The University of Jordan, Comp. Eng. Dept.
Networks lab: Handout: Experiment 7
Dynamic Routing: Open Shortest Path First (Theory and Practice)

Instructors: Prof. Khalid A. Darabkh and Eng. Muna Al-Akhras

Parts Included - Part I: OSPF Discussion with sections (Definitions and Motivation, OSPF Neighbors, OSPF Neighborship Requirement, OSPF Neighborship States, OSPF Tables and Packets – Summarized, OSPF Areas, and OSPF Metric Cost Calculation), Part II: OSPF Configuration Commands, and Part III: Practical Problems

Part I: Open Shortest Path First Routing Protocol Discussion

A) Definitions and Motivation

There are some interesting definitions related to the name of OSPF, which are:

- ***Link:*** A link is a router's interface connected to an IP subnet. When we add an interface to the OSPF process, **OSPF considers the interface as a link.**
- ***State:*** Since a link is an interface, it has two states: up and down. The up state shows the link (interface) is operational and OSPF can reach the IP subnet connected to the link. The down state shows the link is not operational and OSPF cannot reach the IP subnet connected to the link.
- ***Link state protocol:*** OSPF is a link-state protocol. Link state protocols use the Shortest Path First (SPF) algorithm to calculate the best path to a destination. To run this algorithm, link-state protocols learn the complete topology of the network.

As a matter of fact, OSPF is one of the Interior Gateway Protocol (IGP), which helps to find the best routing path between the source and the destination router using its own shortest path first algorithm. It is a link-state routing protocol that is used to distribute routing information about data packets within a large Autonomous System. Because it is an open standard, it is implemented by a variety of network vendors. OSPF will run on most routers that doesn't necessarily have to be Cisco routers (unlike EIGRP which can be run only on Cisco routers). **Here are the most important features of OSPF:**

- **a classless routing protocol**
- **supports CIDR, manual route summarization, equal cost load balancing**
- **incremental updates are supported**
- **uses only one parameter as the metric – the interface cost.**
- **the administrative distance of OSPF routes is, by default, 110.**
- **uses multicast addresses 224.0.0.5 and 224.0.0.6 for routing updates.**

Routers running OSPF have to establish neighbor relationships before exchanging routes. Because OSPF is a link state routing protocol, neighbors don’t exchange routing tables. Instead, they exchange information about network topology. Each OSPF router then runs SPF or Dijkstra algorithm to calculate the best routes and adds those to the routing table. Because each router knows the entire topology of a network, the chance for a routing loop to occur is minimal.

When it is configured, it listens to its neighbours in the networks, and it gathers all the link state data available. This data is then used to make a topology map that contains all available paths in the network. This database is saved for use, and we call it Link State Database. Once the Link State Database is made, it is used to calculate the shortest path to subnets/networks using an algorithm known as Shortest Path First, developed by Edsger W Dijkstra. However, OSPF creates three tables:

- **Routing Table:** It contains currently working best paths that will be used to forward traffic between two neighbours.
- **Neighbour Table:** This contains all discovered Open Short Path First neighbours.
- **Topology Table: This one contains the entire road map of the network (i.e., topology structure of a network).** This road map includes all the available Open Short Path First routers and keeps calculated data about best and alternative paths.

B) OSPF Neighbors

OSPF routers need to establish a neighbor relationship before exchanging routing updates. OSPF neighbors are dynamically discovered by sending Hello packets out each OSPF-enabled interface on a router. **Hello packets are sent to the multicast IP address of 224.0.0.**5. The process is explained in the following figure:

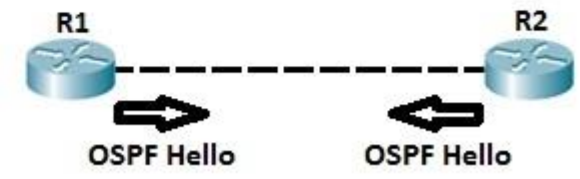


Figure 1. OSPF Hello messages

Routers R1 and R2 are directly connected. After OSPF is enabled both routers send Hellos to each other to establish a neighbor relationship. You can verify that the neighbor relationship has indeed been established by typing the ***show ip ospf neighbors*** **command**.

```
R1#show ip ospf neighbor
Neighbor ID Pri State Dead Time Address Interface
2.2.2.2 1 FULL/DR 00:00:30 192.168.0.2 FastEthernet0/0
```

In the example above, you can see that the router-id of R2 is 2.2.2.2. Each OSPF router is assigned a router ID. A router ID is determined by using one of the following:

- ✓ Using the router-id command under the OSPF process.
- ✓ Using the highest IP address of the router’s loopback interfaces.
- ✓ Using the highest IP address of the router’s physical interfaces.

C) OSPF Neighborship Requirement

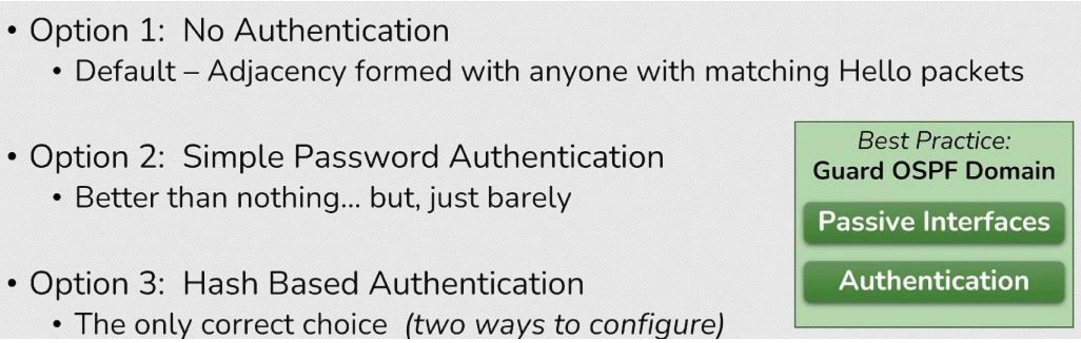
In order to become OSPF neighbor, the following values must be match on both routers. In other words, **the following fields in the Hello packets must be the same on both routers in order for routers to become neighbors:**

- Subnet, totally clear
- Area ID, which will be discussed shortly
- Authentication (optional but highly recommended)
- Hello and Dead Intervals
- Area Type, which will be discussed shortly
- Interface MTU Size

1. Authentication

To enhance the security of network, OSPF allows us to configure the password for specific areas. Routers who have same password will be eligible for neighborship. If you want to use this facility (as the default is no authentication, as seen in Figure 2), you need to configure password on all routers which you want to include in network. **If you skip any router, that will not be able to form an OSPF neighborship.**

Suppose that our network has two routers R1 and R2. Both routers are connected with direct link and meet all criteria mentioned in first requirement. What if I configure password in R1 and leave R2 as it is? Will it form neighborship with R2? **In this situation neighborship will not take place** because when both routers see each other's hello packet in segment, they try to match all configure values **including password field. One packet has a value in password filed while other has nothing in it. In this case routers will simply ignore each other's packet. However, kindly pay attention to the OSPF authentication option 2, as shown in Figure 3.**



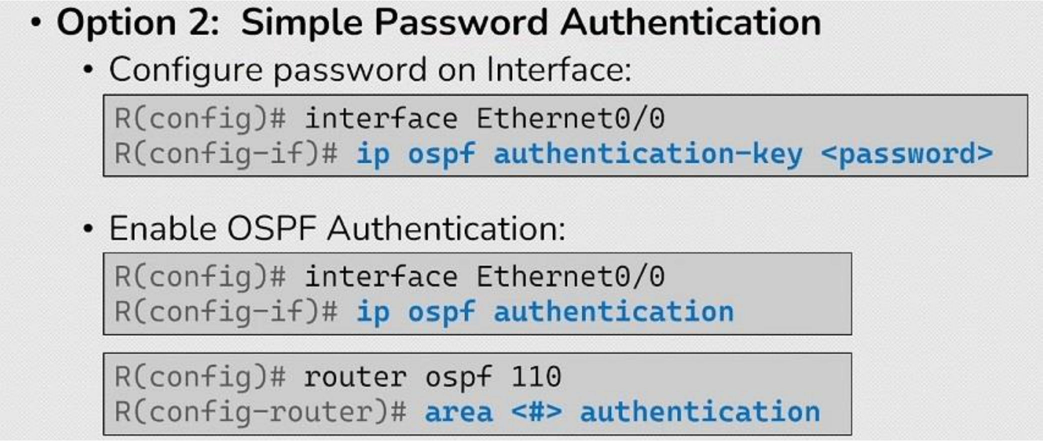
- Option 1: No Authentication
  - Default – Adjacency formed with anyone with matching Hello packets
- Option 2: Simple Password Authentication
  - Better than nothing... but, just barely
- Option 3: Hash Based Authentication
  - The only correct choice *(two ways to configure)*

Best Practice:
Guard OSPF Domain

Passive Interfaces

Authentication

Figure 2. Authentication Options



- **Option 2: Simple Password Authentication**
  - Configure password on Interface:

```
R(config)# interface Ethernet0/0
R(config-if)# ip ospf authentication-key <password>
```

  - Enable OSPF Authentication:

```
R(config)# interface Ethernet0/0
R(config-if)# ip ospf authentication
```

```
R(config)# router ospf 110
R(config-router)# area <#> authentication
```

Figure 3. OSPF Authentication Option 2

2. Hello and Dead Intervals

By default, OSPF sends hello packets every 10 second on an Ethernet network (Hello interval). A dead timer is four times the value of the hello interval, so if a router on an Ethernet network doesn’t receive at least one Hello packet from an OSPF neighbor for 40 seconds, the routers declares that neighbor to be down. Figure 4 shows a description for OSPF Hello and dead intervals.

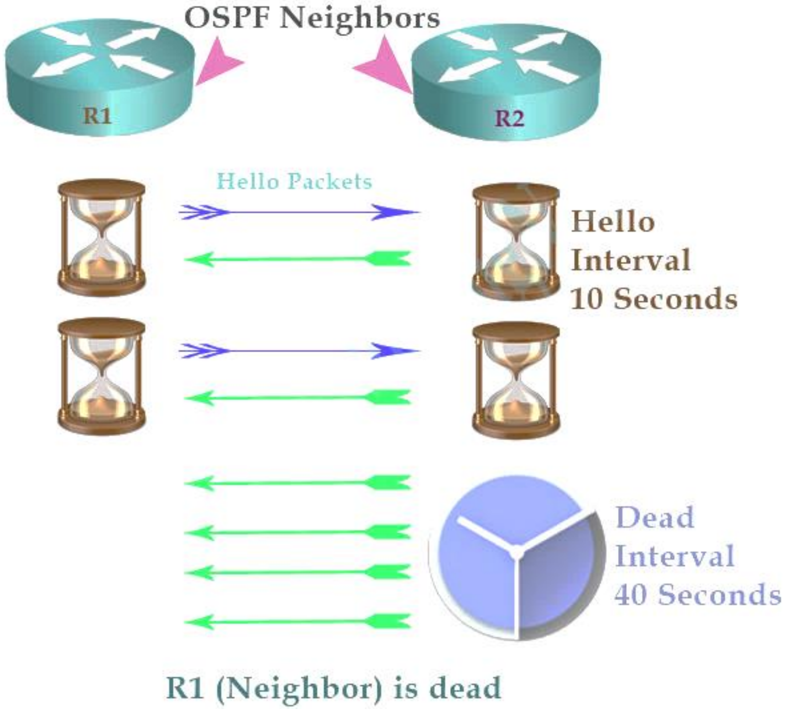


Figure 4. OSPF Hello and Dead Intervals

3. Interface MTU

Technically, the MTU (Maximum Transmission Unit) has to be verified. **It includes the MTU value of the outgoing interface. If this value does not match the value of the ingoing interface (at the receiving end), routers may have stuck in Exstart/Exchange exchange stage.** Particularly, consider a situation where MTU setting between two OSPF routers does not match.

If the router with the higher MTU sends a packet larger than the MTU set on the neighboring router, the neighboring router will ignore this packet. The reason is that this function creates serious problem for database updates. Database updates are heavier in nature. Once an update becomes larger than the configured MTU setting, it needs to be spilt. In a case of miss match MTU, database update may lose few bytes. Due to this, OSPF will ignore that update and cannot synchronize with database.

D) OSPF Neighborship States

a. Neighborship States

OSPF routers go through the seven states while building neighborship with other routers, which are as follows:

1. **Down state**
2. **Attempt/Init state**
3. **Two ways state**
4. **Exstart state**
5. **Exchange state**
6. **Loading state**
7. **Full state**

They are explained thoroughly as follows:

1. **Down state**

At this point both routers have no information about each other. Following Figure 5, R1 does not know which protocol is running on R2. Vice versa R2 have no clue about R1. In this stage OSPF learns about the local interfaces which are configured to run the OSPF instance.

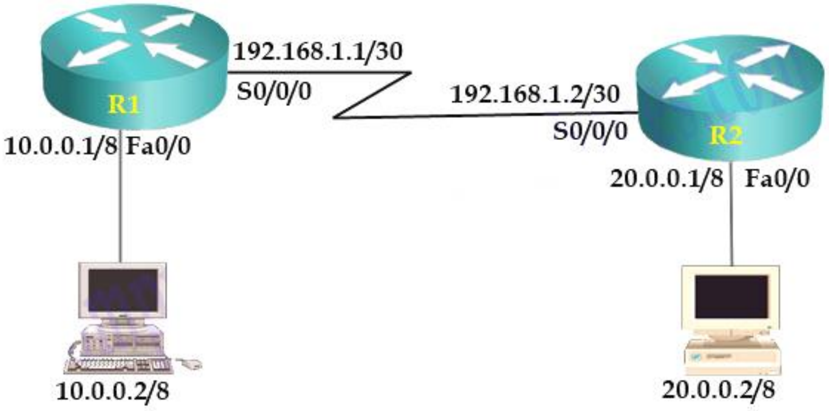


Figure 5. OSPF Neighborship States - Down State

In down state routers prepares themselves for neighborship process. In this state routers choose RID (Router ID). **RID plays a big role in OSPF process**. **If OSPF fails to select the RID, it will halt the OSPF process. We cannot use OSPF process without RID (where there are three different ways to assign RIDs, as shown in Figure 6).** In fact, RID is a unique identifier of Router in OSPF network. It must be unique within the autonomous system. Routers identify each other through the RID in AS.

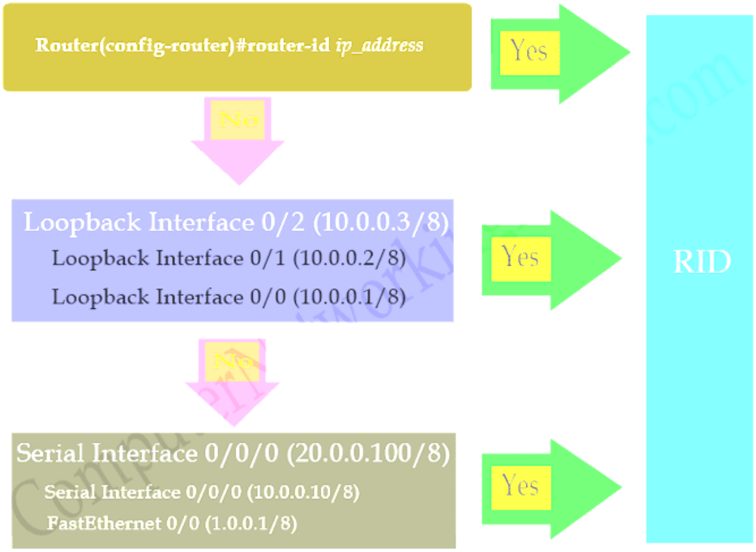

Router(config-router)#router-id ip_address
Yes
No
Loopback Interface 0/2 (10.0.0.3/8)
Loopback Interface 0/1 (10.0.0.2/8)
Loopback Interface 0/0 (10.0.0.1/8)
Yes
No
Serial Interface 0/0/0 (20.0.0.100/8)
Serial Interface 0/0/0 (10.0.0.10/8)
FastEthernet 0/0 (1.0.0.1/8)
Yes
RID

Figure 6. OSPF Router ID Options

In down state, router do following:

- Choose RID and initialize the OSPF process
- Run OSPF instance on local interfaces which are configured through the network command such as **R1(config-router)#network *10.0.0.0 0.0.255.255* area *0*.**
- Collect necessary information for Hello packet such RID and configuration values which are required to build the neighborship.

2. Attempt/Init states

They are illustrated as follows:

i. Init state

Neighborship building process starts from this state. Following Figure 7, R1 multicasts first hello packet so other routers in network can learn about the existence of R1 as an OSPF router. **This hello packet contains Router ID and some essential configuration values such as area ID, hello interval, hold down timer, stub flag and MTU. Essential configuration values must be same on routers who want to build an OSPF neighborship.** Let us assume that these values match on both routers. If essential configuration values match, R2 will add R1 in his neighbor Table.

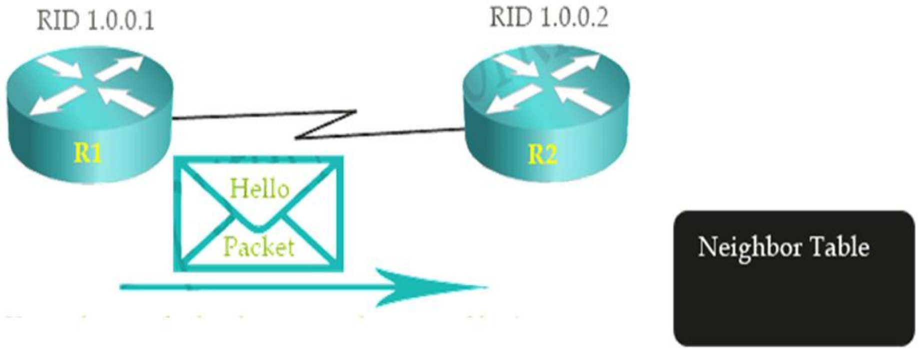


Figure 7. OSPF Neighborship States - Init state

To summarize, in Init state, routers do following

- R1 will generate a hello packet with RID and essential configuration values and send it out from all active interfaces.
- The hello packets are sent to the **multicast address *224.0.0.5***.
- R2 will receive this packet.
- R2 will read RID from packet and look in neighbor table for existing entry.
- If a match found, R2 would skip neighborship building process and **reset the dead interval timer for that entry.**
- If OSPF does not find a match in neighbor table, it will consider R1 (sender router) as a **possible** OSPF neighbor and start neighborship building process.
- R2 will match its essential configuration values with values listed in packet.
- If all necessary configuration values match, R2 will add R1 in its neighbor table.

At this moment R1 has no idea about R2. R1 will learn about R2 when it will respond. Let's have a quick look on attempt state.

ii. Attempt

In non-broadcast multi-access environment such as Frame Relay and X.25, OSPF uses Attempt state instead of Init state. OSPF uses this state only if neighbors are statically configured with neighbor command. In this situation, it does not have to discover them dynamically. As it already knows the neighbors, it will use unicast instead of multicast in this state.

Once neighborship is built, OSPF uses hello packets as keep alive. **If a router does not receive a hello packet from any particular neighbor in dead interval, it will change its state to down from full. After changing the state, it will make an effort to contact the neighbor by sending Hello packets. This effort is made in Attempt state.**

3. Two ways state

If essential configuration values match, R2 will add R1 in neighbor table and reply with its hello packet, as shown in Figure 8. As R2 knows the exact address of R1, it will use unicast for reply. Beside RID and configuration values, this packet also contains the R2's neighbor table data. As we know R2 has already added R1 in its neighbor table. Therefore, when R1 will see R2's neighbor table data, R1 would also see its name in this data. This will assure R1 that R2 has accepted its neighborship request. At this point, the following checked:

- R2 has checked all essential configuration values listed in hello packet which it received from R1.
- R2 is ready to build neighborship with these parameters.
- R2 has added R1 in its neighbor table.
- **To continue the neighborship process, R2 has replied with its hello packet.**
- R1 has received a reply from neighbor, with its own RID listed in R2's neighbor table.

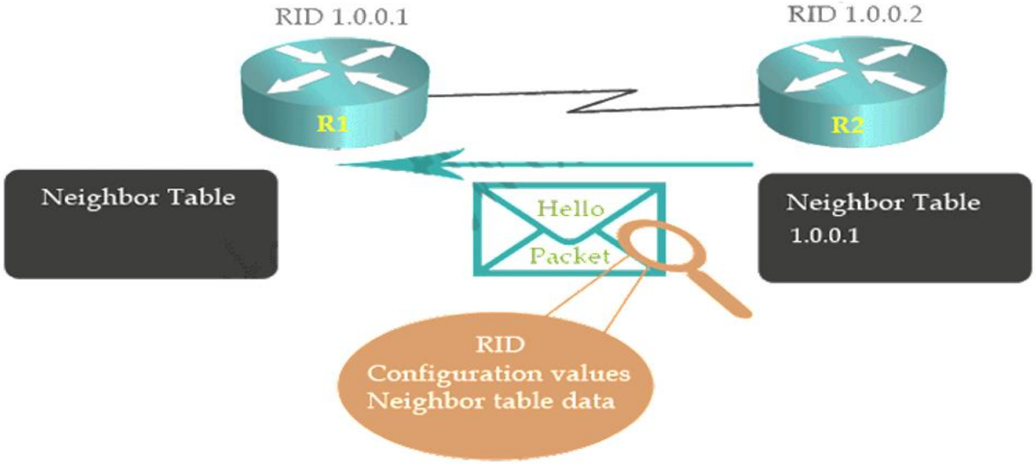


Figure 8. OSPF Neighborship States - Two Ways State - In Process

Now, it is R1's turn to take action on R2's reply. This reply would be based on hello packet which it received from R2. As we know that this hello packet contains one additional field; Neighbor table data field which indicates that this is not a regular neighbor discovery hello packet. This packet is a reply of its own request. It is interesting to note that R1 will take following actions:

- It will read RID from hello packet and look in its neighbor table for existing entry.
- If a match for RID found in neighbor table, it would reset the dead interval timer for that entry.
- If a match is not found in neighbor table, it would read the essential configuration values from packet.
- It will match configuration values with its own values. If values match, it will add R2's RID in neighbor table.
- **If packet contains neighbor table data with its own RID, it will consider that as request to enter in two-way state.**
- **R1 will reply with a hello packet which contains its neighbor table data.**
- **This packet is a confirmation of two ways state, as shown in Figure 9.**

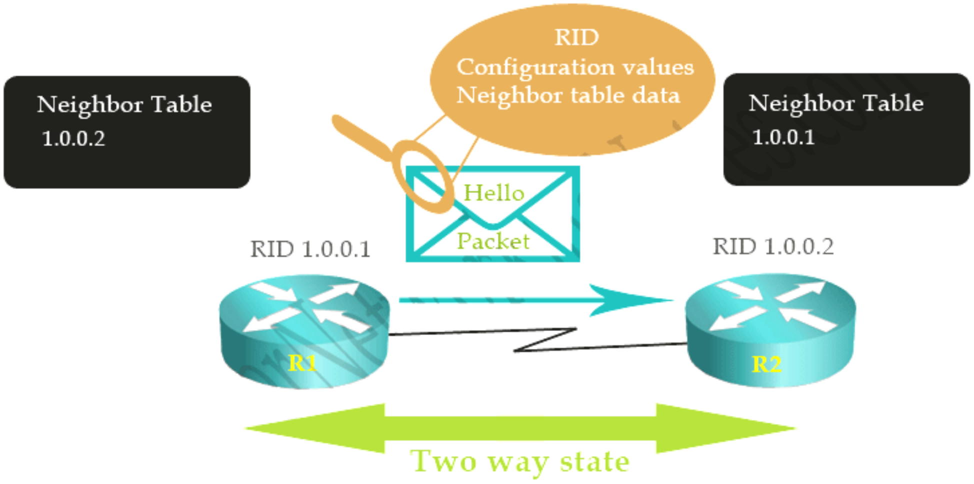
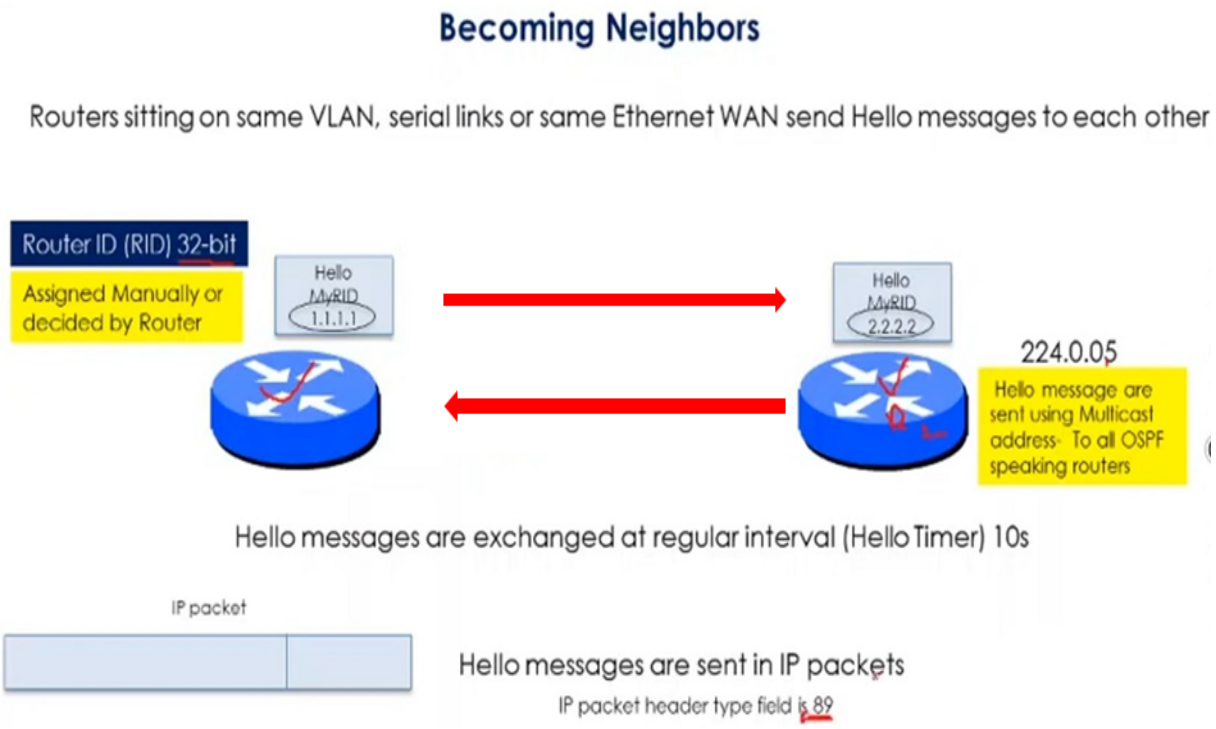


Figure 9. OSPF Neighborship States - Two Ways State - done

The routers are neighbor now. They are ready to exchange the routing information. Another scenario of reaching the two-way state is illustrated in Figures 10 and 11.



Becoming Neighbors

Routers sitting on same VLAN, serial links or same Ethernet WAN send Hello messages to each other

Router ID (RID) 32-bit

Assigned Manually or decided by Router

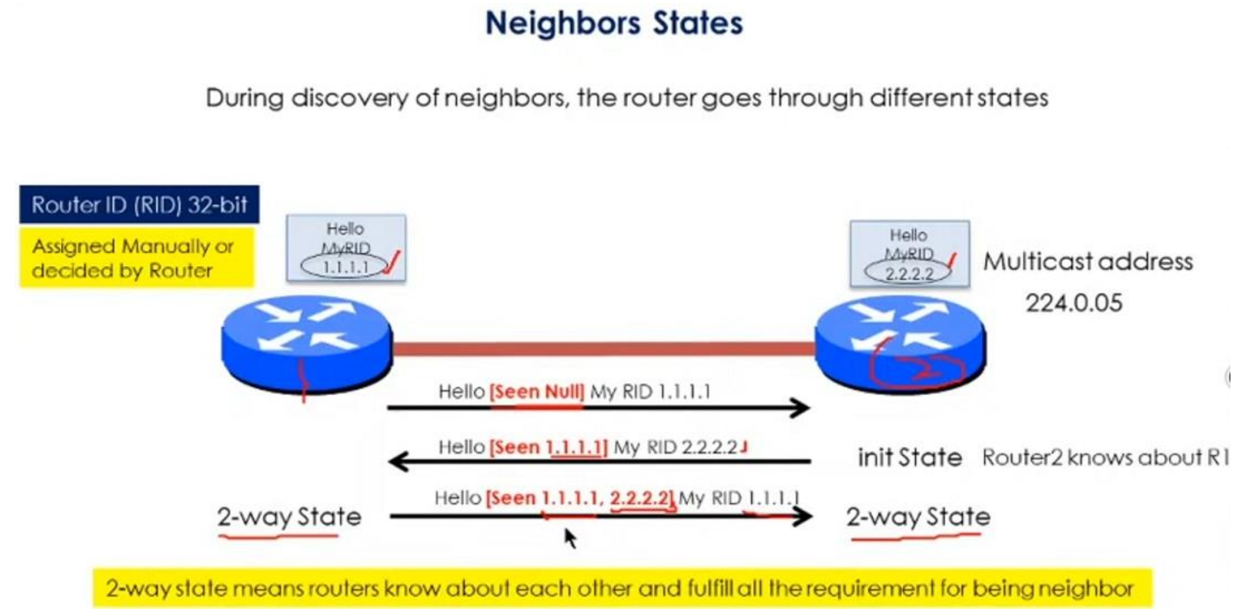
Hello message are sent using Multicast address- To all OSPF speaking routers

Hello messages are exchanged at regular interval (Hello Timer) 10s

Hello messages are sent in IP packets

IP packet header type field is 89

Figure 10. OSPF Neighborship - Becoming Neighbors



Neighbors States

During discovery of neighbors, the router goes through different states

Router ID (RID) 32-bit

Assigned Manually or decided by Router

Multicast address

224.0.05

Hello [Seen Null] My RID 1.1.1.1

Hello [Seen 1.1.1.1] My RID 2.2.2.2

init State Router2 knows about R1

Hello [Seen 1.1.1.1, 2.2.2.2] My RID 1.1.1.1

2-way State

2-way State

2-way state means routers know about each other and fulfill all the requirement for being neighbor

Figure 11. OSPF Neighborship States - Two ways State - Perfect

4. Exstart state

Routers who decided to build adjacency will form a master/slave relationship. In each adjacency router who has higher RID will become master and other will become slave. In fact, Master/Slave relationship is built between two interfaces which need to exchange routing information. Interestingly, the neighbor routers are going to exchange EMPTY Database Descriptors (DBDs) (do not include a list of LSAs), which will be discussed shortly (in the exchange state). The purpose of these DBD packets is to do the master/slave election which is required to govern the reliable exchange of further DBDs. Initially, both routers think that they are the Master Router. Therefore, they send DBDs indicating that they are the Master (i.e., the master bit (MS) of the DBD header is one for both), as shown in Figure 12. Then, the router with the higher numerical ID is elected as the Master. In Figure 12, R2 is the Master and R1 is the slave. What happens next is that the slave (R1) is going to drop into exchange state and consequently send a confirmation DBD (i.e., the MS bit is 0) indicating it does acknowledge it is the slave in this relationship.

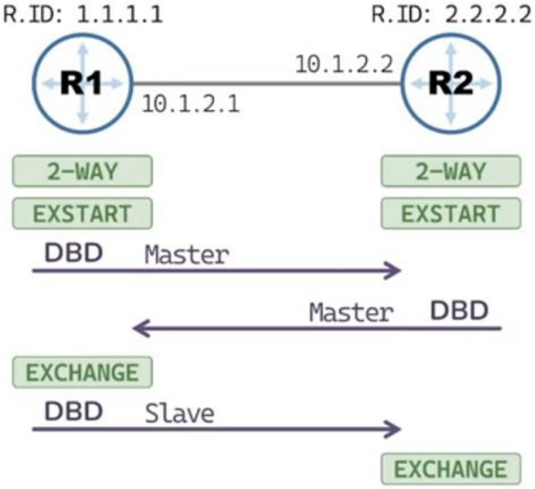


Figure 12. OSPF - Master/Slave Relationship

5. Exchange state

In the exchange state, the Master/Slave election is complete. As mentioned in the Exstart state, the slave will send a confirming DBD indicating it does acknowledge it is the slave in this relationship. What happens next is that both routers are going to exchange a summary of their link state database taking into account that the Master should start first. It is good to mention that a router that has more than one interface may learn same network information from different sources. An OSFP router is smart enough to filter the updates before receiving it. It will ask only for the updates which it does not have. In this state, routers will filter the updates which need be to exchange. Before we learn how routes will filter this information, let’s understand few relative terms:

- **LSA**

Link state advertisement (LSA) is a control packet which contains link-state and routing information. OSPF uses it to share and learn network information. It is noteworthy to mention that a router will flood its LSA all over the network domain (i.e., through all of its interfaces), as shown in Figure 13. Unlike the other OSPF packets, this packet (LSA), in particular, has noting with the neighborship.

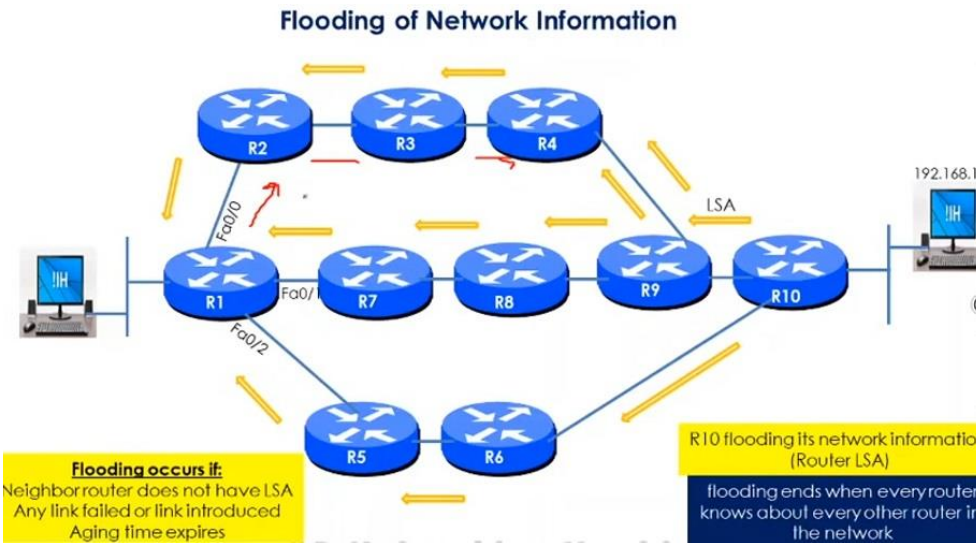

Flooding of Network Information
R2
R3
R4
Fa0/0
192.168.1
LSA
R1
Fa0/1
R7
R8
R9
R10
Fa0/2
R5
R6
Flooding occurs if:
Neighbor router does not have LSA
Any link failed or link introduced
Aging time expires
R10 flooding its network informatio
(Router LSA)
flooding ends when every router
knows about every other router i
the network

Figure 13. OSPF – Router LSA Flooding

- **LSDB**

Every OSPF router maintains a Link state database (LSDB). LSDB is collection of all LSAs received by a router. Every LSA has a sequence number. OSPF stores LSA in LADB with this sequence number.

- **DBDs**

Database description packets (also referred as DDPs) contain the list of LSA. This list includes link state type, cost of link, ID of advertising router and sequence number of link. Make sure you understand this term correctly. It is only a list of all LSAs from its respective database (i.e., LSAs checklist). It does not include full LSAs.

In this state, routers exchanges DBDs. Through DBDs routers can learn which LSAs they already have. For example, in Figure 14, R1 has A1, A2 and B2 LSAs in its LADB. Therefore, it will send a list of these LSAs to R2. This list is a DBDs. R2 will send an acknowledgment of receiving the list with LSACK signal. Same as R2 will send its DBDs to R1 and R1 would acknowledge that with its LSACK single.

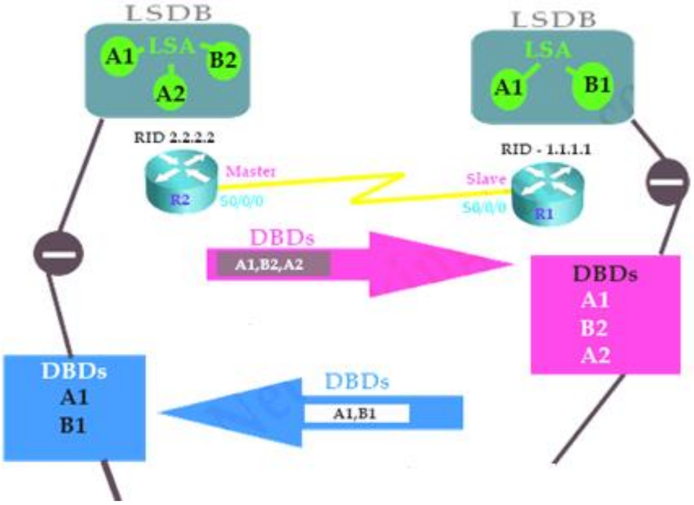


Figure 14. OSPF - Exchange State

At the end of this state, both routers have a list of LSAs which need to be exchanged.

6. **Loading state**

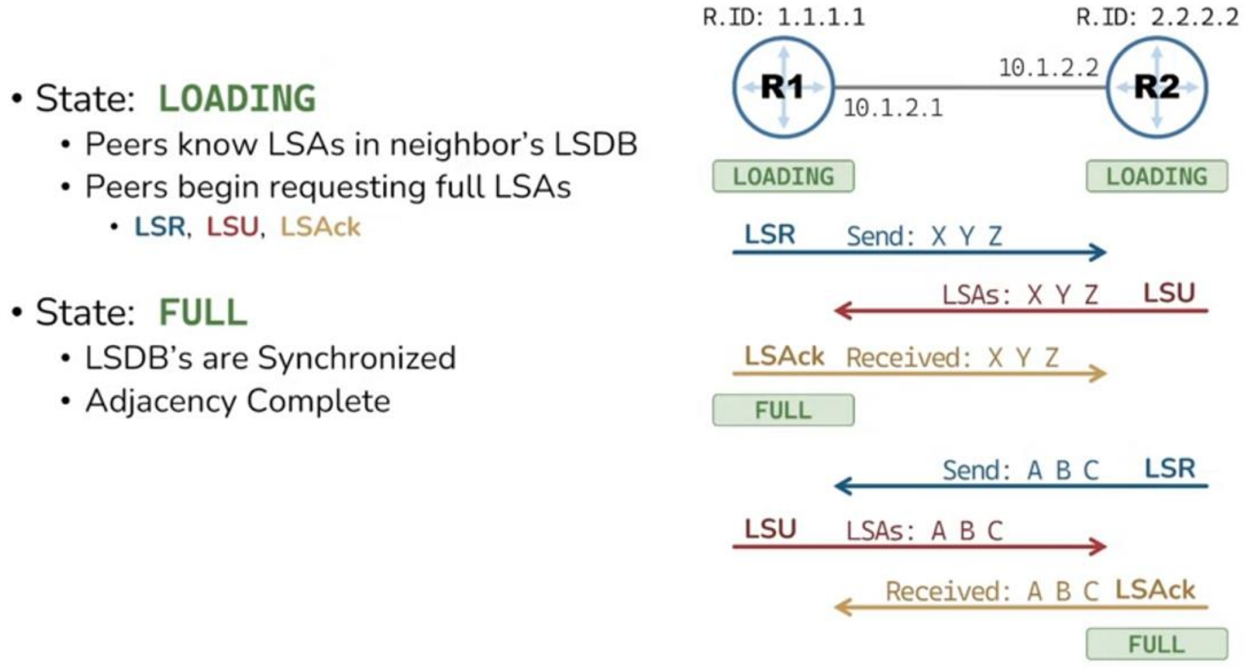
In this state, actual routing information is exchanged, as shown in Figure 15. Routers exchange LSAs from LSR list. In fact, routers will use LSU (Link state update) to exchange the LSAs. Each LSA contains routing information about a particular link. Routers also maintain a retransmission list to make sure that every sent LSA is acknowledged. For example, following figure illustrates loading state of above example. R1 sent a LSU which contain two LSAs but it received acknowledgement of only one, so it had to resend lost LSA again.

- **LSR**

Upon receiving DBDs, routers will compare it with their own LSDB. Thus, they will learn what they need to order. Referring to Figure 14, R1 received a check list (DBDs) of A1 and B1. When it will compare this list with its own LSA database (LADB), it will learn that it already has A1. Hence, it does not need to order this LSA again. But it does not have B1, so it needs to order for this LSA. After a complete comparison, both routers will prepare a list of LSAs which they do not have in their own LADB. This list is known as LSR (Link State Request). It is important to mention that after exchanging Database Description packets with a neighboring router, a router may find that parts of its topological database are out of date. The LSR is further used to request the pieces of the neighbor's database that are more up to date.

- **Link-State Update (LSU), which is a packet that contains fully detailed LSAs, typically sent in response to an LSR message.**
- **Link-State Acknowledgment (LSAck), which is sent to confirm the receipt of an LSU message.**

Figure 15 illustrates how aforementioned packets work over the scope of this state and eventually how to get to the full state.



- State: LOADING
  - Peers know LSAs in neighbor's LSDB
  - Peers begin requesting full LSAs
    - LSR, LSU, LSAck
- State: FULL
  - LSDB's are Synchronized
  - Adjacency Complete

Figure 15. OSPF - OSPF Adjacency (loading and getting to full state)

7. Full state

Full state indicates that both routers has been exchanged all LSAs from LSR list. Referring to Figure 14, both routers now have identical LSDB, as shown in Figure 16.

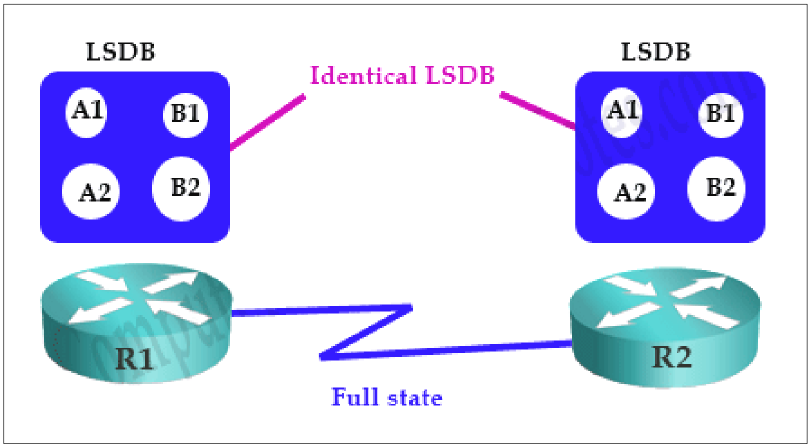


Figure 16. OSPF – Full State

Adjacent routers remain in this state for life time. **This state also referred as adjacency.** If any change occurs in network, routers will go through this process again.

b. Maintaining adjacency (i.e. sticking to Full state)

The following bullets are considered to maintain the OSPF adjacency:

- **Routers will send hello messages in hello interval.**
- **If a router does not receive hello message from neighbor in dead interval, it will declare that neighbor as dead.**
- **Once a neighbor is dead, router will flood this change to other connected neighbors.**
- **Beside of this, if a router detects any change in network or receives any update, it will flood that change (i.e., LSA (belong to this change) will be flooded).**
- **A LSA has a default lifetime of 30 minutes. Any unchanged LSAs must be reflooded in every 30 minutes.**

Please pay attention that the neighboring routers are the routers that have interfaces in common network. Adjacency is a relationship formed between neighboring routers for the purpose of exchanging routing information. Not every pair of neighboring routers becomes adjacent.

E) OSPF DR and BDR

a. Preliminaries

There is a need to understand the types of network before detailing OSPF DR (Designated router) and BDR (Backup designated router). In fact, OSPF uses different types of exchange process for different types of network, as shown in Figure 17 (a)-(d).

i. Point to point network

It is a Cisco specific network type. It connects a single pair of routers. HDLC and PPP are example of point to point network type. In this type of network:

- **All routers form full adjacencies with each other.**
- **Hello packets are sent using a multicast address 224.0.0.5**
- **No DR and BDR are required.**
- **All routers are considered as AllSPFRouters.**

The terms adjacencies, DR, BDR and AllSPFRouters will be explained shortly.

ii. Broadcast Networks

Broadcast networks are capable in connecting more than two devices. Ethernet and FDDI are the example of broadcast type network. In this type of network:

- **A single transmitted packet can be received by all attached devices.**
- **DR and BDR are required.**
- **All routers form full adjacencies only with DR and BDR.**
- **Routers use a multicast address 224.0.0.6 to update the DR.**
- **DR uses a multicast address 224.0.0.5 to update the all routers.**

iii. NMBA

Non-broadcast Multi-access networks are also capable in connecting more than two devices but they do not have broadcast capability. X.25 and Frame Relay are the example of NMBA type network. In this type of network:

- **As network does not have broadcast capability, dynamic network discovery will not be possible.**
- **OSPF neighbors must have to define statically (relies on manual neighbor setup).**
- **All OSFP packets are unicast (as NBMA does not support multicast either).**
- **DR and BDR are required.**

iv. Point to multipoint

Point to multipoint is a special implementation of NMBA network where networks are configured as a collection of point to point links. In other words, rather than trying to emulate the broadcast capability, it seeks to organize the Permanent Virtual Circuit (PVCs) into a collection of point-to-point networks. In this type of network:

- **All routers attached to this kind of non-broadcast network must be manually configured to recognize it as a point-to-multipoint segment.**
- **No DR and BRD are selected in this type of network.**
- **OSPF packets are multicast.**

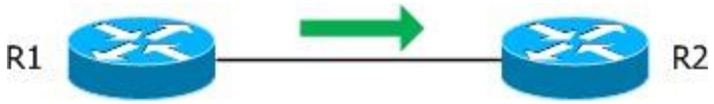


Figure 17(a). OSPF - Network Types - Point-to-Point

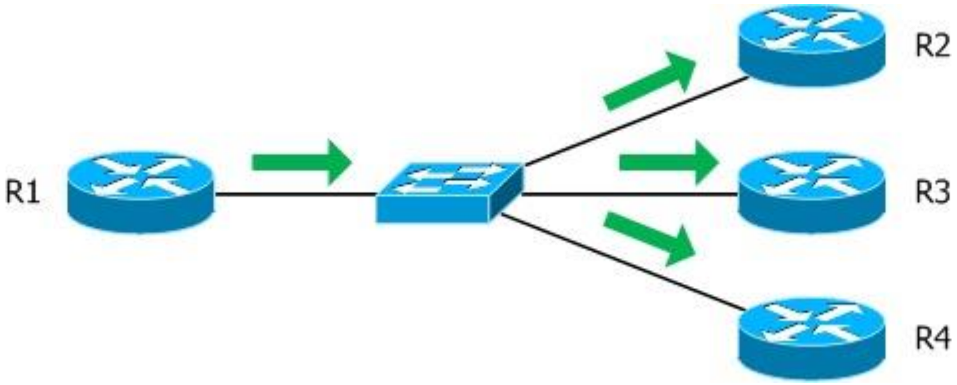


Figure 17(b). OSPF - Network Types - Broadcast

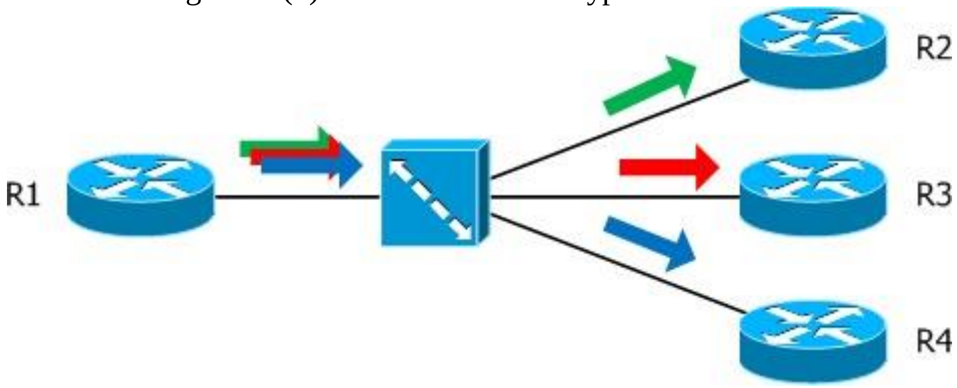


Figure 17(c). OSPF - Network Types - NBMA

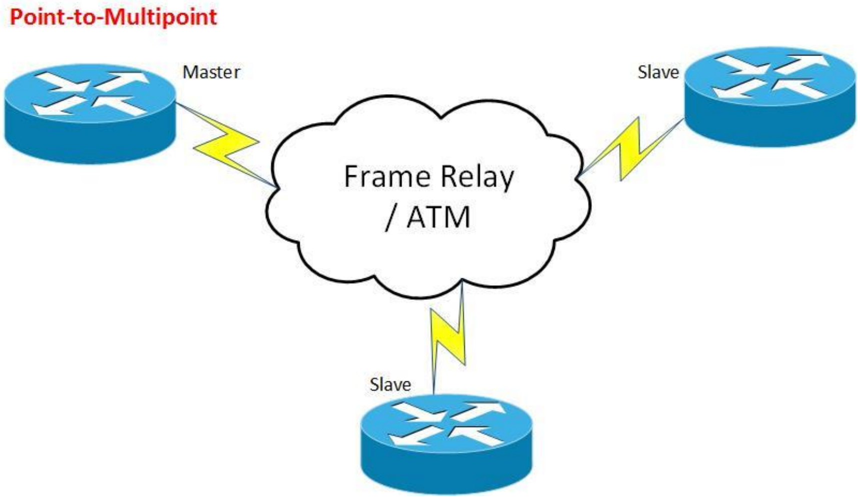


Figure 17(d). OSPF - Network Types – Point-to-Multipoint

The bottom-line, we can divide these networks in two types:

1. **Networks which need DR and BDR such as broadcast and NBMA**
2. **Networks which do not need DR and BDR such as point to point and point to multipoint**

Questions? What does DR and BDR actually do? Why do we need them in our network?

b. DR and BDR (Adjacent is Neighbor while Neighbor is not Adjacent)

OSPF routers in a network, which need DR and BDR, do not share routing information directly with all each other's. To minimize the routing information exchange, they select one router as designated router (DR) and one other router as backup designated router (BDR). Remaining routers are known as DROTHERs.

All DROTHERs share routing information with DR. DR will share this information back to all DROTHERs. BDR is a backup router. In case DR is down, BDR will immediately take place the DR and would elect new BDR for itself. Main reason behind this mechanism is that routers have a central point for routing information exchange. Thus they need not to update each other's. A DROTHER only need to update the central point (DR) and other DROTHERs will receive this update from DR. Practically this will cut the numbers of routing information exchange from *O(n*n)* to *O(n)* where n is the number of routers in a multi-access segment.

For example, Figure 18 illustrates a simple OSPF network. In this network R4 is selected as DR and R5 is selected as BDR. DROTHERs (R1, R2 and R3) will share routing information with R4 (DR) and R5 (BDR), but they will not share routing information with each other. Later DR will share this information back to all DROTHERs.

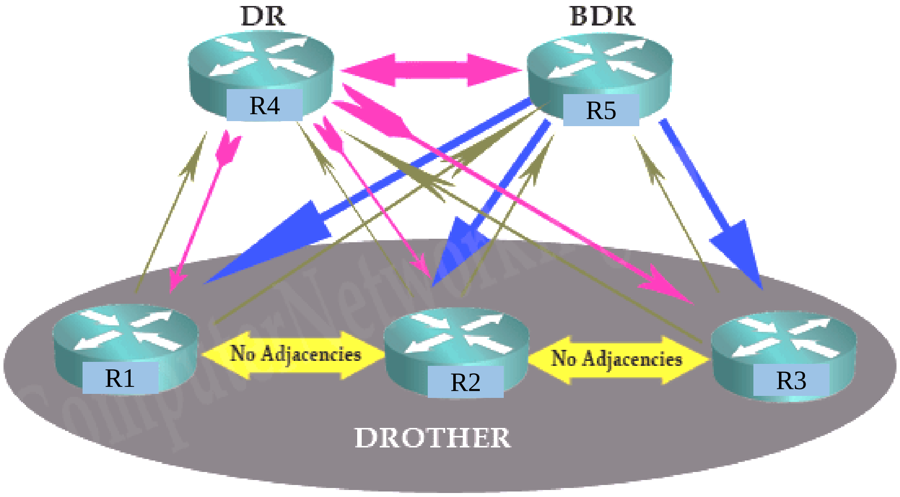


Figure 18. Simple OSPF Network (DR, BDR, and DROTHERs)

c. DR and BDR Election process

OSPF uses priority value to select DR and BDR. OSPF router with the highest priority becomes DR. Router with second highest priority becomes BDR. If there is a tie, router with the highest RID will be chosen.

Priority value is 8 bit in length. Default priority value is 1. We can set any value from range 0 to 255. We can change it from Interface Sub-configuration mode with *ip ospf priority* command. Interestingly, we can force any router to become DR (Highest) or BDR

(Second highest) by changing its priority value. If we set priority value to 0, it will never become DR or BDR. For example, Figure 19 illustrates a simple OPSF network. In this network we have five routers. We do not want that R3 becomes DR or BDR. Therefore, we changed its default priority value to 0. Now, let's see how these routers select DR and BDR.

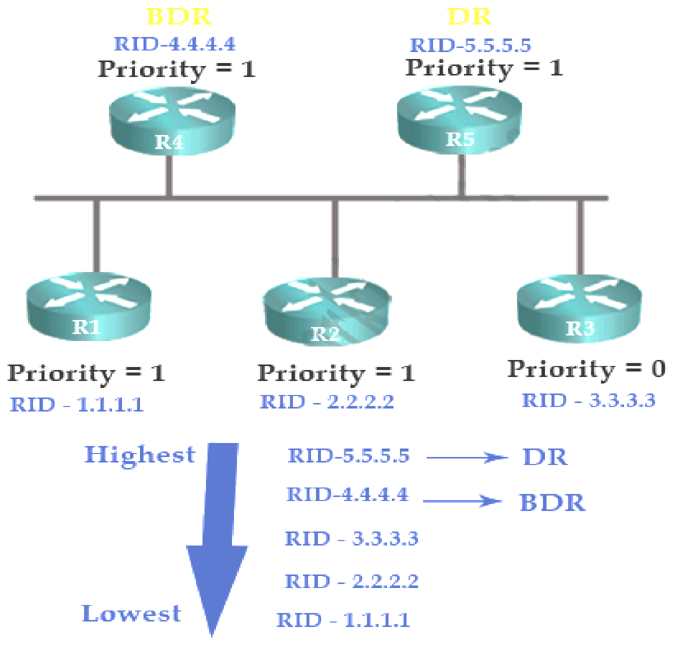


Figure 19. DR and BDR Election Process in Multi-access Networks

- **Condition 1:** ***Use the highest priority value***

This condition says "Arrange all routes in high to low order and pick the highest for DR and second highest for BDR". If we arrange our routers in high to lower order, R3 will stand at last. Remaining routers have equal priority value. Hence, at the end of this condition, we have a tie between four routers.

- **Condition 2:** ***If there is a tie use the highest RID***

This condition says "If there is a tie, use RID value to choose". In our network we have a tie between four routers, so our routers will use RID to elect the DR and BDR. Arranging routers in high to low order will give us the DR and BDR.

As we know that there are two types of network; networks which do not require DR and BDR for exchange process and networks which require DR and BDR for exchange process. In the first type, all routers will exchange routing information with each other's. In the second type, DROTHERs will exchange routing information with DR and BDR. Routers which will exchange routing information are known as adjacent. Relationship between two adjacent is known as adjacency. This terminology is associated with interfaces.

A router which has two interfaces can be adjacent in one interface and DROTHER in other interface. For example, Figure 20 illustrates an OSPF running NBMA network. In this

network, R3 will build adjacency with R1, so in this relationship they will be considered as Adjacent. Now, R3 will not build adjacency with R4, so in this relationship they will be considered only DROTHER.

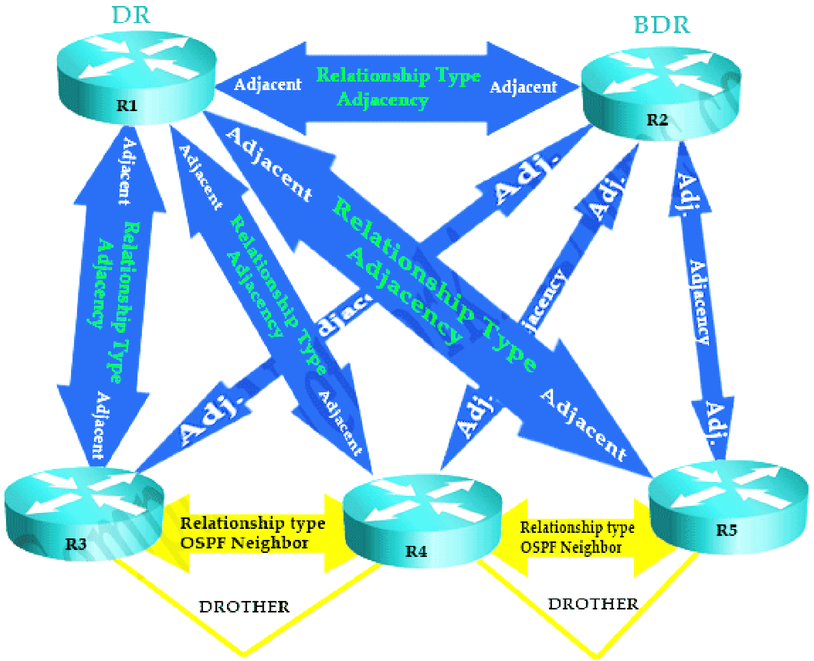
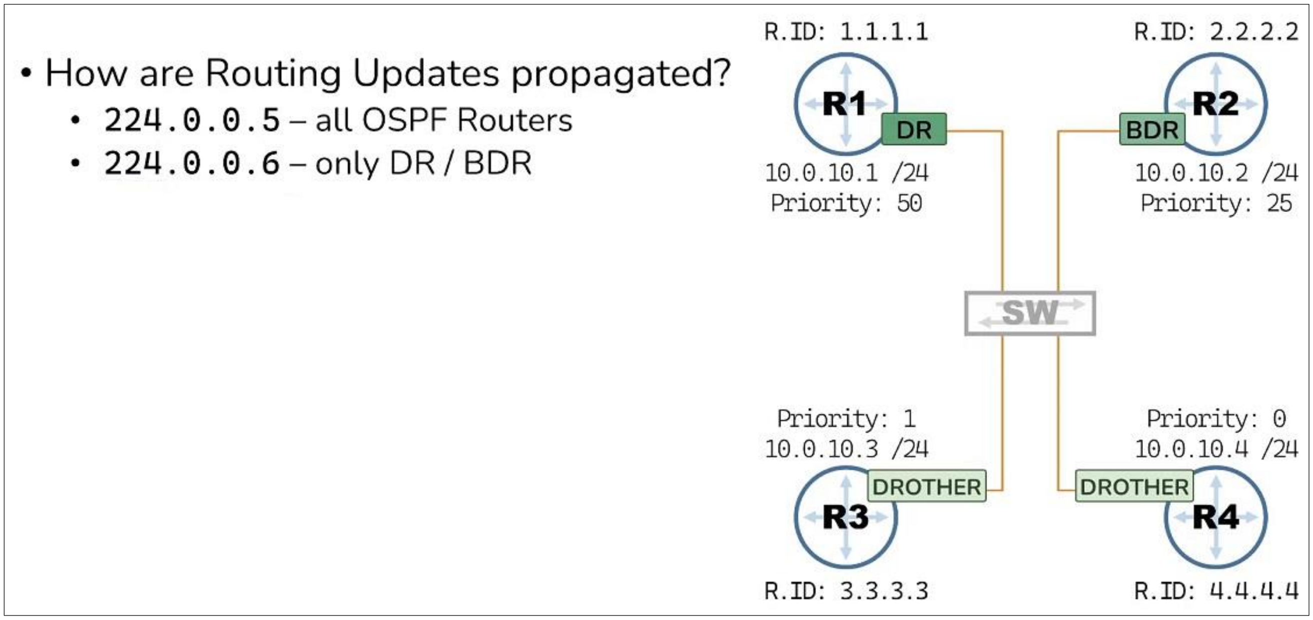


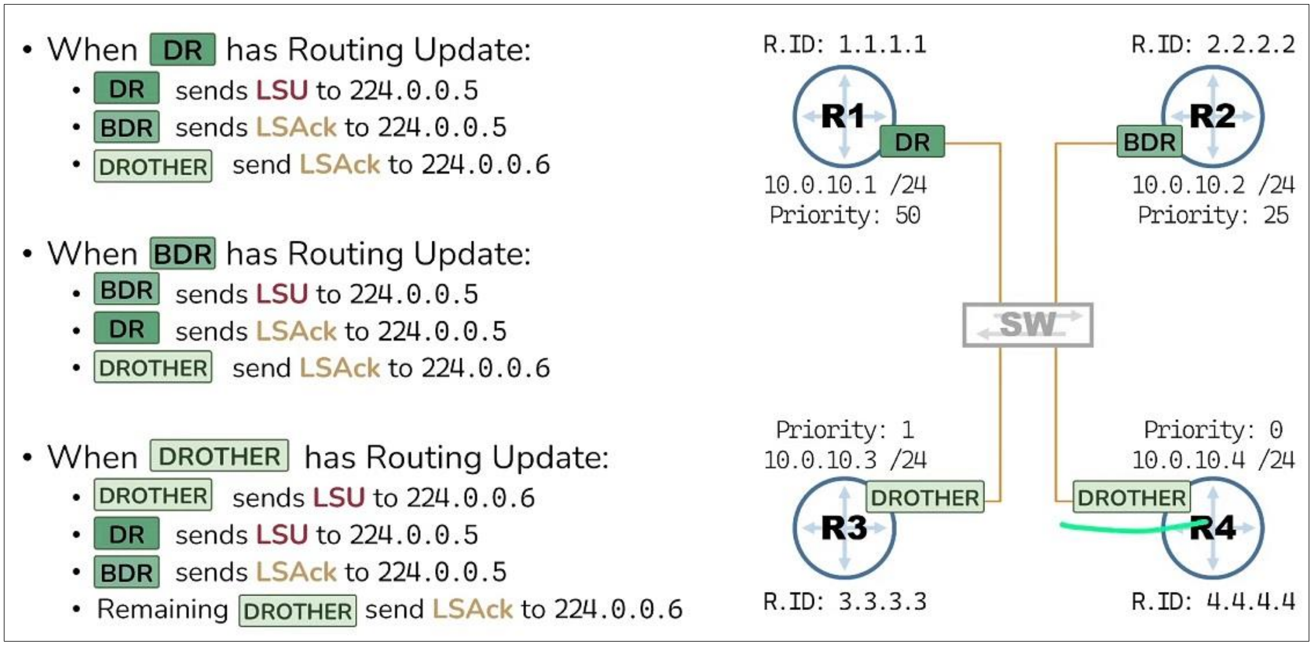
Figure 20. OSPF Running NBMA Network

In a network which doesn't require DR and BDR, all routers will be considered as Adjacent and relationship between them will be considered as Adjacency. Interestingly, only adjacent routers will enter in next states to build the adjacency. It is good to mention that Figures 21 (a) through (c) provide another descriptive scenario of using DR, BDR, and DROTHERs.



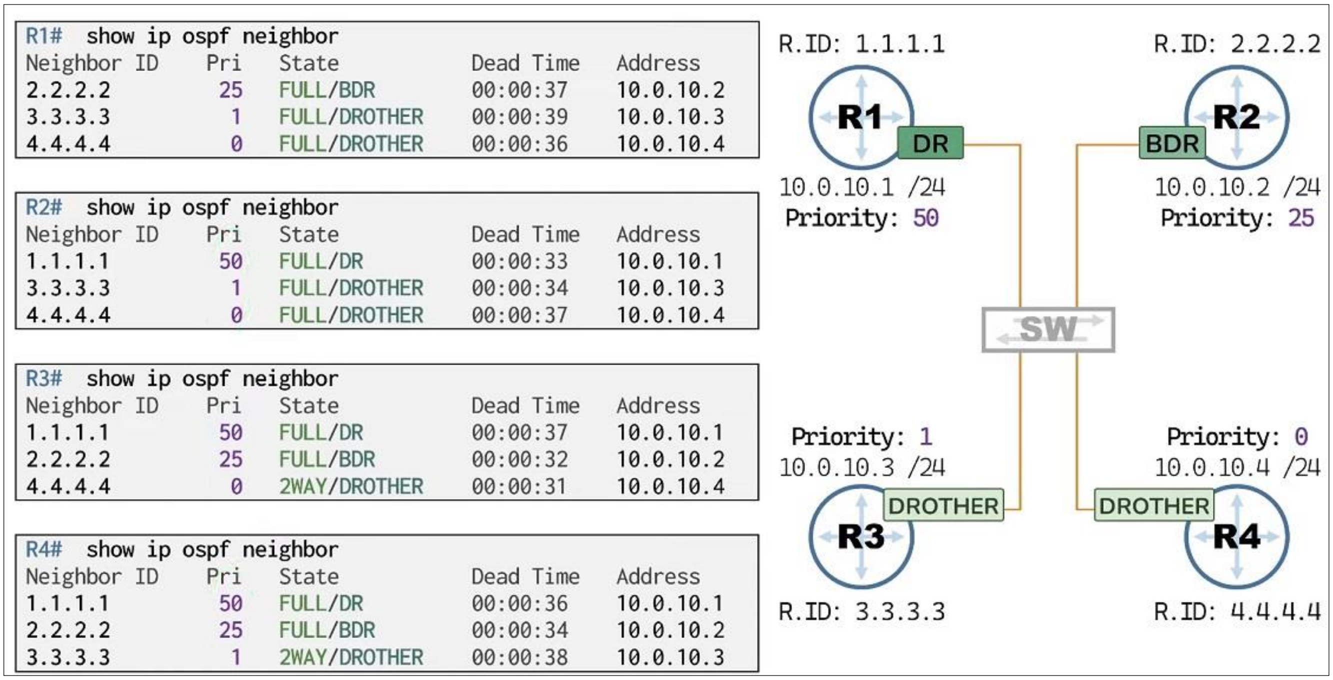
- How are Routing Updates propagated?
  - 224.0.0.5 – all OSPF Routers
  - 224.0.0.6 – only DR / BDR

Figure 21 (a). OSPF Scenario - DR, BDR, and DROTHERs



- When DR has Routing Update:
  - DR sends LSU to 224.0.0.5
  - BDR sends LSAck to 224.0.0.5
  - DROTHER send LSAck to 224.0.0.6
- When BDR has Routing Update:
  - BDR sends LSU to 224.0.0.5
  - DR sends LSAck to 224.0.0.5
  - DROTHER send LSAck to 224.0.0.6
- When DROTHER has Routing Update:
  - DROTHER sends LSU to 224.0.0.6
  - DR sends LSU to 224.0.0.5
  - BDR sends LSAck to 224.0.0.5
  - Remaining DROTHER send LSAck to 224.0.0.6

Figure 21 (b). OSPF Scenario - DR, BDR, and DROTHERs



```
R1#  show ip ospf neighbor
Neighbor ID     Pri   State           Dead Time    Address
2.2.2.2          25   FULL/BDR        00:00:37     10.0.10.2
3.3.3.3           1   FULL/DROTHER    00:00:39     10.0.10.3
4.4.4.4           0   FULL/DROTHER    00:00:36     10.0.10.4
```

```
R2#  show ip ospf neighbor
Neighbor ID     Pri   State           Dead Time    Address
1.1.1.1          50   FULL/DR         00:00:33     10.0.10.1
3.3.3.3           1   FULL/DROTHER    00:00:34     10.0.10.3
4.4.4.4           0   FULL/DROTHER    00:00:37     10.0.10.4
```

```
R3#  show ip ospf neighbor
Neighbor ID     Pri   State           Dead Time    Address
1.1.1.1          50   FULL/DR         00:00:37     10.0.10.1
2.2.2.2          25   FULL/BDR        00:00:32     10.0.10.2
4.4.4.4           0   2WAY/DROTHER    00:00:31     10.0.10.4
```

```
R4#  show ip ospf neighbor
Neighbor ID     Pri   State           Dead Time    Address
1.1.1.1          50   FULL/DR         00:00:36     10.0.10.1
2.2.2.2          25   FULL/BDR        00:00:34     10.0.10.2
3.3.3.3           1   2WAY/DROTHER    00:00:38     10.0.10.3
```

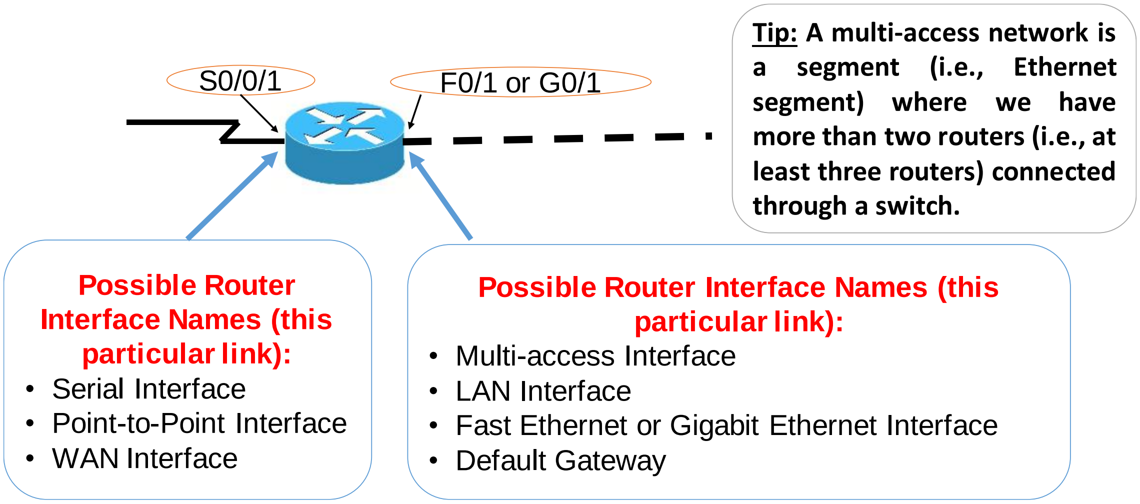
Figure 21 (c). OSPF Scenario - DR, BDR, and DROTHERs

d. DR and BDR – Multi-access Network

It is worth mentioning that OSPF uses a DR and BDR on each multi-access network. **A multi-access network is a segment** (i.e., **Ethernet segment**) **where we have more than two routers. OSPF figures this out by looking at the interface type. For example, an Ethernet interface is**

considered a multi-access interface, and a serial interface is considered a point-to-point interface, as shown in Figure 21 (d).

Tip: Something you need to be aware of is that the DR/BDR election is per multi-access segment…not per area!)



Tip: **A multi-access network is a segment (i.e., Ethernet segment) where we have more than two routers (i.e., at least three routers) connected through a switch.**

Possible Router Interface Names (this particular link):

- Serial Interface
- Point-to-Point Interface
- WAN Interface

Possible Router Interface Names (this particular link):

- Multi-access Interface
- LAN Interface
- Fast Ethernet or Gigabit Ethernet Interface
- Default Gateway

Figure 21 (d). Router Interface Types

Figure 21 (e) shows an example of a network with 3 OSPF routers on a FastEthernet network. **They are connected to the same switch (i.e., multi-access network)** so there will be a DR/BDR election. OSPF has been configured, so all routers have become OSPF neighbors.

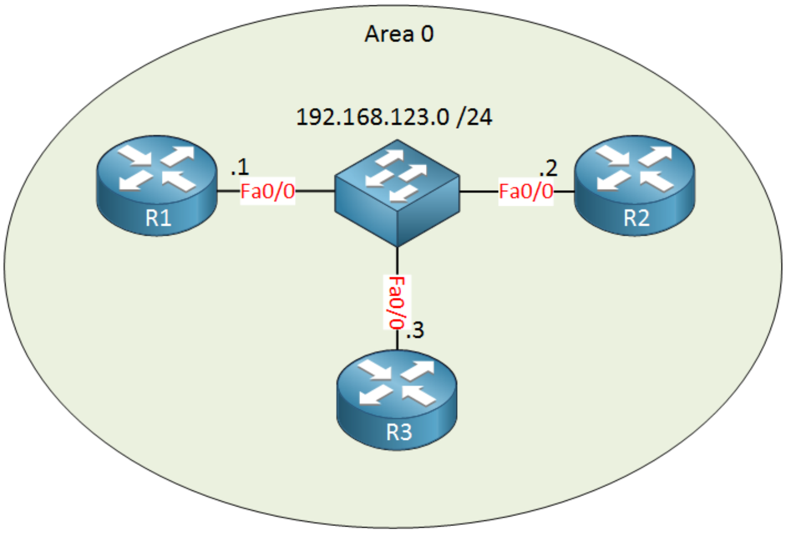


Figure 21 (e). Multi-access Network - Example 1

Interestingly, from R1's perspective, R2 is the BDR, and R3 is the DR. When a router is not the DR or BDR, it's called a **DROTHER**. Referring to the following neighbor tables, we can see that R1 is a DROTHER. Besides, R2 (the BDR) sees the DR and DROTHER.

```
R1#show ip ospf neighbor
Neighbor ID      Pri   State          Dead Time   Address         Interface
192.168.123.2    1    FULL/BDR        00:00:32    192.168.123.2 FastEthernet0/0
192.168.123.3    1    FULL/DR         00:00:31    192.168.123.3 FastEthernet0/0
```

```
R2#show ip ospf neighbor
Neighbor ID      Pri   State          Dead Time   Address           Interface
192.168.123.1    1    FULL/DROTHER    00:00:31    192.168.123.1 FastEthernet0/0
192.168.123.3    1    FULL/DR         00:00:32    192.168.123.3 FastEthernet0/0
```

```
R3#show ip ospf neighbor
Neighbor ID      Pri   State          Dead Time   Address           Interface
192.168.123.1    1    FULL/DROTHER    00:00:36    192.168.123.1 FastEthernet0/0
192.168.123.2    1    FULL/BDR        00:00:39    192.168.123.2 FastEthernet0/0
```

In Figure 21 (f), we have two multi-access segments. Between R2 and R1, and between R2 and R3. For each segment, there will be a DR/BDR election.

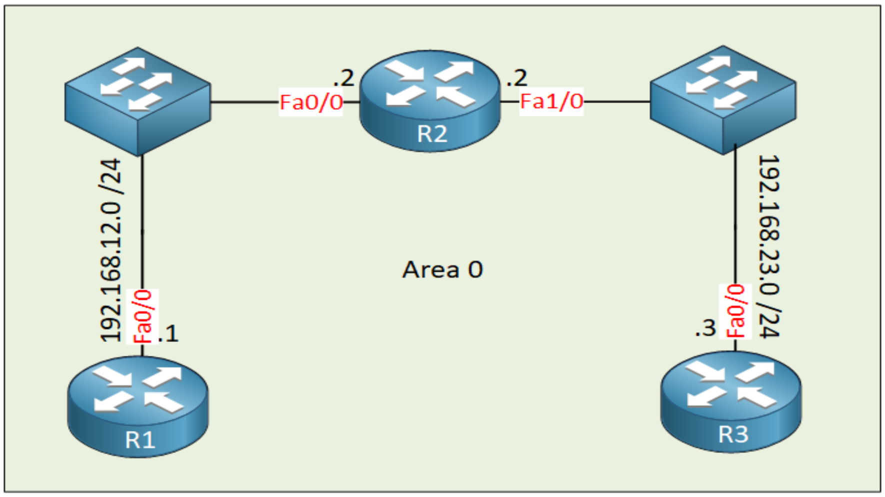
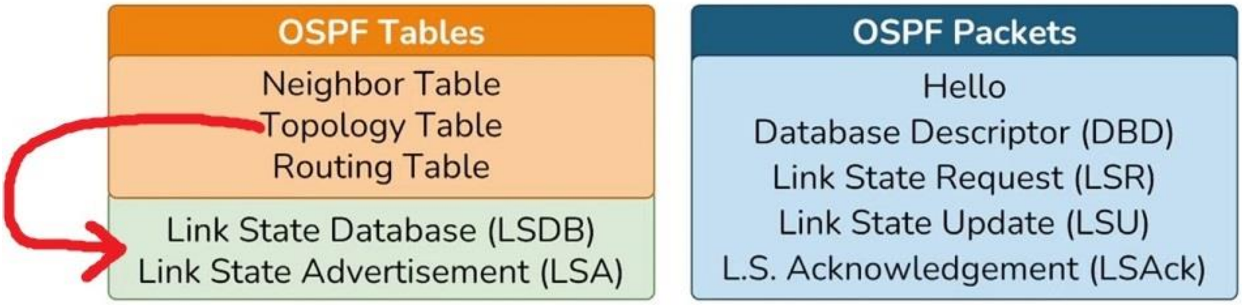


Figure 21 (f). Multi-access Network - Example 2

It is good to mention that you change the priority if you like by using the **ip ospf priority** command taking into account that the default priority is "1" and a priority of 0 means you will never be elected as DR or BDR.

F) OSPF Tables and Packets – Summarized

The following figures (Figures 22 through 28) summarize descriptively all important information about the OSPF tables and packets taking into account that the router ID is symmetric and assumed to match the router order. For example, the router IDs of R3 and R4 are 3.3.3.3 and 4.4.4.4, respectively.



OSPF Tables

Neighbor Table

Topology Table

Routing Table

Link State Database (LSDB)

Link State Advertisement (LSA)

OSPF Packets

Hello

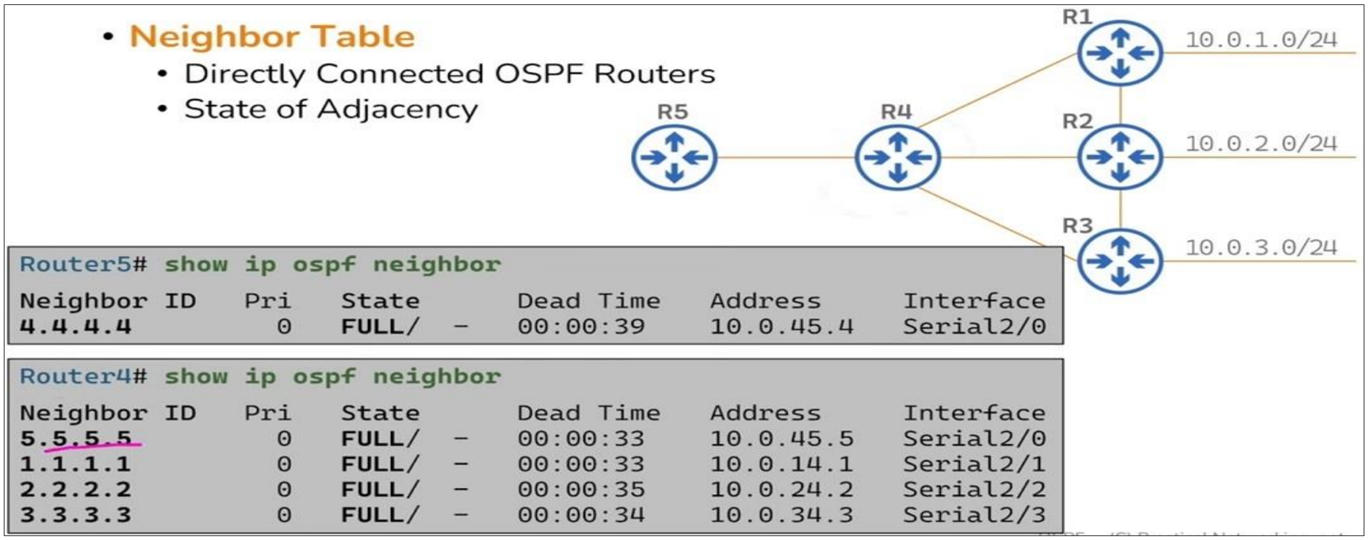
Database Descriptor (DBD)

Link State Request (LSR)

Link State Update (LSU)

L.S. Acknowledgement (LSAck)

Figure 22. OSPF Tables and Packets – Summary

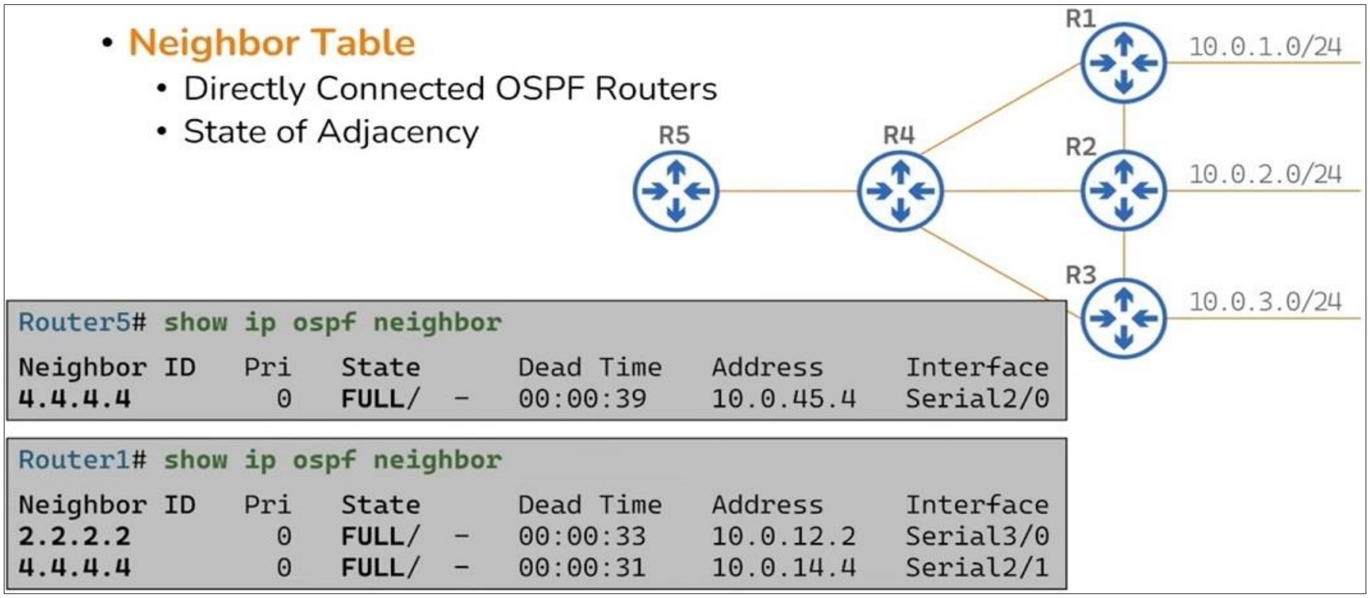


- **Neighbor Table**
  - Directly Connected OSPF Routers
  - State of Adjacency

```
Router5# show ip ospf neighbor
Neighbor ID     Pri   State         Dead Time   Address        Interface
4.4.4.4           0   FULL/  -      00:00:39    10.0.45.4      Serial2/0
```

```
Router4# show ip ospf neighbor
Neighbor ID     Pri   State         Dead Time   Address        Interface
5.5.5.5           0   FULL/  -      00:00:33    10.0.45.5      Serial2/0
1.1.1.1           0   FULL/  -      00:00:33    10.0.14.1      Serial2/1
2.2.2.2           0   FULL/  -      00:00:35    10.0.24.2      Serial2/2
3.3.3.3           0   FULL/  -      00:00:34    10.0.34.3      Serial2/3
```

Figure 23. OSPF Tables – Summary 1

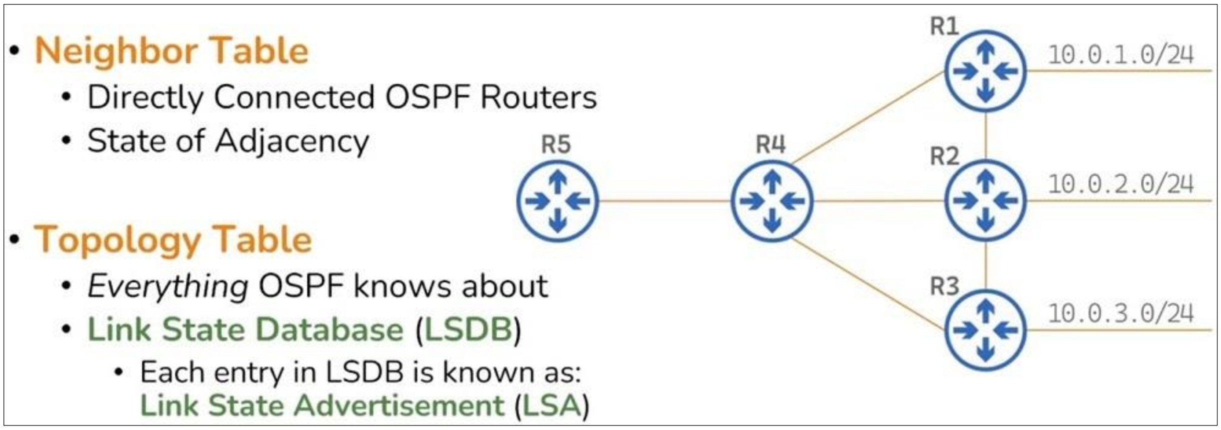


- **Neighbor Table**
  - Directly Connected OSPF Routers
  - State of Adjacency

```
Router5# show ip ospf neighbor
Neighbor ID     Pri   State         Dead Time   Address        Interface
4.4.4.4           0   FULL/  -      00:00:39    10.0.45.4      Serial2/0
```

```
Router1# show ip ospf neighbor
Neighbor ID     Pri   State         Dead Time   Address        Interface
2.2.2.2           0   FULL/  -      00:00:33    10.0.12.2      Serial3/0
4.4.4.4           0   FULL/  -      00:00:31    10.0.14.4      Serial2/1
```

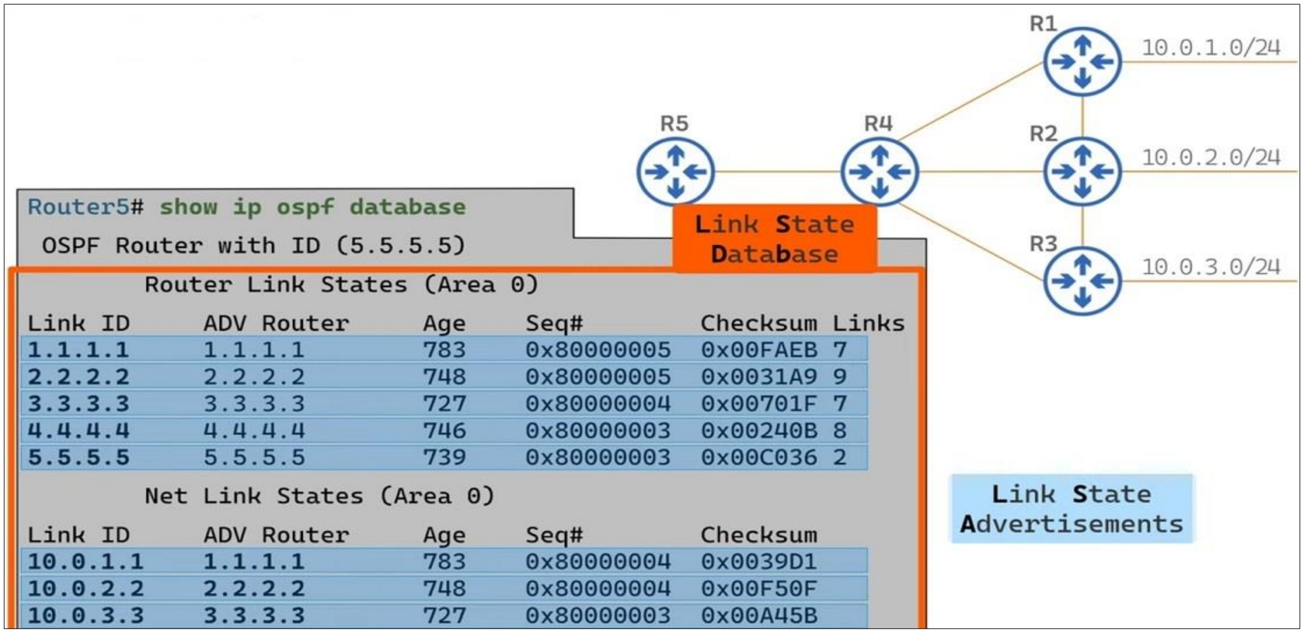
Figure 24. OSPF Tables – Summary 2



- Neighbor Table
  - Directly Connected OSPF Routers
  - State of Adjacency
- Topology Table
  - *Everything* OSPF knows about
  - Link State Database (LSDB)
    - Each entry in LSDB is known as: Link State Advertisement (LSA)

Figure 25. OSPF Tables – Summary 3

Figure 29 (a) shows a portion of R4 LSDB where each entry describes a certain LSA. Additionally, the LSA header is shown in Figure 29 (b) along with necessary discussion of its fields.



```
Router5# show ip ospf database
 OSPF Router with ID (5.5.5.5)
          Router Link States (Area 0)
Link ID      ADV Router     Age    Seq#         Checksum Links
1.1.1.1      1.1.1.1        783    0x80000005   0x00FAEB 7
2.2.2.2      2.2.2.2        748    0x80000005   0x0031A9 9
3.3.3.3      3.3.3.3        727    0x80000004   0x00701F 7
4.4.4.4      4.4.4.4        746    0x80000003   0x00240B 8
5.5.5.5      5.5.5.5        739    0x80000003   0x00C036 2
          Net Link States (Area 0)
Link ID      ADV Router     Age    Seq#         Checksum
10.0.1.1     1.1.1.1        783    0x80000004   0x0039D1
10.0.2.2     2.2.2.2        748    0x80000004   0x00F50F
10.0.3.3     3.3.3.3        727    0x80000003   0x00A45B
```

Figure 26 (a). OSPF Tables – Summary 4

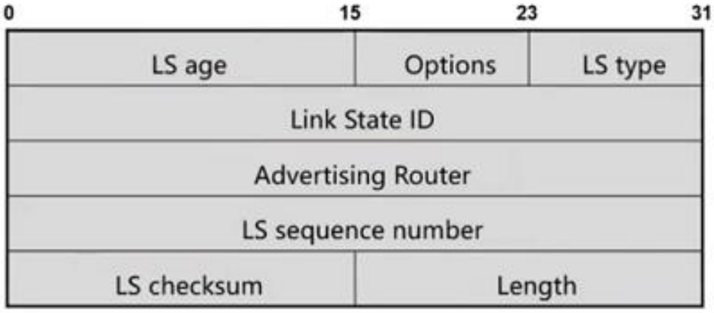
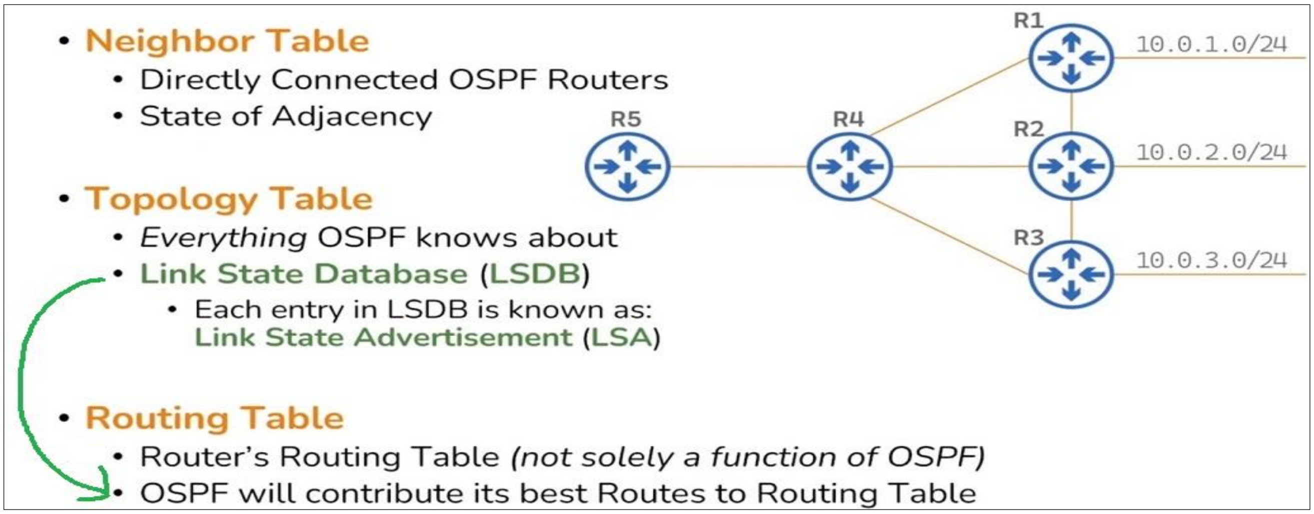


Figure 26 (b). LSA Packet Header

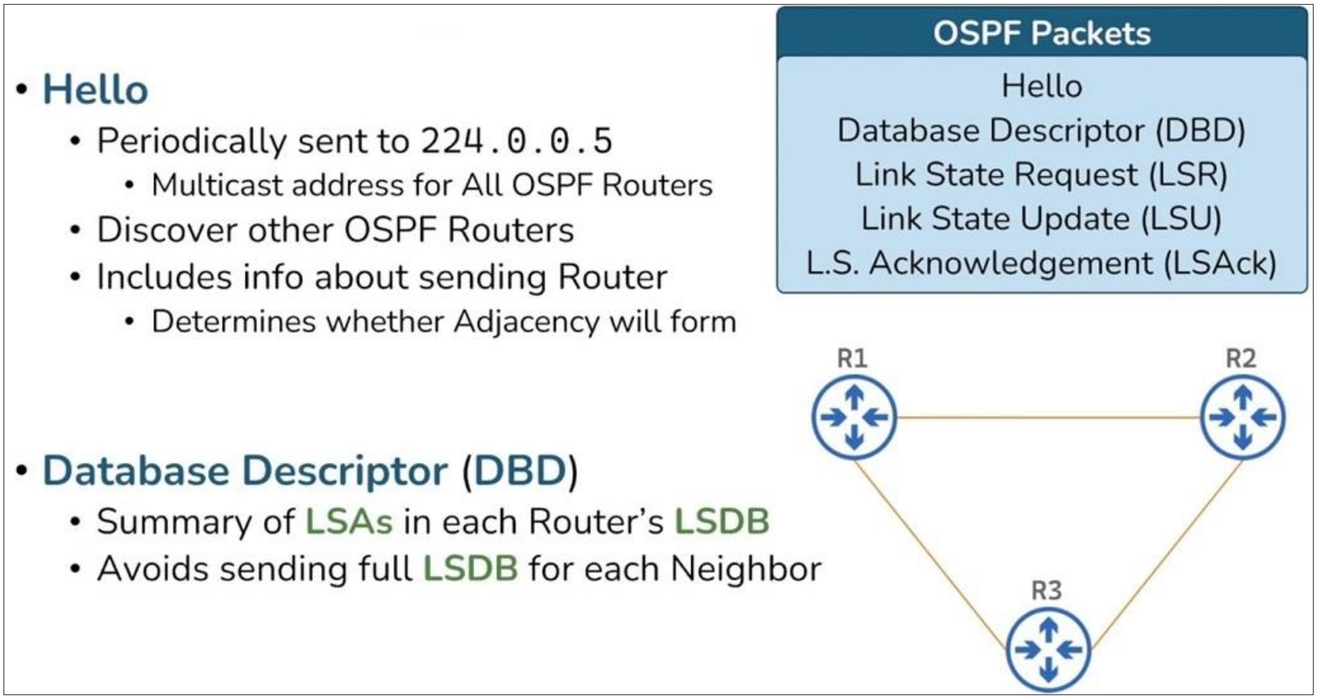
The LSA head fields are described as follows:

1) LS Age, is the number of seconds that have elapsed since the LSA was originated.
2) OSPF Options, a series of flags that identify the various optional services that an OSPF network can support.
3) LS Type, identifies which of the possible types the LSA contains.
4) Link State ID, identifies the specific portion of the network environment that the LSA describes. In different words, it describes the address of the interface generating this LSA.
5) Advertising Router, is the router ID that originated this LSA.
6) LS Sequence Number, where OSPF routers increment the sequence number for each LSA generated. Therefore, a router that receives two instances of the same LSA has two options for determining which of the two is the most recent (i.e., the LSA age or LS sequence number). The question still remains. Which one the router should rely on? It is known that the LSA Age field can be checked to determine how long the LSA has been traversing the network. It is theoretically possible for a newer LSA to have a greater LSA age than an older LSA, particularly in large and complex OSPF networks. Therefore, recipient routers must compare the LS sequence number. The higher number was the most recently generated.
7) LS Checksum, used to detect if there is a damage in LSAs during the transmissions to their destinations. Checksums are simple mathematical algorithms. The source node runs an algorithm known as the Fletcher Algorithm and stores the results in the LS Checksum field. The destination node performs the same mathematical exercise and compares its result to the result stored in the Checksum field. If the values are different, it is relatively safe to assume that damage has occurred in transit. Consequently, a retransmission request is generated.
8) LS Length—Predictably, the LS Length field informs the recipient of the LSA's length, in octets including LSA header.



- Neighbor Table
  - Directly Connected OSPF Routers
  - State of Adjacency
- Topology Table
  - *Everything* OSPF knows about
  - Link State Database (LSDB)
    - Each entry in LSDB is known as: Link State Advertisement (LSA)
- Routing Table
  - Router's Routing Table *(not solely a function of OSPF)*
  - OSPF will contribute its best Routes to Routing Table

Figure 27. OSPF Tables – Summary 5

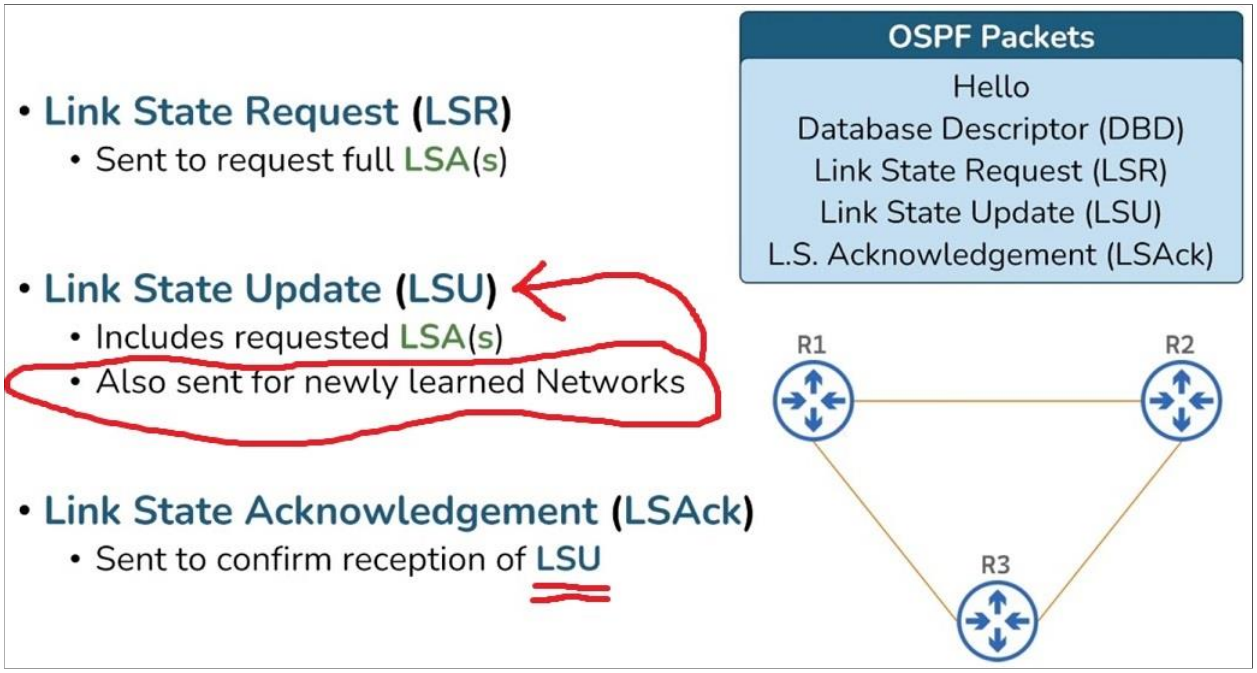


- **Hello**
  - Periodically sent to 224.0.0.5
    - Multicast address for All OSPF Routers
  - Discover other OSPF Routers
  - Includes info about sending Router
    - Determines whether Adjacency will form
- **Database Descriptor (DBD)**
  - Summary of LSAs in each Router's LSDB
  - Avoids sending full LSDB for each Neighbor

OSPF Packets

Hello
Database Descriptor (DBD)
Link State Request (LSR)
Link State Update (LSU)
L.S. Acknowledgement (LSAck)

Figure 28. OSPF Packets – Summary 1



- **Link State Request (LSR)**
  - Sent to request full LSA(s)
- **Link State Update (LSU)**
  - Includes requested LSA(s)
  - Also sent for newly learned Networks
- **Link State Acknowledgement (LSAck)**
  - Sent to confirm reception of LSU

OSPF Packets

Hello
Database Descriptor (DBD)
Link State Request (LSR)
Link State Update (LSU)
L.S. Acknowledgement (LSAck)

Figure 29. OSPF Packets – Summary 2

G) OSPF Areas

a. Preliminaries

OSPF uses the concept of areas. An area is a **logical** grouping of contiguous networks and routers. All routers in the same area have the same topology table, but they don't know about routers in the other areas. **The main benefits of creating areas is that the size of the topology and the routing table on a router is reduced, less time is required to run the SPF algorithm and routing updates are also reduced.**

Each area in the OSPF network has to connect to the backbone area (area 0). In other words, the area 0 is mandatory in OSPF. All router inside an area must have the same area ID to become OSPF neighbors. A router that has interfaces in more than one area (area 0 and area 1, for example) is called **Area Border Router (ABR)**. A router that connects an OSPF network to other routing domains (EIGRP network, for example) is called **Autonomous System Border Router (ASBR)**. More details about OSPF areas and routers will be discussed shortly. To better understand the concept of areas, consider the following example, which is illustrated in Figure 30. Particularly, all routers are running OSPF. Routers R1 and R2 are inside the backbone area (area 0). Router R3 is an ABR, because it has interfaces in two areas, namely area 0 and area 1. Router R4 and R5 are inside area 1. Router R6 is an ASBR, because it connects OSPF network to another routing domain (an EIGRP domain in this case). If the R1's directly connected subnet fails, router R1 sends the routing update only to R2 and R3, because all routing updates all localized inside the area.

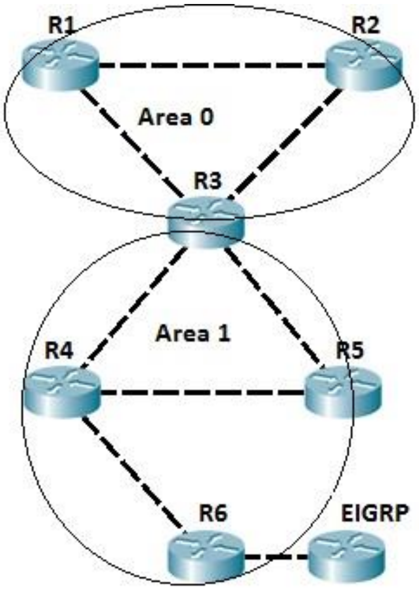


Figure 30. OSPF Areas

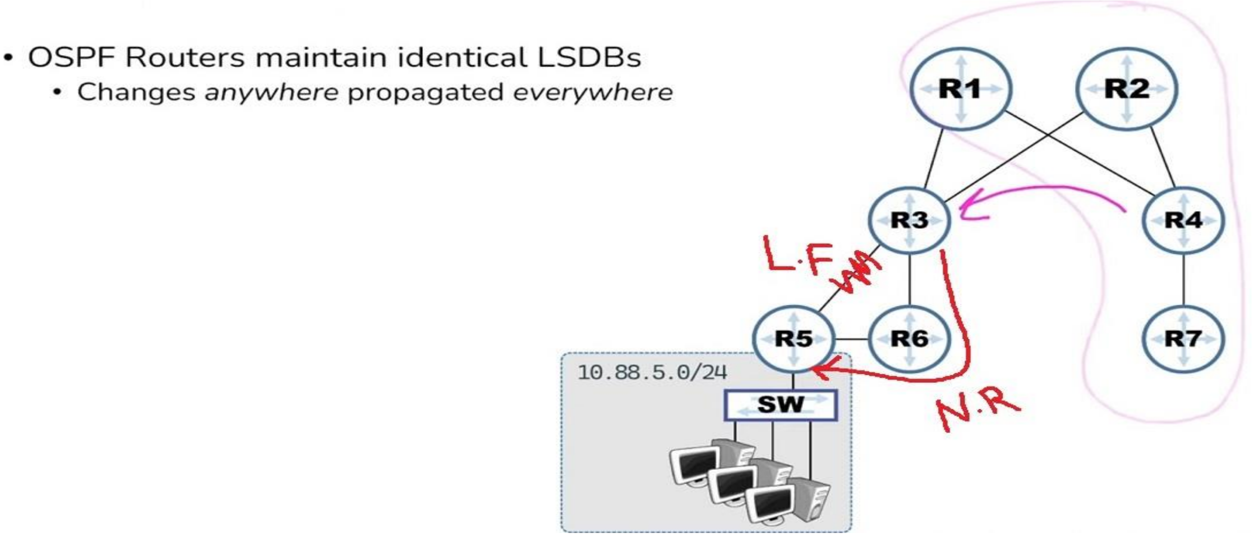
b. OSPF Areas and Router Types

In this section, we will provide illustrative example along with detailed figures (Figures 31 through 33) which make the need of having areas in OPSF so clear. Relevantly, we will describe all types of routers used in OSPF.

Kindly consider the topology, shown in Figure 31 and particularly look to the position of R3. Imagine there is a link failure (or down) between R5 and R3, which was the selected route to reach 10.88.5.0 network. Since R3 knows about the entire topology, it already knows of an

alternate path to get to the 10.88.5.0 network, which is through R6. But, since something changed in the network router 3, R3 will tell R1 and R2 and likewise R2 will tell R4 and R4 will tell R7 (that something has changed). This is going to force all other routers in this topology to recalculate the network map so that they all have an updated version of the network map, which includes that this link is down. If this link backs up, R3 will again start to use that link but again we will have to update every single router in this topology to let them know that something has changed.

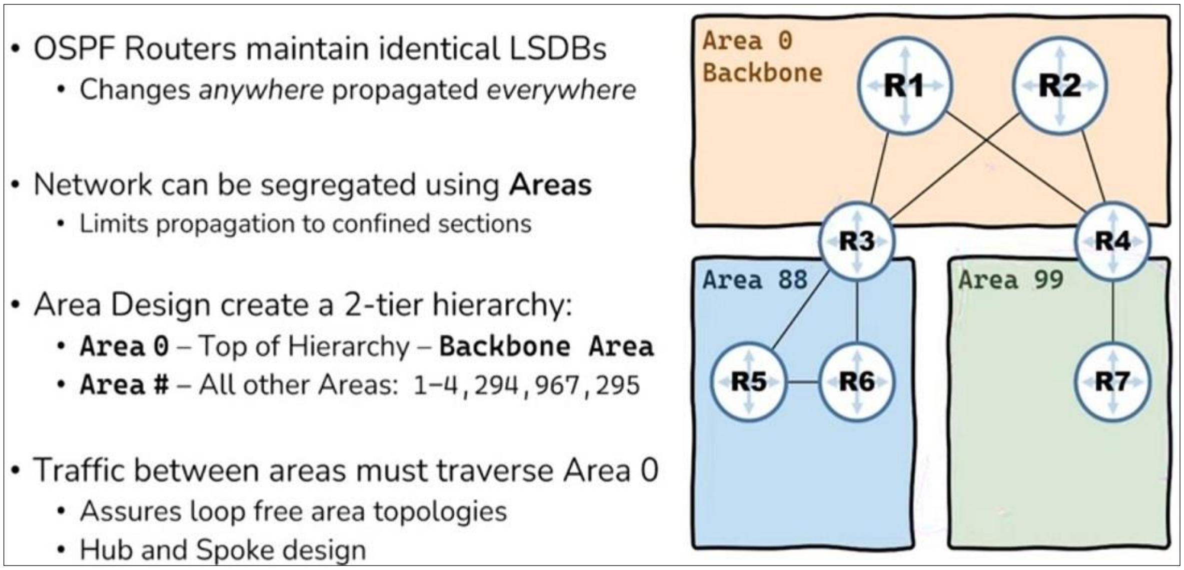
It is worth mentioning that in order that R1, R3, R4, and R7 get into 10.88.5.0 network, they should reach R3 first, which means that updating all of these routers (i.e., R1, R2, R4, and R7) every time something changes in here (i.e., links between R3 and R5 or R3 and R6) is completely redundant and unnecessary. That is where areas come into play. Areas allow you to segregate your network for the sake of limiting LSA propagation to confined sections. Now, R3 might change the path that is using to get to the 10.88.5.0 but since R3 knows it does not affect any of the other routers in the topology, therefore, it is not going to propagate every single change that happens in this area. That is the basic idea behind the areas.



- OSPF Routers maintain identical LSDBs
  - Changes *anywhere* propagated *everywhere*

Figure 31. The Need for OSPF Areas

Kindly consider the areas division, shown in Figure 32. As a result, the communication from area to area must traverse through area 0. In other words, in order for the routers in area 88 to speak with those at area 99, that communication must flow through the backbone area (area 0). The purpose of this is that it assures a loop free area topology. Imagine the case where there is a router, that located down, which connects R6 and R7, you will then have a potential loop because of the routers' advertisements between these areas (due to circular communications). With this OPSF rule, all traffic between areas 88 and 99 must flow through the backbone area, ensuring a loop-free area topology. Note that this creates a hub and spoke design in OSPF areas (or sometimes, we refer it to a star topology). In different words, all of these areas are going to be connected into the backbone area (we call it hup) and each of these areas access a different spoke.

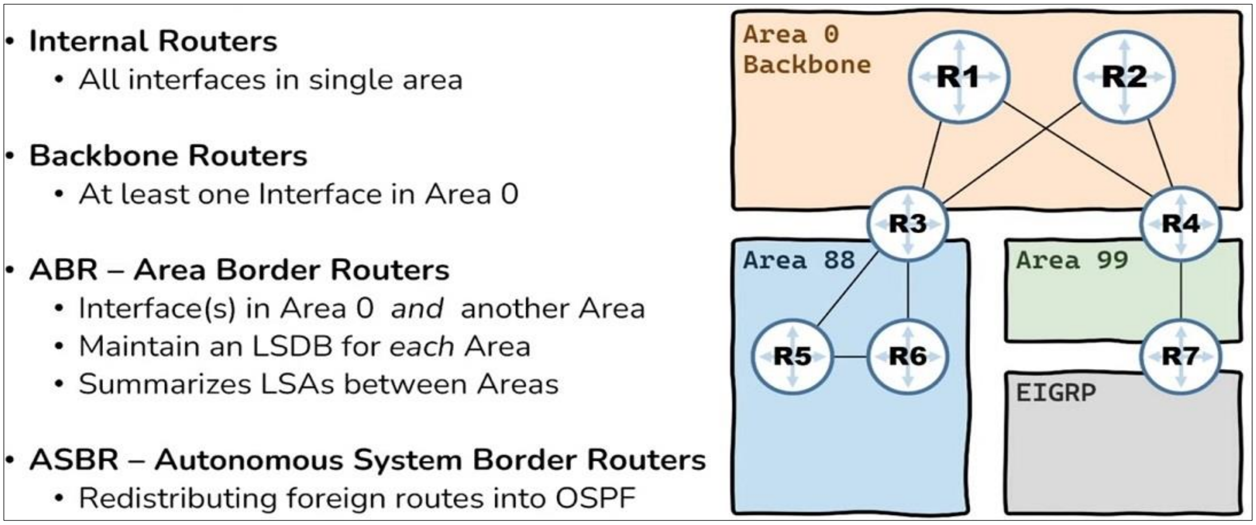


- OSPF Routers maintain identical LSDBs
  - Changes *anywhere* propagated *everywhere*
- Network can be segregated using **Areas**
  - Limits propagation to confined sections
- Area Design create a 2-tier hierarchy:
  - **Area 0** – Top of Hierarchy – **Backbone Area**
  - **Area #** – All other Areas: 1-4,294,967,295
- Traffic between areas must traverse Area 0
  - Assures loop free area topologies
  - Hub and Spoke design

Figure 32. OSPF Areas – Division and Illustration

To understand and recognize the types of routers used in OSPF, kindly refer to Figure 33(a). Consequently, the internal routers are R1, R2, R5, R6, and R7. The backbone routers are R1, R2, R3, and R4. The area border routers are R3 and R4. The autonomous system border routers are R7.

Important Tip: The ABR is the router that has interfaces in multiple OSPF areas (i.e., not necessarily to have at least interface in area 0).



- **Internal Routers**
  - All interfaces in single area
- **Backbone Routers**
  - At least one Interface in Area 0
- **ABR – Area Border Routers**
  - Interface(s) in Area 0 *and* another Area
  - Maintain an LSDB for *each* Area
  - Summarizes LSAs between Areas
- **ASBR – Autonomous System Border Routers**
  - Redistributing foreign routes into OSPF

Figure 33 (a). OSPF Types of Routers

c) OSPF Areas Usage Summary – Important

In a nutshell, there are three advantages of using OSPF areas. The first one related to the capability of ABR. In particular, it is capable of summarizing all contiguous subnets of an area, it belongs to, to send just one summarized LSA packet and this can happen in both directions. The second one related also to ABR capability. As a matter of fact, it is able to limit the propagation of link state updates into a confined section. In other words, it will reduce the wide flooding of link state updates, thereby maintaining identical LSDB for just every area (i.e., not for the whole network topology) which in turn does not waste bandwidth and resources for recalculating the network map. Figure 33 (b) describes the first advantage of OSPF areas. There is a question arises, does the ABR R3 care if there is a link failure in subnet 172.16.3.0/24 taking into account that it already summarized these 8 contiguous subnets into one subnet or network 172.16.0.0/20 (not /16). Certainly, no, bearing in mind that the only way for area 2 to reach foreign areas is through this ABR (R3). The last advantage related to the areas usage. In specific, the use of OSPF areas, which is identified by having direct communications with just the backbone area (area 0) (i.e., not other areas), prevents any potential loop and assures a loop free area topology.

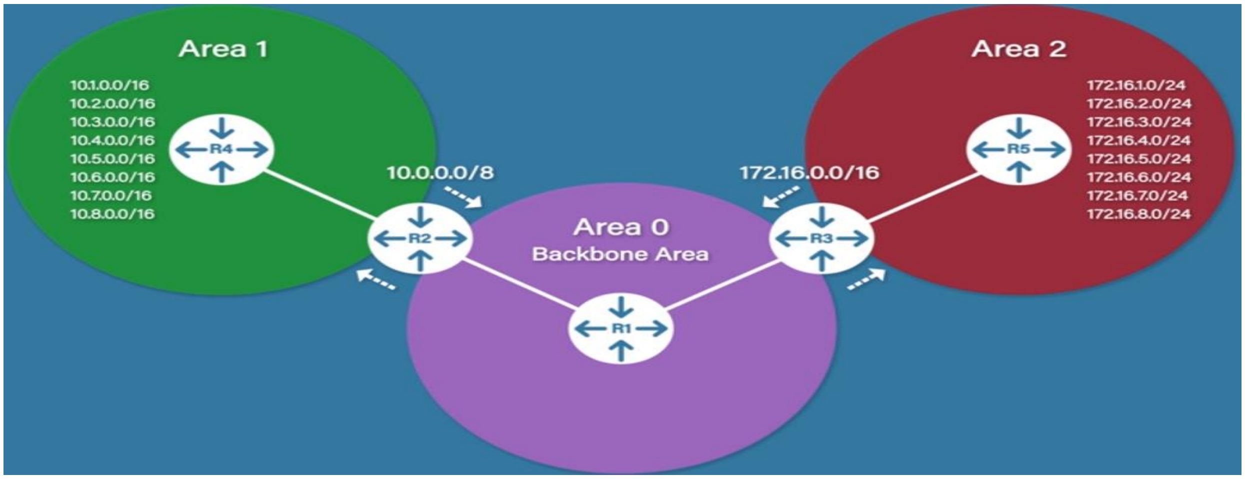


Figure 33 (b). OSPF Areas – ABR – Contiguous Subnets – Summarization

Important Tip: OSPF can do summarization, but is impossible to summarize within an area. This means we have to configure summarization on an ABR or ASBR. In other words, OSPF can only summarize LSA types 3 and 5. If we want summarization for OSPF, we will have to configure it ourselves. This means also that there is no auto-summary feature in OSPF. In different words, OSPF, unlike EIGRP, does not support automatic summarization.

d) Multiple areas and virtual links

- Each OSPF network that is divided into different areas must follow these rules:
    - A backbone area - which combines a set of independent areas into a single domain - must exist.
    - Each non-backbone area must be directly connected to the backbone area (though this connection might be a simple logical connection through a virtual link).

- **The backbone area must not be partitioned - divided into smaller pieces - under any failure conditions, such as link or router down events.**

❖ **Virtual Links:**

- **An OSPF virtual link is a connection between two ABRs. The ABR connects the isolated area to the OSPF backbone area 0 through a transit area or a non-backbone area.**
- **Virtual links are used for two purposes:**
  - ✓ **To an area that does not have a physical connection to the backbone**
  - ✓ **To patch the backbone in case discontinuity of area 0 occurs.**

❖ **Areas Not Physically Connected to Area 0:**

- **As mentioned earlier, area 0 has to be at the center of all other areas. In some rare case where it is impossible to have an area physically connected to the backbone, a virtual link is used.**
- **The virtual link provides the disconnected area a logical path to the backbone. The virtual link has to be established between two ABRs that have a common area, with one ABR connected to the backbone.**
- **Virtual links to connect two parts of a partitioned backbone through a non-backbone area. The area through which you configure the virtual link, known as a transit area, must have full routing information. The transit area cannot be a stub area.**
- **Once you have identified the OSPF ABRs, you need to configure a virtual link using the** `area area-id virtual-link router-id` **command, where the area-id is the area ID assigned to the transit area (this can be either a valid IP address or a decimal value), and where router-id is the router ID associated with the virtual link neighbor.**
- **The OSPF router-id is usually the highest IP address on the box, or the highest loopback address if one exists.**
- **The router-id is only calculated at boot time or anytime the OSPF process is restarted. To find the router-id, use the** `show ip ospf interface` **command.**

❖ **Figure 33 (c) shows an example, where the virtual link connects area 7 to the backbone through area 5, In this example, the virtual link is created between the routers with router ID 1.1.1.1 and router ID 2.2.2.2. In order to create the virtual link, configure the area 5 virtual-link 2.2.2.2 subcommand on router 1.1.1.1 and the area 5 virtual-link 1.1.1.1 subcommand on router 2.2.2.2.**

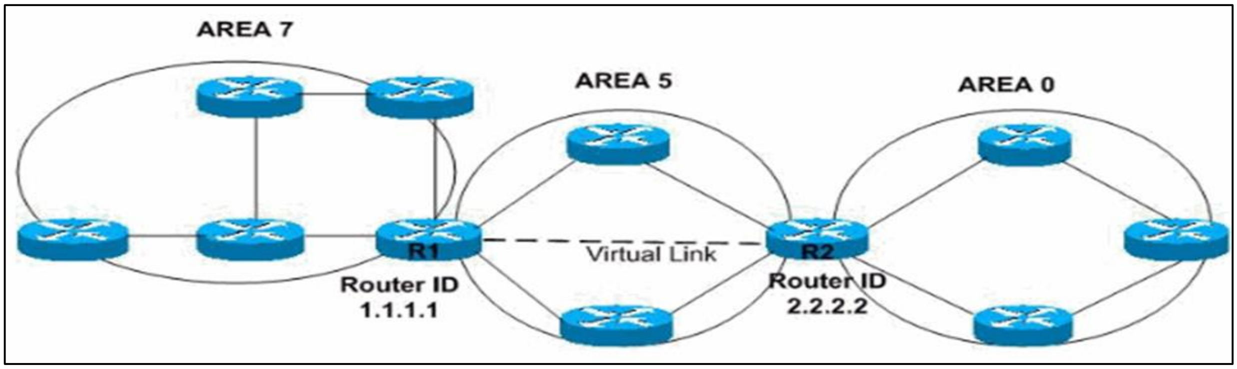


Figure 33 (c): Virtual link connects area 7 to the backbone through area 5

- **In a nutshell, through OSPF virtual link, you connect an isolated OSPF area to the OSPF backbone area 0, and you must configure Virtual links on ABRs.**
- **In summary, there are three main steps:**
  - **Step 1: Identify an isolated area that needs to connect to area 0.**
  - **Step 2: Identify appropriate ABRs and make sure the above requirements are met.**
  - **Step 3: Configure under the OSPF process using the above command on both ABRs.**

H) OSPF Metric Cost Calculation

In this part, we will show how SPF (Shortest Path First) algorithm calculates cumulative cost of route to build the Shortest Path Tree (SPT) and how OSPF Metric Formula can be manipulated by changing reference bandwidth value.

a. Shortest Path First (SPF) Algorithm

As we know upon initialization or due to any change in routing information an OSPF router generates a LSA. This LSA contains the collection of all link-states on that router. Router propagates this LSA in network. Each router that receives this LSA would store a copy of it in its LSA database then flood this LSA to other routers. **After database is updated, router selects a single best route for each destination from all available routes. Router uses SPF algorithm to select the best route. Just like other routing algorithm, SPF also uses a metric component called cost to select the best route for routing table.**

i. OSPF Metric Cost

Logically, a packet will face more overhead in crossing a 56Kbps serial link than crossing a 100Mbps Ethernet link. Respectively, it will take less time in crossing a higher bandwidth link than a lower bandwidth link. OSPF uses this logic to calculate the cost. Cost is the inverse proportional of bandwidth (i.e., link speed). Higher bandwidth has a lower cost. Lower bandwidth has a higher cost. OSPF uses following formula to calculate the cost

$$\text{Cost} = \text{Reference bandwidth} / \text{Interface bandwidth in bps} \qquad (1)$$

Reference bandwidth was defined as arbitrary value in OSPF documentation (RFC 2338). Network vendors need to use their own reference bandwidth. Cisco uses 100Mbps (10^8) bandwidth as reference bandwidth. With this bandwidth, our equation would be:

$$\text{Cost} = 10^8/\text{interface bandwidth in bps} \qquad (2)$$

It is important to consider the following key points

- Cost is a positive integer value.
- Any decimal value would be rounded back in nearest positive integer.
- Any value below 1 would be considered as 1.

ii. SPT (Shortest Path Tree)

OSPF router builds a Shortest Path Tree. SPT is just like a family tree where router is the root and destination networks are the leaves. SPF algorithm calculates the branch cost between leaves and root. Branch with lowest cost will be used to reach at leaf. In technical language, route that has lowest cumulative cost value between source and destination will be selected for routing table.

Cumulative cost = *Sum of all outgoing interfaces cost in route*

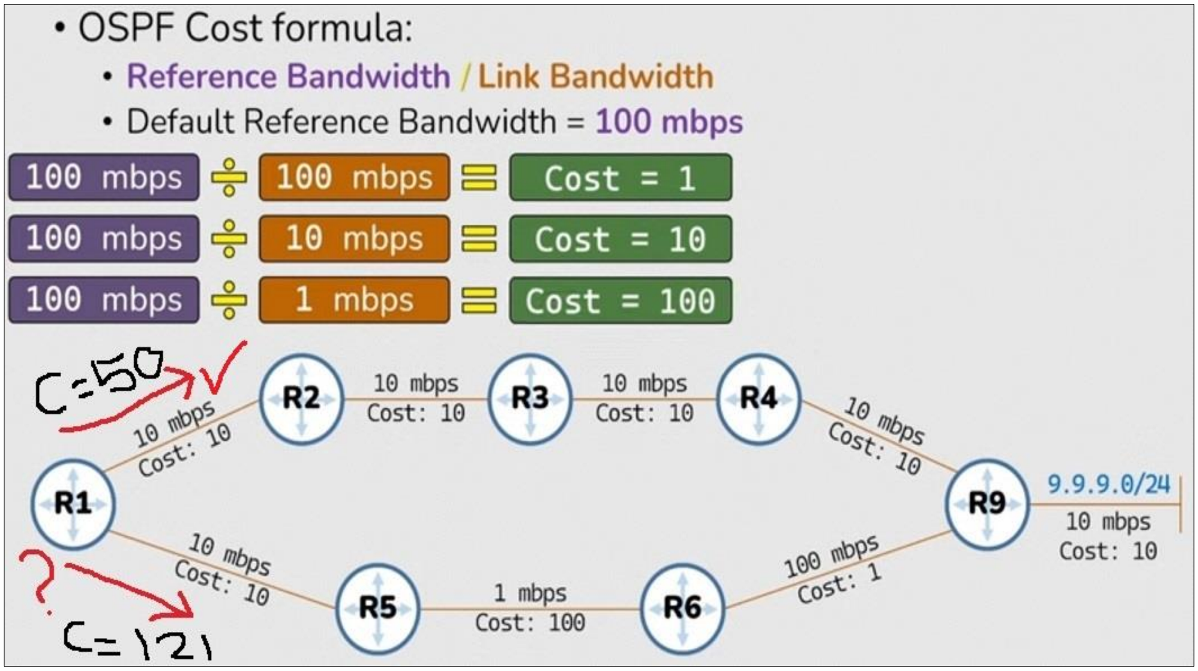
Best route for routing table = *Route which has the lowest cumulative cost*

The summary is as follows:

- OSPF uses SPT tree to calculate the best route for routing table.
- A SPT tree cannot grow beyond the area. Therefore, if a router has interfaces in multiple areas, it needs to build separate tree for each area.
- SPF algorithm calculates all possible routes from source router to destination network.
- Cumulative cost is the sum of the all costs of the outgoing OSPF interfaces in the path.
- While calculating cumulative cost, OSPF consider only outgoing interfaces in path. It does not add the cost of incoming interfaces in cumulative cost.
- If multiple routes exist, SPF compares the cumulative costs. Route which has the lowest cumulative cost will be chosen for routing table.

b. OSPF –Route Cost Calculation – Examples

There are three examples, illustrated through Figures 34, 35, and 36, which detail well how the OSPF link cost is calculated (kindly refer to equations (1) and (2)) and ultimately how the best route is selected, which is certainly based on the least accumulated cost to the destined network.



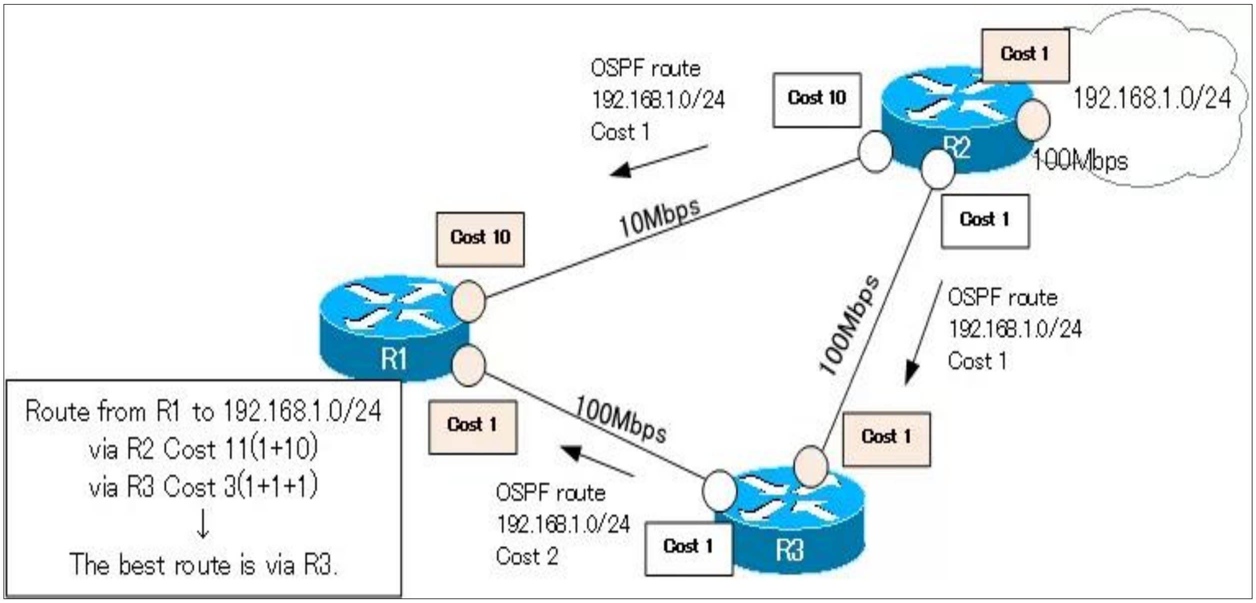
- OSPF Cost formula:
  - Reference Bandwidth / Link Bandwidth
  - Default Reference Bandwidth = 100 mbps

100 mbps ÷ 100 mbps = Cost = 1

100 mbps ÷ 10 mbps = Cost = 10

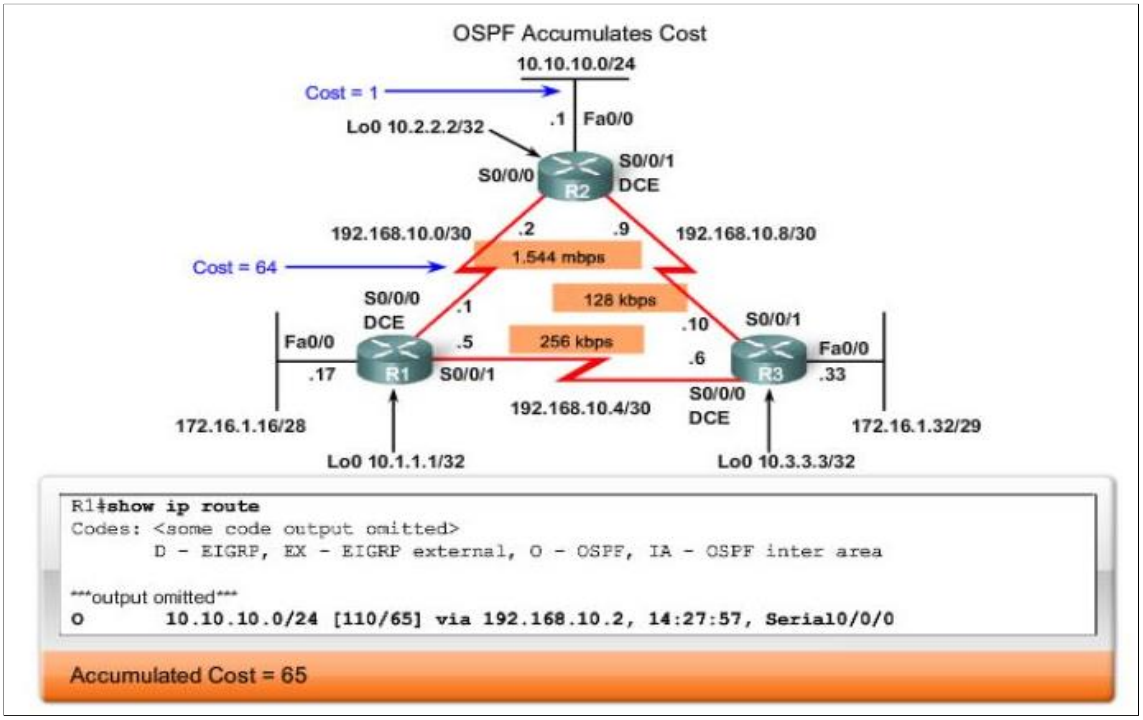
100 mbps ÷ 1 mbps = Cost = 100

Figure 34. OSPF Cost Calculation and Best Route Selection – Example 1



Route from R1 to 192.168.1.0/24
via R2 Cost 11(1+10)
via R3 Cost 3(1+1+1)
↓
The best route is via R3.

Figure 35. OSPF Cost Calculation and Best Route Selection – Example 2



```
R1#show ip route
Codes: <some code output omitted>
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

***output omitted***
O        10.10.10.0/24 [110/65] via 192.168.10.2, 14:27:57, Serial0/0/0
```

Figure 36. OSPF Cost Calculation and Best Route Selection – Example 3

Part II: OSPF Configuration Commands

A. OSPF configuration commands for IPv4 (OSPFv2):

- The Configuration of OSPF for IPV4 is a pretty simple and requires only two major steps:

1. Enabling the OSPF routing process on all routers by using the `router OSPF {process ID}` command in the global configuration command. **The process ID is a positive number between 1 and 65535** and it's locally significant, which means that it does not have to match other OSPF routers in order to establish adjacencies with those neighbors. We can run multiple OSPF process on same router. Process ID is used to differentiate between them. Process ID need not to match on all routers. This command does not directly start the OSPF process. Instead, it provides access to the router configuration mode where the OSPF routing settings are configured. Routing updates are not sent until OSPF is enabled using the network command in router configuration mode for the directly connected networks.

```
Router (config)# router ospf {process ID}
Router (config-router)#
```

2. Configuring the network addresses to be included in routing updates or specifying the interfaces to participate in routing updates with the wildcard mask and the area number. In other words, network command allows us to specify the interfaces which we want to include in OSPF process. This command accepts three arguments network number, wildcard mask and area number.

```
Router(config-router)# network {A.B.C.D} {wild-card mask} area {area ID}
```

Thus, Entering the network address for each directly connected network indicates that:

- Enabling OSPF on all interfaces that belong to a specific network.
- Associated interfaces will now both send and receive OSPF updates.
- **Advertising the specified network** (A.B.C.D) in OSPF routing updates sent to other routers.
- **The wild card mask** is used in the network command to specify the interface or range of interfaces that will be enabled for OSPF. Additionally, Wildcard mask is used with network ID to filter the interfaces. Wildcard mask is different from subnet mask. Subnet mask is used to separate the network portion and host portion in IP address. While wildcard mask is used to match corresponding octet in network portion. Wildcard mask tells OSPF the part of network address that must be matched.
- 0 (Decimal–octet format) Wildcard mask indicates that corresponding octet in network address must be matched exactly.
- 255 (Decimal–octet format) Wildcard mask indicates that we don't care about corresponding octet in network address.
- **OSPF area ID:** Third argument which network command accept is area number. This parameter say router to put matched interface in specified area. All routers

inside an area must have the same area ID. When a single routing domain is existed, the backbone area (Area 0) is used.

- **Note that** the OSPF process ID doesn't have to be the same on all routers in order for the routers to establish a neighbor relationship, but the area parameter has to be the same on all neighboring routers in order for the routers to become neighbors.

- The first command is easy to comprehend, but the second command requires a little bit more thought. With the network command you specify which interfaces will participate in the routing process, the wild card mask, and the backbone area 0. Figure 2.1 shows the directly connected networks in both R1 and R2.

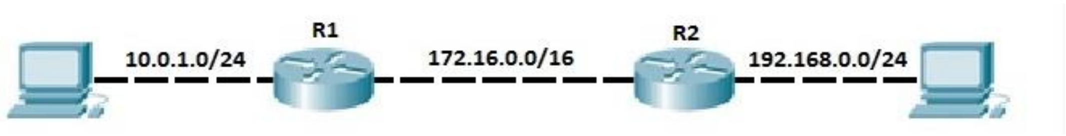


Figure 2.1. Directly connected networks to R1 and R2.

So, the configuration on R1 should look like this:

```
R1(config)#router ospf 1
R1(config-router)#network 10.0.1.0 0.0.0.255 area 0
R1(config-router)#network 172.16.0.0 0.0.255.255 area 0
```

The configuration on R2 looks similar, but with different network number for the directly connected subnet:

```
R2(config)#router ospf 1
R2(config-router)#network 192.168.0.0 0.0.0.255 area 0
R2(config-router)#network 172.16.0.0 0.0.255.255 area 0
```

- To set the router ID use the following command:

```
Router(config-router)#router-id 1.1.1.1
```

- The clear ip ospf process command is used to activate the RID on a router that is already running OSPF:

```
Router#clear ip ospf process
```

- **The passive-interface command** in the router mode allows a router to receive routing updates on an interface but not send updates via that interface. In other words, Use the passive-interface to prevent the transmission of routing messages through a LAN router interface, but still allow that network to be advertised to other routers.

```
Router(config-router) # passive-interface {interface name}
```

- In real life environment **clock rate** parameter controls the data flow between serial links and need to be set at service provider's end. In lab environment we need not to worry about this value. We can use any valid clock rate.

```
Router(config-if)#clock rate 64000
```

- Priority value is 8 bits in length. Default priority value is 1. We can set any value from range 0 to 255. **We can change it from Interface Sub-configuration mode with `ip ospf priority` command**. Interestingly, we can force any router to become DR (Highest) or BDR (Second highest) by changing its priority value. If we set priority value to 0, it will never become DR or BDR.

```
Router(config-if)# ip ospf priority{ priority-
value}
```

- Bandwidth works as an influencer. It is used to influence the metric calculation of OSPF or any other routing protocol which uses bandwidth parameter in route selection process. Serial interface has default bandwidth of **1544Kbps**. The bandwidth command is used to modify the bandwidth value used by the Cisco IOS software in calculating the OSPF cost metric.

```
Router(config) # interface g0/0
Router(config-if)# bandwidth {kilobits}
```

For instance, to explain, how bandwidth influence route selection process we will configure (64Kbps) bandwidth for any serial in the network topology.

```
Router(config) # interface s0/0/0
Router(config-if)# bandwidth 64
```

- Redistributing an OSPF Default Route: In most topologies, there is an edge or gateway router. This router located between an OSPF routing domain and a non-OSPF network is called the Autonomous System Boundary Router (ASBR). Like RIP, OSPF requires the use of the `default-information originate` command to advertise the static default route (0.0.0.0/0) to the other routers in the area.

```
Router(config-router) # default-information originate
```

B. OSPF troubleshooting commands for IPv4 (OSPFv2):

1) `show ip ospf database` → Displays a list of the LSAs and types them into a link-state database. This list shows only the information in the LSA header.
2) `show ip ospf database [router] [link-state-id]` → Displays a list of all of the LSAs of a router in the database. LSAs are produced by every router. These fundamental LSAs list all of the links of the routers or interfaces, along with the states and outgoing costs of the links, and they are flooded only within the area in which they originate.
3) `show ip ospf [process-id [area-id]] database [summary] [link-state-id]` → Displays information only about the network summary LSAs in the database.

4) `show ip ospf database [summary] [self-originate]` → Displays only self-originated LSAs (from the local router).
5) Notice that LSAs learned through the virtual link have the DoNotAge (DNA) option. The virtual link is treated like a demand circuit.
6) `show ip ospf neighbor` → Displays the OSPF neighbor. the OSPF must first establish the neighbor. It is important to verify that the OSPF has established the neighbor correctly with the show ip ospf neighbor command. The State part is "FULL/ -", indicating that it is a point-to-point interface and therefore does not require DR election.
7) `debug ip ospf adj` →Displays the events involved to build or break OSPF adjacency. The routers become adjacent and exchange LSAs via the virtual link, similar to a physical link.
8) `debug ip ospf hello` →Displays the exchange of OSPF Hello packets in real time on the console.
9) `show ip ospf virtual-links` → Notice that adjacencies over virtual links are not displayed in the `show ip ospf neighbor` command output. The only way to see them is to look at the router LSA and observe debug commands as the adjacency comes up, or issue the `show ip ospf virtual-links` command.
10) `show ip ospf interface <interface>` → This command is a quick check to determine if all of the interfaces belong to the areas, they are supposed to be in. The sequence in which the OSPF network commands are listed is very important. This output shows very important information, including area number, the process ID, and the router ID. The state of the interface is (DR, BDR, or DROTHER).
11) `show ip protocols` → Displays general information about the routing protocol, not just OSPF. for OSPF, you can verify the process number, router IDs, the network command configurations, and the numerator value of the cost formula.
12) `show ip ospf` →Verifies the various timer values and other information for OSPF processing. You can also see the number of times the SPF calculation is performed per area.
13) `show ip route ospf` →Displays only the OSPF routes in the routing table. There are several types of OSPF route codes, as shown in the table below (Table 2.1).

Table 2.1. The types of OSPF route codes

Code	**Summary**
O	OSPF routes in the same area
O IA	OSPF routes in other areas
O E1	Routes in non-OSPF domains (metric type 1)
O E2	Routes in non-OSPF domains (metric type 2)
S*	The default route.
O*	Means that there is a default route is distributed and can be reached. The route source of this path is the OSPF routing protocol.

14) `show ip route` →Verifies the routing table, where **O** in the output indicates the remote networks routes that discovered via OSPF updates. Also note that the administrative distance of 110 is shown, together with the cost of 128.

```
Router#show ip route
O    192.168.10.8 [110/128] via 192.168.10.6, 14:27:57, Serial0/0/1
                  [110/128] via 192.168.10.2, 14:27:57, Serial0/0/0
```

C. OSPF commands for IPv6 (OSPFv3):

OSPFv3 is the OSPFv2 equivalent for exchanging IPv6 prefixes, it exchanges routing information to populate the IPv6 routing table with remote prefixes.

1. Always remember that the **ipv6 unicast-routing** global configuration command must be configured to enable the router to forward IPv6 packets (i.e., packet forwarding). That's the starting point before you can work with IPv6 addresses on a router.

```
Router(config)# ipv6 unicast-routing
```

2. Configure OSPFv3 on the routers.

To do this, we will simply, create an OSPFv3 process using **ipv6 router ospf process ID** command from the global config mode. This is just the same way we created an OSPF process in IPv4. We will also add router ID using **router-id router ID** command from the router configuration mode. For instance,

```
Router(config)#ipv6 router ospf {process ID }
R1(config-rtr)#router-id {router ID}
```

3. Enabling OSPFv3 on all router interfaces

Add router interfaces to the OSPFv3 process just created. This will activate(enable) route advertisement on the interfaces. We will use the following command on interface configuration mode.

```
Router(config)# interface g0/0
Router(config-if)# ipv6 ospf {process ID} area {area ID}
```

For instance, the number 10 is the OSPF process ID, 0 is the area ID.

```
Router(config-if)# ipv6 ospf 10 area 0
```

4. To configure the OSPFv3 network type use the ipv6 ospf network command in interface configuration mode. To return to the default type, use the no form of this command.

```
Router(config-if)# ipv6 ospf network point-to-point
Router(config-if)# no ipv6 ospf network
```

5. Configure an IPv4 Loopback Interface: Another common configuration of Cisco IOS routers is enabling a loopback interface. **The loopback interface is a logical interface internal to the router. It is not assigned to a physical port and can therefore never be**

connected to any other device. It is considered a software interface that is automatically placed in an “up/up” state, as long as the router is functioning. The loopback interface is useful in testing and managing a Cisco IOS device because it ensures that at least one interface will always be available. **For example, it can be used for testing purposes, such as testing internal routing processes, by emulating networks behind the router.** Additionally, the IPv4 address or IPv6 assigned to the loopback interface can be significant to processes on the router that use an interface IPv4 or IPv6 address for identification purposes, such as the OSPF routing process. By enabling a loopback interface, the router will use the always available loopback interface address for identification, rather than an IP address assigned to a physical port that may go down. The steps to configure a loopback interface on a router are:

- ✓ Step 1. Create the loopback interface using the interface loopback number global configuration command.
- ✓ Step 2. Configure the IP address.

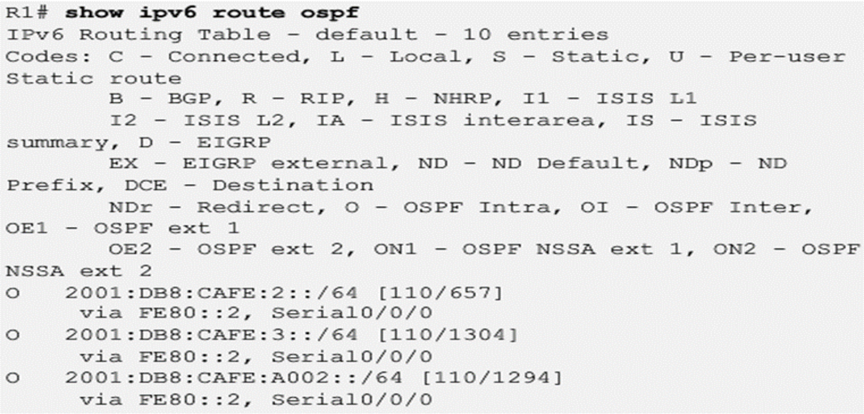
For example, the following commands configure a loopback interface of the R1 router:

```
R1# configure terminal
R1(config)# interface loopback0
R1(config-if)# ip address 10.0.0.1 255.255.255.0
R1(config-if)# exit
R1(config)#
```

```
R1# configure terminal
R1(config)# interface loopback0
R1(config-if)# ipv6 address 2001:100:4::4/64
R1(config-if)# ipv6 address FE80::4 link-local
R1(config-if)# exit
R1(config)#
```

A loopback interface is always enabled and therefore does not require a no shutdown command. Multiple loopback interfaces can be enabled on a router. The IPv4 address for each loopback interface must be unique and unused by any other interface.

6. Verifying OSPFv3: Commands used to verify the OSPV3 is the same command that are used to verify the IPv4, but use ipv6 instead of ip in all commands, for instance, Figure 2.2 shows the show ipv6 route ospf command. Table 2.1 summarizes the OSPF Routing protocol configuration commands.



```
R1# show ipv6 route ospf
IPv6 Routing Table - default - 10 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user
Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS
summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND
Prefix, DCE - Destination
       NDr - Redirect, O - OSPF Intra, OI - OSPF Inter,
OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF
NSSA ext 2
O   2001:DB8:CAFE:2::/64 [110/657]
     via FE80::2, Serial0/0/0
O   2001:DB8:CAFE:3::/64 [110/1304]
     via FE80::2, Serial0/0/0
O   2001:DB8:CAFE:A002::/64 [110/1294]
     via FE80::2, Serial0/0/0
```

Figure 2.2. Verify OSPFv3

Table 2.1 Summary of OSPFv3 configuration commands

Command	**Description**
Router(config-if)# ipv6 ospf {process ID} area {area ID}	This command when executed in **interface configuration mode** enables OSPFv3 per specified process id and area id.
Router(config)#ipv6 router ospf {process ID}	This command when executed in **global configuration mode** places you into IPv6 OSPF (OSPFv3) router configuration mode where you can configure the router-id, distribute list, default information options and more.
Router(config-rtr)#router-id A.B.C.D	This command is executed in **OSPFv3 router configuration mode** to statically set a router-id. If you're in a complete IPv6 network with no IPv4 addresses assigned to any interface on a router you must have a Router-ID assigned due to OSPF not being able to use the highest IPv4 address assigned to a logical or connected physical interface.
Router #show ipv6 ospf {process ID}	This command when executed in user or privileged mode will display current timers, router-id and reference bandwidth.
Router #show ipv6 ospf neighbor	This command when executed in user or privileged mode displays established neighbor relationships and their router type (DR,BDR, DROTHER)
Router #show ipv6 ospf database	This command when executed in user or privileged mode displays the current OSPFv3 database contents including the sequence number.
Router #show ipv6 protocols	This command displays the parameters about the state of any active IPv6 routing protocol processes configured on the router.
Router # show ipv6 route	This command is used to examine the IPv6 routing table. OSPF for IPv6 routes are denoted in the routing table with a O.

Part III: Configuring OSPFv2 and OSPFv3 (Practical part)

A. OSPFv2 Configuration (Packet Tracer)

In this example, you should configure OSPF routing protocol in multiple areas with virtual links. Suppose that the routers' interfaces are configured. Thus, all routers have information about the networks that they have on their own interfaces. Routers will not exchange this information between them on their own. We need to implement OSPFv2 routing protocol that will insist them to share this information.

Look at Figure 2.3. We have three areas and on the left side is area 0. Area 2 is behind area 1. **Normally this is not going to work since area 2 has to be directly connected to area 0.** We can make this work by using a virtual link. By using a **virtual link,** we can extend area 0 through area 1 so area 2 will be "directly connected" to area 0. Let's take a look at how a virtual link can solve this problem. This is basically how a virtual link works. It’s like a tunnel through area 1 to reach area 2. This way area 2 will be directly connected.

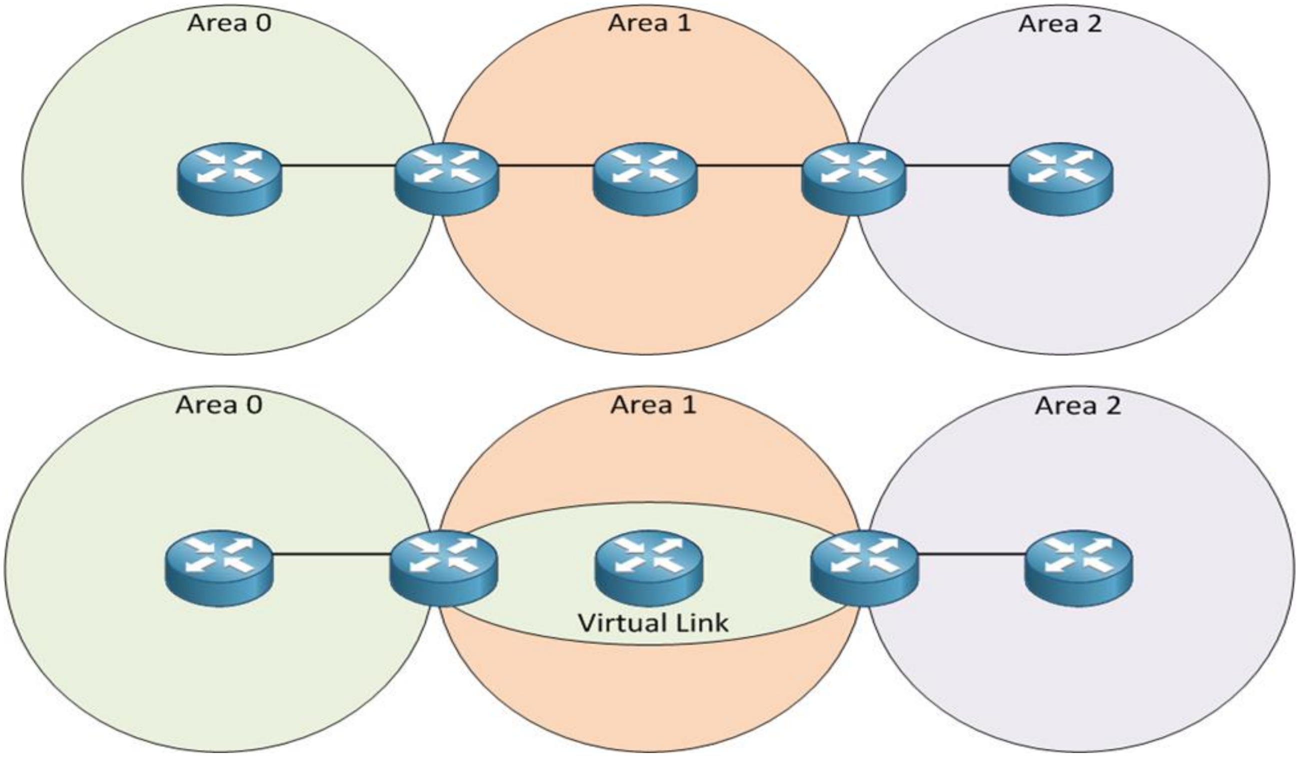


Figure 2.3 . Network topology for OSPFv2 routing protocol without interface names

Now let we show you how to configure a virtual link. In the example above area 2 is not directly connected to area 0 so we will have to use a virtual link between routers R1 and R2, as shown on Figure 2.4. here is how we do it: we will start with a default OSPF configuration.

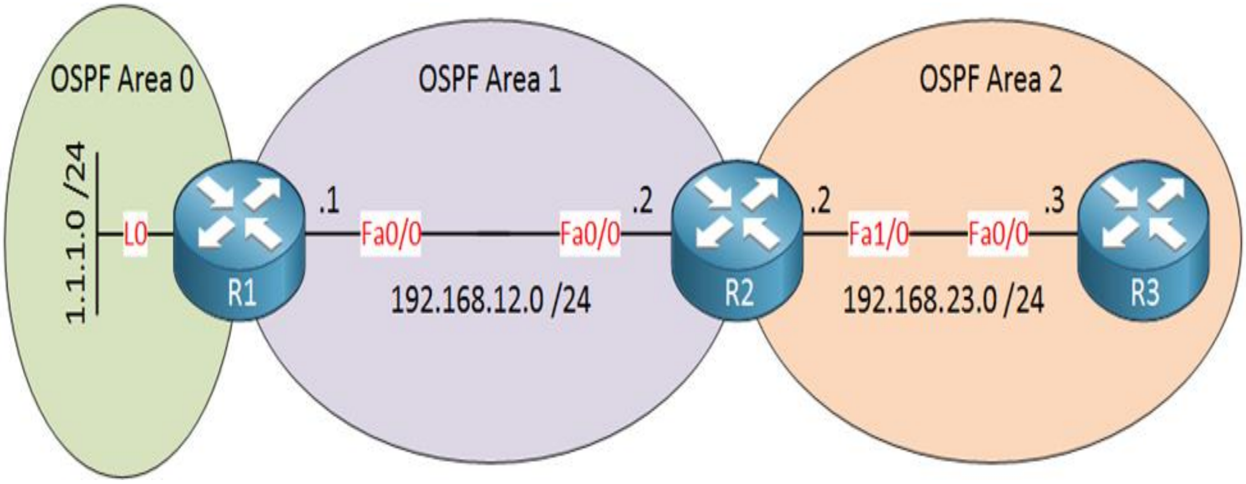


Figure 2.4 . Network topology for OSPFv2 routing protocol

1. Configure OSPFv2 routing protocol

Configuration of OSPFv2 protocol is much easier than you think. It requires only four steps to configure the OSPFv2 on routers.

1) Enable OSPF routing protocol from global configuration mode with the assigned process ID.
2) Tell OSPF routing protocol which networks you want to advertise, with the wild card mask, and the area number (i.e., area 0).
3) Configure the LAN interface that contains no routers so that it does not send out any routing information (i.e., passive interface).

```
R1(config)#router ospf 1
R1(config-router)#network 1.1.1.0 0.0.0.255 area 0
R1(config-router)#network 192.168.12.0 0.0.0.255 area 1
```

```
R2(config)#router ospf 1
R2(config-router)#network 192.168.12.0 0.0.0.255 area 1
R2(config-router)#network 192.168.23.0 0.0.0.255 area 2
```

```
R3(config)#router ospf 1
R3(config-router)#network 192.168.23.0 0.0.0.255 area 2
```

2. Configure the virtual links between ABRs:

We configure the virtual link between ABRs and we use the area virtual-link command. First, you need to specify the area where we need the virtual link which is area 1 in our example. The second step is to configure the OSPF router ID of the other ABR. Keep this in mind you need to configure **the OSPF router ID and NOT the IP address of the ABR**. If everything is OK area 2 will be directly connected to area 0 through our virtual link.

```
R1(config)#router ospf 1
R1(config-router)#area 1 virtual-link 192.168.23.2
```

```
R2(config)#router ospf 1
R2(config-router)#area 1 virtual-link 1.1.1.1
```

You will see the message below that tells us the virtual link is established correctly.

```
R1# %OSPF-5-ADJCHG Process 1, Nbr 192.168.23.2 on OSPF_VL0 from
LOADING to FULL, Loading Done
```

```
R2# %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on OSPF_VL0 from
LOADING to FULL, Loading Done
```

3. Verify your configuration:

- You can use the `show ip ospf virtual-links` command to check if your virtual link is working on R1 and R2.

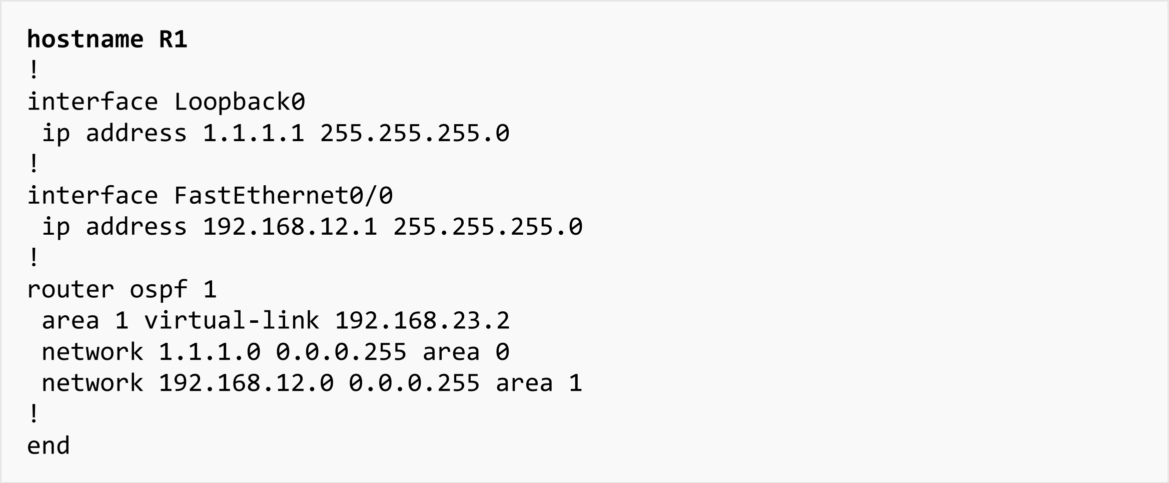
```
R1#show ip ospf virtual-links
Virtual Link OSPF_VL0 to router 192.168.23.2 is up
  Run as demand circuit
  DoNotAge LSA allowed.
  Transit area 1, via interface FastEthernet0/0, Cost of using 1
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:06
    Adjacency State FULL (Hello suppressed)
    Index 1/2, retransmission queue length 0, number of retransmission 0
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
    Last retransmission scan length is 0, maximum is 0
    Last retransmission scan time is 0 msec, maximum is 0 msec
```

```
R2#show ip ospf virtual-links
Virtual Link OSPF_VL0 to router 1.1.1.1 is up
  Run as demand circuit
  DoNotAge LSA allowed.
  Transit area 1, via interface FastEthernet0/0, Cost of using 1
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:05
    Adjacency State FULL (Hello suppressed)
    Index 1/3, retransmission queue length 0, number of retransmission 0
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
    Last retransmission scan length is 0, maximum is 0
    Last retransmission scan time is 0 msec, maximum is 0 msec
```

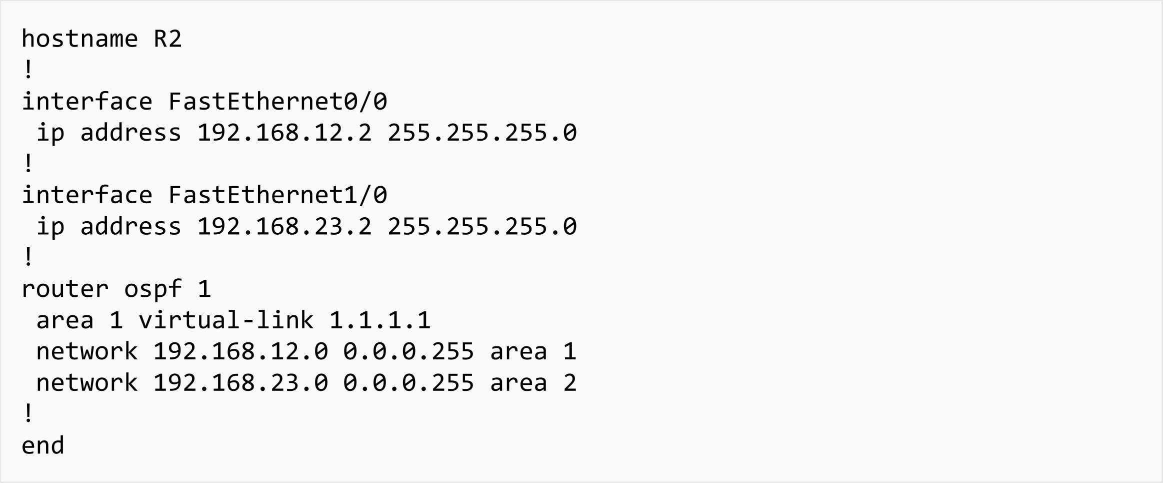
- If you look at the LSDB you will see that the virtual link shows up as a type 1 router LSA. You can also see **DNA which means do not age.**

```
R1#show ip ospf database
            OSPF Router with ID (1.1.1.1) (Process ID 1)

                Router Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum Link count
1.1.1.1         1.1.1.1         189         0x80000004 0x00E333 2
192.168.23.2    192.168.23.2    1     (DNA) 0x80000002 0x009816 1
R2#show ip ospf database

            OSPF Router with ID (192.168.23.2) (Process ID 1)

                Router Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum Link count
1.1.1.1         1.1.1.1         1     (DNA) 0x80000004 0x00E333 2
192.168.23.2    192.168.23.2    159         0x80000002 0x009816 1
```

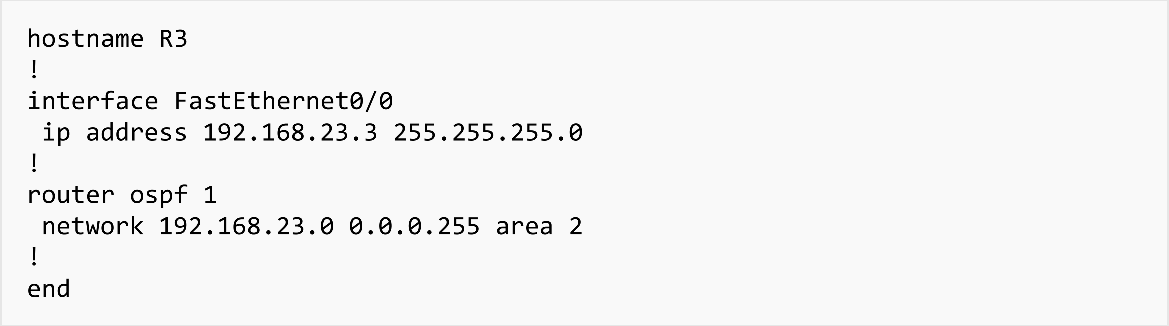
This is the complete configuration on the three routers.



```
hostname R1
!
interface Loopback0
 ip address 1.1.1.1 255.255.255.0
!
interface FastEthernet0/0
 ip address 192.168.12.1 255.255.255.0
!
router ospf 1
 area 1 virtual-link 192.168.23.2
 network 1.1.1.0 0.0.0.255 area 0
 network 192.168.12.0 0.0.0.255 area 1
!
end
```



```
hostname R2
!
interface FastEthernet0/0
 ip address 192.168.12.2 255.255.255.0
!
interface FastEthernet1/0
 ip address 192.168.23.2 255.255.255.0
!
router ospf 1
 area 1 virtual-link 1.1.1.1
 network 192.168.12.0 0.0.0.255 area 1
 network 192.168.23.0 0.0.0.255 area 2
!
end
```



```
hostname R3
!
interface FastEthernet0/0
 ip address 192.168.23.3 255.255.255.0
!
router ospf 1
 network 192.168.23.0 0.0.0.255 area 2
!
end
```

4. If any router is connected to the Internet or ISP, we must add two extra commands,

4) Use the appropriate command to create a static default route on that router for all Internet traffic to reach the network.
5) Advertise the default route configured in the previous step with other OSPF routers.

5. ISP router is configured with summarized static route and LAN static routes.

That's it. Our network is ready to take the advantage of OSPFv2.

6. Verify OSPFv2 Configurations

To verify the setup, we will use ping command. ping command is used to test the connectivity between two devices.

B. OSPFv3 Configuration (Packet Tracer)

To configure OSPFv3 routing protocol, IPv6 addressing and unicast-routing must be already configured on the network devices. For our example configuration, we have two routers, R1 and R2, and we will configure OSPFv3 routing between them, as shown in Figure 2.5.

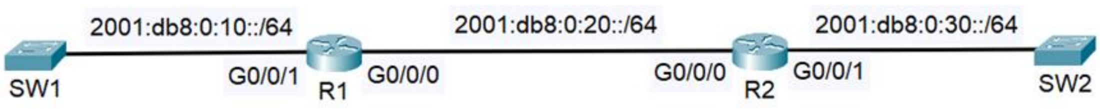


Figure 2.5. Verify OSPFv3

1. Configure OSPFv3 routing protocol

1) We have to configure unicast-routing on R1 and R2 so that routing protocols will work for IPv6.

```
R1(config)#ipv6 unicast-routing

R2(config)#ipv6 unicast-routing
```

2) Next, we will set the IPv6 loopback addresses.

```
R1(config)#interface loopback 0
R1(config-if)#ipv6 address 2001::1/128
```

```
R2(config)#interface loopback 0
R2(config-if)#ipv6 address 2001::2/128
```

3) Then, we have to enable IPv6 on our interfaces and configure their IPv6 addresses as well.

```
R1(config)#interface G0/0/0
R1(config-if)#ipv6 enable
R1(config-if)#ipv6 address 2001:db8:0:20::1/64
R1(config)#int G0/0/1
R1(config-if)#ipv6 enable
R1(config-if)#ipv6 address 2001:db8:0:10::1/64
```

```
R2(config)#interface G0/0/0
R2(config-if)#ipv6 enable
R2(config-if)#ipv6 address 2001:db8:0:20::2/64
R2(config)#interface G0/0/1
R2(config-if)#ipv6 enable
R2(config-if)#ipv6 address 2001:db8:0:30::1/64
```

4) So that's our basic configuration done. Now let's enable OSPFv3 on our routers using the '`ipv6 router ospfv3 <process-id>`' command at the global configuration

mode. We also have to assign a router ID for the OSPF routing process if there are no IPv4 addresses configured on the router. The command to do this is ‘router-id <router-id>’, and it is entered under the OSPF router configuration mode. The router ID resembles an IPv4 address and it can have any value as long as it is unique within the OSPF domain.

```
R1(config)#ipv6 router ospf 1
R1(config-rtr)#router-id 1.1.1.1
R1(config-rtr)#exit
```

```
R2(config)#ipv6 router ospf 1
R2(config-rtr)#router-id 2.2.2.2
R2(config-rtr)#exit
```

5) Finally, we have to assign our interfaces to their corresponding OSPFv3 areas. Under the interface configuration mode, enter the command ‘ipv6 ospf <process id> area <area number>’. For our example, we will assign all interfaces into area 0.

```
R1(config)#interface range G0/0/0-1
R1(config-if)#ipv6 ospf 1 area 0
R1(config)#interface loopback 0
R1(config-if)#ipv6 ospf 1 area 0
```

```
R2(config)#interface range G0/0/0-1
R2(config-if)#ipv6 ospf 1 area 0
R2(config)#interface loopback 0
R2(config-if)#ipv6 ospf 1 area 0
```

2. IPv6 OSPFv3 Verification

1) To check the IPv6 addresses on the router, we can use the ‘show ipv6 interface brief’ command.

```
R1#show ipv6 interface brief
GigabitEthernet0/0/0 [up/up]
FE80::5054:FF:FE0F:F10A
2001:DB8:0:20::1
GigabitEthernet0/0/1 [up/up]
FE80::5054:FF:FE0B:CEFB
2001:DB8:0:10::1
GigabitEthernet0/0/2 [administratively down/down]
unassigned
GigabitEthernet0/0/3 [administratively down/down]
unassigned
Loopback0 [up/up]
FE80::5054:FF:FE0F:F10A
2001::1
```

2) To view the OSPFv3 neighbor, we can use the command ‘show ipv6 ospf neighbor.

```
R1#show ipv6 ospf neighbor
OSPFv3 Router with ID (1.1.1.1) (Process ID 1)
```

```
Neighbor ID Pri State Dead Time Interface ID
Interface
2.2.2.2 1 FULL/DR 00:00:38 2 GigabitEthernet0/0
```

```
R2#show ipv6 ospf neighbor
OSPFv3 Router with ID (2.2.2.2) (Process ID 1)
Neighbor ID Pri State Dead Time Interface ID
Interface
1.1.1.1 1 FULL/BDR 00:00:38 2 GigabitEthernet0/0
```

3) We can also have a detailed view of the OSPFv3 interface configurations using the '`show ospfv3 interface`' command.

```
R1#show ospfv3 interface
Loopback0 is up, line protocol is up
Link Local Address FE80::5054:FF:FE0F:F10A,
Interface ID 9
Area 0, Process ID 1, Instance ID 0, Router ID
1.1.1.1
Network Type LOOPBACK, Cost: 1
Loopback interface is treated as a stub Host
GigabitEthernet0/1 is up, line protocol is up
Link Local Address FE80::5054:FF:FE0B:CEFB,
Interface ID 3
Area 0, Process ID 1, Instance ID 0, Router ID
1.1.1.1
Network Type BROADCAST, Cost: 1
Transmit Delay is 1 sec, State DR, Priority 1
Designated Router (ID) 1.1.1.1, local address
FE80::5054:FF:FE0B:CEFB
No backup designated router on this network
Timer intervals configured, Hello 10, Dead 40, Wait
40, Retransmit 5
Hello due in 00:00:00
Graceful restart helper support enabled
Index 1/2/2, flood queue length 0
Next 0x0(0)/0x0(0)/0x0(0)
Last flood scan length is 0, maximum is 0
Last flood scan time is 0 msec, maximum is 0 msec
Neighbor Count is 0, Adjacent neighbor count is 0
Suppress hello for 0 neighbor(s)
GigabitEthernet0/0 is up, line protocol is up
Link Local Address FE80::5054:FF:FE0F:F10A,
Interface ID 2
Area 0, Process ID 1, Instance ID 0, Router ID
1.1.1.1
Network Type BROADCAST, Cost: 1
Transmit Delay is 1 sec, State BDR, Priority 1
Designated Router (ID) 2.2.2.2, local address
FE80::5054:FF:FE10:8031
```

```
Backup Designated router (ID) 1.1.1.1, local
address FE80::5054:FF:FE0F:F10A
Timer intervals configured, Hello 10, Dead 40, Wait
40, Retransmit 5
Hello due in 00:00:07
Graceful restart helper support enabled
Index 1/1/1, flood queue length 0
Next 0x0(0)/0x0(0)/0x0(0)
Last flood scan length is 1, maximum is 2
Last flood scan time is 0 msec, maximum is 1 msec
Neighbor Count is 1, Adjacent neighbor count is 1
Adjacent with neighbor 2.2.2.2 (Designated Router)
Suppress hello for 0 neighbor(s)
```

4) The command ‘`show ipv6 route ospf`’ shows the OSPFv3 IPv6 routing table.

```
R1#show ipv6 route ospf
IPv6 Routing Table - default - 8 entries
Codes: C - Connected, L - Local, S - Static, U -
Per-user Static route
B - BGP, HA - Home Agent, MR - Mobile Router, R -
RIP
H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS
interarea
IS - ISIS summary, D - EIGRP, EX - EIGRP external,
NM - NEMO
ND - ND Default, NDp - ND Prefix, DCE -
Destination, NDr - Redirect
RL - RPL, O - OSPF Intra, OI - OSPF Inter, OE1 -
OSPF ext 1
OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF
NSSA ext 2
la - LISP alt, lr - LISP site-registrations, ld -
LISP dyn-eid
lA - LISP away, a - Application

O 2001::2/128 [110/1]
via FE80::5054:FF:FE10:8031, GigabitEthernet0/0
O 2001:DB8:0:30::/64 [110/2]
via FE80::5054:FF:FE10:8031, GigabitEthernet0/0
```

Important Note: Some parts of this handout have been collected from several trustable sites, books, and published videos/slides and other parts have been prepared and written by the instructors. As a matter of fact, this handout is made to be so straight forward, understandable, and so attractive whereas the students can do the required activities and solve the problems in a systematic and easy way, but still the instructors are expected to discuss some important material during the labs’ sessions.