discontinous networks since RIPv1 and IGRP summarize
  - **Class-less routing:** those protocols that advertise masks (e.g.; RIPv2, OSPF, BGP, EIGRP, etc)
    - Subneting is allowed using VLSM (Variable Length Subnet Mask)
    - Take care of discontinous networks since most protocols (RIPv2, OSPF, EIGRP) summarize

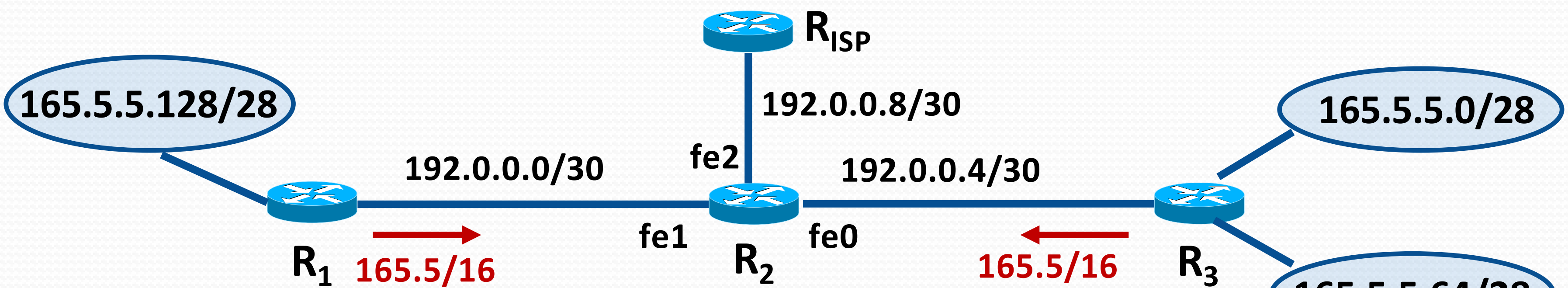
Summarization versus aggregation:

- **Major network:** the class (A,B,C) of an IP network
  - Class A: 0/8 to 127/8
  - Class B: 128.0/16 to 191.255/16
  - Class C: 192.0.0/24 to 223.255.255/24

- **Summarization:** a network is driven towards its major network.
  - E.g. 143.56.78.0/25→ 143.56.0.0/16 (it is written downs as 143.56/16)

- **Aggregation** (suppernetting): two networks are aggregated forming one with Net-ID. Prefixes have to be contiguous.
  - E.g. 143.56.78.0/25 and 143.56.78.128/25 → 143.56.78.0/24

- **Summarization versus aggregation:**
  - **Discontinuous Networks:** major networks that are separated by a router
  - In general, when a router has a discontinuous network, <u>it summarizes to the class (major network) in the other interface</u>, except when the network administrator specifies (via a specific command, e.g., not summarize).



Routing table of R2

C 192.0.0.0/30, directly connected, fe1
C 192.0.0.4/30, directly connected, fe1
O 165.5.0.0/16, via 192.0.0.2, fe1
S 192.0.0.8/30, via 192.0.0.10, fe2

- **Summarization versus aggregation:**

  - **Intra-domain routing** (RIP, OSPF, IS-IS) **summarizes** by default if the networks are separated by different major networks (discontinuous networks) and they do not aggregate networks except that the IETF standard specifies the contrary.

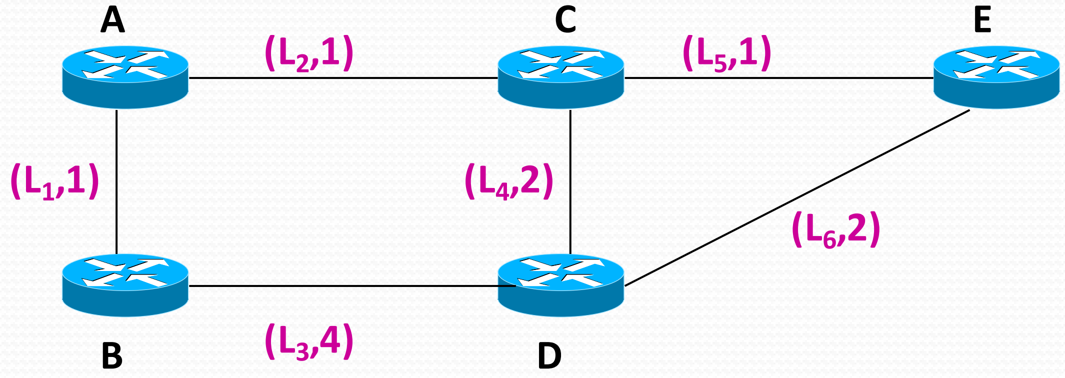
    - Example: **OSPF allows aggregation** (it calls it summarization) in **multi-area networks** but not in single-area,

  - **Inter-domain routing** (BGPv4) allows **aggregation** and then also **summarization** as a particular case of aggregation.

- **Link State Routing Protocols:**
  - Those routing protocols that **react to changes in the link** (**up/down**) sending connectivity information in contrast to vector-distance that sends the current distance to that node (i.e., routing table information).
  - Link Connectivity has associated a name and a cost.
  - A link state protocol is thus characterized, in general, by:
    - **Discovering of neighbors** (e.g., using a **protocol called HELLO**)
    - Every node learns the topology of the network (Link State Database) **flooding** **Link State Packets (LSP)**
    - A **minimum cost algorithm** (**e.g., Dijkstra**) that calculates the best next hop (routing table) using the data base

Link state general issues:

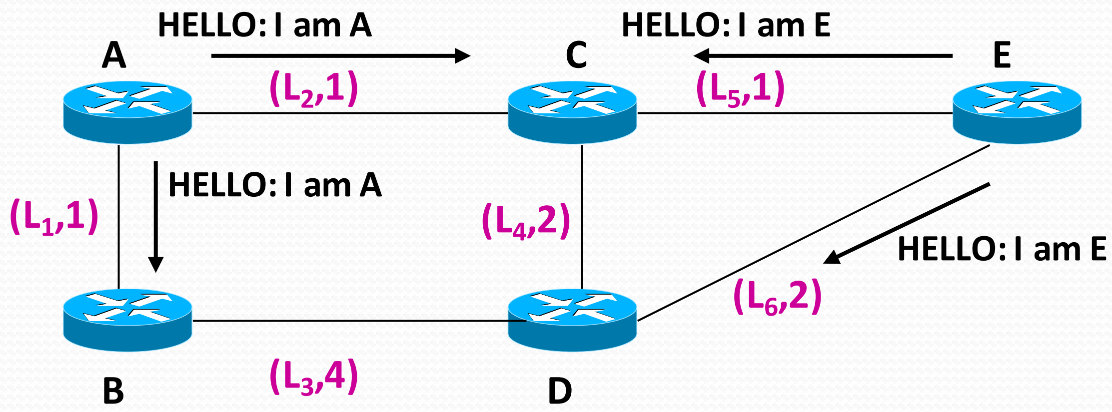
(Link, Cost)



- Do not compute routes in a distributed manner.
- Create a common network topological data base
- Locally each router computes its routing table

Link state general issues:

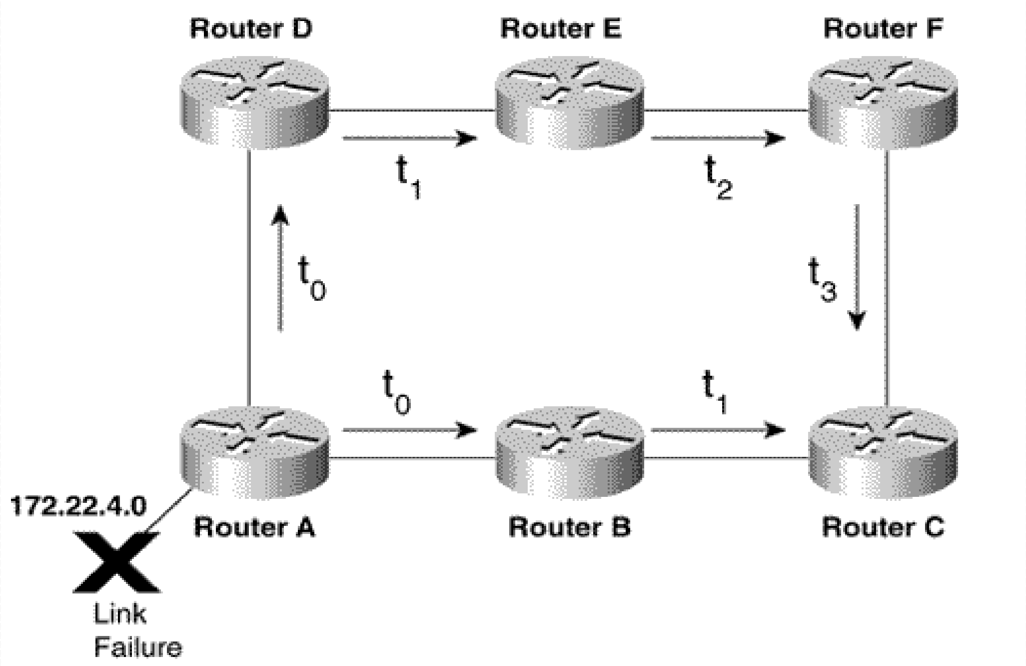
- **Discovering neighbours**: we have to identify each router uniquely from other rotuers, then, it uses typically an active IP@ of a router to identify it.



1. Each router has to learn who are they neighbors → use a **HELLO protocol**

- **Flooding Link State information:**
  - **Flooding:** propagate a Link State Packet (LSPs) to all routers of the network,
  - Every router k sends Link State Packets (LSPs) to all of its neighbors N(k) (neighborhood set of router k)
  - LSPs arrive and wait in buffers to be "accepted" and forwarded upon conditions:
    1. If node j receives a LSP from node k it compares the **sequence numbers**. Only the most recent LSP from k, can be forwarded to its neighborhood N(j)-{k}.
    2. Furthermore, each LSP travels with an **aging field** that adds reliability → each router that floods a message adds a value to the age of the LSP (1 second). Ages are included in the database.
       - the protocol defines an MaxAgeDiff. If a router receives an LSP with different ages and same sequence number and the difference between ages is lower than MaxAgeDiff then the LSP is not flooded, otherwise, the newest LSP is recorded and flooded.

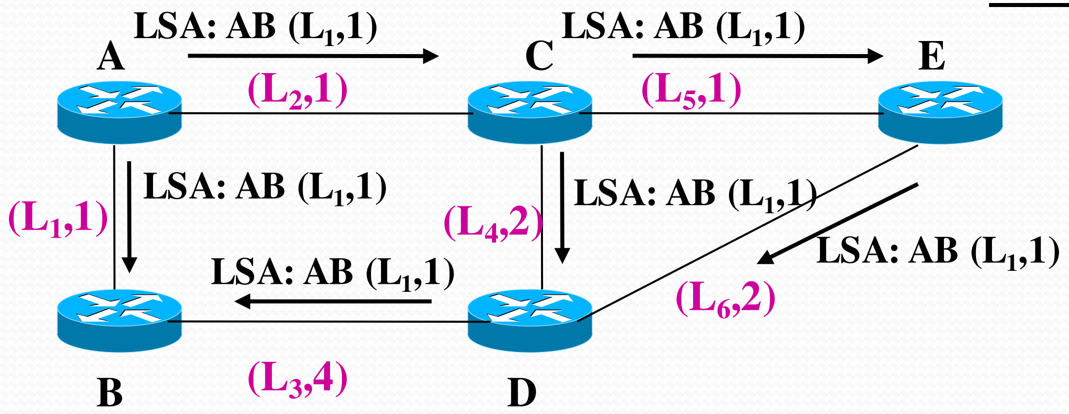
- **Flooding Link State example:**



1. Route 172.22.4.0 fails
2. R_A floods an LSP with SeqNum=x
3. The LSP arrives to R_C from R_B at t_1
4. A delayed LSP arrives to R_C from R_F at t_3
5. R_C uses the SeqNuym=x to learn that both LSP are the same and since t_3-t_1<MaxAgeDiff, the first LSP is keept in the database with its age and the new LSP is not flooded.

Link state general issues:

- Drawing the Topological Network Database



Database (DB) at each router

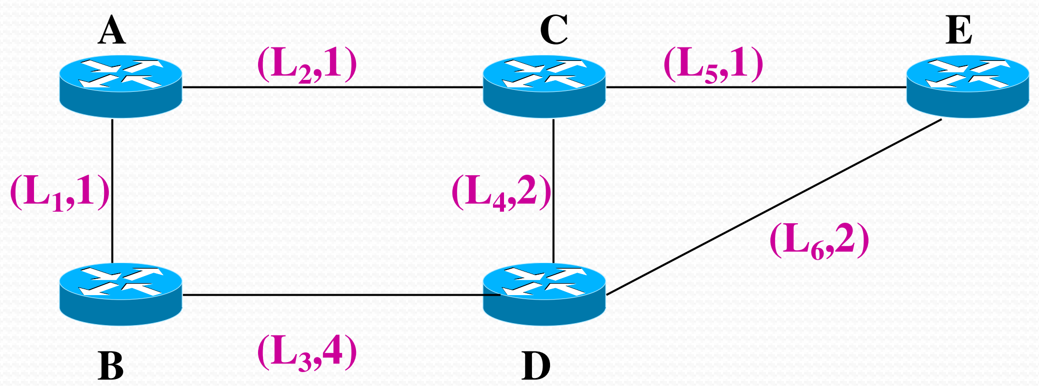
From	To	Link	Cost	Age
A	B	L_1	1	1
A	C	L_2	1	1
B	A	L_1	1	1
B	D	L_3	4	1
C	A	L_2	1	1
C	D	L_4	2	1
C	E	L_5	1	1
D	B	L_3	4	1
D	C	L_4	2	1
D	E	L_6	2	1
E	C	L_5	1	1
E	D	L_6	2	1

2. Use a **flooding protocol** to send LSA (Link State Advertisements) in order all router gets the same topological data base

Link state general issues:

- Use **Dijkstra** at each router to get the routing table

3. Use a **minimum cost algorithm (e.g. Dijkstra)** to calculate the routing table. Each router has a different routing table although all share the same Data Base entries



Routing table at router A

To	Link	Cost
B	L_1	1
C	L_2	1
D	L_2	3
E	L_2	2

Routing table at router B

To	Link	Cost
A	L_1	1
C	L_1	2
D	L_3	4
E	L_1	3

Routing table at router D

To	Link	Cost
A	L_4	3
B	L_3	4
C	L_4	2
E	L_6	2

Dijkstra algorithm:

- **Complexity:** originally $O(|V|^2)$, actually **$O(|E|+|V| \log |V|)$**
- C=node identifiers=$\{1,2,\ldots,N\}$
- $F(j)$ shortest path from node 1 to j
- $D(j,i)$ cost from node i to node j
- $A(j)=\{i \in C: D(j,i)<\infty\}$ set of successor nodes from j

---

Initialization:

$j=1$; $F(1)=0$; $F(i)=\infty$; $i \in \{2,\ldots,n\}$; $U=C$

Iteration:

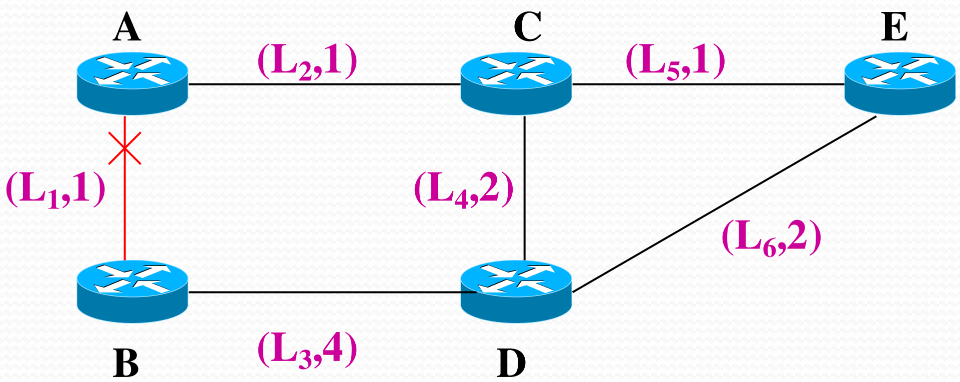
While($j \neq n$ and $F(j) < \infty$) do:

Update $U=U-\{j\}$

Update F: $F(i)=\min\{F(i), F(j)+D(j,i)\}$, $i \in A(j) \cap U$

Update j: $j=\text{argmin}\{F(i); i \in U\}$

- **Link state general issues:**
  - **Maintain** the routing information



Routing table at router A

To	Link	Cost
B	L_2	7
C	L_2	1
D	L_2	3
E	L_2	2

If **a link state change is detected**, the topological data base has to be updated. The router/s that detect the link state change has to inform to the rest of routers. For that the routers send LSA's (Link State Advertisement) using **Flooding protocols** to disseminate information.

A and B detect a link state change, modify their DBs and transmit the change to update the neighbor DB → **From A to B, link 1, cost = ∞**

All routers re-compute and refresh their routing table.

OSPF (Open Shortest Path First):

- Link State protocol
  - State is **up or down**

- Routing messages are **encapsulated in IPv4 packets** and identified from transport protocols using number 89 (remember that TCP=6, UDP=17)

- OSPF may be used in:
  - **BMA** (Broadcast Multi-Access) topologies: e.g., on Ethernet switches,
  - **Point-to-point** topologies: e.g., dedicated lines (e.g.; E1),
  - **NBMA** (Non-Broadcast Multi-Access) topologies: ATM or Frame Relay

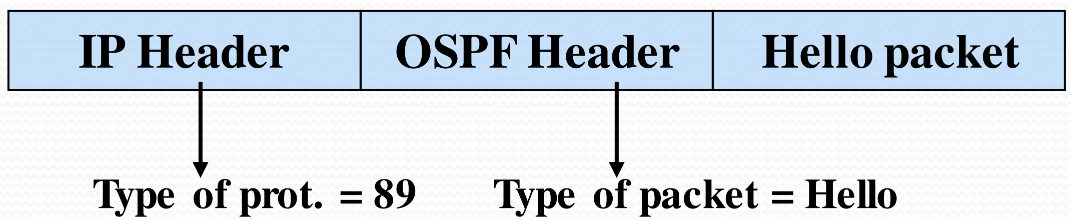
OSPF general issues:

- In general, <u>LS protocols have to</u>:
  - **Discover neighbors**, e.g. using a HELLO protocol
  - Each router **keep a map** (i.e., a DataBase) with the whole network. Each router sends information **<u>to ALL network routers</u>** when a link state change is detected
  - From this information each router re-calculates the routing table using **a minimum cost algorithm.**
- Thus, <u>OPSF is characterized by</u>:
  - **Discovering neighbors** (HELLO protocol), defining a Router-ID's, selecting optimal routers DR/BDR (Designated Routers/Back-up DR) for flooding,
  - **Sending LSAs** (Link State Advertisements) with all the changes detected using the DR/BDR routers as optimal **flooding mechanism**,
  - **Maintain a data base (DB)** with the network topology (Link State Database) at each router, **this DB has 5 levels** with different kind of information,
  - A **minimum cost algorithm** (Dijkstra) that calculates the best next hop (shortest-path routing) using the information of the data base.

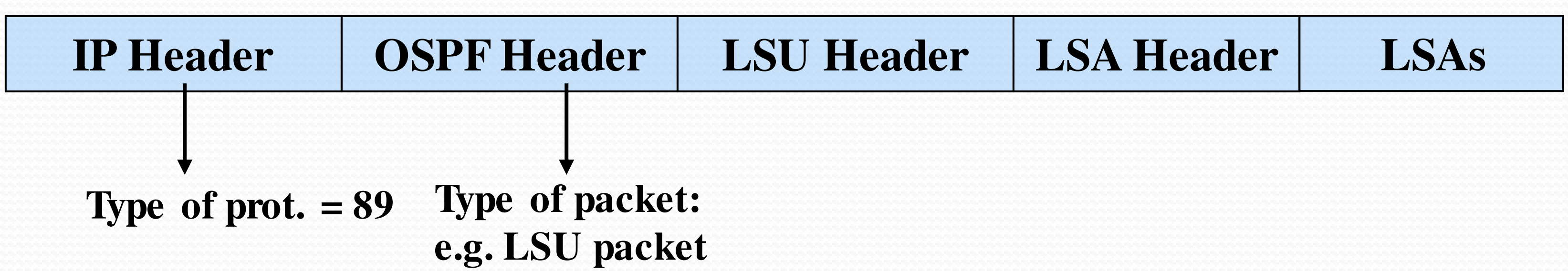
OSPF packet format

- Use IP encapsulation with transport protocol 89
- Different OSPF packets (**HELLO, UPDATE, REQUEST, DataBase Description and acknoledgements**) defined inside the OSPF header and identified in the “type of packet” field.
- Then further encapsulation for the different OSPF packets.

HELLO packets



DBD, LSU, LSR and LSack packets

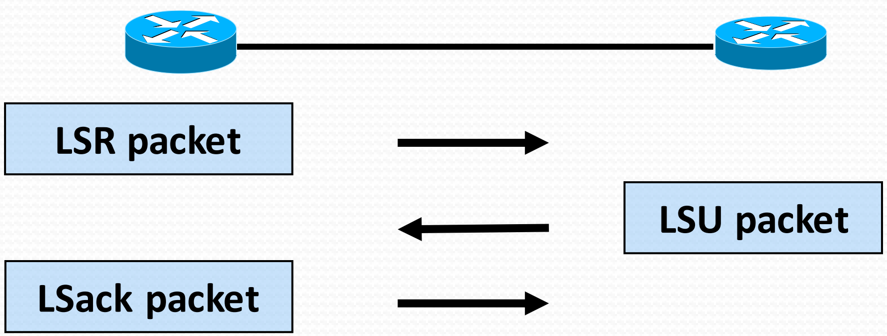


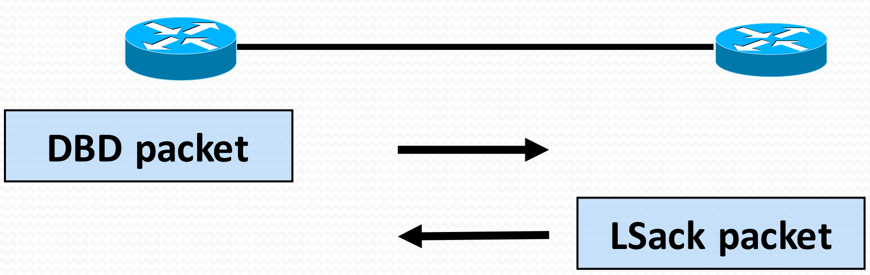
- # OSPF Header:

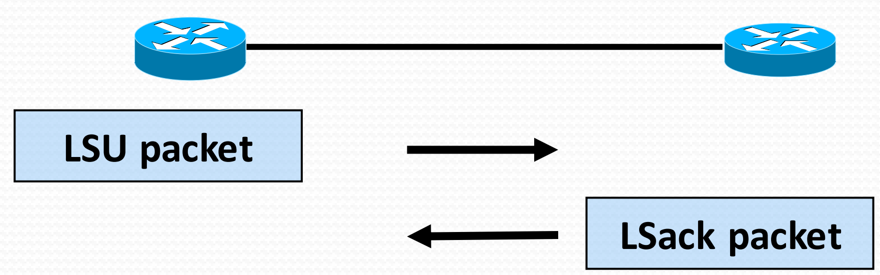
```
          1                   2                   3                   4
 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Version      |     Type      |          Packet Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Router ID                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Area ID                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Checksum            |  Authentication Type          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Authentication information                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Data (variable)                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- **OSPF packet types (TYPE field in OSPF header)**
  - **Types of packets**
    - **HELLO packets (Type=1):** establish and maintain neighbor relationships
    - **Database Description (DBD) packets (Type=2):** describe DB (DataBase) contents
    - **Link-State Request (LSR) packets (Type=3):** request portions of the DB
    - **Link-State Update (LSU) packets (Type=4):** answer with DB portions
    - **Link-State ACK (LSAck) packets (Type=5):** acknowldege reception of LSR, LSU or DBD OSPF packets
  - **LSAs (Link-State Advertisements):** Data unit that describes the network or router local state.
    - LSA are **encapsulated in** DBD, LSU, LSR o LSAck packets, **never in** Hello packets,
    - There are 11 types of LSA's, but only **LSA's type 1 to 5 and type 7** are commonly found. The other ones, type 6 and from 8 to 11 are rarely found in OSPF,
    - The content of each LSA depends on the type and some will be studied at the end of the chapter when we see multi-area networks.

- **OSPF Header:**
  - Several combinations of packets. Some examples:

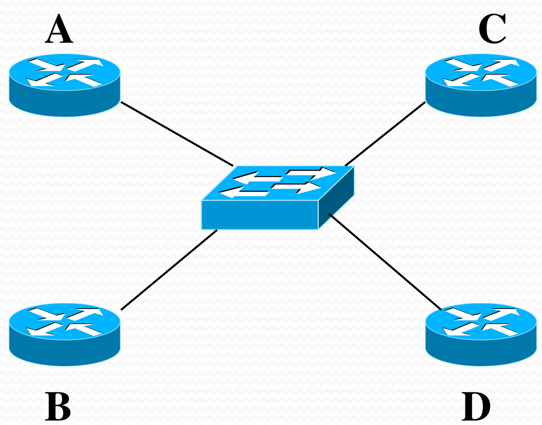






- **OSPF Header:**
  - **Packet Length in Bytes**: including OSPF header, gives the number of bytes of the rest of the OSPF packet, since the size is variable depending on the “Type of packet”
  - **Router ID (RID)**: identifies the OSPF packet owner **(each router chooses RID as the largest @IP among the active router @IP and some choose the loopback if active)**
  - **Area ID**: identifies the routing OSPF area **(default: area = 0)**, larger in multi-area OSPF networks
  - **Checksum** as in IPv4 header
  - **Authentication type**:
    - Type 0: no authentication
    - Type 1: clear-text password or simple authentication
    - Type 2: cryptographic or MD5 authentication
  - **Authentication information**: contains the authentication info
  - **Data**: encapsulates routing information, depends on the “Type of packet”

- **DR (Designated Router) and BDR (Backup-DR):**
  - The **main objective of the DR** is to minimize the amount of flooding (forwarding) and the DBs synchronization mechanism centralizing the exchange of information
    - Routers do not exchange link state information among them since the amount of packets would be very high → exchange link state packets only with DR and BDR
    - DR maintains synchronized the DBs (DataBases) of all routers in the BMA that is responsible
    - **BDR** do nothing while there is an active DR, only acts if the DR fails
    - **Adjacency**: relationship established among a router and its DR and BDR



In a LAN with N links there are N*(N-1)/2 adjacencies. Each router has to announce these adjacencies. In order to optimize the number of adjacencies announced to the N-1 routers a **designated router (DR)** is selected to receive information from its neighbors. DR will have to announce the DB so all routers are synchronized.

The **HELLO protocol** is used to elect the DR and BDR router

- **HELLO packets:**
  - Each router tests that the line (state) with a neighbor is operative and thus may interchange packets (link state up or down),
  - Each router periodically sends (**HelloInterval=10 seconds**) using the ***multicast address 224.0.0.5* (All-OSPF-routers)**,
  - If a HELLO packet is not received in **DeadInterval = 4*10 = 40 seconds**, then the neighbour is considered lost (link state down), elsewhere the link state is considered accessible (link state up),
  - **It also allows initiates the selection of** a **DR** (Designated Router) and a **BDR** (Backup Designated Router) at each BMA (Broadcast Multi-Access network) or at each point-to-point network.

- **HELLO Header:**

```
        1                   2                   3                   4
  0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |          OSPF packet header with Type= 1 (HELLO packet)       |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                        Network Mask                           |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |         HelloInterval         |    Options    |    Rtr Pri    |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                     RouterDeadInterval                        |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                      Designated Router                        |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                   Backup Designated Router                    |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                          Neighbor                             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                              ...                              |
```

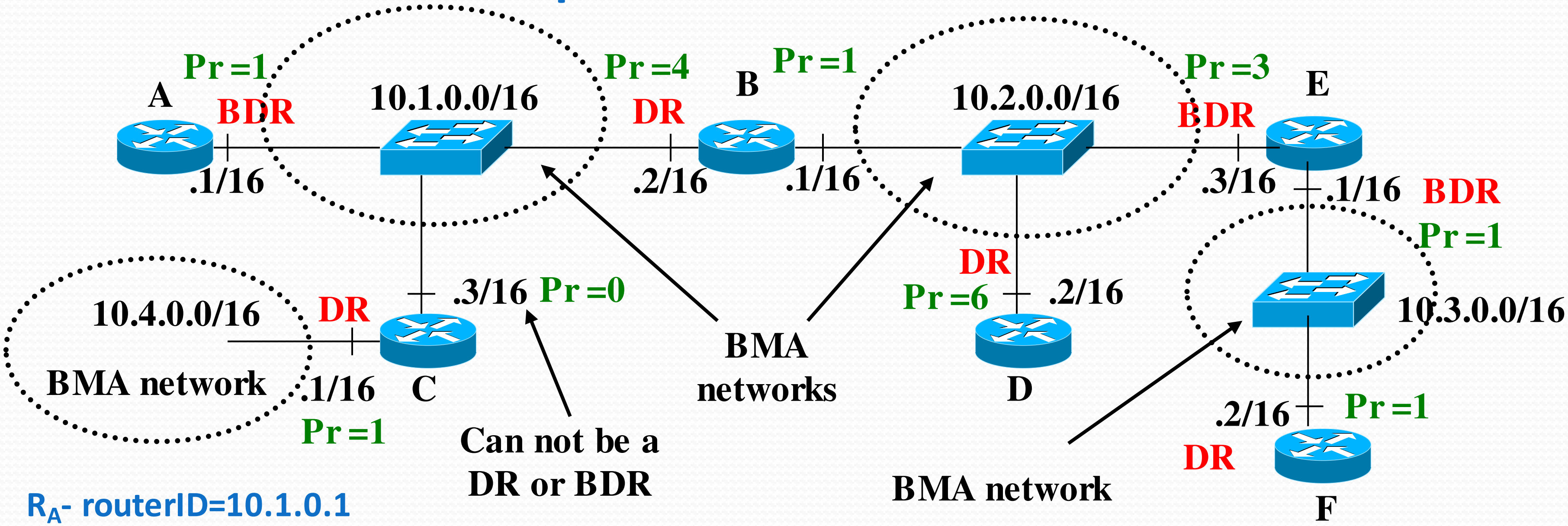
HELLO Header:

- **Network mask**: mask associated with that interface
- **Hello Interval**: interval in which se HELLO's packets are sent (10 seconds)
- **Options**: documented in the OSPF RFC
- **Router Priority**: priority (default =1)
- **Router-Dead-Interval**: time that a router waits to consider that a neighbor is or it is not active (4*HelloInterval 40 seconds)
- **DR y BDR**: both IP addresses (0.0.0.0 if unknown)
- **Neighbors**: list of neighbor's Router-ID (RID) listened in the last Router-Dead-Interval seconds

DR and BDR election process:

- **Each BMA network segment has a DR and a BDR**, thus, a router connected to multiple networks may act as DR in a BMA segment and act as a normal router in other segment. Thus, **DR and BDR are set at INTERFACE level**
- The router with highest priority level is chosen as DR while the second is chosen as BDR,
- A router that is not a DR or BDR is called a **DROTHER**,
- **Default router priority = 1**, highest RID is used to break ties (where the router ID use to be the highest active @IP in the router)
- Routers with **priority=0** cannot be elected as DRs or BDRs

- DR and BDR election process:



R_A- routerID=10.1.0.1

R_B- routerID=10.2.0.1

R_C- routerID=10.4.0.1

R_D- routerID=10.2.0.2

R_E- routerID=10.3.0.1

R_F- routerID=10.3.0.2

<u>For each router of the OSPF network:</u>

1. Obtain the Router-ID (RID) for each router

<u>For each BMA network:</u>

1. Check who has the highest priority (DR) and second highest priority (BDR)
2. In case of priority tie, choose the one with highest RID

DR or BDR is lost:

- If a router with higher priority than a DR or BDR joins the network, **the DR and BDR does not change**. These only change when the DR fails → the BDR becomes DR, and a new BDR is elected
- Other protocols such as IS-IS do not keep this rule and higher priorities change automatically DR/BDR's),
- A BDR detects that a DR fails by HELLO DeadInterval's
- Example: R1, R2 and R3 with priorities (1, 2, 3)
  - <u>If R1, R2 and R3 configure at the same time</u>, R3 (DR), R2 (BDR), R1 (DROTHER)
  - <u>If R1 is configured before R2 or R3</u> –> R1 will be DR, although with lower priority
  - <u>Assuming R3 (DR), R2 (BDR), R1 (DROTHER)</u>: if R3 fails –> R2 (DR) and R1 (BDR). Now, if R3 recovers and R3 (DR), R1 (BDR) and R2 (DROTHER). That means that OSPF has DR memory but not BDR memory
  - <u>Assuming R3 (DR), R2 (BDR), R1 (DROTHER)</u>: if R1 changes its priority to 4 (higher than R3 and R2), nothing changes → R3 (DR), R2 (BDR), R1 (DROTHER). The reason is that R3 already was a DR and R2 a BDR.

Building the routing table:

- Once the DR and BDR are elected, routers have to learn network routes using the flooding mechanism. OSPF defines an **Exchange Protocol** for populating the DataBase. The Exchange Protocol consists in:

1. DR and DBR form an adjacency with each router of the BMA network **(Exstart State).** Assume N routers in the BMA network:

   At each adjacency, one of the routers acts as "master" (usually the DR) and the other as "slave". DR and BDR have each N-1 adjacencies, a DROTHER has 2 adjacencies (one with the DR and another with the BDR),

2. The master and slave routers send DBD summary packets (only LSA headers) **(Exchange State)**. <u>DBD packets contain the list of LSA</u>. This list includes link state type, cost of link, RID of advertising router and sequence number of link. DBD packets do not include full LSAs.

   If the DR or the DROTHER discover some link (route) in the Database that is missing they exchange LSR/LSU/LSack to complete the list of LSA's that each know, until this list converges,

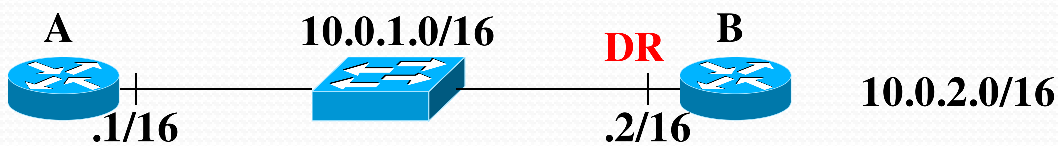
Building the routing table:

3. Upon receiving DBDs, DROTHER's will compare it with their own DataBase. Thus they will learn what they need to order. The routers (DROTHERs) look at their database and request to the master (DR) for those routes to which the DROTHERs haven't enough information **(Loading State).**

   Routers will use LSU (Link state update) to exchange the LSAs. Each LSA contains routing information about a particular link. Routers also maintain a retransmission list to make sure that every sent LSA is acknowledged.

4. Each router builds its own routing table **(Full State)** using Dijkstra algorithm

- **Building the routing table:**



Exstart state

Hello: I will start exchange since I have routerID=10.0.1.1

No I will start exchange since I have a higher routerID=10.0.1.2 :Hello

Exchange state

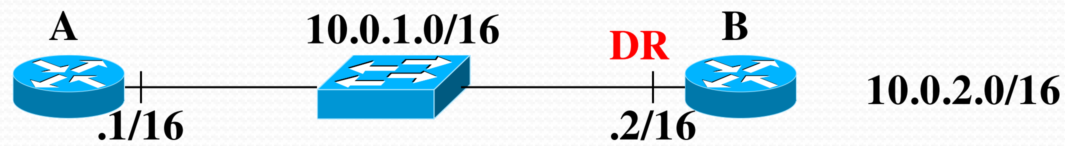
Here is a summary of my link-state DataBase :DBD

DBD: Here is a summary of my link-state DataBase

LSAck:

LSAck: Thanks for the info

- **Building the routing table:**



Loading state

LSR: I need info about net 10.0.2.0/16

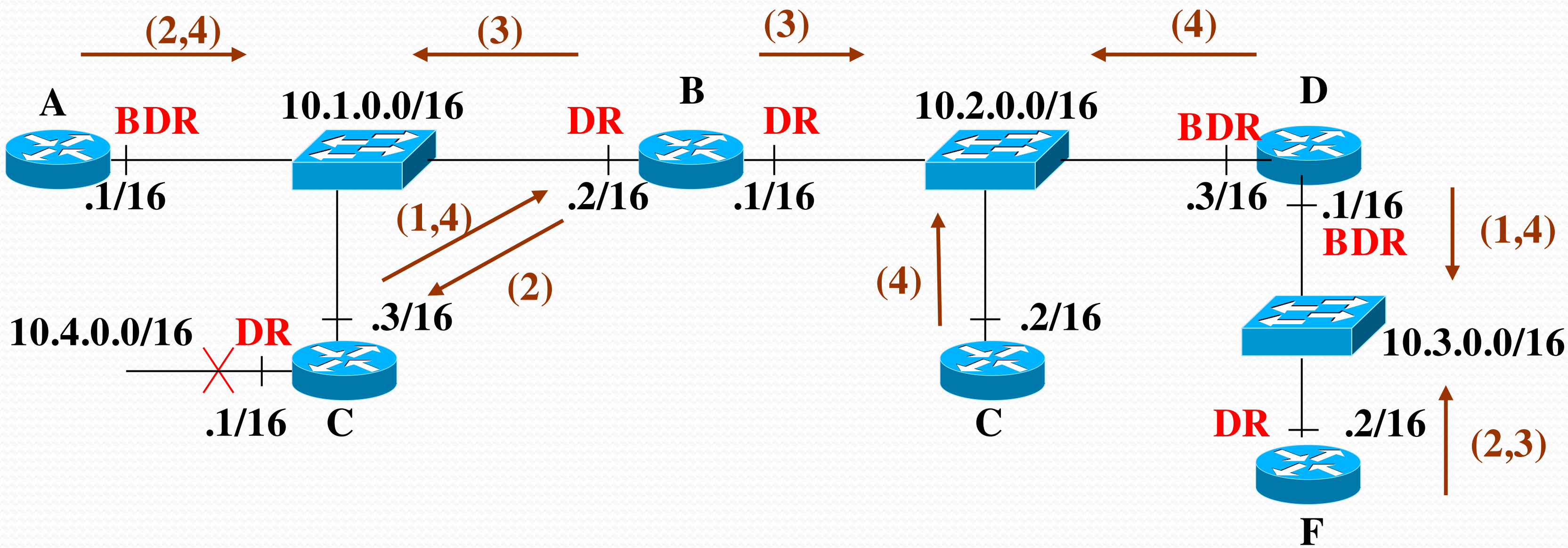
Here is the entry of net 10.0.2.0/16 **:LSU**

LSAck: Thanks, all is OK

Full state

Run Dijkstra and calculate the routing table

- Routing Maintenance:



All routers must have the same DB

(1) Router C sends an LSU to its DR using multicast address 224.0.0.6 (All-DR-DBR-routers)

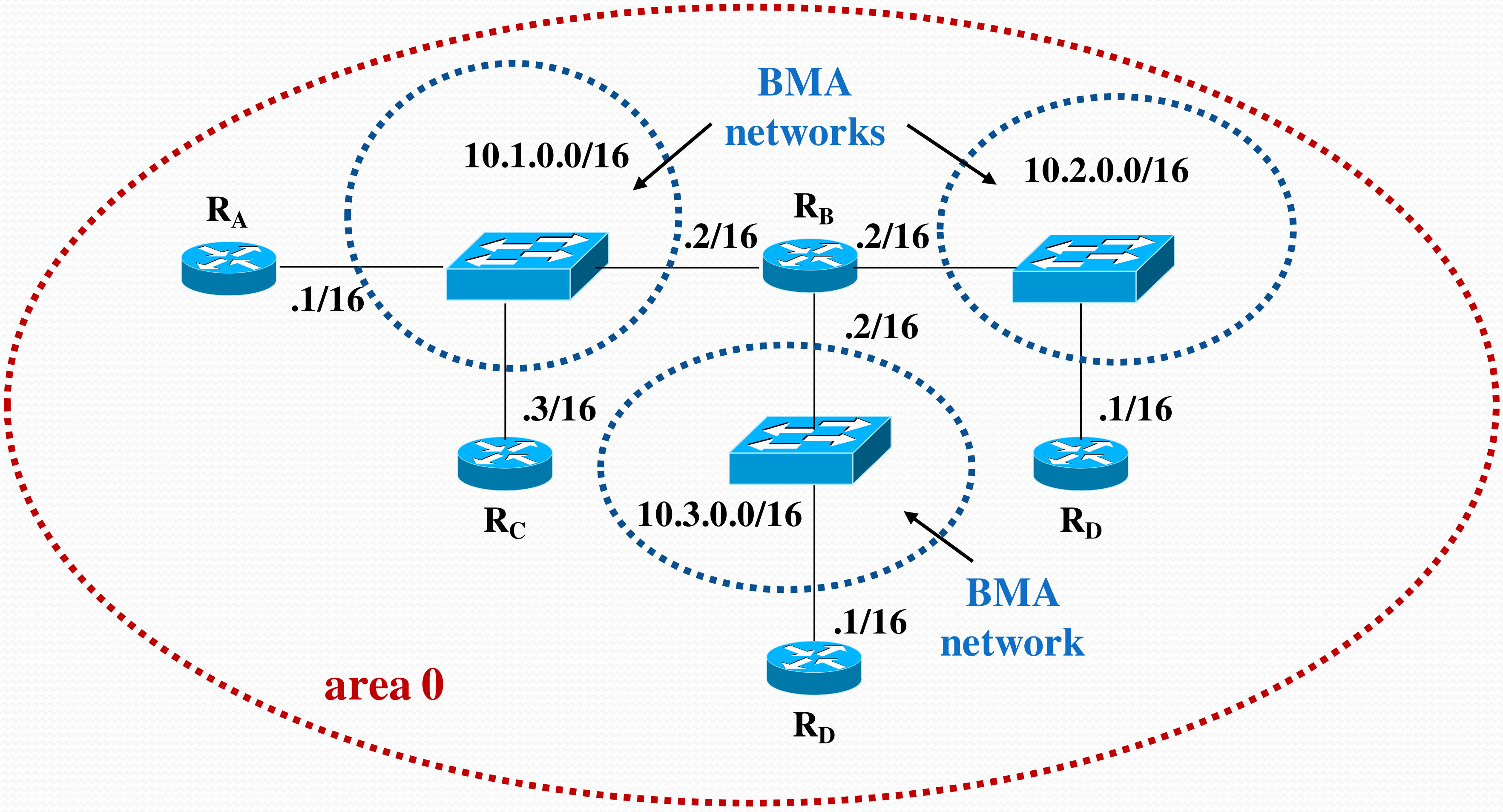
(2) DR ACKs the LSU with a LSAck,

(3) DR floods the LSU using multicast address 224.0.0.5 (All-OSPF-routers) to its BMA, network. Steps (1) and (2) are repeated in the rest of MBA's networks, e.g. router D sends an LSU to its DR of the other MBA network using 224.0.0.6 (All-DR-DBR-routers),

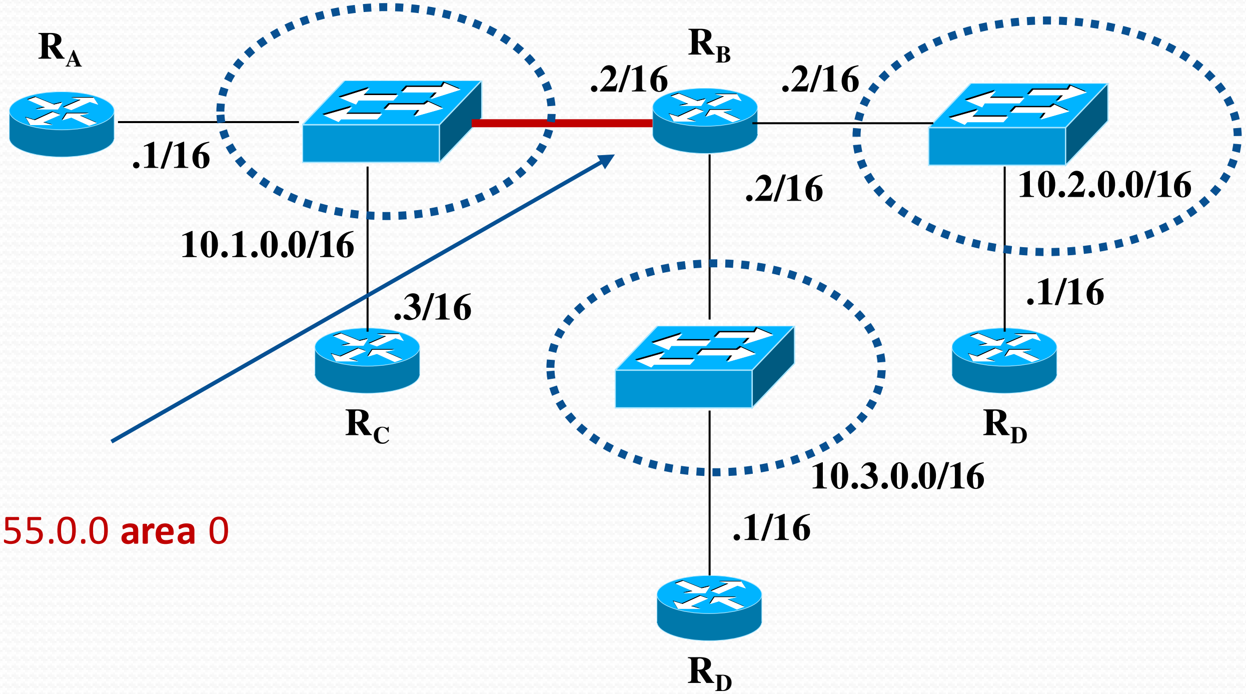
(4) All routers in the MBA network, ACK that LSU to their MBA DR,

(5) All routers in the BMA recalculate their routing table.

- OSPF configuration in a single area:



- **OSPF configuration in a single area:**



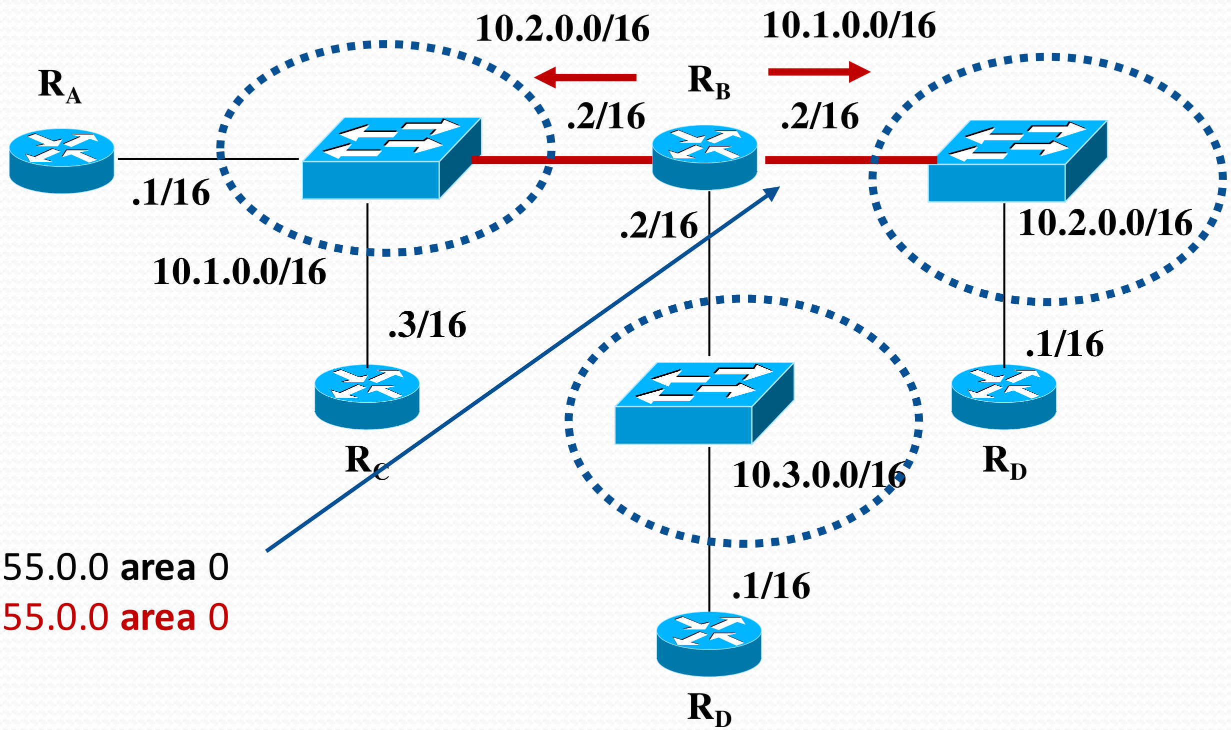
!!!! Configure Router RB
R_B# configure terminal
R_B(conf)# **router** ospf 1
R_B(conf-r)# **network** 10.1.0.0 255.255.0.0 **area** 0

Interface 10.1.0.2 is OSPF active in area 0.

The interface understands multicast address 224.0.0.5 (all-ospf-routers)

The router sends HELLOS on this interface. The routers selects DR/BDR in this interface (10.1.0.2 is DR). Not IP nets are announced in this interface.

- **OSPF configuration in a single area:**



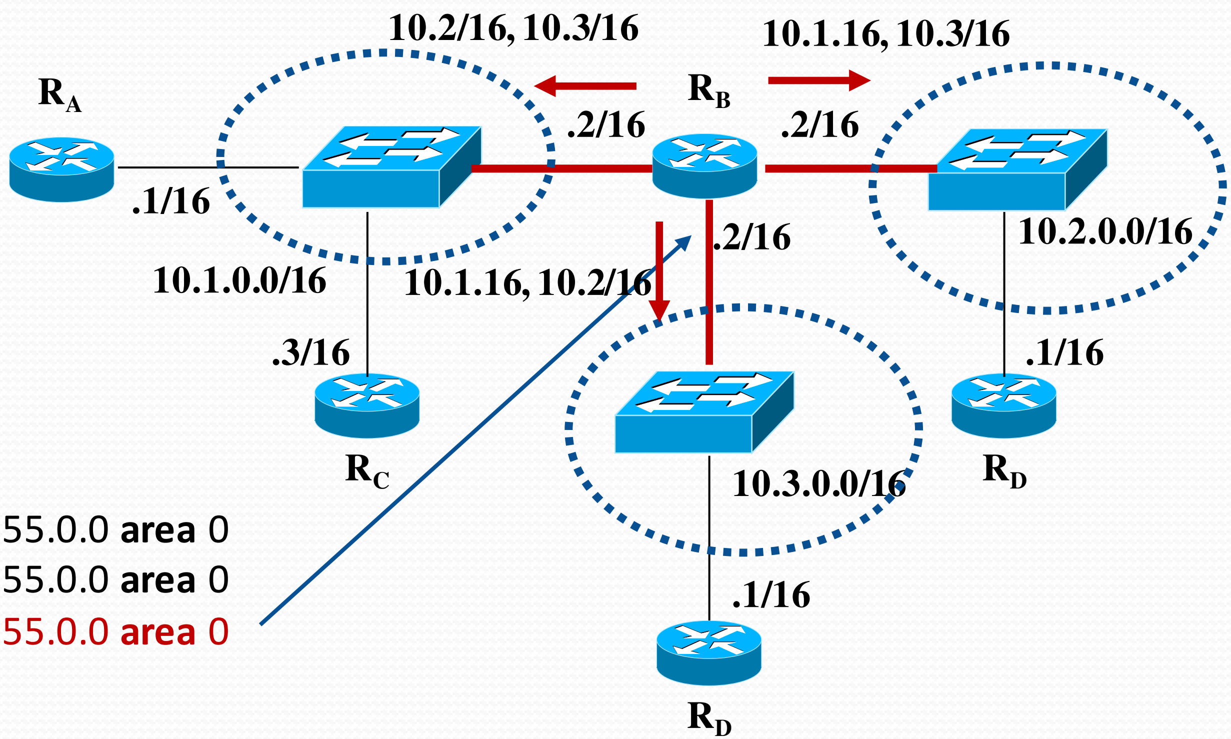
!!!! Configure Router RB
R_B# configure terminal
R_B(conf)# **router** ospf 1
R_B(conf-r)# **network** 10.1.0.0 255.255.0.0 **area** 0
R_B(conf-r)# **network** 10.2.0.0 255.255.0.0 **area** 0

Interfaces 10.1.0.2 and 10.2.0.2 are OSPF active in area 0.

The 2 interfaces understand multicast address 224.0.0.5 (all-ospf-routers)

The router sends HELLOS on both interfaces. The routers selects DR/BDR in the new active interface (10.2.0.2 is DR). Interface 10.1.0.2/16 announces route 10.2.0.0/16 and interface 10.2.0.2 announces route 10.1.0.0/16.

- **OSPF configuration in a single area:**



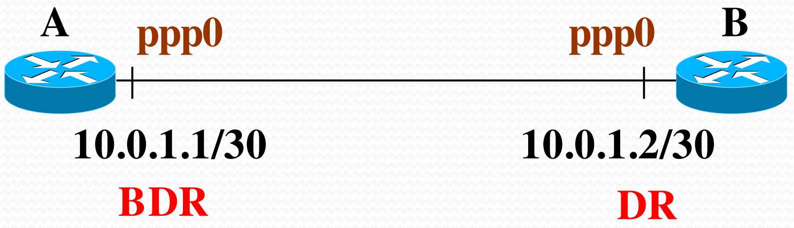
!!!! Configure Router RB
R_B # configure terminal
R_B(conf)# **router** ospf 1
R_B(conf-r)# **network** 10.1.0.0 255.255.0.0 **area** 0
R_B(conf-r)# **network** 10.2.0.0 255.255.0.0 **area** 0
R_B(conf-r)# **network** 10.3.0.0 255.255.0.0 **area** 0

Interfaces 10.1.0.2 and 10.2.0.2 are OSPF active in area 0.

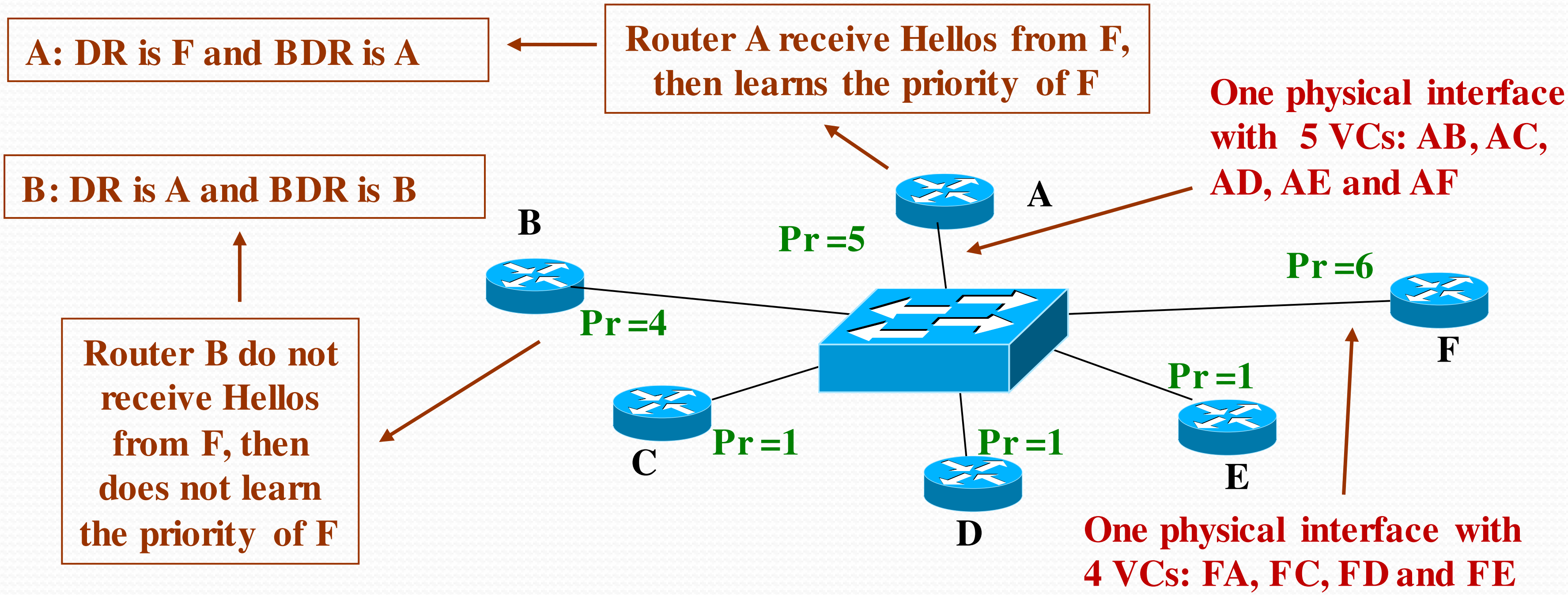
The 2 interfaces understand multicast address 224.0.0.5 (all-ospf-routers)

The router sends HELLOS on both interfaces. The routers selects DR/BDR in the new active interface (10.3.0.2 is DR). Interface 10.1.0.2/16 announce routes (10.2.0.0/16, 10.3.0.0/16), interface 10.2.0.2 announce routes (10.1.0.0/16, 10.3.0.0/16), and interface 10.2.0.3 announce routes (10.1.0.0/16, 10.2.0.0/16),

- **Point-to-Point topologies** (e.g.; dedicated lines E1 or T1)
  - It can be considered as a particular case of the BMA case with 2 routers,
  - Each router is adjacent by definition since there only are two routers. **One is DR and the other BDR**,
  - Both use HELLO packets to discover each other (using multicast address 224.0.0.5 "All-OSPF-routers").



- **NBMA (Non-broadcast Multi-Access) topologies:**
  - NBMA topologies are those ones in which a **non broadcast WAN** communicates the routers, e.g.; Frame Relay or ATM use Virtual Circuits (VC) that are point-to-point L2 communications and does not exist broadcast L2 communications,
  - Problem: difficulty in the DR/BDR election since **there no exists** L2 broadcast
    - E.g.; WAN as Frame Relay or ATM (Virtual Circuit switching)

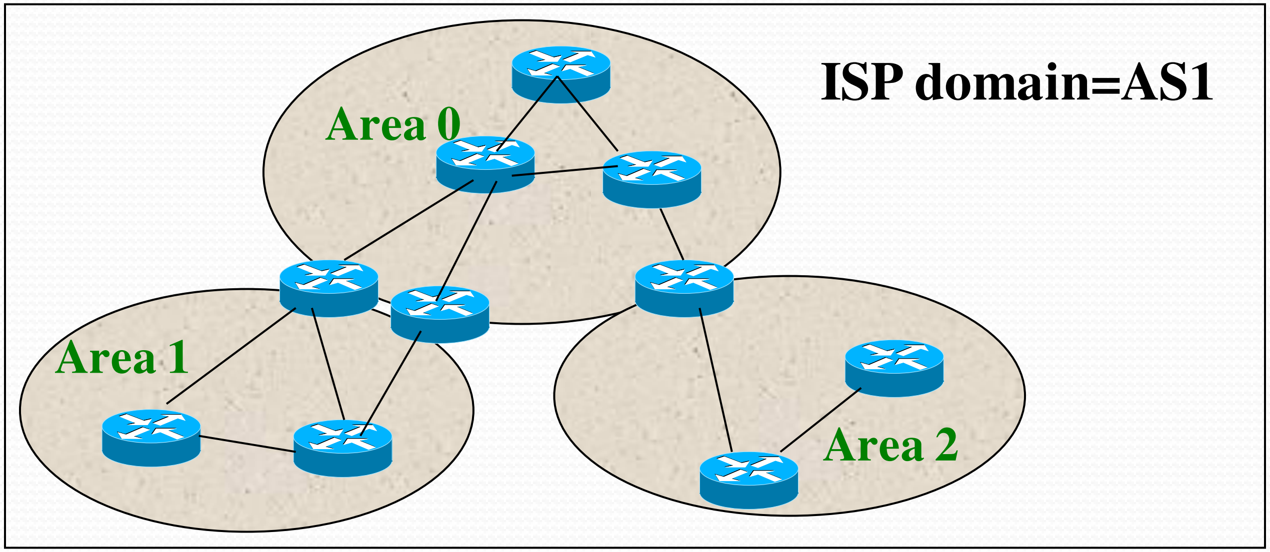


- **NBMA (Non-broadcast Multi-Access) topologies:**
  - Solutions (non-standard, manufacturer dependent):
    1. **NBMA architecture**: simulate OSPF in a MBA network → If N routers in the NBMA, then it needs a full meshed N*(N-1) Virtual Circuits and sends a Hello per VC:
       - CPU consuming !!! and bandwidth intensive !!!
       - Useful in small networks (does not scale well)
       - <u>One DR/DBR for the whole NBMA</u>.
    2. **Point-to-MultiPoint**: treat the non-broadcast as a collection of point-to-point connections:
       - <u>One DR/DBR for point-to-point link</u>. Then, N*(N-1)/2 DR/BDR's
  - In general, the idea is that routers with L2 interfaces with NBMA of these technologies provide some "middleware" to work with OSPF HELLO's that need to be transmitted to all routers attached to the NBMA L2 network.

OSPF Scalability:

- Assume a large network (e.g., 400 routers !!!)
  - Routers have to send LSU's: **large amounts of LSUs** (bandwidth intensive) **in the OSPF network**,
  - **Routing table re-computation** each time there is a change in the network (link up/down), Dijkstra consumes a lot of CPU,
  - The **DataBase may be quite large** (store all links and costs for the whole network)
  - **Routing table can be very large**.
- Solution:
  - **Multi-area OSPF network:** hierarchical routing in an OSPF network using the concept of "**areas**"
  - **Area**: <u>set of routers and networks in the same AS </u>(Autonomous System) <u>using a single OSPF area and connected to other OSPF areas</u>.

- **OSPF Scalability:**



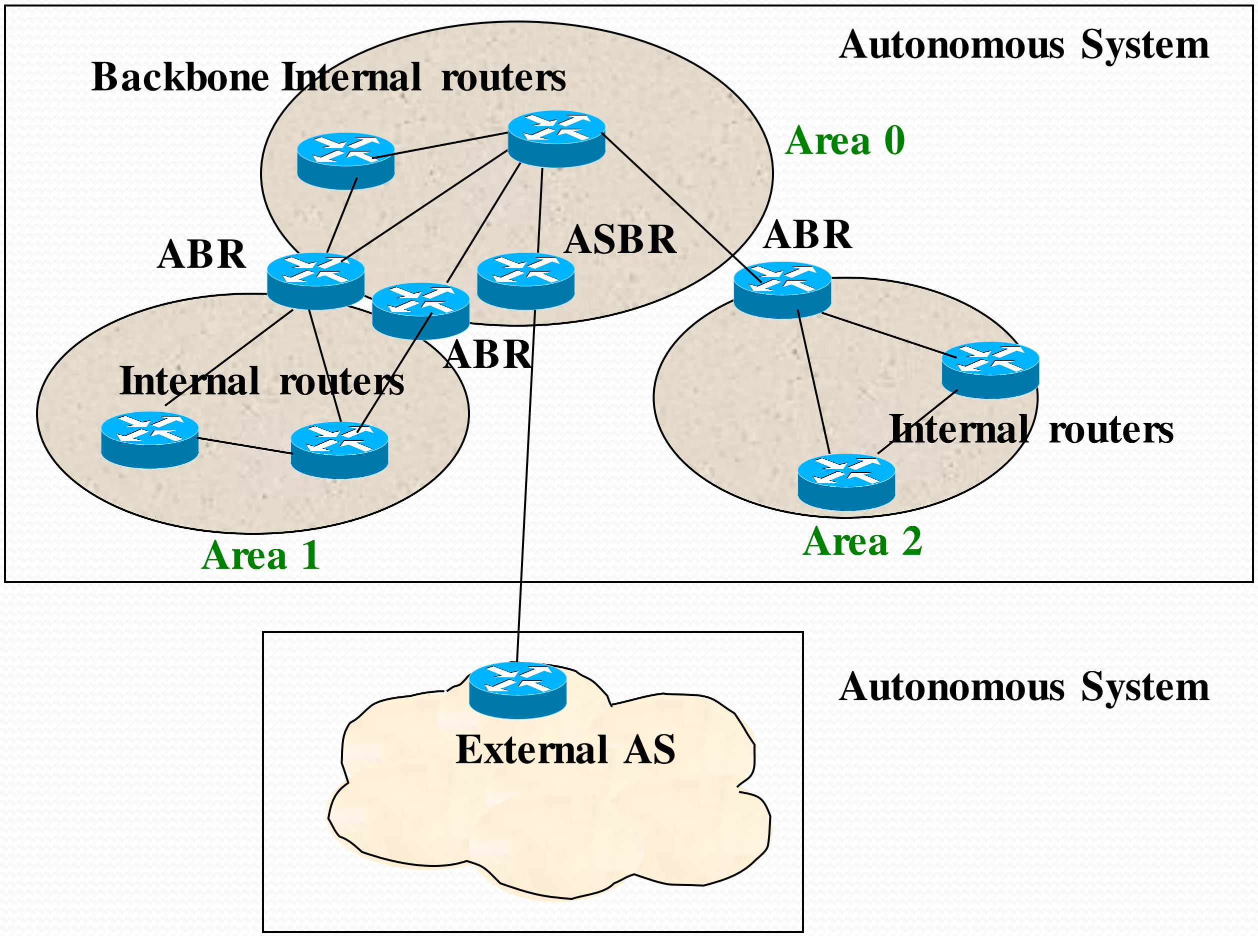
	Minimum	Mean	Maximum
Routers in a Domain	20	510	1000
Routers in a single Area	20	160	350
Areas per Domain	1	23	60

Standard Area: single area as we have previously seen

Backbone Area (transit area): area 0 that interconnects other areas in a multi-area system.

- **OSPF Scalability:**
  - Router types in a multi-area system:
    - **Internal Router:** router with all interfaces inside the same area
      - maintain only 1 DB (identical to other routers in the same area)
    - **Backbone or transit Router:** routers with all their interfaces in area 0
      - area 0 serves as backbone (transit) with other areas, maintain only 1 DB,
    - **Area Border Router (ABR):** routers with interfaces in multiple areas:
      - maintain one BD per area to which they are connected including the backbone (consume memory and CPU),
      - summarize area information and distribute them at each area using the backbone area,
      - when an ABR receives information from one area, calculates paths towards that area,
    - **Autonomous System Boundary Router (ASBR):** routers with one interface towards other AS
      - maintain as many DB as interfaces in different areas as an ABR router.

- **OSPF Scalability:**



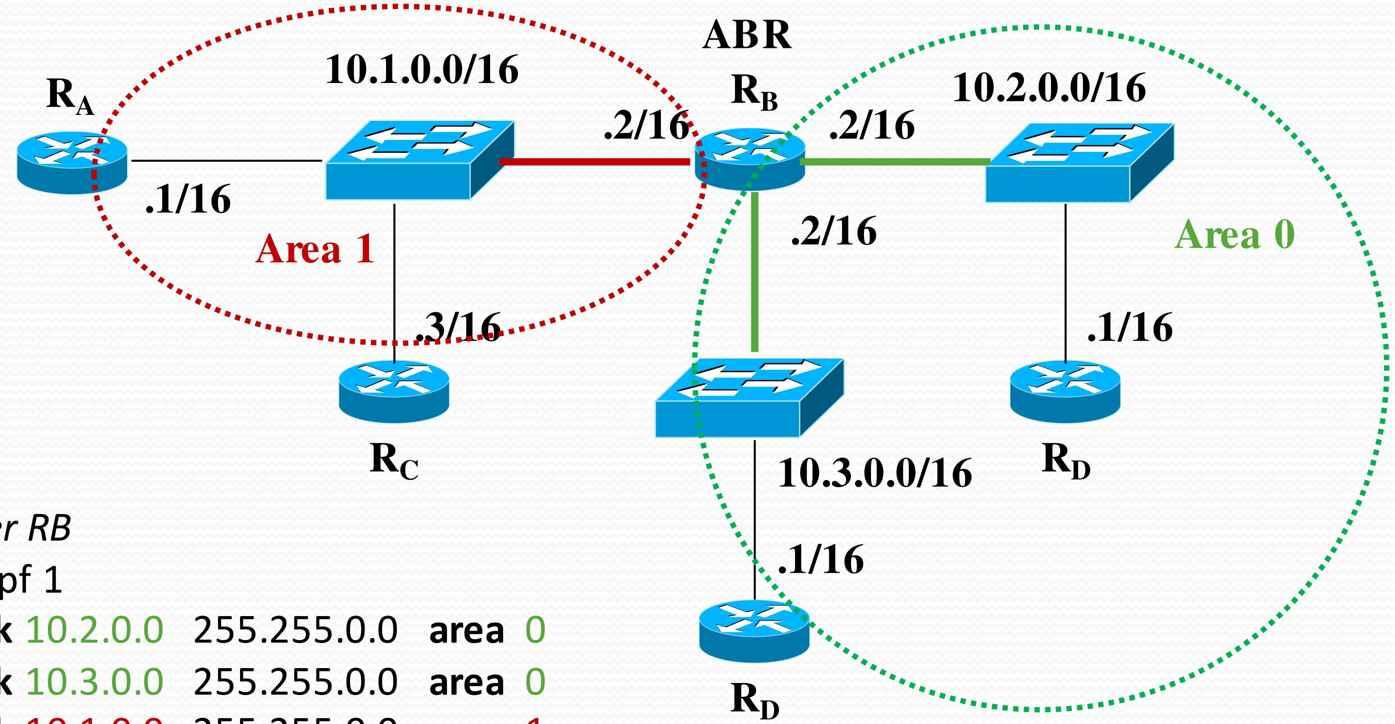
- **OSPF Scalability:**
  - Data packet transmission:
    - **Intra-area routing:** packets directed to an internal OSPF network, that means inside an area,
    - **Inter-area routing:** packets directed to an external OSPF network, that means to other OSPF areas:
      - The packet goes to ABR routers (intra-area routing)
      - The ABR router sends the packet to other ABR connected to the destination area (backbone routing)
      - This ABR router sends the packet inside the destination area (intra-area routing)

OSPF Scalability:

- There are **11 LSAs types** in OSPF multi-area network, 6 are the most used:
  - **Type 1: Router LSA:**
    - generated by each router inside its area towards each internal router,
    - describes each link state and cost for each internal router,
  - **Type 2: Network LSA:**
    - generated by each DR (in a BMA network), describes the set of routers connected to that BMA network (only sent inside the area),
  - **Type 3: Summary LSA:**
    - generated by each ABR router, describe external routes (got from other ABR's),
    - There is an entry per each subnetwork (summarized),
  - **Type 4: ASBR summary LSA:**
    - generated by each ABR router, describe routes towards ASBR's (to get out the AS),

- **OSPF Scalability:**
  - LSAs types in OSPF multi-area network
    - **Type 5: AS external LSA:**
      - generated by ASBR routers, describe external routes belonging to other AS (e.g., default network to get out the AS),
      - used in standard areas that are not "Stub Areas" or "No So Stubby Areas"
    - **Type 7: NSSA (Not So Stubby Area) external LSA:**
      - generated by ASBR routers, describe external routes belonging to other AS (e.g., default network to get out the AS),
      - only used by Not So Stubby areas (NSSA).
  - **Stub Area:** area with only 1 connection to the backbone area 0, and that **blocks type 5 "AS external LSA's"** → It is not allowed to have ASBR routers connected to other AS's,
  - **Not So Stubby Area (NSSA): block type 5 "AS external area"**, but allows to have ASBR routers: **Uses type 7 "NNSA external LSA"** to get the external routes.

OSPF configuration in a multiple area:



!!!! Configure Router RB

RB(conf)# **router** ospf 1

RB(conf-r)# **network** 10.2.0.0 255.255.0.0 **area** 0

RB(conf-r)# **network** 10.3.0.0 255.255.0.0 **area** 0

RB(conf-r)# **network** 10.1.0.0 255.255.0.0 **area** 1

- ISP design: Intra-domain routing