

# OSPF 笔记

## 红茶三杯 CCIE 学习文档

文档版本： 3.0

更新时间： 2013-01-26

文档作者： 红茶三杯

文档地址： <http://ccietea.com>

文档备注： 请关注文档版本及更新时间

## 1 基础知识

### 1.1 知识点

#### 1. COST 的计算

接口 COST=100M /接口带宽，一条路由的 COST 由该路由从来源一路过来的所有入口方向的接口 cost 值的总和。这意味着更高带宽的传输介质在 OSPF 协议中将会计算出一个小于 1 的分数，这在 OSPF 协议中是不允许的（会被四舍五入为 1）。可使用 auto-cost reference-bandwidth xx 来更改参考值。注意如果要调整，就要在全 OSPF 域的所有 OSPF router 上都进行配置。

#### 2. 组播地址

224.0.0.5 （所有的 OSPF router）来向所有其他 router 发送链路状态信息。

224.0.0.6 用来表示所有的指定 router

#### 3. 数据库类型

在同一个 OSPF 区域中的每台 Router 都相同的数据状态数据库，带有多个接口的 router 可以加入多个区域（必须至少有一个接口属于 area0），这些 router 称为 Area Border Routers，它会给每个区域维护一个单独的拓扑数据库。

毗邻数据库 adjacency database	列出所有已经与 router 建立双向通信关系的 router
链路状态数据库 LSDB link-state database	列出关于网络中所有其他 router 的信息，该数据库显示出了网络的拓扑结构，一个区域中所有的 router 都有相同的该数据库
路由选择表	Routing table

Forward database

## 1.2 RouterID

OSPF Router\_ID 的选定是若有 loopback 口，选最大的 loopback 口地址，若无则选活动的物理接口中 IP 地址最大的作为 RouterID。

在路由器运行了 OSPF 并确定了 ROUTER-ID 之后，如果该 ID 对应的接口 DOWN 掉或者接口消失或出现一个更大的 IP，OSPF 仍然使用该 ROUTER-ID（也就是说，非抢占，稳定第一），即使此时 `clear ip ospf process` 重启 OSPF 进程，RouterID 也不会发生改变；除非重新手工配置 RouterID（OSPF 进程下手工敲 `router-id xxx`），并重启 OSPF 进程方可。

## 1.3 Timers

Hello Interval	CISCO 设备在广播型网络上默认 10s，非广播型网络默认 30s，修改 HELLO 时间，Dead 时间默认也会随之修改
Routerdead interval	多长时间没收到邻居的分组，认为其失效。默认为 hello interval 的 4 倍
LSA refresh timer	默认每隔 30 分钟发送链路状态条目的摘要，刷新之后 LSA 序列号加 1
LSAmaxAge	60 分钟，该时间内如果没有刷新，则从 LSDB 中删除
Wait timer	等待计时器 在开始选取 DR 和 BDR 之前，路由器等待邻居路由器的 HELLO 数据包通告 DR 和 BDR 的时长（收集所有候选人的信息），等待计时器的时间长度就是 RouterDeadIntervd 的时间
Retransmit	指在没有得到确认的情况下，路由器重传 OSPF 数据包将要等待的时间，缺省 5s
pollInterval	这个值只用于 NBMA 网络上相关的邻居路由器。因为在 NBMA 网络上，邻居路由器可能无法自动地 本地路由器发现，因此，如果邻居状态是失效 DOWN 的，那么路由器将每经过 pollInterval 的时间就会发送一个 Hello 数据包给它的邻居路由器。这里的 pollInterval 的时长比 HelloInterval 的时间要长一些。

## 1.4 DR BDR

### 1. 选举过程

当一台 OSPF 路由器有效(active)并去发现它的邻居路由器时，它将去检查有效的 DR 和 BDR 路由器。如果 DR 和 BDR 路由器存在的话，这台路由器将接受已经存在的 DR 和 BDR 路由器。如果 BDR 路由器不存

在，将执行一个选取过程，选出具有最高优先级的路由器作为 BDR 路由器。如果存在多台路由器具有相同的优先级，那么在数值上具有最高路由器 ID 的路由器将被选中。如果没有有效的 DR 路由器存在，那么 BDR 路由器将被选举为 DR 路由器，然后再执行一个选取过程选取 BDR 路由器。

每台路由器的每一个多点访问接口都有一个路由器的优先级 0-255 默认 1。

## 2. 要点总结

- 如果在一个多路访问网络上只有惟一的一台具有选取资格的路由器相连，那么这台路由器将成为 DR 路由器，而且在这个网络上没有 BDR 路由器。其他所有的路由器都将只和这台 DR 路由器建立邻接关系
- 如果没有具有选取资格的路由器和一个多路访问网络相连，那么这个网络上将没有 DR 或者 BDR 路由器，而且也不建立任何邻接关系。
- DR、BDR 非抢占。当 DR 失效，BDR 成为 DR，同时再产生 BDR。
- DR 与 BDR、DR 与 Drother、BDR 与 Drother 之间都是 FULL 状态，Drother 之间是 2way 状态。所有的 Drother 都只和 DR 以及 BDR 建立全毗邻关系。

## 3. 验证实验

### 【实验 1】



- R1、R2 接口 ip ospf pri 0，则 R1、R2 不建立邻居关系
- R1 接口优先级为 0，R2 为 1，则 R2 成为 DR，并且该多路访问网络中没有 BDR
- 在上一步的基础上，R1 接口优先级改为 100，对网络没有影响（非抢占）
- 在上一步的基础上，R2（此时为 DR）重启 OSPF 进程，则 R1 成为 DR，R2 成为 BDR

### 【实验 2】

在上图中，如果 R1、R2 接口优先级相等，但 R2 的 routerid 大，一般情况下 R2 应该是 DR，但如果 R1 先配置，并且 R2 在 R1 配置后的 40s 之后才配置，那么这个时候即使 R2 的 routerid 大，R1 已经成为了 DR。原因在于 Wait timer 等待计时器，这个计时器的是在开始选取 DR 和 BDR 之前，路由器等待邻居路由器的 HELLO 数据包通告 DR 和 BDR 的时长（收集所有候选人的信息），等待计时器的时间长度就是 RouterDeadIntervd 的时间。因此只要 R2 在 40s 之后才配的，那么在这 40s 内，R1 就抢夺了 DR 的身份。

## 1.5 序列号

DBD 的序列号用于 master 和 slave 的同步，LSA 的序列号用来比较 LSA 的新或旧，这里讨论的是 LSA 的序列号，

是一个 4 个字节的数字。

### 1. 线性空间

从 0x80000001 到 0x7ffffff

### 2. 循环序列号空间

循环序列编号建立了一个不合逻辑的奇特的位。如果  $x$  是 1 到 4294 967 295 之间任意的一个数,那么  $0 < x < 0$ 。在运行正常的网络中通过声明两条规则可以维持这种条件,其中声明的规则用来确定什么时候一个序列号大于或小于另一个序列号。假设序列号空间为  $n$ ,有两个序列号  $a$  和  $b$ ,如果满足以下任意一种条件,则认为  $a$  更新(数量更大)

$a > b$  且  $(a - b) \leq n/2$

$a < b$  且  $(b - a) > n/2$

例如假设使用一个 6 位序列号空间

$n=2$  的 6 次方 = 64 那么  $n/2=32$

若有 3 和 48 两个序列号,则根据规则 2,可得 3 较新

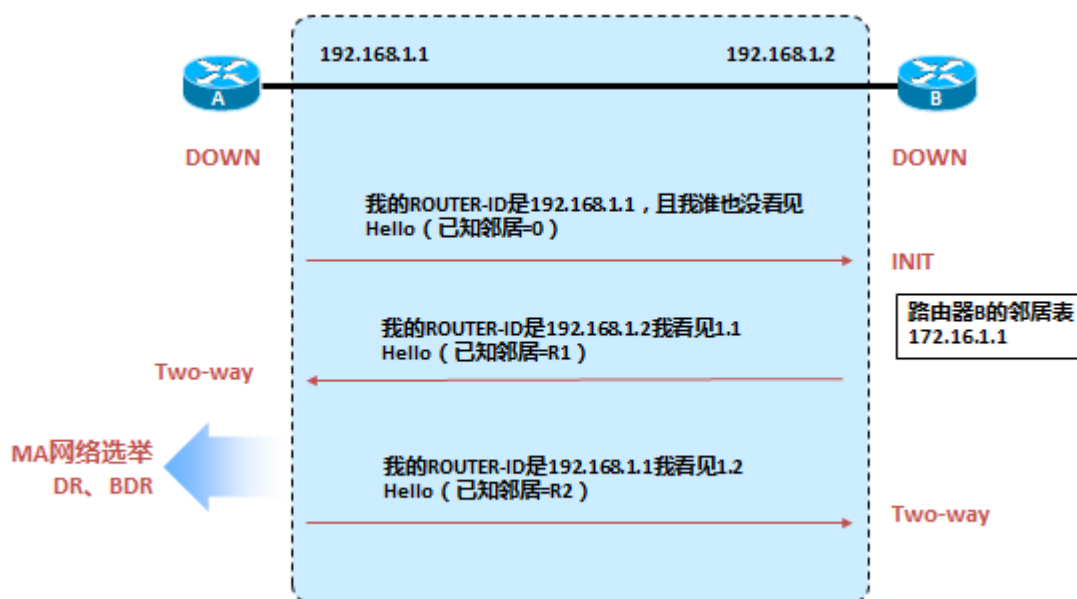
### 3. 棒棒糖

圆形空间的缺点是不存在一个数小于其他所有的数。

线性空间的缺点是不能循环使用序列号,即序列号是有限的

## 2 邻居关系建立及维护

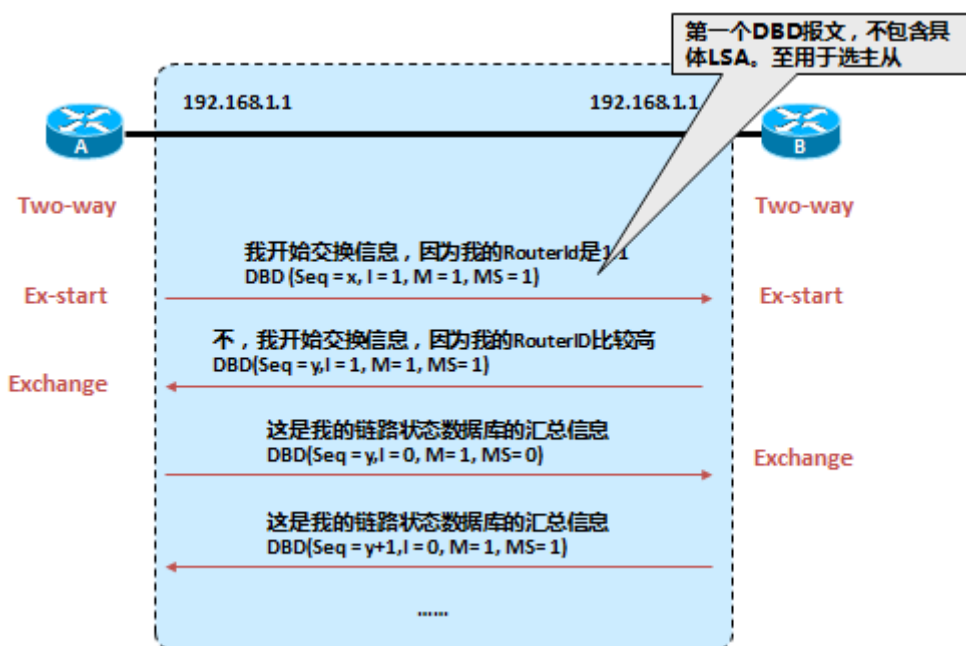
### 2.1 邻居建立过程



在初始情况下，A、B 在某个接口激活了 OSPF 后，都会开始在这个接口上去发组播的 HELLO 包，目的是发现 OSPF 邻居。HELLO 包里，有个 active neighbor 字段，用来存储路由器在某个 OSPF 接口上发现的邻居，当然，初始情况下，这个 HELLO 包里是不包含任何活跃的邻居的（也就没有 active neighbor 字段），因为他谁也没发现。

当 OSPF 路由器（B）在某个 OSPF 接口上收到邻居发来的 HELLO 包（里面没有装 active neighbor），它会记录下 A（在自己的 OSPF 接口数据结构中）并且将 A 的状态视为 **init**，然后将 A 的 RouterID 存储在自己将要发送的 HELLO 包的 active neighbor 字段里发送出去，这样 A 就会收到这个 hello 包，并且在这个 hello 包当中找到自己的 RouterID，那么 A 会认为，与 B 已经完成了双边关系的建立，因此 A 会将 B 的邻居状态置为 **two-way**。

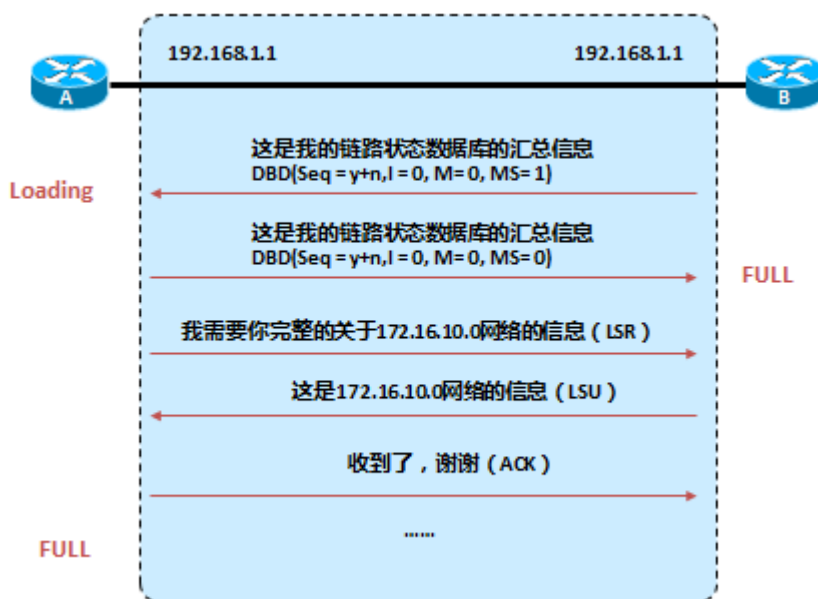
与此同时，A 也会继续发送 HELLO 包，并且将 B 的 routerID 放置于 HELLO 包中，而 B 收到这个 hello 包并看见了自己的 RouterID 后，B 也会将 A 的状态置为 two-way，至此 OSPF 的第一个稳态就达到了。



接下去 A、B 会进入 **ex-start** 状态并开始进行 master、slave 的协商，协商 M/S 的目的是为了决定在后续的 LSA 交互中，谁来决定 DD 的序列号，而 RouterID 大的那个 OSPF 路由器的接口将会成为 master，由它来决定 DD seq，对端成为 slave。这里要注意 master 不是 DR，要注意与 DR 的概念进行区分。这个协商过程，是由交互 DBD 包实现的，注意这里使用的是空的 DBD 包，也就是不包含任何 LSA 头部的 DBD 包，这个包当中，有三个位非常关键：I、M、MS。用于 ex-start 阶段协商 master、slave 的 DBD 包，I 位（或叫做 init 位）都是置 0 的，另外 MS 位如果置 1，表示 DBD 报文始发路由器认为自己的 master，当然起初大家都这么认为，在一系列 DBD 交换后，就会得到选举结果，被选举为 slave 的 OSPF 接口，会将发送的 DBD 包 MS 位置为 0；另外 M 位表示 more，如果一个 OSPF 接口发送的 DBD 包 M 位置 1，在表示这不是最后一个 DBD，后续还有 DBD 包待发送。

当 OSPF 接口收到一个 DBD 包且其中 I 位置 0 的时候，它就知道与该邻居的 ex-start 阶段已经过去了，于是将邻居的状态置为 **ex-change**，并存储对端发来的 DBD 包所包含的 LSA 头部，当然，他自己也发送关于自己 OSPF DB 的摘要给邻居。如此一来，双方都能通过 DBD 的交互，了解到对方 OSPF DB 中的摘要情况。在这个过程中，可能交互数个 DBD 报文，并要注意，这些报文的 I 位都置 0，且 M 位一般也置 0，除非这是某个 OSPF

接口发送的最后一个 DBD 包。



当 Router A 收到一个 M 位置 0 的 DBD 包的时候，它就知道，这是邻居发来的最后一个 DBD 包了，如果它搜集完这个邻居（假设是 B）发来的 DBD 并且发现，这些 DBD 里有它感兴趣的 LSA，它期望更详细的 LSA 信息时，它将 B 置为 **Loading** 状态，并且开始发送 LSR 报文去请求特定 LSA 的详细信息。B 收到这个 LSR 后，会以 LSU 进行回应，其中就包含了对方请求的 LSA 详细信息，因此，只有在 LSU 报文中，才能看到 LSA 的完整信息。收到 LSU 后，A 将 LSU 中所包含的 LSA 放进自己的 LSDB，并且给 B 发一个 Lsack 进行确认。当 OSPF 接口上所有的待请求的 LSA 全部收到更新后，它会将邻居置为 FULL。至此，OSPF 邻接关系的建立达到全毗邻。

在这里我们有个地方需要留意，我们通常说，OSPF 路由器 A 与 B 进入了 xx 状态，其实这句话并不严谨，原因之一，是因为 OSPF 是接口敏感型协议，许多的操作都是以 OSPF 接口作为立足点去考虑的，譬如邻居关系的建立，再如 DR 和 BDR，我们不能说一台路由器是 DR，准确的说，应该是某路由器的某个接口是 DR；再者，说两台路由器之间是 xx 状态，这个也不严谨，所谓的邻居状态，必须是以某台路由器为观察点，在其某个接口上观察到的某个邻居的状态，因此可能出现的一个情况是，在 A 上，看到的 B 的状态为 Loading，但是在 B 上，A 的状态已经是 FULL 了。

## 2.2 状态机

### 1. Down

在 DOWN 状态下，OSPF 接口仍然有尝试发现邻居的意愿，因此会不断的发送组播 hello 包。

### 2. Attempt

如果一个路由器，它邻居处于这种状态，则表示它从邻居没有收到任何信息，但是做了努力来与邻居联系。

仅在 NBMA 网络上存在，当 NBMA 网络上具有 DR 选取资格的路由器和其邻居路由器相连的接口开始变为有

效(Activ/e)时,或者当这台路由器成为 DR 或 BDR 时,这台具有 DR 选取资格的路由器将会把邻居路由器的状态转换到 Attempt 状态。在 Attempt 状态下,路由器将使用 hellointerval 时间代替 pollinterval 的时间来作为向邻居发送 hello 数据包的时间间隔。

### 3. init

当 OSPF 接口收到链路上某个邻居发来的第一个 HELLO 包的时候,它会在接口上将该邻居置为 init 状态,注意这个 hello 包中可能并未包含任何的邻居信息。但是这至少证明,我这个 OSPF 接口在这个链路上,至少有个活的邻居。

下面是一个没有发现任何 active neighbor 的 hello 包：

```
Open Shortest Path First
[ ] OSPF Header
    OSPF Version: 2
    Message Type: Hello Packet (1)
    Packet Length: 44
    Source OSPF Router: 1.1.1.1 (1.1.1.1)
    Area ID: 0.0.0.0 (Backbone)
    Packet Checksum: 0xea9c [correct]
    Auth Type: Null
    Auth Data (none)
[ ] OSPF Hello Packet
    Network Mask: 255.255.255.0
    Hello Interval: 10 seconds
    [ ] Options: 0x12 (L, E)
        Router Priority: 1
        Router Dead Interval: 40 seconds
        Designated Router: 0.0.0.0
        Backup Designated Router: 0.0.0.0
    [ ] OSPF LLS Data Block
```

### 4. Two-way

当 OSPF 路由器在某个链路上发现了邻居后,它自己发送的 hello 包里就会增加 active neighbor 字段,用于存储在该链路上发现的 OSPF 邻居。当一台 OSPF router 看到自己(的 RouterID)出现在邻居发过来的 hello 分组中,它就会将该邻居置为 Two-way。该状态是 OSPF 邻居之间可以具有的最基本的关系,也是第一个稳态,但是此时两者还不能共享路由信息。

下面是一个已经在链路上发现了邻居 1.1.1.1 的 hello：



```
Open Shortest Path First
  OSPF Header
    OSPF Version: 2
    Message Type: Hello Packet (1)
    Packet Length: 48
    Source OSPF Router: 2.2.2.2 (2.2.2.2)
    Area ID: 0.0.0.0 (Backbone)
    Packet Checksum: 0xe694 [correct]
    Auth Type: Null
    Auth Data (none)
  OSPF Hello Packet
    Network Mask: 255.255.255.0
    Hello Interval: 10 seconds
    Options: 0x12 (L, E)
    Router Priority: 1
    Router Dead Interval: 40 seconds
    Designated Router: 0.0.0.0
    Backup Designated Router: 0.0.0.0
    Active Neighbor: 1.1.1.1
  OSPF LLS Data Block
```

## 5. ExStart

一台 OSPF 路由器在将某个邻居置为 2way 状态后, 就开始发送空的 DBD 包, 用于协商 master/slave。这个就是 ex-start 状态。两台 router 间用空的 BDB 分组确定 master 和 slave 关系 (注意不是 DR 和 BDR), 在 DBD 包中有 3 个标记位用来管理邻接关系的建立过程:

- I 位 或称为初始位 (initial bit) 用于 ex-start 协商主从关系的初始化协商的 DBD 包, 该位置 1
- M 位 或称为后继位 (More bit) 如果这不是 OSPF router 发送的最后一个 DBD, 该位置 1
- MS 位 或称为主/从位 (Master/slave bit) 如果始发路由器是 Master, 则该位置 1

如果某台 OSPF router 收到邻居发来的 DBD, I 位也就是 init 位置 0, 则意味着 ex-start 状态结束, 并且 MS/slave 已经选出来了, 那么该路由器会将邻居置为 exchange 状态, 开始用包含 LSA 头部的 DBD 交换各自的 LSBD。

下面是一个用于初始化协商的 DBD 消息:

```
Open Shortest Path First
  OSPF Header
  OSPF DB Description
    Interface MTU: 1500
    Options: 0x52 (O, L, E)
    DB Description: 0x07 (I, M, MS)
      .... 0... = R: OOBResync bit is NOT set
      .... .1.. = I: Init bit is SET
      .... ..1. = M: More bit is SET
      .... ...1 = MS: Master/slave bit is SET
    DD Sequence: 7994
  OSPF LLS Data Block
```

## 6. Exchange 状态

这个过程, 双方使用包含自己 LSA 头部的 DBD 报文进行交互, 并且将对方发过来的 LSA 头部、并且自己感兴趣的 LSA (或自己没有的 LSA) 存储在一个本地 OSPF 接口的队列里, 以便在下一个阶段进行 LSA 详细



信息的请求。当某个 OSPF 接口收到邻居发来的 DBD, M 位置 0, 则表示对方已经发完 DBD 了, 与此同时, 如果该路由器的这个 OSPF 接口上存在待请求的 LSA, 那么它会将这个邻居置为 loading 状态。

下面是一个装载了 LSA 头部的 DBD 消息:

```
Open Shortest Path First
+ OSPF Header
+ OSPF DB Description
  Interface MTU: 1500
  + Options: 0x52 (O, L, E)
  + DB Description: 0x02 (M)
    .... 0... = R: OOBResync bit is NOT set
    .... .0.. = I: Init bit is NOT set
    .... ..1. = M: More bit is SET
    .... ...0 = MS: Master/slave bit is NOT set
  DD Sequence: 7994
+ LSA Header
  LS Age: 19 seconds
  Do Not Age: False
  + Options: 0x22 (DC, E)
  Link-State Advertisement Type: Router-LSA (1)
  Link State ID: 1.1.1.1
  Advertising Router: 1.1.1.1 (1.1.1.1)
  LS Sequence Number: 0x80000002
  LS Checksum: 0x3e21
  Length: 48
+ OSPF LLS Data Block
```

LSA头部

## 7. loading 状态

OSPF router 使用 LSR 去请求 LSA 的详细信息, 对方使用 LSU 发来更新, 因此只有 LSU 里才有 LSA 的完整信息。在收到 LSU 后, 一方面本地使用 LSAck 进行确认, 另一方面将 LSU 中包含的 LSA 装载进自己的 LSDB。

以下是一个 LSR 消息, 非常的简单:

```
Open Shortest Path First
+ OSPF Header
+ Link State Request
  Link-State Advertisement Type: Router-LSA (1)
  Link State ID: 1.1.1.1
  Advertising Router: 1.1.1.1 (1.1.1.1)
```

接着是一个 LSU 消息, 里头包含了 LSA 的完整信息, LSA 这里暂时不做详细介绍, 请看下文:

```
Open Shortest Path First
+ OSPF Header
+ LS Update Packet
  Number of LSAs: 1
  + LS Type: Router-LSA
```

## 8. Full Adjacency 状态

Loading 状态结束后, 也就是本地 OSPF 接口上再没有待更新的 LSA 队列后, 将邻居置为 FULL。

## 2.3 邻居数据结构

Router#sh ip ospf neighbor 192.168.1.2

Neighbor 192.168.1.2, interface address 9.9.12.2

In the area 0 via interface Serial2/0

Neighbor priority is 1 (configured 100), State is FULL, 18 state changes

DR is 9.9.12.1 BDR is 9.9.12.2

Poll interval 120

Options is 0x52

LLS Options is 0x1 (LR)

Dead timer due in 00:01:53

Neighbor is up for 00:00:19

Index 1/1, retransmission queue length 0, number of retransmission 2

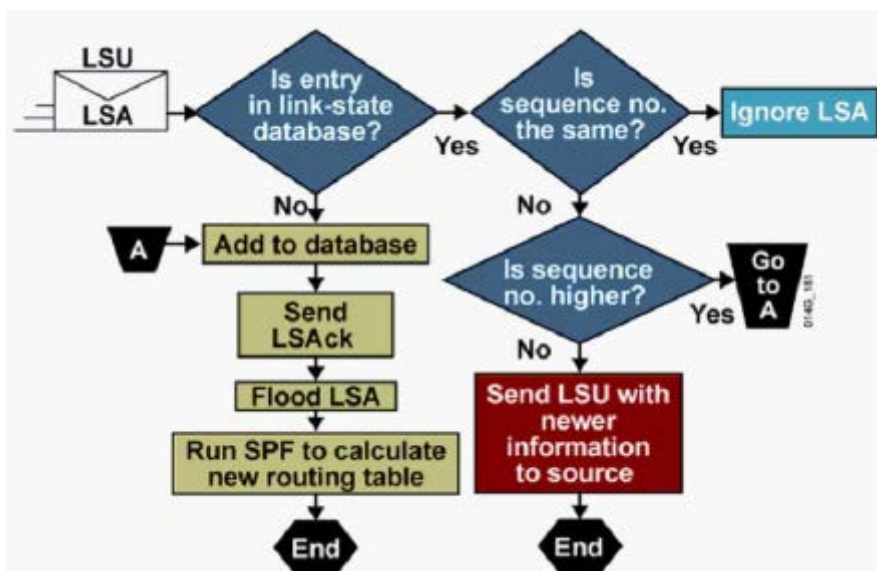
First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)

显示的是 NBMA 环境下，OSPF 邻居的数据结构，其中邻居的实际接口优先级是 1，

本地使用 neighbor 9.9.12.2 priority 100

本命令只对 DR、BDR 选举过程有效

## 2.4 LSA 的更新



LSA 可以通过下面两种方法之一来确认：

显式确认(Explicit Acknowledgment) 确认收到包含这个 LSA 头部的链路状态确认数据包。

隐式确认(Implicit Acknowledgment) 确认收到包含这个 LSA 的相同实例(没有其他更加新的 LSA)的更新数据包。

主路由器将控制数据库的同步过程，并确保每次只有一个数据库描述数据包是未处理的。当从路由器收到一个从主路由器发出的 DBD 后，从路由器将通过发送一个具有相同序列号的 DBD 报文来确认(这个过程一直延续，直到主路由器发出 m 位置 0 的报文)。如果 rxmtInterval 到期后，从路由器还没确认，主路由器会重传。

LSDB 里每个 LSA 都有一个序列号，用来标示 LSA 的新旧，从 0x80000001 到 0x7fffff，当 LSA 触发更新的时候，其泛洪 LSA 的时候会增加序列号，且随着 30min 一次的泛洪，序列号也会加 1 (由产生该 LSA 的 router)。每个 LSA 条目都有一个老化时间 aging timer，存储在 age 字段中，默认的 30 分钟。LSA 失效后，最初发送该条目的路由器会发送一个 LSU，其中包含序列号更高的 LSA。

## 2.5 OSPF 邻居关系建立中的问题

### 1. MTU 问题

ip os mtu-ignore

在接口模式下配置可以忽略 MTU，注意是在 MTU 小的那一个 OSPF 路由器接口上(如果两端 MTU 不一样)

### 2. 接口掩码问题

a) 在一个 MA 网络中，如果邻居两端接口掩码不一致，则邻居建立不起来。

```
*Mar 1 09:20:00.173: OSPF: Mismatched hello parameters from 192.168.12.1
*Mar 1 09:20:00.173: OSPF: Dead R 40 C 40, Hello R 10 C 10 Mask R 255.255.255.0 C
255.255.255.224
```

原因在于 2 类 LSA 中，是有掩码字段的

Routing Bit Set on this LSA

LS age: 251

Options: (No TOS-capability, DC)

LS Type: Network Links

Link State ID: 192.168.23.2 (address of Designated Router)

Advertising Router: 192.168.12.2

LS Seq Number: 80000001

Checksum: 0x144

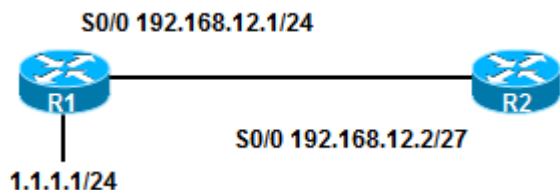
Length: 32

**Network Mask: /24 // 如果掩码不一致，DR 就无法用 2 类 LSA 来描述你这台路由器**

Attached Router: 192.168.12.2

Attached Router: 192.168.23.3

- b) 在 p2p 的网络中，如果邻居两端接口掩码不一致，邻居能建立，并且路由也能学习到  
玩玩而已，别真特么去搞两个掩码不一样。



则 R1 的路由表：

```

192.168.12.0/24 is variably subnetted, 2 subnets, 2 masks
O      192.168.12.0/27 [110/128] via 192.168.12.2, 00:01:34, Serial0/0
C      192.168.12.0/24 is directly connected, Serial0/0
1.0.0.0/24 is subnetted, 1 subnets
C      1.1.1.0 is directly connected, Loopback0
  
```

则 R2 的路由表：

```

192.168.12.0/24 is variably subnetted, 2 subnets, 2 masks
C      192.168.12.0/27 is directly connected, Serial0/0
O      192.168.12.0/24 [110/128] via 192.168.12.1, 00:03:14, Serial0/0
1.0.0.0/32 is subnetted, 1 subnets
O      1.1.1.1 [110/65] via 192.168.12.1, 00:03:14, Serial0/0
  
```

### 3. 网络类型问题

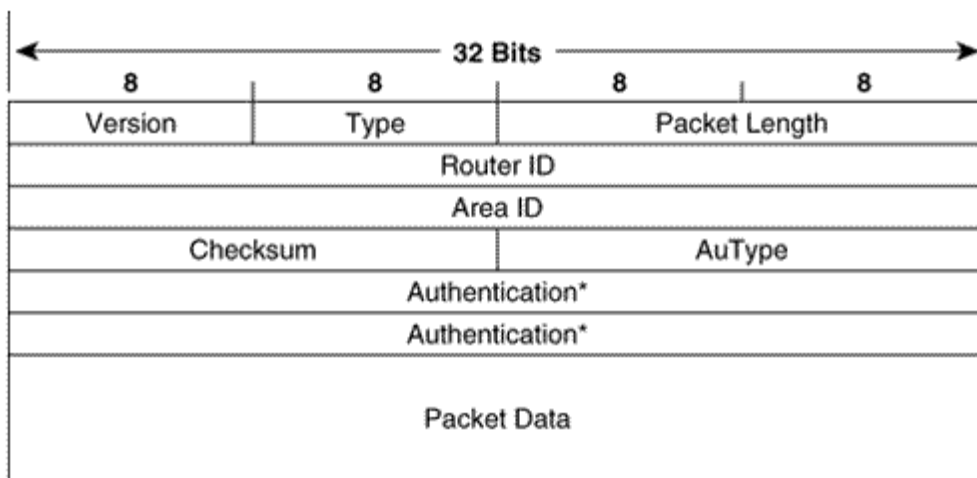
两端如果 OSPF 网络类型不同，那么邻居关系能建立起来，路由器甚至能收到对方的 LSA，但是路由学不到。

## 3 OSPF 数据包

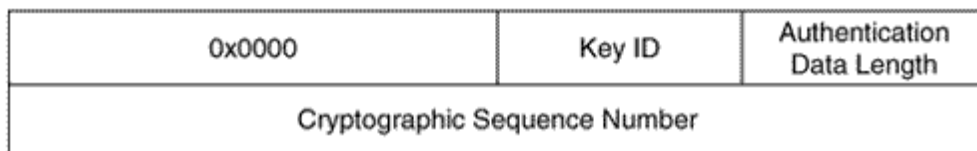
### 3.1 数据包类型

Hello	与邻居建立毗邻关系。默认每 10s 交换一次 HELLO 分组
DBD 数据库描述分组	描述一个 OSPF router 链路状态数据库的内容
LSR 链路状态请求	请求邻居 router 发送其链路状态数据库中的特定项
LSU 链路状态更新	向邻居 router 发送链路状态通告
LSACK 链路状态确认	由于 OSPF 使用 IP 分组，并不使用 TCP 或 UDP，而其又对可靠性有要求，因此增加该包来保证可靠性。

## 3.2 数据包头部



\*If AuType = 2, the Authentication field is:

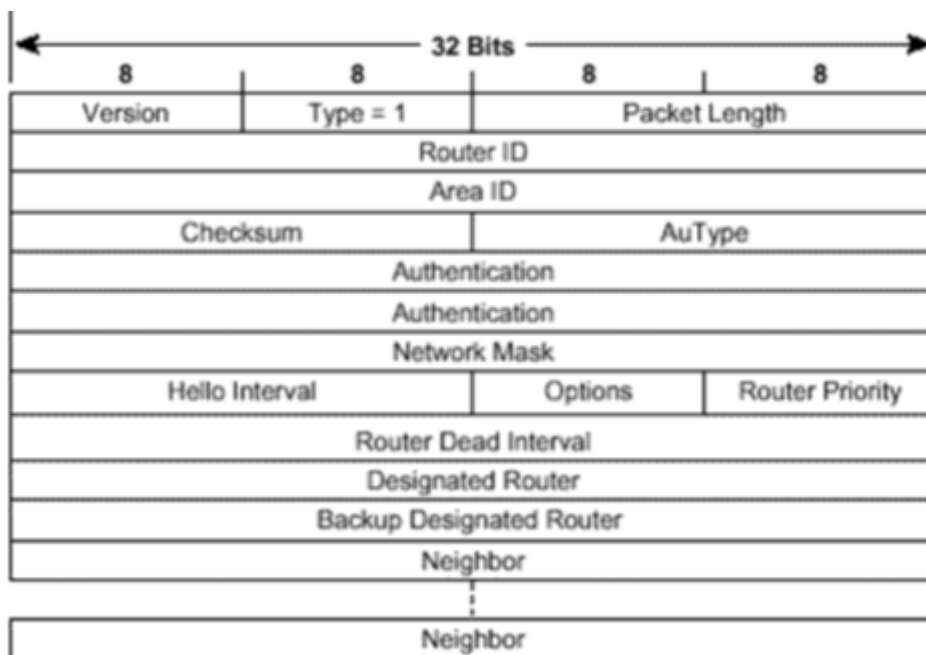


AreaID 是一个点分十进制表示的数在这里，如 area256，则报文中存放的是 0.0.1.0（转换成二进制看）

AuType=0 不认证 =1 明文 =2 密文

## 3.3 数据包格式

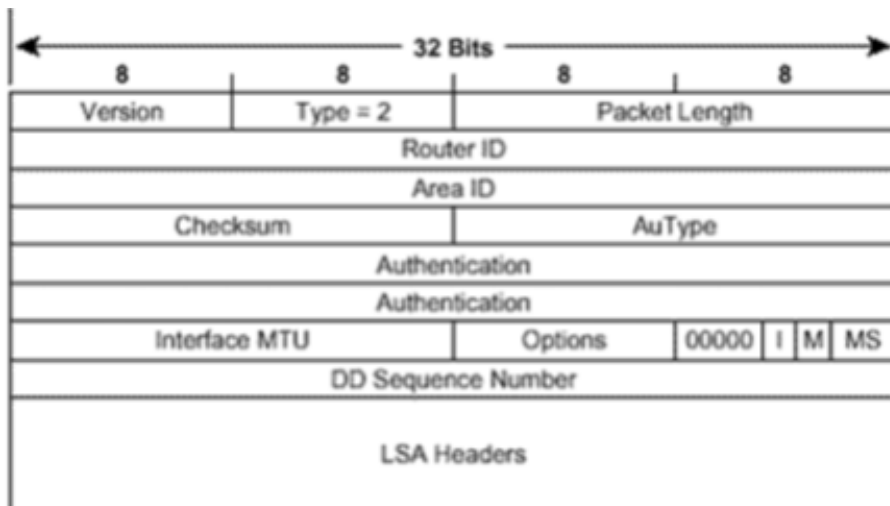
### 1. HELLO 包格式



```

❏ OSPF Hello Packet
  Network Mask: 255.255.255.0
  Hello Interval: 10 seconds
  ❏ Options: 0x12 (L, E)
    0... .... = DN: DN-bit is NOT set
    .0.. .... = O: O-bit is NOT set
    ..0. .... = DC: Demand circuits are NOT supported
    ...1 .... = L: The packet contains LLS data block
    .... 0... = NP: Nssa is NOT supported
    .... .0.. = MC: NOT multicast capable
    .... ..1. = E: ExternalRoutingCapability
  Router Priority: 1
  Router Dead Interval: 40 seconds
  Designated Router: 0.0.0.0
  Backup Designated Router: 0.0.0.0
  
```

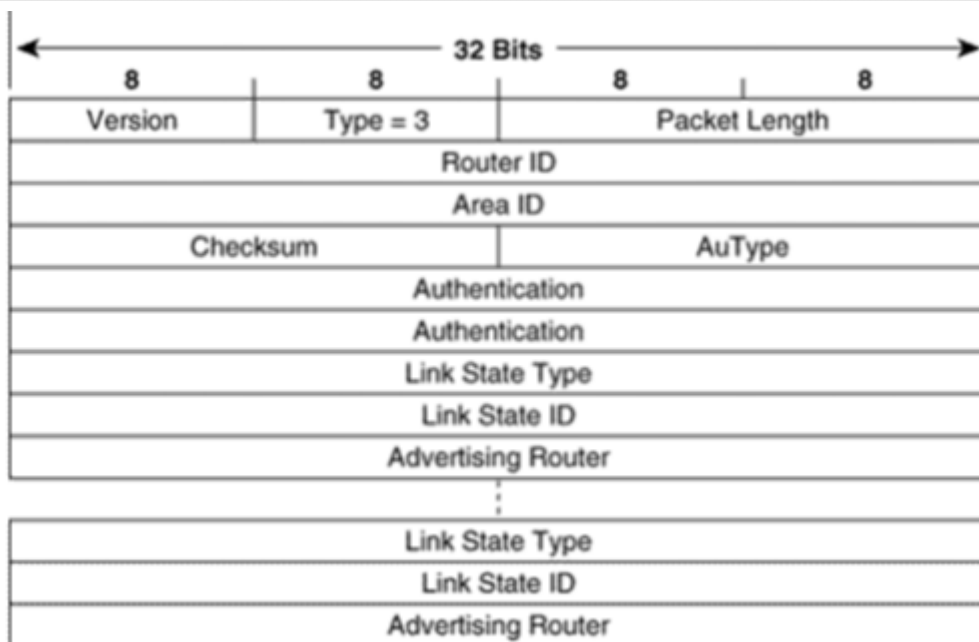
## 2. DBD 报文格式



```

OSPF DB Description
  Interface MTU: 1500
  ❏ Options: 0x52 (O, L, E)
    0... .... = DN: DN-bit is NOT set
    .1.. .... = O: O-bit is SET
    ..0. .... = DC: Demand circuits are NOT supported
    ...1 .... = L: The packet contains LLS data block
    .... 0... = NP: Nssa is NOT supported
    .... .0.. = MC: NOT multicast capable
    .... ..1. = E: ExternalRoutingCapability
  ❏ DB Description: 0x07 (I, M, MS)
    .... 0... = R: OOBResync bit is NOT set
    .... .1.. = I: Init bit is SET
    .... ..1. = M: More bit is SET
    .... ...1 = MS: Master/slave bit is SET
  DD Sequence: 6444
  
```

## 3. LSR 报文格式



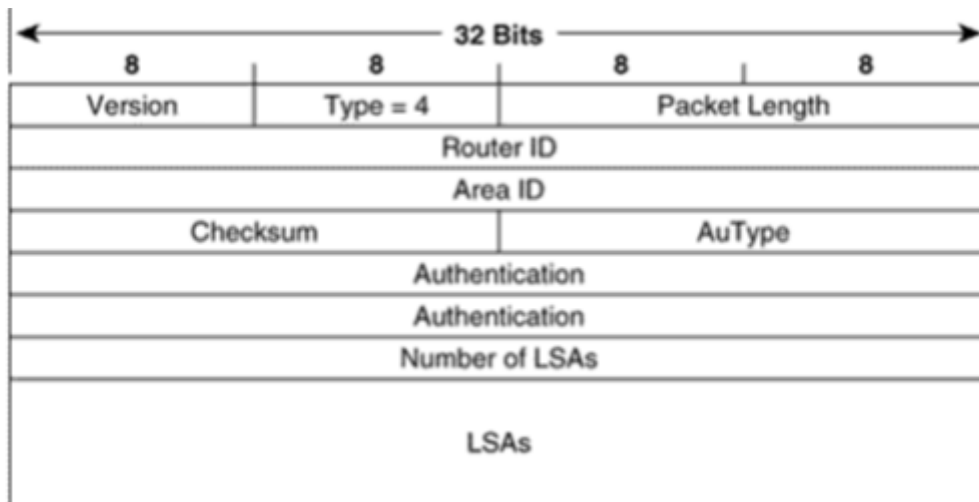
#### Link State Request

Link-State Advertisement Type: Router-LSA (1)

Link State ID: 9.9.12.2

Advertising Router: 9.9.12.2 (9.9.12.2)

#### 4. LSU 报文格式



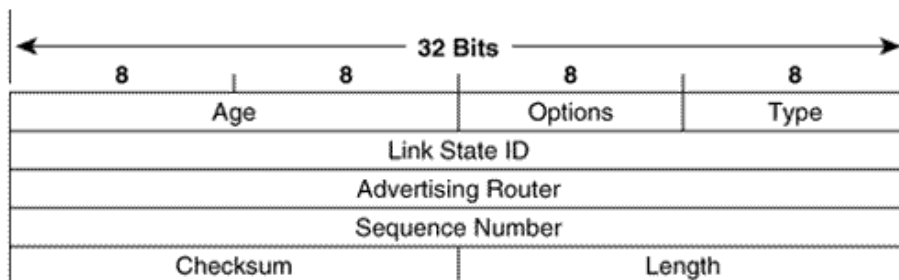


```

LS Update Packet
  Number of LSAs: 1
  [ ] LS Type: Router-LSA
    LS Age: 1 seconds
    Do Not Age: False
    [ ] Options: 0x22 (DC, E)
      Link-State Advertisement Type: Router-LSA (1)
      Link State ID: 9.9.12.2
      Advertising Router: 9.9.12.2 (9.9.12.2)
      LS Sequence Number: 0x80000003
      LS Checksum: 0xd54a
      Length: 48
    [ ] Flags: 0x00 ( )
      Number of Links: 2
    [ ] Type: PTP      ID: 1.1.1.1      Data: 9.9.12.2      Metric: 64
    [ ] Type: Stub    ID: 9.9.12.0    Data: 255.255.255.0 Metric: 64
  
```

### 3.4 LSA 格式

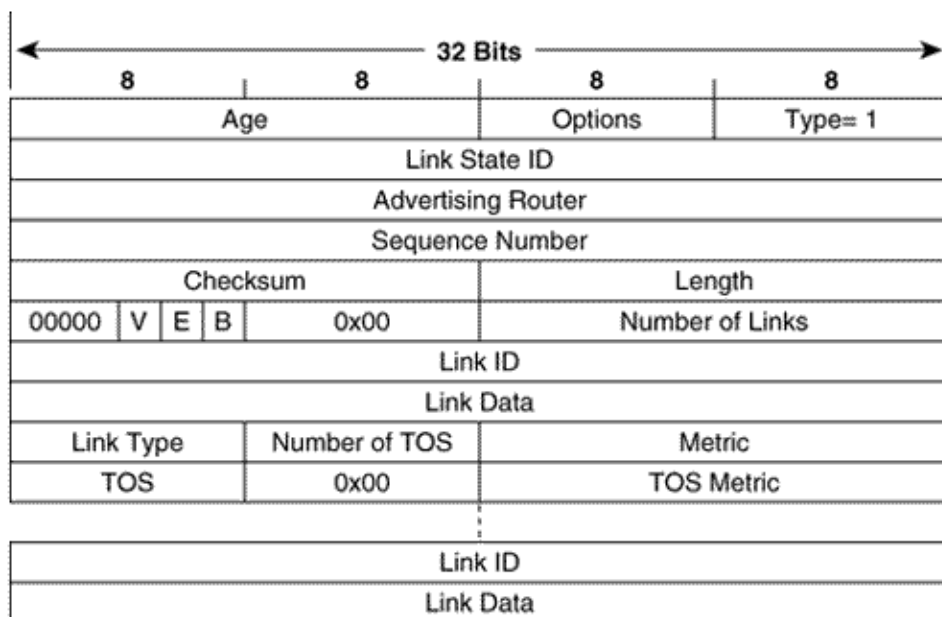
LSA 头部格式如下：



LSA 头部后面跟的，就是具体的 LSA 了，具体如下：

#### 1. Router LSA

这个最基本的 LSA 通告列出了路由器所有的链路或接口，并指明了它们的状态和沿每条链路方向出站的 cost。



**LinkstateID** 始发路由器的 RouterID

**V 位** 置为 1 时说明始发路由器是一条或多条具有完全邻接关系的虚链路的一个端点 ,这里被描述的区域是传送区域。

**E 位** 当始发路由器是一个 ASBR 路由器时,设置该位为 1

**B 位** 当始发路由器是一个 ABR 路由器时,设置该位为 1

从链路 ID 到链路数据字段 , 这段即为具体的 LSA 内容

LINKID、Link Date 的取值根据 OSPF link 类型不同而不同 :

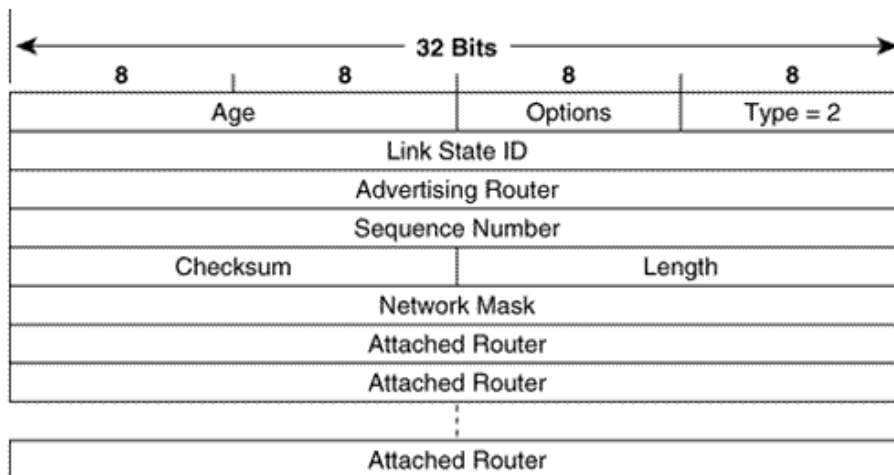
链路类型	连接	链路 ID 的值	链路数据
1	点到点连接到另一台路由器	邻居路由器的路由器 ID	和网络相连的始发路由器接口的 IP 地址
2	连接到一个传送网络 ( 常见以太网 )	DR 路由器的接口的 IP 地址	和网络相连的始发路由器接口的 IP 地址
3	连接到一个末梢网络	IP 网络或子网地址	网络的 IP 地址或子网掩码
4	虚链路	邻居路由器的路由器 ID	始发路由器接口的 MIB-II ifIndex 值

**TOS 号** RFC2328 不在支持。如果没有 Tos 度量和一条链路相关联,那么这个字段就设置为 0x00

**METRIC** 度量值

**TOS** CISCO 只支持为 0

## 2. Network LSA



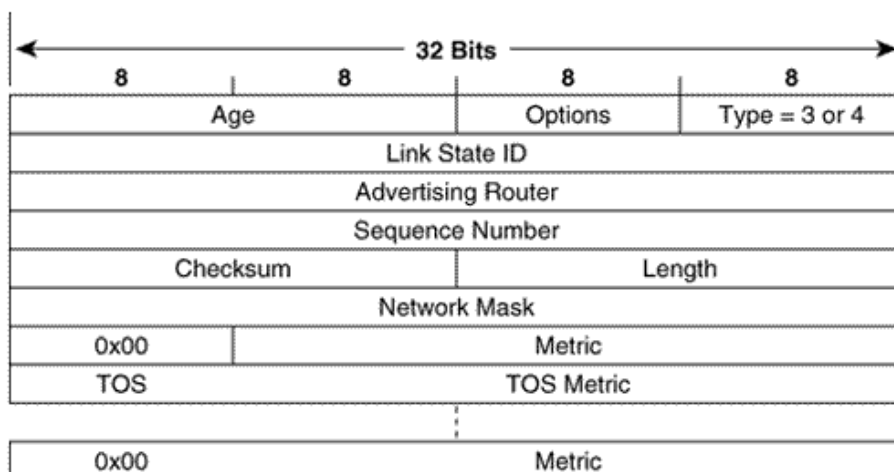
**链路状态 ID (LINK stat ID)** DR 路由器接口上的 IP 地址

**相连的路由器** 相连的路由器的 routerID (包括 DR 他自己)

注意：2 类 LSA 是不带 metric 的

### 3. Network summary LSA 和 ASBR summary LSA

两者有同样的格式，惟一的不同之处是它们所指的类型和链路状态 ID。

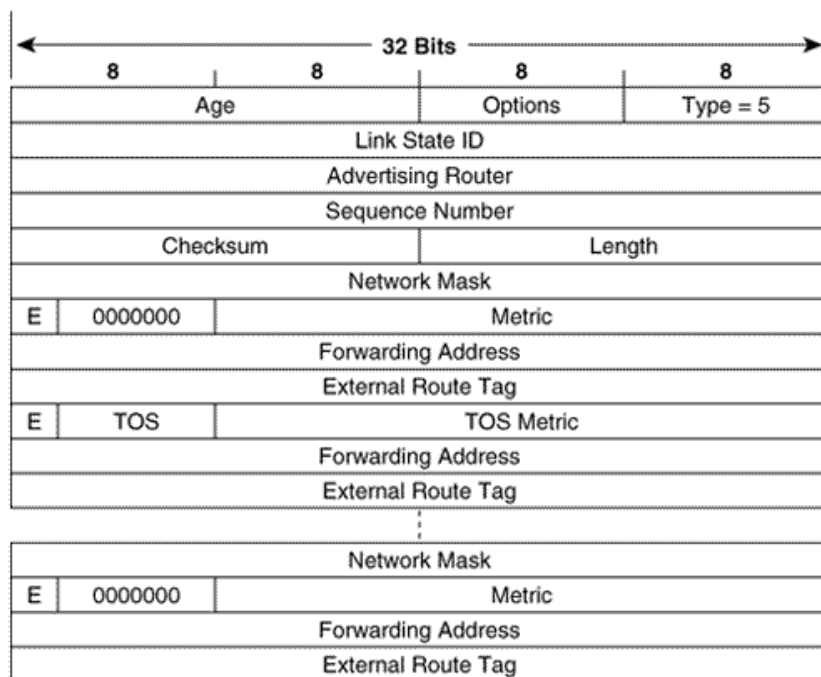


**链路状态 ID(Link state ID)** 对于类型 3 的 LSA 来说,它是所通告的网络或子网的 IP 地址。

对于类型 4 的 LSA 来说,链路状态 ID 是所通告的 ASBR 路由器的路由器 ID

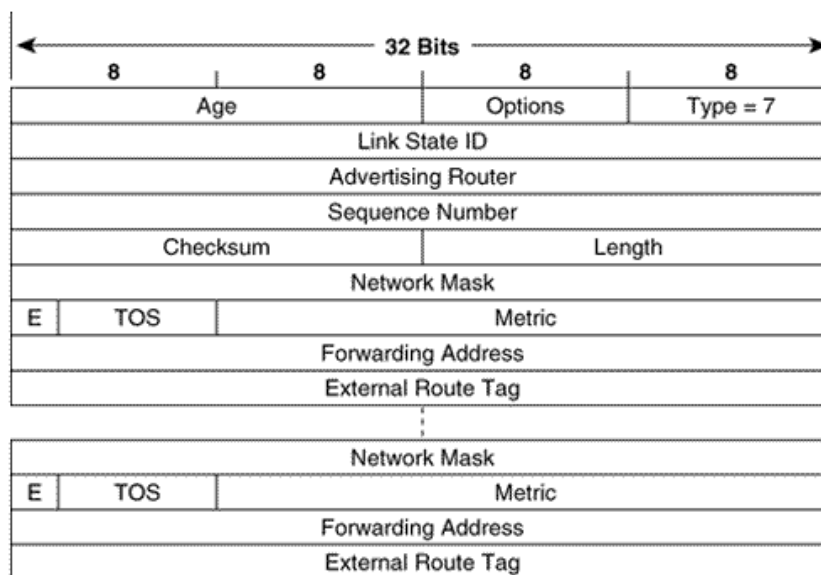
3 类 LSA 是由 ABR 产生的，ABE 指的是，必须有一个接口属于区域 0，并且有另外一个接口属于其他区域

### 4. 自治系统外 LSA



**链路状态 ID** AS 外部 LSA 的链路状态 ID 是指目的地的 IP 地址  
**E 或称外部度量位** 设置为 1，则为 E2 类型；为 0 则 E1 类型

## 5. NSSA 外部 LSA



## 3.5 OPTION 字段

Option 字段，出现在每一个 HELLO、DBD 以及每一个 LSA 中。

•	O	DC	EA	N/P	MC	E	T
---	---	----	----	-----	----	---	---

```
Options: 0x52 (O, L, E)
0... .... = DN: DN-bit is NOT set
.1.. .... = O: O-bit is SET
..0. .... = DC: Demand circuits are NOT supported
...1 .... = L: The packet contains LLS data block
.... 0... = NP: Nssa is NOT supported
.... .0.. = MC: NOT multicast capable
.... ..1. = E: ExternalRoutingCapability
```

## 1. N/P 位 (RFC1587)

其中 N/P 位为 1 位，N 位及 P 位分别只出现在 HELLO 及 LSA 报文中：

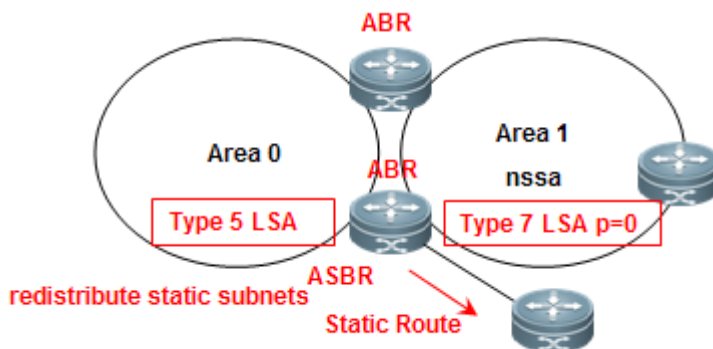
- 在 Hello 报文中：N bit 指示该路由器为 NSSA 区域路由器。当 N bit 被置 1 时 E bit 就必须被清零。
- 在 LSA 报文中：P bit 仅在 7 类 LSA 中出现

### 1) 置 1 时指示 NSSA 区域的 ABR 能够将这条 7 类 LSA 转成 5 类 LSA。

```
r1#sh ip os da nssa-external
OSPF Router with ID (1.1.1.1) (Process ID 1)
Type-7 AS External Link States (Area 1)
LS age: 9
Options: (No TOS-capability, No Type 7/5 translation, DC) //即 P 位被置 0
```

### 2) 为 0 时，ABR 将不能将该 7 类 LSA 转换成 5 类 LSA

只有 NSSA 区域的 ABR 重发布路由时，通告的 7 类 LSA 中的 P bit 才为 0



NSSA 区域 ABR 在重发布时，将 7 类 LSA 的 P 位置 0，通知其他 NSSA 区域 ABR 不要对该 LSA 进行 7 转 5，因为骨干区域内的其他路由器已经从骨干区域内收到该路由的 5 类 LSA，因此 NSSA 区域其他 ABR 执行 7 转 5 是没有意义的。

## 2. E 位

当始发路由器具有接受 AS 外部 LSA 的能力时,该位将被设置。

在所有的 AS 外部 LSA 和所有始发于骨干区域以及非末梢区域的 LSA 中，该位将设置为 1。而在所有始发于末梢区域的 LSA 当中，该位设置为 0。另外，可以在 Hello 数据包中使用该位来表明一个接口具有接收和发送类型 5 的 LSA 的能力。

E 位配置错误的邻居路由器将不能形成邻接关系，这个限制可以确保一个区域的所有路由器都同样地具有支持末梢区域的能力。

### 3. MC 位

该位描述是否组播扩展 OSPF

## 4 OSPF 网络类型及链路类型

### 4.1 网络类型 network-type

指的是 **OSPF 协议在接口上针对不同的二层数据链路层介质或封装而定义的**，例如如果接口二层封装协议是以太网，那么 OSPF 在这个接口的网络类型为 broadcast，如果接口的二层封装是 HDLC 或者 PPP，那么 OSPF 的网络类型是 P2P。OSPF 在不同的接口网络类型下，操作方式是不尽相同的。

使用 show ip ospf interface x 可以查看到接口的网络类型，如下：

```
R4#sh ip os int s 0/0
Serial0/0 is up, line protocol is up
Internet Address 10.1.24.4/24, Area 1
Process ID 1, Router ID 4.4.4.4, Network Type POINT_TO_POINT, Cost: 64
Transmit Delay is 1 sec, State POINT_TO_POINT,
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
oob-resync timeout 40
Hello due in 00:00:03
```

**OSPF 定义了如下几种网络类型：**

- 点到点 P2P
- 广播 Broadcast
- 非广播 Non-Broadcast

非广播又包括了 5 种运行模式：

- NBMA (RFC)
- P2MP (RFC)

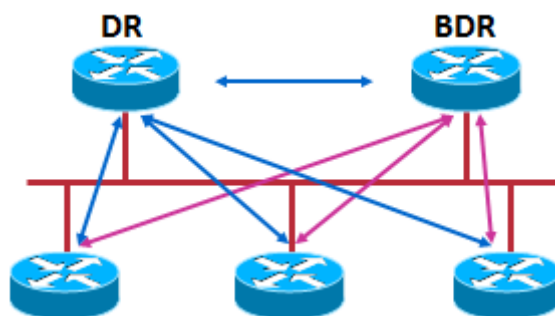
- P2MP nonbroadcast(CISCO)
- Broadcast(CISCO)
- P2P(CISCO)

### 1. 点到点类型



- 如果二层的协议为 PPP、HDLC 等，则 OSPF 网络类型为 P2P
- 如果帧中继接口类型为 P2P 的，则 OSPF 网络类型也为 P2P
- 不选举 DR、BDR
- 使用组播地址 224.0.0.5
- OSPF 能够根据二层封装自动检测到 P2P 网络类型

### 2. 广播多路访问型



- 通常出现在以太网
- 选举 DR、BDR
- 所有路由器均与 DR 及 BDR 建立邻接关系
- 使用组播地址 224.0.0.5 及 224.0.0.6

### 3. 非广播

详见红茶三杯 OSPF 在 NBMA 环境下的操作

## 4.2 链路类型 link-type

OSPF 除了定义网络类型，还定义了链路类型，注意链路类型与网络类型是两个概念，不要混淆。链路类型主要用于描述 OSPF 路由器的接口或邻居。在 1 类 LSA 中，可以看到始发该 LSA 的路由器所连接的所有链路( Link )、链路的类型以及相关的内容。



LS age: 1355  
Options: (No TOS-capability, DC)  
LS Type: Router Links  
Link State ID: 1.1.1.1  
Advertising Router: 1.1.1.1  
LS Seq Number: 8000001F  
Checksum: 0xFF44  
Length: 48  
Area Border Router  
AS Boundary Router  
Number of Links: 2

Link connected to: **another Router (point-to-point)**

(Link ID) Neighboring Router ID: 2.2.2.2

(Link Data) Router Interface address: 10.1.12.1

Number of TOS metrics: 0

TOS 0 Metrics: 64

Link connected to: **a Stub Network**

(Link ID) Network/subnet number: 10.1.12.0

(Link Data) Network Mask: 255.255.255.0

Number of TOS metrics: 0

TOS 0 Metrics: 64

**1 类 LSA 中，用于描述 Link 的 LINKID、Link Data 的取值根据 OSPF link 类型不同而不同：**

链路类型	连接	链路 ID 的值	链路数据
1	点到点连接到另一台路由器	邻居路由器的路由器 ID	和网络相连的始发路由器接口的 IP 地址
2	连接到一个传送网络（常见以太网）	DR 路由器的接口的 IP 地址	和网络相连的始发路由器接口的 IP 地址
3	连接到一个末梢网络	IP 网络或子网地址	网络的 IP 地址或子网掩码
4	虚链路	邻居路由器的路由器 ID	始发路由器接口的 MIB-II ifIndex 值

OSPF 链路类型分为以下几种：

### 1. Stub Network Link

在一个网段中只有一台 OSPF 路由器的情况下，该网段被 OSPF 链路类型定义为 Stub Network Link；因为一个网段中只有一台 OSPF 路由器，所以在这个网段就不可能有 OSPF 邻居，一个接口被通告进 OSPF，无论其二层链路是什么介质，只要在该接口上没有 OSPF 邻居，那么就是 Stub Network Link；Loopback 接口永远被定义为 Stub Network Link，默认使用 32 位掩码表示，无论将 Loopback 接口改为哪种 OSPF 网络类型（Network Type），始终改变不了它的 OSPF 链路类型（Link Type）属性，但可以改变它在 LSA 中的掩码长度。

## 2. Point-To-Point Link

OSPF 网络类型（Network Type）为 Point-To-Point 的接口，OSPF 链路类型（Link Type）为 Point-To-Point Link，但 Loopback 接口除外；而网络类型为点到多点（Point-To-Multipoint）的接口，同样链路类型也为 Point-To-Point Link。

Point-To-Point Link 可以是手工配置的地址（Numbered），也可以是借用的地址（Unnumbered），也可以是物理接口或逻辑子接口。

## 3. Transit Link

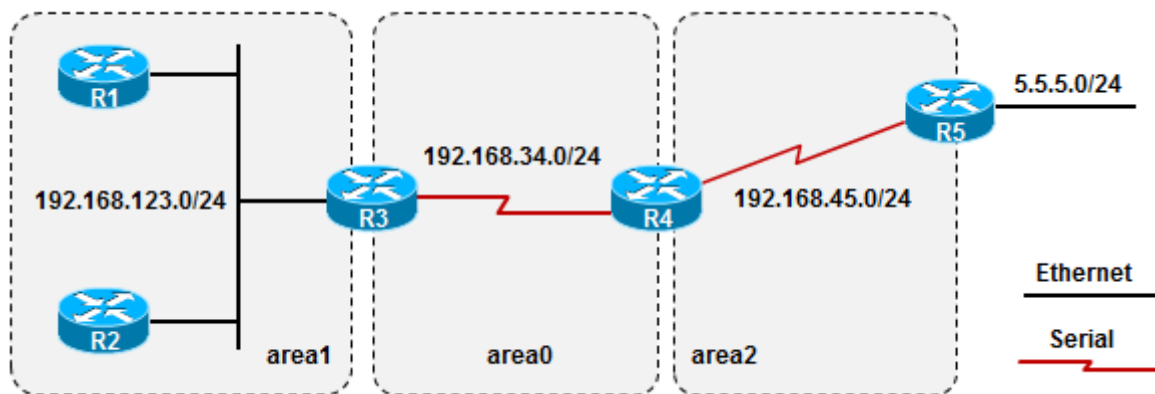
拥有两台或两台以上 OSPF 路由器的链路，简单理解为有邻居的 OSPF 接口就是 Transit Link，但网络类型为 Point-To-Point 和点到多点（Point-To-Multipoint）的接口除外，因为它们被定义为 Point-To-Point Link。

## 4. Virtual link

就是 OSPF 虚链路（Virtual Link），但稀奇的是，虚链路（Virtual Link）被定义为手工配置的地址（Numbered）的 Point-To-Point Link。

## 5 LSA 详解

### 5.1 各类 LSA 解析



我们本次 LSA 详解采用的拓扑结构相对比较简单，如上图所示，R1、R2、R3 之间连接到一个广播多路访问的介质上（以太网交换机），网段为 192.168.123.0/24，R1、R2、R3 的接口 IP 分别是 .1、.2、.3；R3、R4 以及 R4、R5 之间通过串行链路直连；所有路由器上均开启 LOOPBACK 接口，R1 的 LOOPBACK 接口 IP 为 1.1.1.1，R2 为 2.2.2.2，其他设备依次类推；所有的上述 LOOPBACK 接口仅作为路由器 Router-ID 的选取使用，并不做 network 宣告；R5 上将 LOOPBACK 接口重发布进 OSPF；

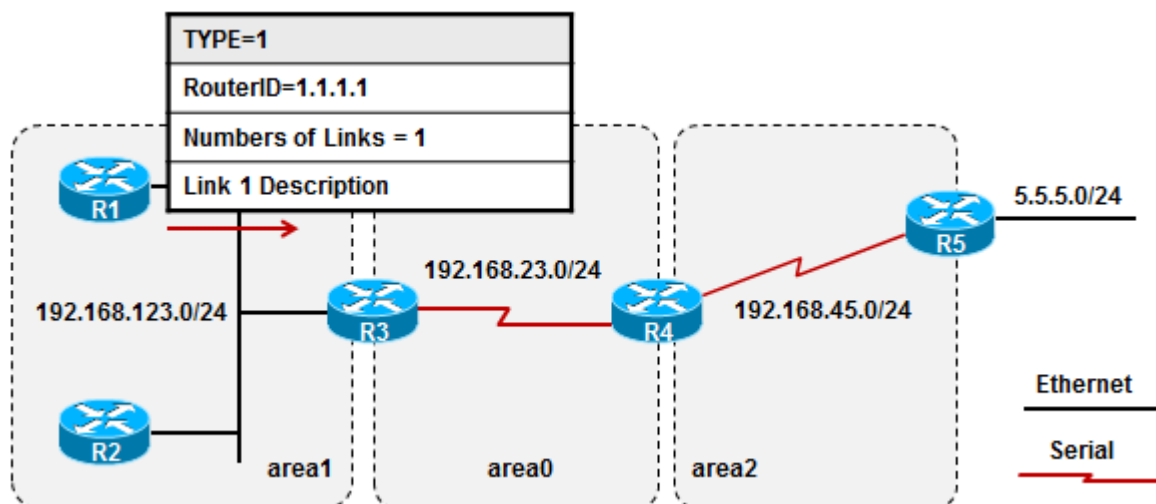
设备互联地址如上图所示。我们开始来逐一认识每种 LSA：

#### 1. Type 1 LSA : Router LSA

每一个运行 OSPF 的路由器均会产生 1 类 LSA，1 类 LSA 怎么理解？其实很简单，就是每台路由器描述一下自己“家门口的状况”，并且只会告诉给“全村的人”（本区域内泛洪）。

主要的功能有以下两点：

- 表示路由器的特殊角色，如 Virtual-link、ABR、ASBR  
这是通过 1 类 LSA 中相关的 V、B、E 位置为来实现的
- 描述本路由器在某个区域内部与的直连链路（接口）及接口 COST 值



在这个环境中，每台路由器均会产生 1 类 LSA，并且这些 1 类 LSA 只会在本区域内泛洪。譬如 area1 内，泛洪的 1 类 LSA 有哪些呢？

**R1#sh ip ospf database**

OSPF Router with ID (1.1.1.1) (Process ID 1)					
Router Link States (Area 1)					
Link ID	ADV Router	Age	Seq#	Checksum	Link count
1.1.1.1	1.1.1.1	59	0x80000002	0x00213B	1
2.2.2.2	2.2.2.2	60	0x80000002	0x00E270	1
3.3.3.3	3.3.3.3	60	0x80000003	0x00A5A2	1

可以看到，R1、R2、R3 均在 area1 内产生了各自的 1 类 LSA，每个 1 类 LSA 中均包含 1 个 LINK，也即三台路由器各有一个直连接口连接到这个区域内。

接着来看 1 类 LSA 详细内容，譬如 R1，R1 有一个直连接口，连接到一个 MA 网络 192.168.123.0，因此 R1 产生的 1 类 LSA（在 area1 内泛洪）中，包含一条链路，详细输出如下：

**R1#sh ip ospf database router 1.1.1.1**

```

LS age: 244
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1 // 产生该 LSA 的路由器的 Router-ID
Advertising Router: 1.1.1.1
LS Seq Number: 80000002
Checksum: 0x1F3C
Length: 36
Number of Links: 1 // R1 连入本区域的接口（链路）数量
    
```

```
Link connected to: a Transit Network      // R1 通过以太网连接到一个 MA 网络 123.0
(Link ID) Designated Router address: 192.168.123.3  // DR 的接口 IP
(Link Data) Router Interface address: 192.168.123.1  // 连接到这个 MA 的本地接口 IP
Number of TOS metrics: 0
TOS 0 Metrics: 1                          // 接口 COST=1
```

这就是 1 类 LSA，“Link connected to” 开始就是该路由器的所有直连链路的相关信息。

R2 及 R3 在 Area1 中的 1 类 LSA 大体上类似，那么 R3 在 area0 中泛洪的 1 类 LSA 是如何？

### R1#sh ip ospf database router 3.3.3.3

```
OSPF Router with ID (1.1.1.1) (Process ID 1)
Router Link States (Area 1)
Routing Bit Set on this LSA
LS age: 584
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 3.3.3.3
Advertising Router: 3.3.3.3
LS Seq Number: 80000003
Checksum: 0xA5A2
Length: 36
Area Border Router      //标识这台路由器为 ABR，这其实是 LSA1 报文的 B 位置 1
Number of Links: 1
```

```
Link connected to: a Transit Network
(Link ID) Designated Router address: 192.168.123.3
(Link Data) Router Interface address: 192.168.123.3
Number of TOS metrics: 0
TOS 0 Metrics: 1
```

我们注意到 R3 产生的 1 类 LSA 中，有着“Area Border Router”字样，标识该路由器是 ABR，而这，正是使得其产生的 1 类 LSA 中 B 位=1

再来看一下 R3 在 area0 中泛洪的 1 类 LSA：

### R3#show ip ospf database router 3.3.3.3

```
OSPF Router with ID (3.3.3.3) (Process ID 1)

Router Link States (Area 0)
```

LS age: 1933  
Options: (No TOS-capability, DC)  
LS Type: Router Links  
Link State ID: 3.3.3.3  
Advertising Router: 3.3.3.3  
LS Seq Number: 80000009  
Checksum: 0x3B1E  
Length: 48  
Area Border Router  
Number of Links: 2

**Link connected to: another Router (point-to-point)**

(Link ID) Neighboring Router ID: 4.4.4.4  
(Link Data) Router Interface address: 192.168.34.3  
Number of TOS metrics: 0  
TOS 0 Metrics: 64

**Link connected to: a Stub Network**

(Link ID) Network/subnet number: 192.168.34.0  
(Link Data) Network Mask: 255.255.255.0  
Number of TOS metrics: 0  
TOS 0 Metrics: 64

R3 在 area0 中只有一个接口，由于是 serial 接口，OSPF 用了两条 LINK 来描述。

我们来总结一下，针对不同的链路类型，OSPF 1 类 LSA 的有不同的内容：

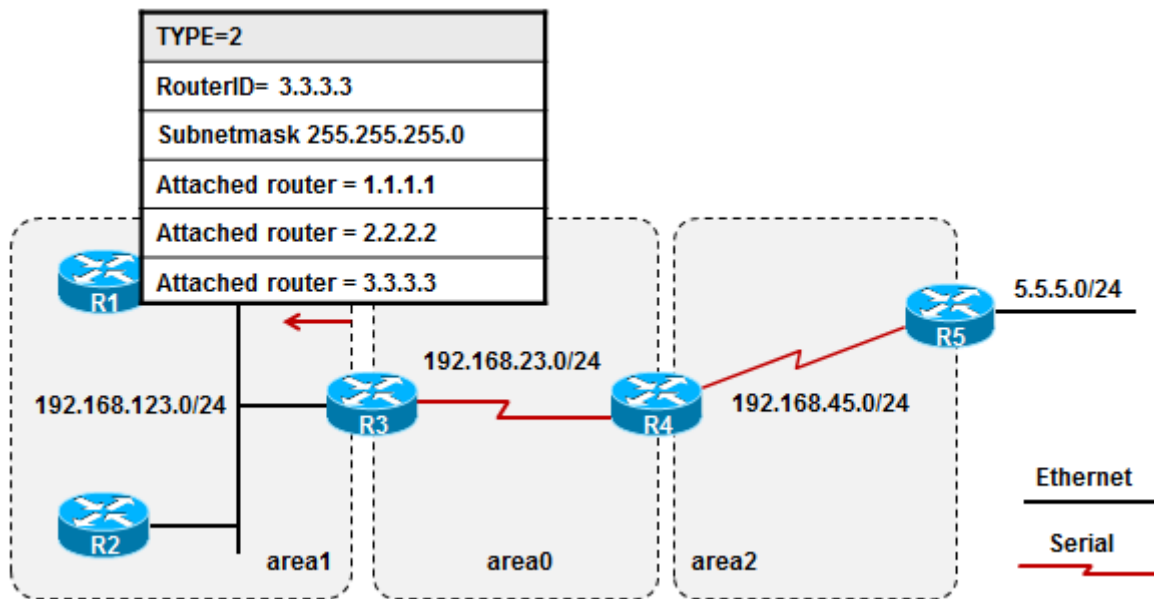
链路类型	连接	LINK ID	LINK DATA
1	点到点连接到另一台路由器	邻居路由器的路由器 ID	和网络相连的始发(本地)路由器接口的 IP 地址
2	连接到一个传送网络(常见以太网)	DR 路由器的接口的 IP 地址	和网络相连的始发路由器接口的 IP 地址
3	连接到一末梢网络	IP 网络或子网地址	网络的 IP 地址或子网掩码
4	虚链路	邻居路由器的路由器 ID	始发路由器接口的 MIB-II ifIndex 值

## 2. Type 2 LSA : Network LSA

在 MA 网络中，会选举 DR、BDR，而所有的 Drother 都只能和 DR 及 BDR 建立邻接关系。

从某种层面上说，DR 实际上代表了这个 MA 网络，在本区域内泛洪 2 类 LSA，来呈现该 MA 网络中的所有路由器。因此 2 类 LSA 仅存在于有 MA 网络的区域中，并且由 DR 发送，用来描述这个 MA 网络中的所有路

由器 ( 的 Router-ID )。



在上例中，R3 作为 123.0 网络的 DR，因此 R1、R2 都只和 R3 建立邻接关系。这时候 R3 就成了这个 MA 网络的代表者，其发送 2 类 LSA，该 LSA 包含的内容如图，详细信息见下，注意该 2 类 LSA 也只是在 area1 内泛洪。

**R1#show ip ospf database network**

```

OSPF Router with ID (1.1.1.1) (Process ID 1)
  Net Link States (Area 1)
    Routing Bit Set on this LSA
    LS age: 952
    Options: (No TOS-capability, DC)
    LS Type: Network Links
    Link State ID: 192.168.123.3 (address of Designated Router)
    Advertising Router: 3.3.3.3
    LS Seq Number: 80000001
    Checksum: 0x4CDD
    Length: 36
    Network Mask: /24

    Attached Router: 3.3.3.3
    Attached Router: 1.1.1.1
    Attached Router: 2.2.2.2
    
```

**总结一下：**

LSA2 描述 TransNet ( 包括 Broadcast 和 NBMA 网络 ) 网络信息；

由 DR 产生，描述其在该 MA 网络上连接的所有路由器的 RouterID ( 包括 DR ) 以及该 MA 网络的掩码



LSA 类型 2 只在本区域 Area 内洪泛，不允许跨越 ABR；

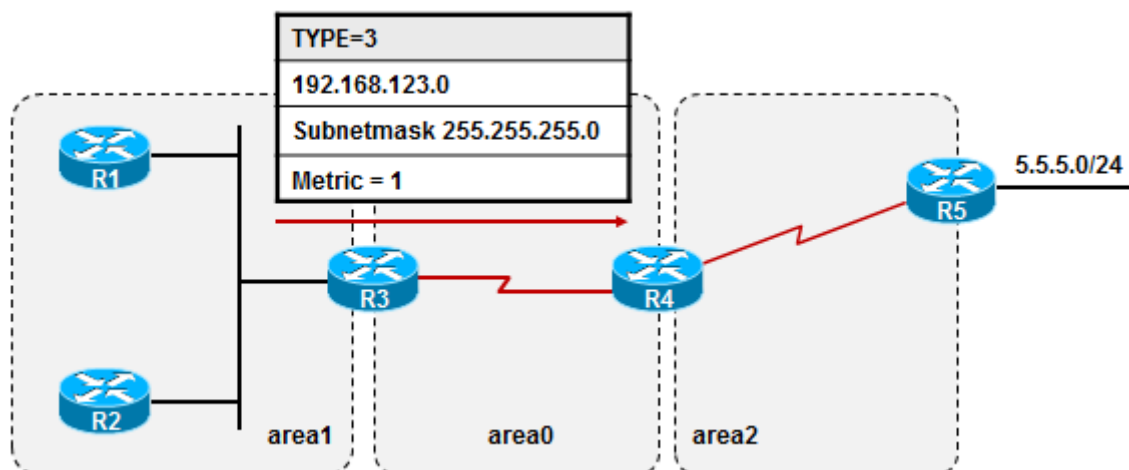
LSA 类型 2 中没有 COST 字段（因此需借助 1 类 LSA 来进行 SPF 算法）

**得益于 1、2 类 LSA，OSPF 在一个区域内的路由计算就没有问题了，由此产生的路由，在路由表中的标记为“O”，表示本区域内的路由。**

### 3. Type 3 LSA : Network summary LSA

前两类 LSA 解决了区域内路由计算的问题，那么区域间呢？如果路由器需要访问其他区域呢？这时就需要 3 类 LSA。3 类 LSA 是网络汇总 LSA，这里的汇总，其实翻译为归纳更贴切，它和路由汇总是完全不同的概念。由 ABR 同时属于两个以上的区域（其中必须有骨干区域），它知道这些区域的 1、2 类 LSA，那么就能做一件事：将某个区域的 1、2 类 LSA，做个归纳，然后为其他区域生成 3 类 LSA 并泛洪到其他区域中，如此一来，区域间的路由计算就没问题了。

因此 3 类 LSA，由 ABR 产生：



上图中，R3 两个接口分属 area1 及 area0，它是一台 ABR，因此它将 area1 内的 LSA 做个归纳，为 area0 生成 3 类 LSA，实际上就是 192.168.123.0 这个网段的信息，然后泛洪到 area0 中，接着，再被 R4 泛洪到 area2 中。这样，R4、R5 就都能根据这个 LSA3 计算到 123.0 的路由。

当然，R1、R2 也都能通过 R3 泛洪过来的 3 类 LSA 了解到 34.0、45.0 的路由信息。

#### R3#show ip ospf database

```
.....
```

Summary Net Link States (Area 0)				
Link ID	ADV Router	Age	Seq#	Checksum
192.168.45.0	4.4.4.4	1533	0x80000001	0x00F065
192.168.123.0	3.3.3.3	1562	0x80000001	0x003912

Area0 中，有 2 个 3 类 LSA 在泛洪，分别描述的是 192.168.45.0、192.168.123.0 两个网络。

可以再详细点查看一下 3 类 LSA 的内容：

### R3#show ip ospf database summary 192.168.123.0

```

OSPF Router with ID (3.3.3.3) (Process ID 1)
  Summary Net Link States (Area 0)

LS age: 1886
Options: (No TOS-capability, DC, Upward)
LS Type: Summary Links(Network)
Link State ID: 192.168.123.0 (summary Network Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000001
Checksum: 0x3912
Length: 28
Network Mask: /24

TOS: 0 Metric: 1
  
```

以上是在 area0 中泛洪的 192.168.123.0 这条三类 LSA 的详细内容，注意到 3 类 LSA 的 link state ID 也就是这个 LSA 所描述的子网号，同时 LSA 中还描述了这个子网的掩码以及到达该子网的 metric，也就是 cost 值，这里 cost=1，其实就是 R3 的 fast0/0 口的接口 cost。

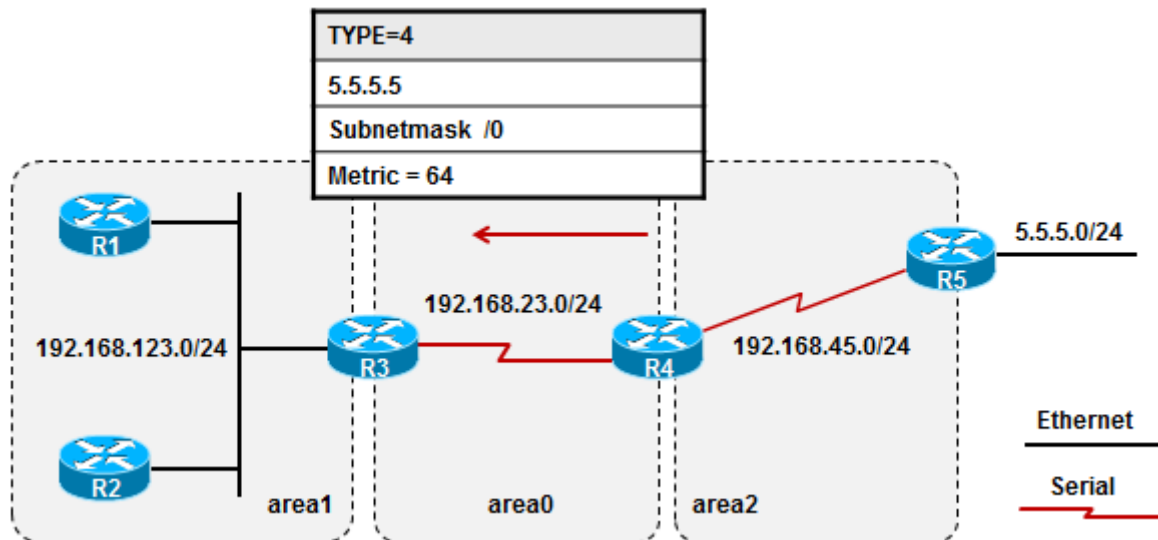
#### 4. Type 4 LSA : ASBR summary LSA

4 类 LSA 是一个指向 ASBR 的 LSA，由该 ASBR 所在的区域中的 ABR 产生（这点要格外留意）。ASBR 作为域边界路由器，将外部的路由通过重发布的方式注入了 OSPF 域，这些外部路由在 OSPF 中进行传递（这些外部路由是以 5 类 LSA 的形式在域内传播），而我们 OSPF 内部的路由器如果想前往这些外部网段，则需要同时具备两个条件：

- **知道外部的路由（这通过重发布的动作，已经完成了注入，借助 5 类 LSA 完成传播）**
- **知道完成这个重分发动作的 ASBR 的位置**

关键在于第二点，与 ASBR 在同一区域的区域内部路由器（例如本实验中的 R4），能通过 ASBR 产生的 1 类 LSA 知道该 ASBR 的位置（1 类 LSA 中 E 位=1，所以与 ASBR 同区域的路由器都知道），但是问题来了，1 类 LSA 的泛洪范围是本区域内，那么该区域外的路由器，如何得知这台 ASBR 的位置呢？那么就需要借助 4 类 LSA 了。

因此 4 类 LSA 由 ABR 产生，用来告诉与 ASBR 不在同一个区域内的其他 OSPF Router 关于 ASBR 的信息。



上图中 R5 做为 ASBR ,在了路由重发布 ,将 5.5.5.0/24 的直连网段发布进了 OSPF。那么 R1、R2、R3、R4 如何能够访问 5.5.5.0 呢？上面已经说了，条件 1，是需要 5.5.5.0 的路由信息，这通过重发布将这些外部路由引入就 OK。条件 2，是需要了解 ASBR，也就是 R5 的信息。

R5 作为 OSPF 路由器，其本身会泛洪 1 类 LSA ( 在 area2 内部 )，并且该 LSA1 的 E 位被置 1，用以表示其本身是一台 ASBR，如此一来，R4 就能通过该 1 类 LSA，知道 ASBR 的存在，那么 R4 本身去往 5.5.5.0 就没问题了，但是 R1、R2、R3 呢？他们是无法收到上述的 1 类 LSA 的。因此 R4，作为 ASBR 所在区域的 ABR，就承担起解决这个问题的关键，它会向 area0 下发一条 4 类的 LSA，该 4 类 LSA 会进一步被泛洪，那么 R1、R2、R3 就都能了解到 R5，也就是 ASBR 了，所以条件 2 就具备了，问题也解决了。下面的输出，即是 4 类 LSA：

#### R4#show ip ospf database asbr-summary

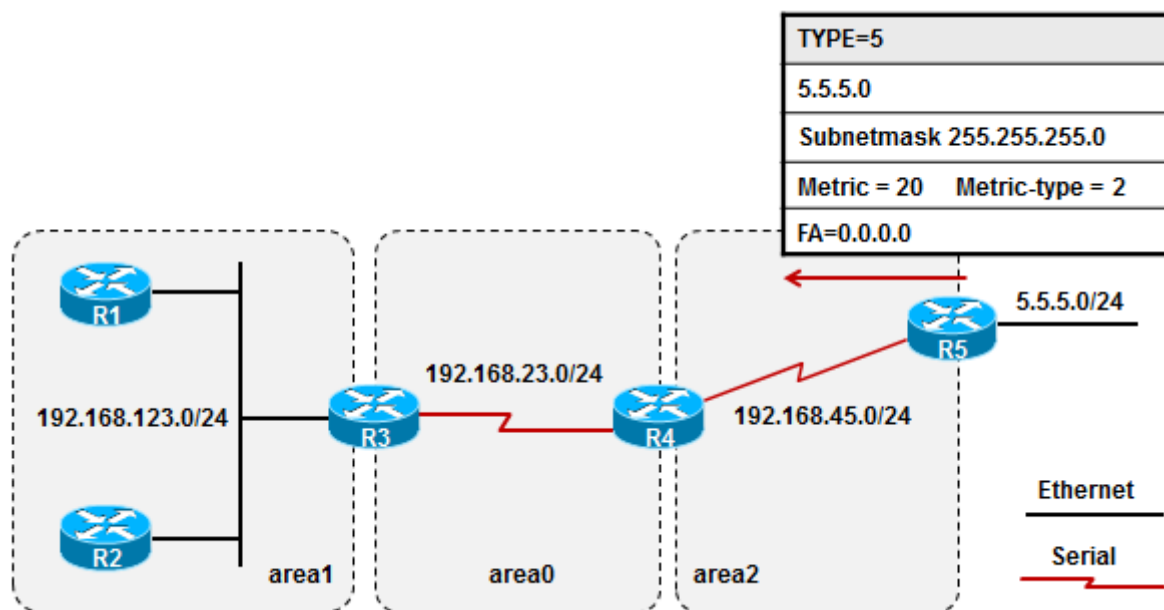
```

OSPF Router with ID (4.4.4.4) (Process ID 1)
  Summary ASB Link States (Area 0)
    LS age: 448
    Options: (No TOS-capability, DC, Upward)
    LS Type: Summary Links(AS Boundary Router)
    Link State ID: 5.5.5.5 (AS Boundary Router address)    // ASBR 的 RouterID
    Advertising Router: 4.4.4.4    //通告该 4 类 LSA 的路由器的 routerID，也就是 R4
    LS Seq Number: 80000002
    Checksum: 0x9C3A
    Length: 28
    Network Mask: /0
    TOS: 0   Metric: 64    //到达 ASBR 的开销
  
```

## 5. Type 5 LSA : AS external LSA

外部路由，被重发布进 OSPF 以后，将产生 5 类的 LSA，在 OSPF 与中进行传递；

因此 5 类 LSA 由 ASBR 产生



上图中，R5 做为 ASBR 将 5.5.5.0 这条直连网段重发布进了 OSPF，因此 R5 产生一个关于该外部路由的 5 类 LSA，其实就是描述的这条外部路由 5.5.5.0/24，体现如下：

**R4#show ip ospf database external**

```

OSPF Router with ID (4.4.4.4) (Process ID 1)
  Type-5 AS External Link States
    Routing Bit Set on this LSA
    LS age: 584
    Options: (No TOS-capability, DC)
    LS Type: AS External Link
    Link State ID: 5.5.5.0 (External Network Number ) //外部路由的网络号
    Advertising Router: 5.5.5.5 //通告该外部路由的 ASBR 的 routerID
    LS Seq Number: 80000002
    Checksum: 0x9AE1
    Length: 36
    Network Mask: /24 //外部路由的掩码
      Metric Type: 2 (Larger than any link state path) //外部路由的 metric-type
      TOS: 0
      Metric: 20 //metric，在重发布的时候指定
      Forward Address: 0.0.0.0 //FA 地址，是一个知识难点
  
```

## External Route Tag: 0

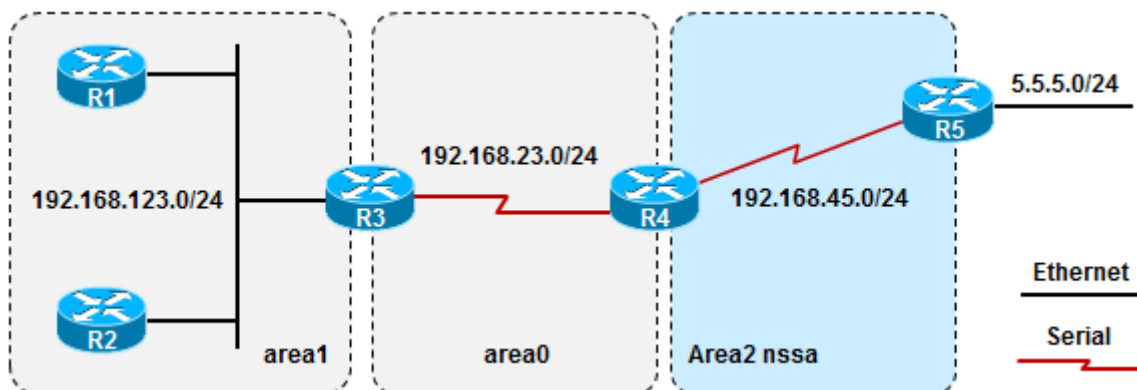
关于 type5 LSA 的 metric type, E1 及 E2 的区别请见下一小节

### 6. Type 6 LSA : Group-membership-LSA

这里不做讨论

### 7. Type 7 LSA : NSSA external LSA

7 类 LSA 是一种非常特殊的 LSA 要注意的是这种 LSA 作为一种描述外部路由的 LSA 它只能被在 NSSA 中进行泛洪,不能跨越 NSSA 进入常规区域。NSSA 区域阻挡从骨干区域 area0 中过来的 5 类 LSA 进入,同时允许 NSSA 本地始发外部路由,这些外部路由以 7 类 LSA 的形式在本地 NSSA 中进行泛洪,当这些 7 类 LSA 到达 NSSA 的 ABR 时,由该 ABR 负责将这些 7 类 LSA 转换成 5 类 LSA,方可注入骨干区域。



留意一下上图,在上图中,我们将 area2 配置为 NSSA,接下去再观察 R5 的 LSDB:

R5#show ip ospf da

OSPF Router with ID (5.5.5.5) (Process ID 1)				
Router Link States (Area 2)				
Link ID	ADV Router	Age	Seq#	Checksum Link count
4.4.4.4	4.4.4.4	73	0x80000006	0x0048E8 2
5.5.5.5	5.5.5.5	71	0x80000007	0x00E249 2
Summary Net Link States (Area 2)				
Link ID	ADV Router	Age	Seq#	Checksum
192.168.34.0	4.4.4.4	83	0x80000004	0x000A4E
192.168.123.0	4.4.4.4	83	0x80000004	0x003DC0
Type-7 AS External Link States (Area 2)				

Link ID	ADV Router	Age	Seq#	Checksum Tag
5.5.5.0	5.5.5.5	75	0x80000001	0x00C90E 0

我们看到，重发布注入进来的 5.5.5.0 这条外部路由，是 7 类 LSA 所描述的。再看一下详细的内容：

#### R5#show ip ospf database nssa-external

```

OSPF Router with ID (5.5.5.5) (Process ID 1)
  Type-7 AS External Link States (Area 2)

LS age: 144
Options: (No TOS-capability, Type 7/5 translation, DC)
LS Type: AS External Link
Link State ID: 5.5.5.0 (External Network Number )
Advertising Router: 5.5.5.5
LS Seq Number: 80000001
Checksum: 0xC90E
Length: 36
Network Mask: /24

Metric Type: 2 (Larger than any link state path)
TOS: 0
Metric: 20
Forward Address: 192.168.45.5
External Route Tag: 0
  
```

我们发现 7 类 LSA 的结构与 5 类 LSA 并无明显差异，其实在数据包方面，也并没有明显的差别。当然，他们的差异从应用或者规划的角度而言，是非常明显的，这里不做详细描述。

我们再看一下 R4 的 OSPF LSDB

#### R4#show ip ospf database

```

OSPF Router with ID (4.4.4.4) (Process ID 1)

  Router Link States (Area 0)

Link ID      ADV Router   Age      Seq#          Checksum Link count
3.3.3.3      3.3.3.3      93       0x80000004    0x004519 2
4.4.4.4      4.4.4.4      29       0x80000005    0x00E86D 2

  Summary Net Link States (Area 0)

Link ID      ADV Router   Age      Seq#          Checksum
192.168.45.0 4.4.4.4      49       0x80000003    0x00EC67
  
```

192.168.123.0	3.3.3.3	93	0x80000003 0x003514	
Router Link States (Area 2)				
Link ID	ADV Router	Age	Seq#	Checksum Link count
4.4.4.4	4.4.4.4	18	0x80000006	0x0048E8 2
5.5.5.5	5.5.5.5	18	0x80000007	0x00E249 2
Summary Net Link States (Area 2)				
Link ID	ADV Router	Age	Seq#	Checksum
192.168.34.0	4.4.4.4	29	0x80000004	0x000A4E
192.168.123.0	4.4.4.4	29	0x80000004	0x003DC0
Type-7 AS External Link States (Area 2)				
Link ID	ADV Router	Age	Seq#	Checksum Tag
5.5.5.0	5.5.5.5	24	0x80000001	0x00C90E 0
Type-5 AS External Link States				
Link ID	ADV Router	Age	Seq#	Checksum Tag
5.5.5.0	4.4.4.4	10	0x80000001	0x007C69 0

我们看到 R4 收到 R5 始发的 7 类 LSA 后，装入的自己的 LSDB，同时他还做了一件利国利民的事情，那就是将这条 LSA “转换” 成了 5 类 LSA，并将它传递给骨干区域，好让其他区域的 OSPF 路由器知道这条外部路由。当然，在 R4 的路由表中，该条 5.5.5.0 的外部路由：

#### R4#show ip route

```
O IA 192.168.123.0/24 [110/65] via 192.168.34.3, 00:09:01, Serial0/0
C    192.168.45.0/24 is directly connected, Serial0/1
    4.0.0.0/24 is subnetted, 1 subnets
C      4.4.4.0 is directly connected, Loopback0
    5.0.0.0/24 is subnetted, 1 subnets
O N2    5.5.5.0 [110/20] via 192.168.45.5, 00:09:01, Serial0/1
C    192.168.34.0/24 is directly connected, Serial0/0
```

## 5.2 E1、E2 的比较 (N1 及 N2 类似)

Type5 LSA 由 ASBR 产生，告诉相同自治区的路由器通往外部 AS 的路由。



## 1. E1/E2 路由的优选问题：

E1 永远优于 E2 路由，也就是说，如果一台 OSPF 路由器，同时学习到去往同一个目的地的两条外部路由，一条为 OE1 一条为 OE2，则不论各自 metric 如何，OE1 的路由永远优选。

### ● E1 路由之间比较

当本地接收到多条描述同一外部路由信息的类型 1 外部 LSA（都是 OE1）时，需要比较每条外部 LSA 中携带的 Metric 与本地到达产生该 LSA 的 ASBR 的 Metric 之和（即路由表中显示的 Metric），并优选值小的类型 1 外部 LSA

如果比较的结果 Metric 之和相等，则产生路由的负载均衡。

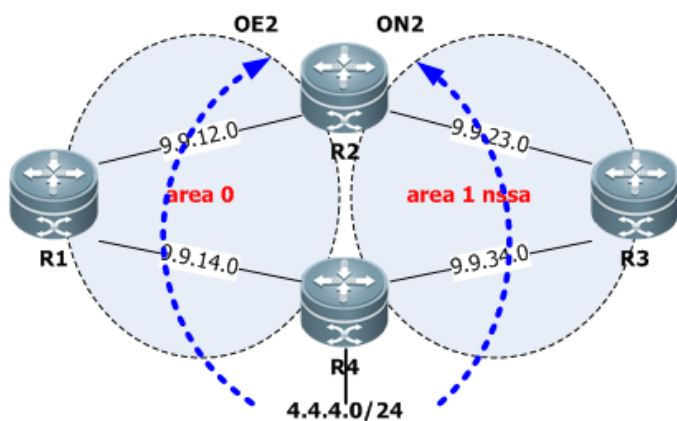
### ● E2 路由之间比较

当本地接收到多条描述同一外部路由信息的类型 2 外部 LSA（都是 OE2）时，按照如下顺序比较不同的 Metric 值，从而实现优选：

- 1、首先比较外部 LSA 本身携带的 Metric，优选值小的外部 LSA。
- 2、如果外部 LSA 携带的 Metric 值相同，则继续按照下面原则比较
- 3、比较本地到达通告每条 LSA 的 ASBR/FA 地址的 Metric，优选值小的。
- 4、如果本地到达通告每条 LSA 的 ASBR/FA 地址的 Metric 值相同，则产生路由负载均衡。

## 2. ON2 和 OE2 的选择问题

当 metric 不一样时，比较各自到 ASBR/FA 地址的内部 metric，如果 metric 一样，CISCO 优选择 OE2

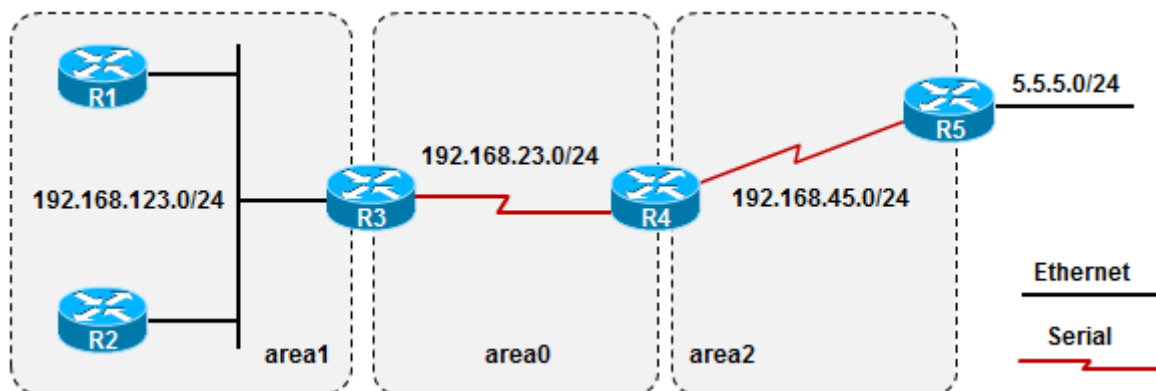


如图 R2 收到 4.4.4.0 的 OE2 和 ON2，在两者 metric 相等时，CISCO 默认选择 OE2，但是更改 R2 接口 cost 后，可以看到相应变化。

当上面的 OE2 和 ON2 两 metric 相等，各自到 ASBR 或 FA 地址的 metric 相当时，CISCO 默认优选 OE2，当两 metric 相等，但各自到 FA 或 ASBR 的 metric 不等时，优选这个距离小的。

## 5.3 查看 LSDB

经过前面的讲解，相信大家对于各种类型的 LSA 都能掌握了，那么下面，我们就来全面的分析一下实验环境中的 LSDB。OSPF 最底层的东西就是 LSA，LSA 在 OSPF 路由选择域中泛洪并发挥着自己的作用，OSPF Router 搜集到 LSA 后都存储在自己的 LSDB（链路状态数据库）里，并且根据这些 LSA 进行 SPF 算法的计算，最终得出前往每个目的地最优的路径。因此在从事 OSPF 的网络建设及维护过程中，学会读懂 LSDB，学会读懂每一类的 LSA 是非常关键的。



还是这张拓扑，完成基本配置后，我们看看 R1 的 LSDB。

**R1#show ip ospf database**

OSPF Router with ID (1.1.1.1) (Process ID 1)					
Router Link States (Area 1) // 区域 1 中泛洪的 1 类 LSA					
Link ID	ADV Router	Age	Seq#	Checksum	Link count
1.1.1.1	1.1.1.1	1203	0x80000003	0x001F3C	1
2.2.2.2	2.2.2.2	1099	0x80000003	0x00E071	1
3.3.3.3	3.3.3.3	1050	0x80000004	0x00A3A3	1
Net Link States (Area 1) // 区域 1 中泛洪的 2 类 LSA					
Link ID	ADV Router	Age	Seq#	Checksum	
192.168.123.3	3.3.3.3	1050	0x80000002	0x004ADE	
Summary Net Link States (Area 1) // 区域 1 中泛洪的 3 类 LSA					
Link ID	ADV Router	Age	Seq#	Checksum	
192.168.34.0	3.3.3.3	1050	0x80000002	0x0086DD	
192.168.45.0	3.3.3.3	1050	0x80000002	0x008F89	
Summary ASB Link States (Area 1) // 区域 1 中泛洪的 4 类 LSA					

Link ID	ADV Router	Age	Seq#	Checksum
5.5.5.5	3.3.3.3	1050	0x80000002	0x003D5D

#### Type-5 AS External Link States // 5 类 LSA

Link ID	ADV Router	Age	Seq#	Checksum Tag
5.5.5.0	5.5.5.5	1007	0x80000002	0x009AE1 0

R1 属于常规区域内部路由器，因此其 LSDB 相对比较好读，分别有 1、2、3、4、5 类 LSA；其中 1 类 LSA 产生自本区域（area1）的三台路由器 R1、R2、R3。2 类 LSA 产生自本 MA 网络的 DR 也就是 R3。3 类 LSA 产生自本区域的 ABR 也就是 R3，目的是告诉本区域内的路由器 OSPF 域内其他 area 的网络信息。4 类 LSA 也由 ABR R3 产生，用于告知本区域内的路由器关于 ASBR 的信息。5 类 LSA 是用于描述域外路由的，由 ASBR R5 产生并最终被泛洪到本区域。R2 的 LSDB 应该和 R1 类似，这里不再赘述。

### R3#show ip ospf database

#### OSPF Router with ID (3.3.3.3) (Process ID 1)

#### Router Link States (Area 0) // 区域 0 中泛洪的 1 类 LSA

Link ID	ADV Router	Age	Seq#	Checksum	Link count
3.3.3.3	3.3.3.3	1338	0x80000003	0x004718	2
4.4.4.4	4.4.4.4	1311	0x80000003	0x00E673	2

#### Summary Net Link States (Area 0) // 区域 0 中泛洪的 3 类 LSA

Link ID	ADV Router	Age	Seq#	Checksum
192.168.45.0	4.4.4.4	1311	0x80000002	0x00EE66
192.168.123.0	3.3.3.3	1338	0x80000002	0x003713

#### Summary ASB Link States (Area 0) // 区域 0 中泛洪的 4 类 LSA

Link ID	ADV Router	Age	Seq#	Checksum
5.5.5.5	4.4.4.4	1311	0x80000002	0x009C3A

#### Router Link States (Area 1) // 区域 1 中泛洪的 1 类 LSA

Link ID	ADV Router	Age	Seq#	Checksum	Link count
1.1.1.1	1.1.1.1	1494	0x80000003	0x001F3C	1
2.2.2.2	2.2.2.2	1388	0x80000003	0x00E071	1
3.3.3.3	3.3.3.3	1338	0x80000004	0x00A3A3	1

Net Link States (Area 1) // 区域 1 中泛洪的 2 类 LSA				
Link ID	ADV Router	Age	Seq#	Checksum
192.168.123.3	3.3.3.3	1338	0x80000002	0x004ADE

Summary Net Link States (Area 1) // 区域 0 中泛洪的 3 类 LSA				
Link ID	ADV Router	Age	Seq#	Checksum
192.168.34.0	3.3.3.3	1340	0x80000002	0x0086DD
192.168.45.0	3.3.3.3	1340	0x80000002	0x008F89

Summary ASB Link States (Area 1) // 区域 0 中泛洪的 4 类 LSA				
Link ID	ADV Router	Age	Seq#	Checksum
5.5.5.5	3.3.3.3	1340	0x80000002	0x003D5D

Type-5 AS External Link States // 5 类 LSA					
Link ID	ADV Router	Age	Seq#	Checksum	Tag
5.5.5.0	5.5.5.5	1297	0x80000002	0x009AE1	0

R3 的 LSDB 与 R1 有一个很直观的变化，那就是 R3 作为 ABR，同时连接到了 area0 及 area1，因此 R3 需为两个区域同时创建并泛洪 LSA。这里注意到 area0 中，并没有 2 类 LSA 存在，这是因为 2 类 LSA 只存在与 MA 网络中，而 area0 中并没有 MA 网络，因此没有 2 类 LSA，而 area1 中的 2 类 LSA，自然是不能超越区域泛洪到 area0 中的，当然，也没有这个必要。另外 R3 关于 area1 的 LSA，与 R1 并无两样，这里不再赘述。

#### R4#show ip ospf database

OSPF Router with ID (4.4.4.4) (Process ID 1)

Router Link States (Area 0)					
Link ID	ADV Router	Age	Seq#	Checksum	Link count
3.3.3.3	3.3.3.3	1661	0x80000003	0x004718	2
4.4.4.4	4.4.4.4	1631	0x80000003	0x00E673	2

Summary Net Link States (Area 0)				
Link ID	ADV Router	Age	Seq#	Checksum
192.168.45.0	4.4.4.4	1631	0x80000002	0x00EE66
192.168.123.0	3.3.3.3	1661	0x80000002	0x003713

#### Summary ASB Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum
5.5.5.5	4.4.4.4	1632	0x80000002	0x009C3A

#### Router Link States (Area 2)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
4.4.4.4	4.4.4.4	1632	0x80000003	0x00A299	2
5.5.5.5	5.5.5.5	1616	0x80000004	0x0043F1	2

#### Summary Net Link States (Area 2)

Link ID	ADV Router	Age	Seq#	Checksum
192.168.34.0	4.4.4.4	1632	0x80000002	0x0068F7
192.168.123.0	4.4.4.4	1633	0x80000002	0x009B6A

#### Type-5 AS External Link States

Link ID	ADV Router	Age	Seq#	Checksum	Tag
5.5.5.0	5.5.5.5	1618	0x80000002	0x009AE1	0

我们再观察一下 R4 的 LSDB，R4 作为一台 ABR，LSDB 中自然也有两个区域的 LSA 描述信息。细心的朋友一定会发现，在 R4 LSDB 中，area2 里并没有 Summary ASB Link States 也就是 4 类 LSA，原因上面已经讲过了，这是因为 R4 与 ASBR R5 同属一个区域，通过 R5 发出的 1 类 LSA 即可知道 ASBR 的所在。

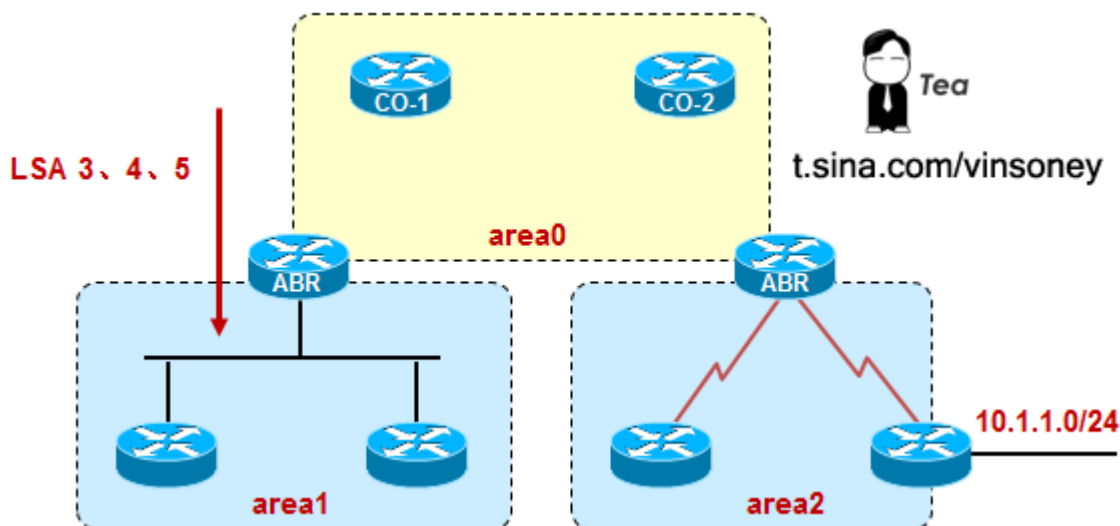
## 5.4 各区域中的 LSA

各区域内所允许出现的 LSA 总结如下：

- 骨干区域： 1、2、3、4、5
- 标准区域： 1、2、3、4、5
- Stub 区域： 1、2、3、3 类 0.0.0.0/0 ( ABR 向区域内发起的一条 3 类缺省路由 LSA )
- 完全 Stub 区域： 1、2、3 类 0.0.0.0/0 ( ABR 向区域内发起的一条 3 类缺省路由 LSA )
- NSSA 区域： 1、2、3、7
- 完全 NSSA 区域： 1、2、7、3 类 0.0.0.0/0 ( ABR 向区域内发起的一条 3 类缺省路由 LSA )

## 6 OSPF 特殊区域

### 6.1 认识特殊区域

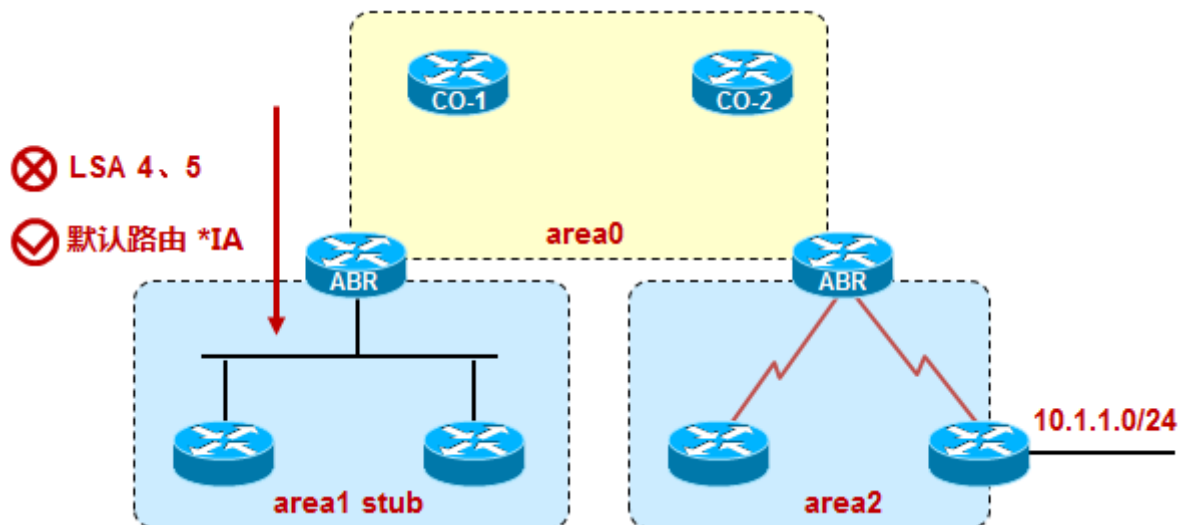


为了让我们的讲解更加的通俗易懂,我设计了上面这个拓扑,这是一个根据客户业务逻辑结构所涉及的 OSPF 网络,共有三个区域(实际上远远不止),骨干区域 area0 为一级行及二级行所运行,三级行运行的是 OSPF 的常规区域,为了保证网络的畅通,我们将网络中的各个角落都宣告进了 OSPF,感觉上很爽,但是其实路由器很压抑,毕竟随着设备越老越多、网络前缀越来越多,路由条目势必逐渐增多,那么路由器就亚历山大了,毕竟庞大的路由表及 LSA 在极大地消耗着路由器的资源。

从网络优化的角度,我们一直在试图在保证网络通畅的情况下减少网络中传递的路由条目及 LSA 的数量,路由汇总就是一种很好的方式,当然,从 OSPF 的设计规划角度,我们还有特殊区域可供我们灵活运用,下面来看看 OSPF 特殊区域是如何帮助我们减少 LSA 泛洪的。

我们拿 area1 做参考区域,当 area1 为常规区域时,区域中会有多少种 LSA 在泛洪呢?1 类是必然有的,由于 area1 中存在 MA 网络,因此 2 类 LSA 也有。其他区域的 3 类 LSA 被 ABR 也都注入进了本区域。另外,由于 area2 的 ASBR 引入的外部路由(5 类 LSA)也会被泛洪进 area1,当然 4 类 LSA 也跟着来了。那么如此一来,area1 中就有 1、2、3、4、5,共计 5 种类型的 LSA,齐活了。但是仔细一想我们就发现,其实 area1 作为“叶”区域,没必要知道外部路由的详细情况,我只需要知道有那么一条路,让我到达域外即可,因此,第一种特殊区域:末梢区域 stub area。

## 6.2 末梢区域 stub area



我们可以通过配置，将一个常规区域设置为 stub 区域。

Stub 区域将禁止 4、5 类 LSA 进入该区域，同时该区域的 ABR 将会自动下发一条默认路由（3 类 LSA）进入该区域，以确保数据通路没有问题。这可以形象的理解为：“外面的世界再怎么精彩，你不用告诉我细节，只要让我出去就行了”。这就是 stub area 的设计思路。当引入大量的外部路由进 OSPF，适当的规划某些区域为 stub，可以起到不错的网络优化作用。

有一点值得注意，你不能将骨干区域 area0 配置为 stub 区域，同时，让一个区域被指定为 stub，区域内将不允许注入外部路由，也就是不能做重发布。

### 配置命令：

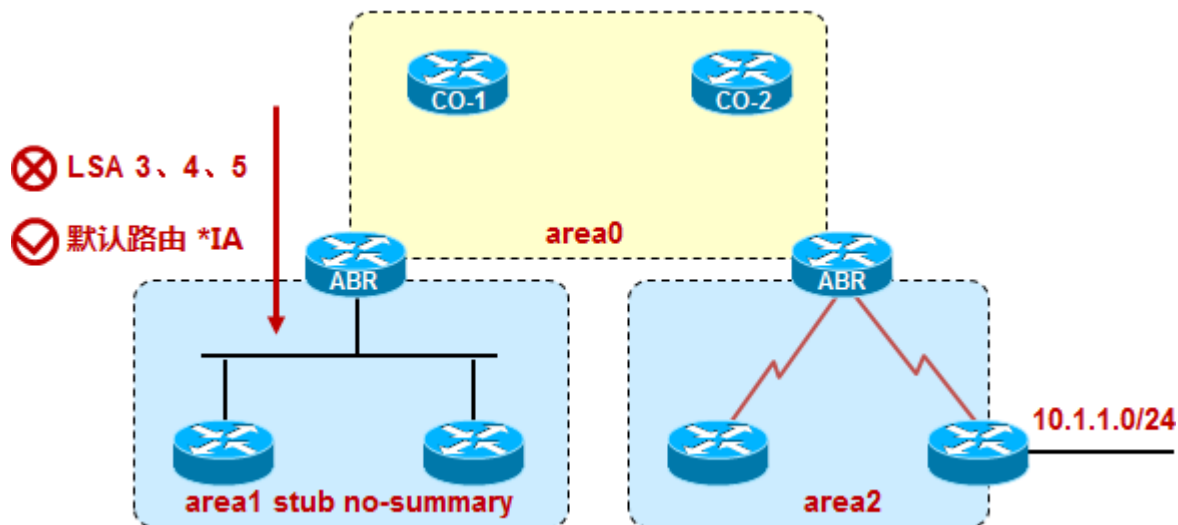
```
router ospf 1
area 1 stub
```

需要注意的是，该命令要配置在 stub 区域中的所有路由器上，如果某台路由器没有配置，那么它将无法去其他 stub area router 建立邻接关系。

### 实现效果：

Area1 中将不会有 4、5 类 LSA，也就是 area2 重发布进来的路由，被 ABR 过滤掉了，同时 area1 中的路由器将获取到一条 3 类 LSA 的默认路由，该默认路由是由 area1 的 ABR 自动下发。

## 6.3 完全末梢区域 totally stub



通过将区域规划为 stub area，可以起到一定的网络优化作用，但是感觉上还不够彻底，除了外部路由，其他区域的路由（LSAs）其实我也没必要知道太多细节，完全用一条默认路由替代也行嘛，那么你可以将 area1 配置为完全末梢区域（totally stub area），当一个区域被配置为完全末梢区域，这个区域将：

- 阻挡 3、4、5 类 LSA 进入本区域
- 区域的 ABR 自动下发一条 3 类 LSA 的默认路由进入本区域

这么一来，area1 内路由器收到的 LSA 将进一步减少，在存储 LSA 及进行 SPF 算法运算的时候，耗费的资源自然也就减少了，另外当区域外拓扑出现变更的时候，对本区域的影响也将变为最小。

与 stub 区域类似，你无法将骨干区域 area0 配置为 totally stub area，当然，如果一个区域被指定为 totally stub area，你将不能在区域中的路由器上做路由重发布动作。

### 配置命令：

```
router ospf 1
area 1 stub
```

在完全末梢区域内的所有路由器，都配置上述命令，与 stub 区域的区别在于 ABR 的配置，下面是配置在完全末梢区域的 ABR 上的。

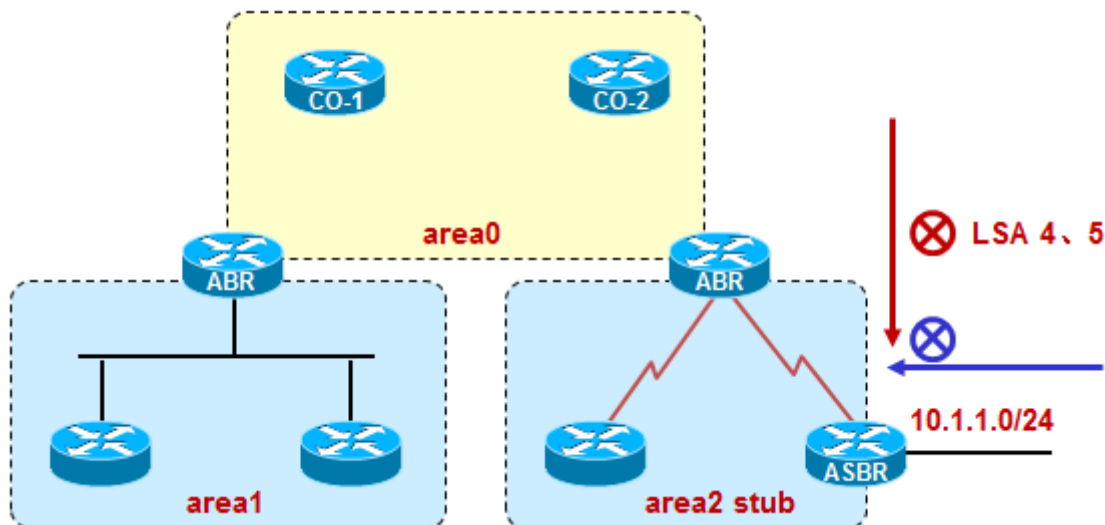
```
router ospf 1
area 1 stub no-summary
```

### 实现效果：

完成上述配置后，area1 内的路由器将只有本区域内的路由（ABR 除外），同时都能获取到 ABR 下发的 3 类的默认路由。也就是说其他区域的路由以及外部注入的路由都被 ABR 阻挡在外，取而代之的是一条默认路由。

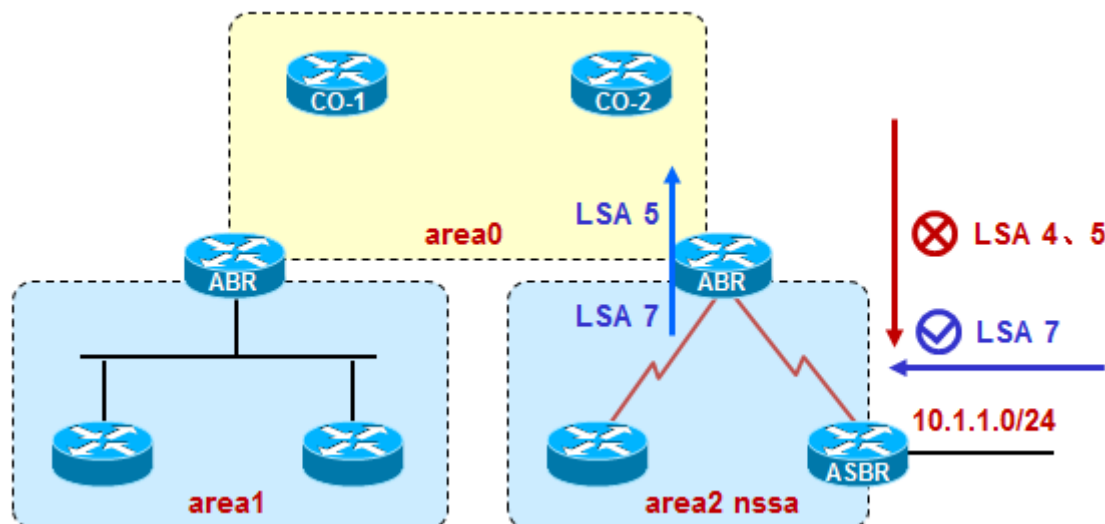


## 6.4 非完全末梢区域 NSSA



在前面的知识基础上，我们已经了解到，在保证网络连通性的情况下，减少 LSA 的泛洪以及精简路由表，我们可以将特定区域配置为末梢区域或完全末梢区域。看上面的拓扑，我们将 area2 配置为 stub，那么，这个区域一来将阻挡来自其他区域的 4、5 类 LSA，同时区域内的路由器禁止重发布外部路由，那么如果此时我期望这个区域保持“阻挡其他区域过来的 4、5 类 LSA”这个特性，同时允许我在区域本地重发布路由呢？

例如，假设 area2 原先是作为一个末梢区域运行的，但突然有一个外部网络，需要接入到我们这个 OSPF 网络中，并且连接在 area2 中，那么这个时候，为了保证路由的可达，就必须向 area2 中注入外部路由了，而这又违反了 stub area 的规则。



这里就引入 NSSA ( not-so-stubby-area ) 的概念，中文翻译过来，可以理解为非完全末梢区域，当你将一个区域配置为 NSSA，那么这个区域一来将阻挡骨干区域过来的 4、5 类 LSA，同时允许区域本地注入外部路由，这些外部路由以一种特殊的 LSA 类型——7 类 LSA 在 NSSA 中泛洪，并且 7 类 LSA 不允许进入骨干区域或常规区域，NSSA 的 ABR 会负责将 7 类 LSA “转换”成 5 类 LSA，从而在常规区域中进一步泛洪。上面的“允许区域本地注入”的意思是，NSSA 这个区域的路由器配置重发布。

值得注意的是，与 stub area 及 totally stub area 不同的是，如果你将一个区域配置为 NSSA，默认情况下 NSSA 的 ABR 不会自动下发默认路由进 NSSA，因此在 NSSA 环境下，需留意网络连通性问题。

#### 配置命令：

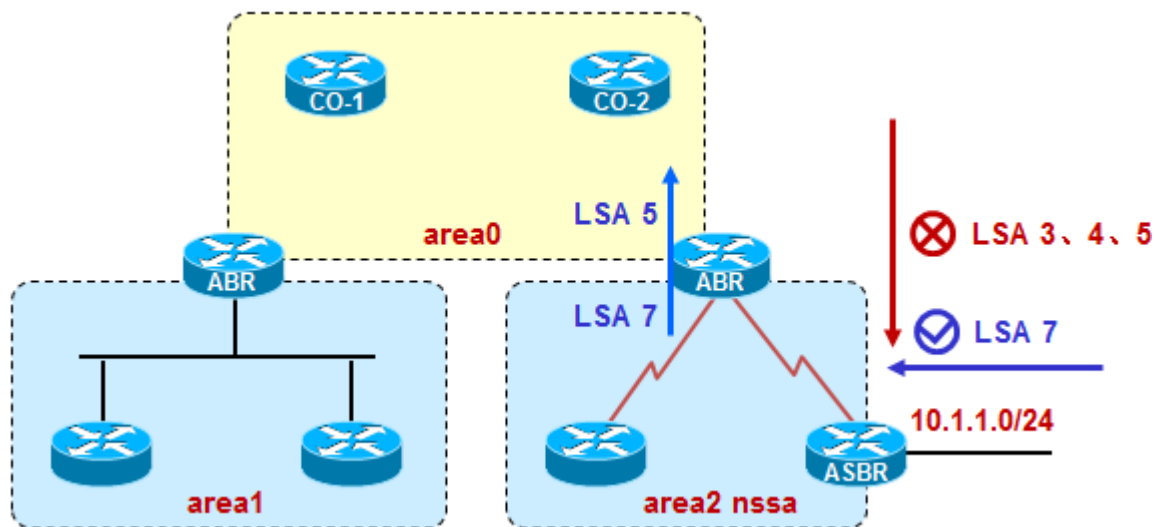
```
router ospf 1
area 2 nssa
```

上述命令需配置在 NSSA 内的所有路由器上。

#### 实现效果：

将 area2 配置为 nssa 后，从骨干区域过来的 4、5 类 LSA 将无法进入 NSSA，也就是说如果 area1 做了重发布，那么这些重发布的外部路由将无法进入 NSSA（这是因为 NSSA 中不允许出现 5 类 LSA），与此同时，area2 允许本地的路由器做重发布动作，重发布进来的路由，以 LSA7 在 NSSA 中泛洪，大家在路由表中看到的这些外部路由，标记为“O N”，而这些 7 类 LSA 在“穿越”NSSA 的 ABR 进入骨干区域之前，由 ABR 负责将 7 类 LSA “转换”成 5 类 LSA。最终 area0 及 area1 也都能学习到这些外部路由，只不过，他们路由表中呈现的是“O E”标记。

## 6.5 Totally NSSA



Totally NSSA，这是一个不太好用中文翻译的词汇，完全 非完全末梢区域？显得有点尴尬，不过这个概念的理解并不像其名字那么唬人，Totally NSSA 是在 NSSA 区域的基础之上，进一步阻挡 NSSA 区域外的其他区域过来的 3 类 LSA，同时 ABR 自动下发一条 3 类的默认路由进 NSSA 区域。

#### 配置命令：

```
router ospf 1
area 2 nssa
```

上述命令需配置在 NSSA 中的所有路由器上，ABR 的配置有所不同，如下：

```
router ospf 1
area 2 nssa no-summary
```

### 实现效果：

Area2 之外的其他 OSPF area 过来的 3、4、5 类 LSA，都会被阻挡在 NSSA 之外，同时 ABR 会自动下发默认路由进 NSSA，另外，NSSA 内的路由器做重发布动作，区域内的其他路由器将会学习到 7 类的外部 LSA，这些外部 LSA 会被 ABR 转成 5 类 LSA 并注入进骨干区域。

## 6.6 特殊区域总结

### ● 骨干区域 Backbone Area 0

本身是一个标准区域，负责连接非骨干区域，其它区域（非骨干区域）必须保证和骨干区域有直接的物理连接，因为区域之间的 LSA 必须经过骨干区域中转

### ● 常规(标准)区域 Standard Area

一个区域缺省是常规区域

### ● 末梢区域 Stub Area

把一个区域配成存根区域的好处是，阻挡 LSA4、LSA5 外部路由进入本地区域，从而精简路由表；同时 ABR 会自动产生 3 类的默认路由 LSA 注入进该区域。

### ● 完全末梢区域 Totally Stubby Area

完全末梢区域是一种对末梢区域的改进，进一步精简路由表；  
阻挡 LSA3、4、5 进入该区域，同时 ABR 自动下发 3 类默认路由。

### ● 非完全末梢区域 Not-so-stubby Area

即想阻挡 LSA5，自身又想引入外部路由，stub 的变种；  
NSSA 既阻挡外部 LSA5 的进入，同时区域内的路由器又可以引入外部路由 LSA7；  
LSA7 在 NSSA 内洪泛，通过 ABR 时转换为 LSA5 并被注入进骨干区域。  
ABR 不会缺省生成 0/0 默认路由进入本地区域，需手工配置

### ● 完全 NSSA totally Not-so-stubby Area

在 NSSA 的基础上 进一步阻挡骨干区域过来的 3、4、5 类 LSA 并且 ABR 自动下发 3 类的默认路由进 NSSA。

### 各区域内所允许出现的 LSA 总结如下：

- 骨干区域： 1、2、3、4、5
- 标准区域： 1、2、3、4、5
- Stub 区域： 1、2、3、3 类 0.0.0.0/0 ( ABR 向区域内发起的一条 3 类缺省路由 LSA )
- 完全 Stub 区域： 1、2、3 类 0.0.0.0/0 ( ABR 向区域内发起的一条 3 类缺省路由 LSA )
- NSSA 区域： 1、2、3、7

- 完全 NSSA 区域： 1、2、7、3 类 0.0.0.0/0 ( ABR 向区域内发起的一条 3 类缺省路由 LSA )

## 6.7 NSSA 详解

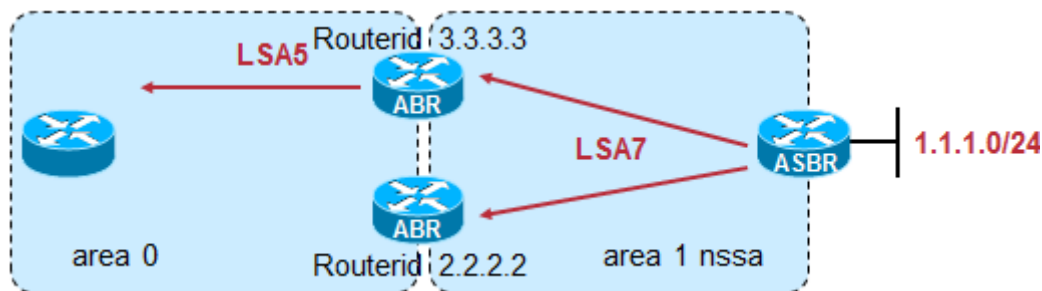
### 1. NSSA ABR 上的路由汇总动作

NSSA ABR 会将 7 类 LSA 转换为 5 类 LSA ,在 7 转 5 的过程中可以使用 summary-address 通告汇总的 5 类 LSA ,注意, 这里汇总的是针对 NSSA 区域外部引入对汇总, 但是如果仍想对 area 1 nssa 区域自己内部对路由做汇总, 则使用 area range。

### 2. NSSA 的 ABR , 在某种程度上可理解为常规区域的 ASBR , 它不会为常规区域生成 4 类 LSA。

NSSA 的 ABR 产生了 5 类 LSA(7 转 5)并通告进骨干区域 ,因此这台 NSSA 的 ABR 对于骨干区域而言就是一台 ASBR。由于存在这个 7 转 5 的过程 ,真正的 ASBR(NSSA 区域中执行重发布的那台)信息就不需要被骨干区域路由器所知道 ,因此 NSSA 区域的 ABR 也不会为这个区域里的 ASBR 产生 4 类 LSA 并通告进骨干区域。

### 3. NSSA 双 ABR 问题



上图中, 两个 ABR 都会收到 7 类 LSA ,但只有 router-id 大的 ABR 执行 7 转 5。

两台 ABR 都会在 NSSA 区域泛洪 1 类 LSA ,并从中了解到对方的存在。

### 4. N/P 位

Hello 报文以及 LSA 报文中的 Option 字段

•	O	DC	EA	N/P	MC	E	T
---	---	----	----	-----	----	---	---

其中 N/P 位为 1bit , 在 HELLO 和 LSA 中都有携带 option 字段 , N/P 位在两者中分别有不同意义

**在 Hello 报文中** : N bit 指示该路由器为 NSSA 区域路由器 , 当 N bit 被置 1 时 E bit 就必须被清零。

**在 LSA 报文中** : P bit 仅在 7 类 LSA 中出现 , 置 1 时指示 NSSA 区域的 ABR 能够将这条 7 类 LSA 转成 5 类 LSA。

```
r1#sh ip os da nssa-external
```

OSPF Router with ID (1.1.1.1) (Process ID 1)

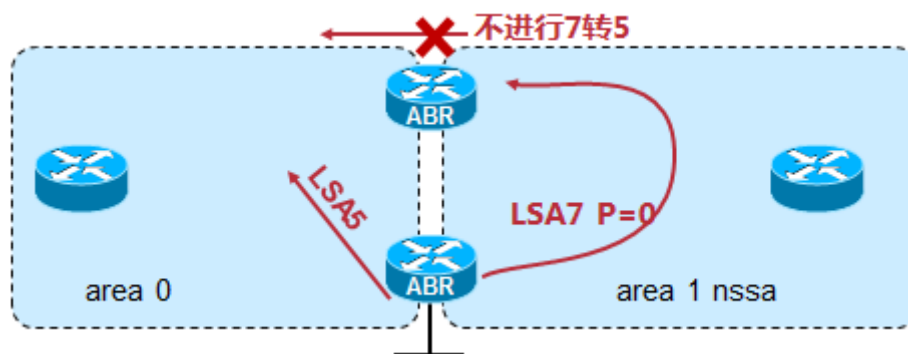
Type-7 AS External Link States (Area 1)

LS age: 9

Options: (No TOS-capability, No Type 7/5 translation, DC) //即 P 位被置 0

- **P bit 为 0 时，ABR 将不能将该 7 类 LSA 转换成 5 类 LSA**

只有 NSSA 区域的 ABR 重发布路由时，通告的 7 类 LSA 中的 P bit 才为 0



Open Shortest Path First

⊕ OSPF Header

⊖ LS Update Packet

Number of LSAs: 1

⊖ LS Type: NSSA AS-External-LSA

LS Age: 1 seconds

Do Not Age: False

⊖ Options: 0x20 (DC)

0... .. = DN: DN-bit is NOT set

.0.. .... = O: O-bit is NOT set

..1. .... = DC: Demand Circuits are supported

...0 .... = L: The packet does NOT contain LLS data block

.... 0... = NP: NSSA is NOT supported

.... .0.. = MC: NOT Multicast Capable

.... ..0. = E: NO External Routing Capability

.... ...0 = MT: NO Multi-Topology Routing

Link-State Advertisement Type: NSSA AS-External-LSA (7)

Link State ID: 2.2.2.0

Advertising Router: 2.2.2.2 (2.2.2.2)

LS Sequence Number: 0x80000001

LS Checksum: 0x1754

Length: 36

Netmask: 255.255.255.0

External Type: Type 2 (metric is larger than any other link sta

Metric: 20

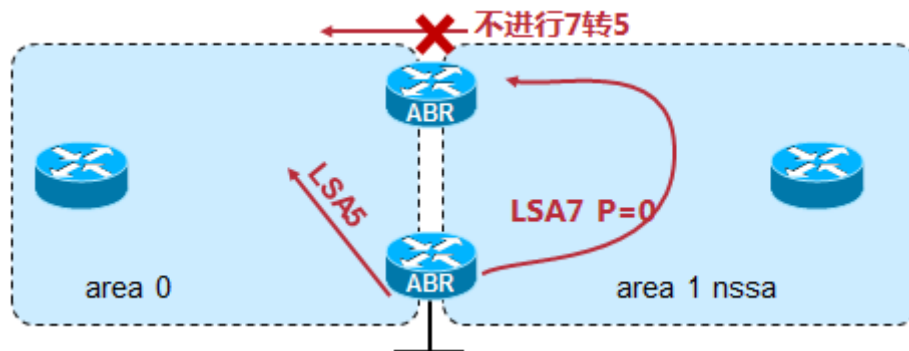
Forwarding Address: 10.1.24.2

External Route Tag: 0

NSSA 区域 ABR 在重发布时，将 7 类 LSA 的 P 位置 0，通知其他 NSSA 区域 ABR 不要对该 LSA 进行

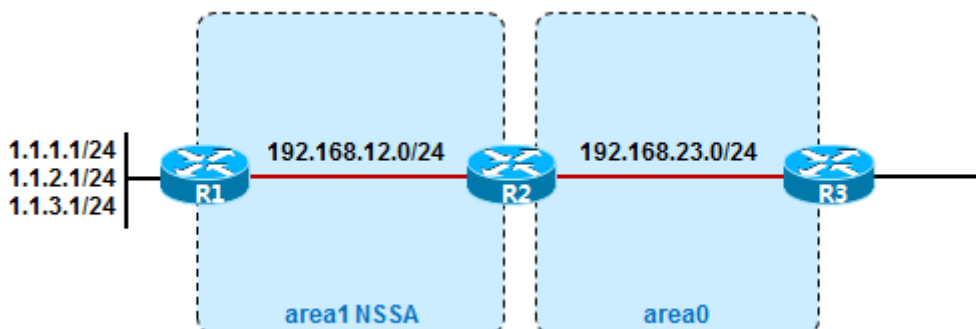
7 转 5，因为骨干区域内的其他路由器已经从骨干区域内收到该路由的 5 类 LSA，因此 NSSA 区域其他 ABR 执行 7 转 5 是没有意义的。

- 在 ABR 上重发布，默认情况，通告 7 类 LSA 进 NSSA 区域，通告 5 类 LSA 进入骨干区域  
如果使用 area x nssa no-redistribution，则不通告 7 类 LSA 进入 NSSA 区域



这个一般在不希望 NSSA 学到太多对路由明细对情况下使用，比如网点路由器。这个时候可在此配置基础上再在 ABR 上对 NSSA 区域发布一条默认路由，从而最大程度上对减少路由表。

## 5. NSSA 中默认路由的传递问题



- NSSA 区域中的 ABR ( R2 ) 通告默认路由
  - 将区域类型配置为完全 NSSA，自动下发 3 类 LSA 的默认路由；同时阻塞 LSA3；
  - 在进程下面使用 area x nssa default-information-originate 命令（本地无需默认路由的存在），自动下发 7 类 LSA 的默认路由，这时候 ABR 并不阻塞从骨干区域来的 LSA3 类进入 NSSA 区域
- NSSA 区域中的 ASBR ( R1 ) 通告默认路由

在进程下面使用 area x nssa default-information-originate 命令，不会自动下发 7 类 LSA 的默认路由，必须事先在路由表中存在一条默认路由。

【注意】在 NSSA 区域中，无论是 ASBR 上，还是 ABR 上，使用静态默认路由加上 default-information-originate [always]命令均无法注入默认路由，原因是采用该命令注入的默认路由，为 5 类 LSA，而 5 类 LSA 在 NSSA 中无法存活。

## 7 OSPF 配置

### 7.1 基本配置

#### 1. 修改 OSPF 接口优先级 ( 0-255 )

取值范围为 0-255，默认为 1，优先级为 0 的 OSPF 接口在任何情况下都不具备选举资格

```
Router (config-if)# ip ospf priority x
```

DR 不会因为新加入的接口在其 HELLO 分组中报告了更高的优先级而放弃其地位，这是 DR 的非抢占性。

#### 2. 修改 OSPF 接口 cost

OSPF 使用 cost 作为 metric ( 10E8 / 带宽 )

```
Router (config-if) # ip ospf cost x
```

#### 3. 配置 OSPF 定时器

在缺省的情况下，失效时间间隔是 Hello 时间间隔的 4 倍

```
Router (config-if) # ip ospf hello-interval seconds
```

```
Router (config-if) # ip ospf dead-interval seconds
```

#### 4. 配置 LSDB 过载保护

```
max-lsa xx [ yy% ] [ warning-only ] [ ignore-time aaa ] [ ignore-count bbb ] [ reset-time zzz ]
```

当 LSA 的个数超过定义的这个阈值，将会.....

**xxx** 为 LSA 最大个数

**yy** 为百分比，默认 75%

**warning-only** 超过最大值，只是警告（仍然收发 LSA）。默认关闭

**ignore-time** 超过所定义的最大值后，超过多长时间，路由器才进入 ignore 状态。默认 5 分钟。

Ignore 状态下会清除所有邻居及 LSA，不会尝试去建立邻居

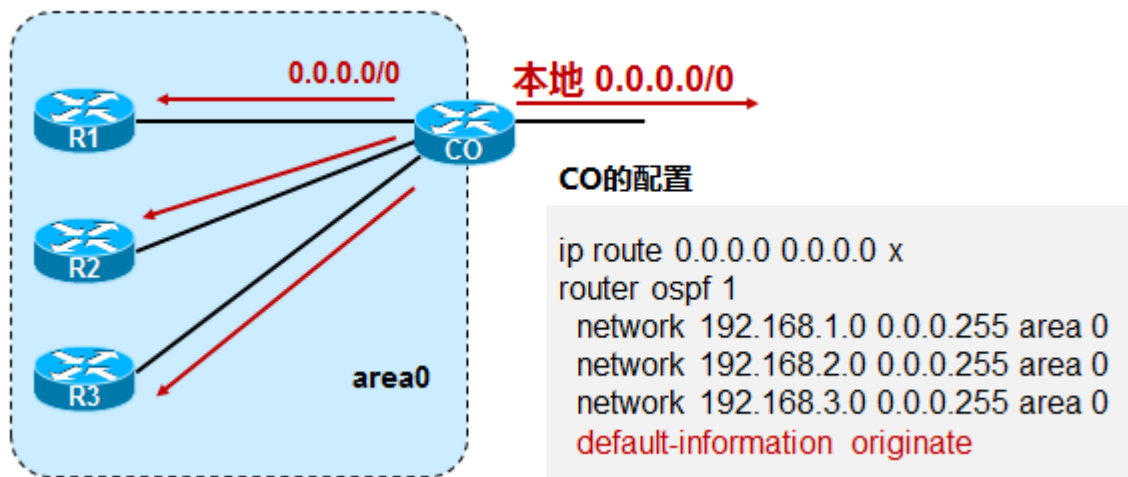
**ignore-count** 默认 5 次，超过这个次数，则 OSPF 永远处于 ignore 状态，除非手工重启 ospf 进程

**Reset-time** 默认 10 分钟，ignore 多长时间，脱离 ignore 状态

### 7.2 注入默认路由

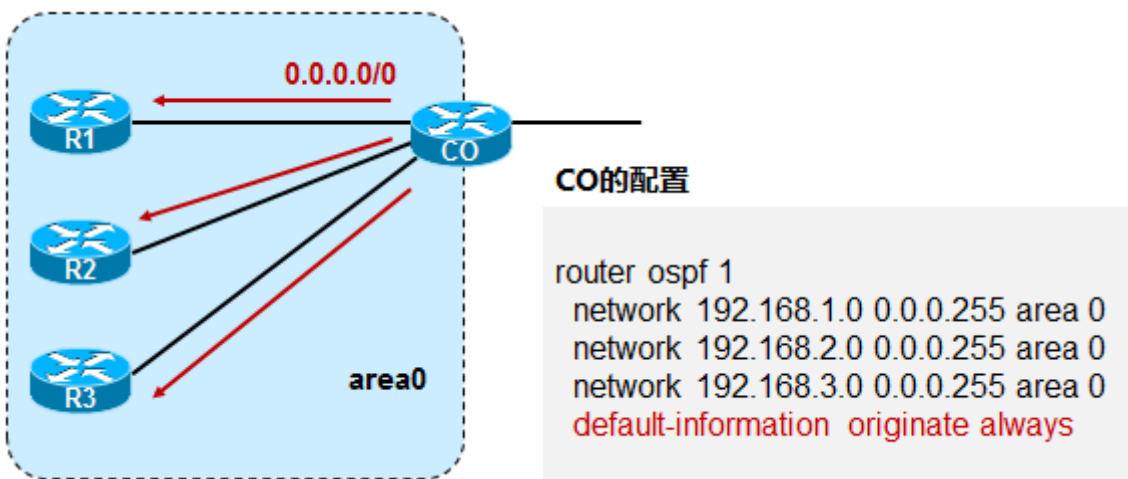
- OSPF 默认路由注入方式一：default-information originate





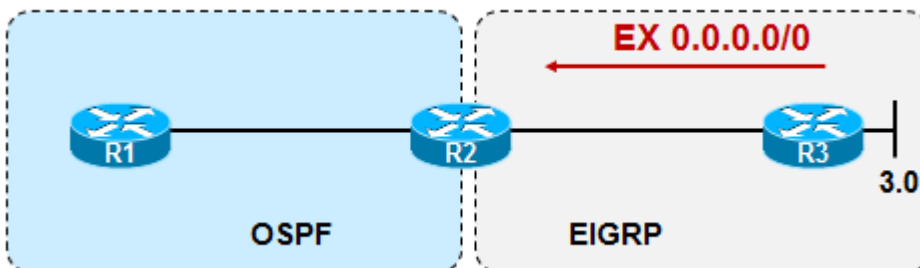
CO 本地必须有一条默认路由，该默认路由可以是静态的，也可以是通过其他动态路由协议获取到的，同时搭配上 `default-information originate`，即可向 OSPF 域中注入 OE 类型的默认路由，且该默认路由在 CO 本地这条默认失效后也随之 DOWN 掉。

- OSPF 默认路由注入方式二：`default-information originate always`



本地无需任何形式的默认路由，使用 `default-information originate always` 将始终向 OSPF 域中注入一条默认路由，并且该默认路由只要 CO 还 UP，则不会 DOWN。

- 使用 `Redistribute` 方式尝试从其他动态路由协议注入默认路由是徒劳的



R1、R2 跑 OSPF；R2、R3 跑 EIGRP：

首先 R2 上配置 `ip route 0.0.0.0 0.0.0.0 10.1.23.3`，并将静态路由重发布进 OSPF，默认路由重发布失败



其次 R3 上向 R2 传递 EIGRP 默认路由及 3.0 网络,然后在 R2 上将 EIGRP 重发布进 OSPF,发现 R3 上 3.0 网络及 R2、R3 之间的链路所在网段都被重发布进 OSPF,但是默认路由重发布失败。

由此可见,使用重发布的方式,无法将本地的静态默认路由或从其他协议学习到的默认路由注入 OSPF

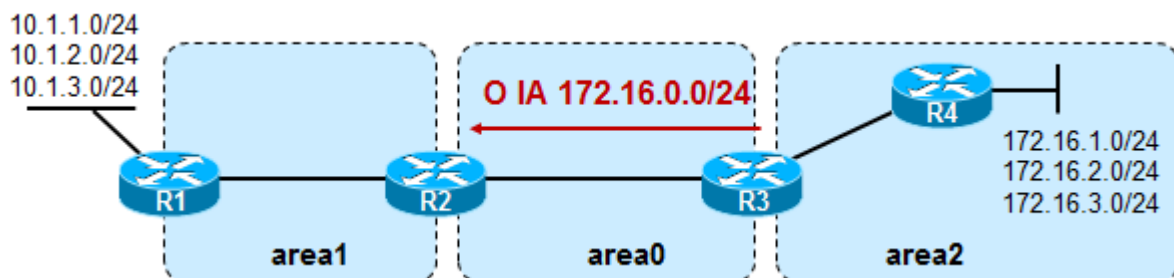
## 7.3 路由汇总

### 1. 区域间路由汇总

Router (config-router)#area 1 range 172.9.0.0 255.255.0.0 ?

<b>cost</b>	User specified metric for this range	//调整汇总的 3 类 LSA 的 Metric
<b>advertise</b>	Advertise this range (default)	//通告汇总但抑制明细,默认就携带了这个关键字
<b>not-advertise</b>	DoNotAdvertise this range	//抑制产生该汇总的 3 类 LSA 以及相关明细 3 类 LSA 也就是汇总和明细都没了,可用于过滤明细

注意该 area range 只对 1 类、2 类 LSA 产生作用,对 3 类 LSA 是不起作用对,也就是对 O IA 是无法汇总的,只对本区域内始发的路由起作用,并且必须配置在 ABR 上,产生的汇总路由将传递给其他区域。

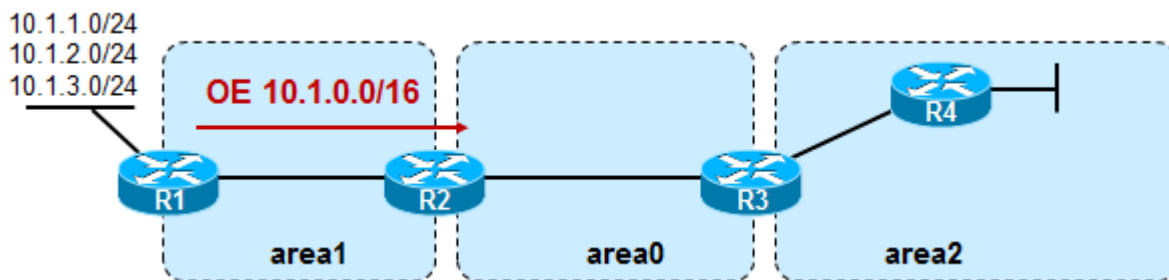


配置如下:

```
router ospf 1
  area 2 range 172.16.0.0 255.255.0.0
```

另外一点,产生的这条汇总路由,metric 是多少呢?譬如 area2 内,由 R4 传递给 ABR R3 的明细路由如果 cost 各不相等,那么 R3 产生的这条汇总路由的 metric 即为这些明细路由中, cost 最小的那个值再加上 R3 路由入接口的 cost,两者之和就是 R3 产生的汇总路由的 metric。

### 2. 外部路由汇总



```
router ospf 1
```

```
summary-address 10.1.0.0 255.255.0.0 ?
```

```
not-advertise      Do not advertise when translating OSPF type-7 LSA
```

```
tag                Set tag
```

对重发布进 OSPF 域的外部路由，需使用上述命令进行汇总，且注意配置的路由器必须是引入这些外部路由的那台 ASBR。

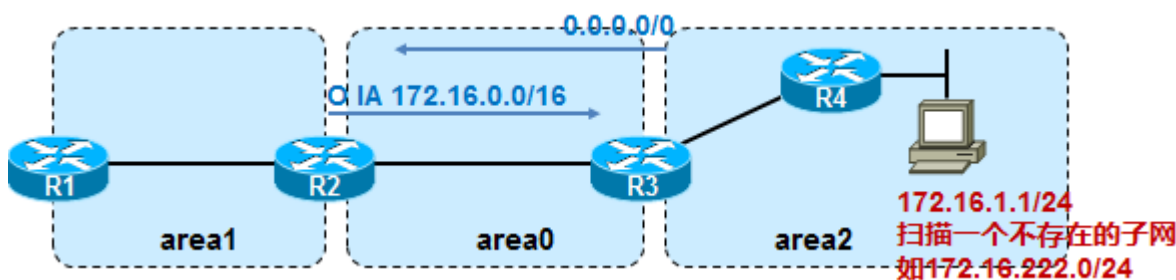
另外，当在 NSSA 中外部路由时，这些 7 类的 LSA 在穿越 NSSA 的 ABR 时，由 ABR 进行 7 转 5，这些 5 类 LSA 的路由，由于是由 NSSA 的 ABR 产生的，这时候此台 ABR 就有点 ASBR 的味道了，因此，亦可使用此命令针对这些路由进行汇总。

### 3. 关于汇总路由的防环问题

许多路由协议在手工汇总后，在执行汇总的 Router 的路由表里会出现一条指向 NULL0 的路由：

```
O 172.16.0.0/16 is a summary, 00:19:40, Null0
```

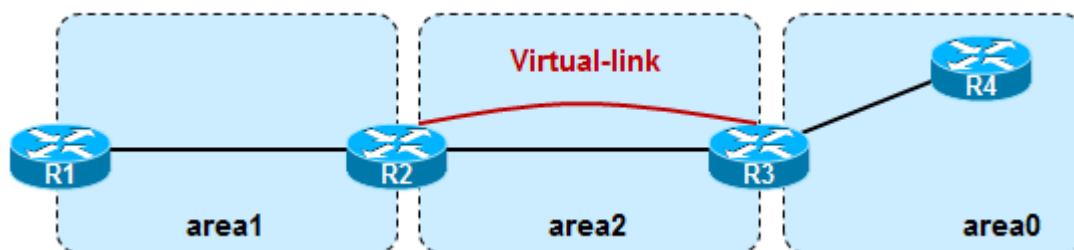
如果没有这个机制，看看会发生什么：



R3、R4 都有默认路由出去，R3 作为 ABR，将 area2 内 172.16.0.0 子网进行汇总，将汇总路由传递给了骨干区域。此时 R4 下某个子网里，有 PC 在扫描一个不存在的子网内的 IP，如 172.16.222.0/24，这些数据包会被默认路由匹配一路传递到 R2，而 R2 上由于收到 R3 传递过来的汇总路由，因此又把这些数据包丢回去给 R3，R3 又丢回 R2，如此反复，直到报文 TTL 为 0。

OSPF 为了解决这个问题，在进行汇总时，OSPF 在 R3 上（汇总路由的产生地）会在本地自动产生一条指向 Null0 的汇总路由，这样一来当再类似事件发生，数据包将在 R3 这就被丢弃。

## 7.4 virtual-link



OSPF 规定每个区域都必须与区域 0 相连，当出现了以上情况，可以在区域 1 中建立一条虚连接来过渡。

Area 过渡区域的 ID virtual-link 虚链路对端 router 的 RouterId

虚链路用 hello 包建立，单播。一旦建立，hello 将不再发送。凡是通过虚链路学到的 LSA，都会标 DNA，表示永远不老化。

例如在 R2 上看到的部分 LSDB：

Router Link States (Area 0)					
Link ID	ADV Router	Age	Seq#	Checksum	Link count
2.2.2.2	2.2.2.2	270	0x80000002	0x00C38D	1
3.3.3.3	3.3.3.3	1	(DNA)	0x002161	3
4.4.4.4	4.4.4.4	87	(DNA)	0x00E775	2

R2 上的邻居表：

R2#sh ip os nei

Neighbor ID	Pri	State	Dead Time	Address	Interface
3.3.3.3	0	FULL/ -	-	192.168.23.3	OSPF_VL0
1.1.1.1	0	FULL/ -	00:00:35	192.168.12.1	Serial0/0
3.3.3.3	0	FULL/ -	00:00:31	192.168.23.3	Serial0/1

虚链路的网络类型是 p2p；中转区域不能是 stub 区域；另外 vl 只是一种临时或者割接的手段，不建议运用在常规网络的实施当中。

注意：虚连接是作为主干区域的一部分，因此如果在 area 0 上配置了验证，那么 virtual-link 也要做。

如上图，area0 开启了认证，那么由于 R2、R3 之间的虚链路在此前已经建立，所以暂时不受影响（不超时），但重启 OSPF 后，邻居关系则无法建立，此时需要分别在 R2 及 R3 上开启虚链路认证，例如 R2 上的配置：

```
area 2 virtual-link 3.3.3.3 authentication-key spoto
```

```
area 2 virtual-link 3.3.3.3 authentication
```

当然，如果 area0 不验证，虚链路上自己做认证，那验证的范围就仅在虚链路上

Virtual-link 具有可传递性。

## 7.5 OSPF 认证

一共三种，链路认证、区域认证、虚链路认证，每种认证都支持明文及 MD5 认证

### 1. 链路（接口）认证

**明文认证：**

```
ip ospf authentication-key spoto //接口下配置明文密钥
ip ospf authentication           //为该接口开启明文认证
```

**密文认证：**

```
ip ospf message-digest-key 1 md5 spoto // 1 为 key ID，接口下配置密文密钥
ip ospf authentication message-digest // 开启密文认证
```



上图中，R2 配置了 2 个 key，R2 会将 key1 及 key2 给 R1，R1 使用 key1 与 R2 完成认证，当 R1 配置上 key2 cisco 后，R1 R2 之间会自动切换到使用 cisco 密钥进行认证，并且邻居关系不受影响，key1 也就不用了。

### 2. 区域认证

一旦给某区域开启认证，则属于给区域的所有接口都要配置密钥（不要求所有的密码一致，只需要直连接口双方一致），这种验证方式相当于区域内的所有接口都开启接口验证。

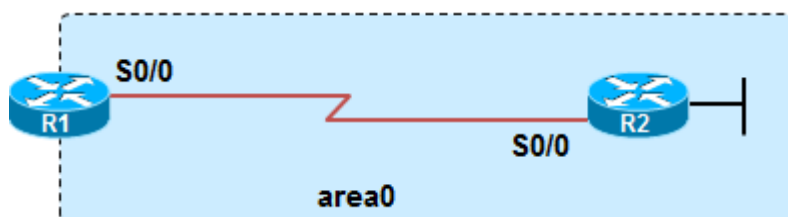
**明文认证：**

```
ip ospf authentication-key spoto //在所有属于开启认证区域的接口下配置明文密钥
area 0 authentication           //在 ospf 进程中开启区域明文认证
```

**密文认证：**

```
ip ospf message-digest-key 1 md5 spoto //在所有属于开启认证区域的接口下配置密文密钥
area 0 authentication message-digest //开启区域密文认证
```

如果区域 0 开启了认证，则虚链路也要起认证，只要虚链路路由器上 area 0 authentication



下图中，R1 及 R2 启用了区域验证后，R1 及 R2 的 s0/0 口都要配置 key，而 R2 的 loopback 接口没有接任何邻居，仅是一个末梢网段，因此该接口即使不配置验证 key，R1 仍然能学习到关于该接口的 LSA，因此也能学习到该 loopback 的路由

### 3. 虚链路的认证

#### 明文认证

```
Area 1 virtual-link 2.2.2.2 authentication-key xx      // 配置明文密钥
Area 1 virtual-link 2.2.2.2 authentication            // 开启明文认证
```

#### MD5 认证

```
area 1 virtual-link 2.2.2.2 message-digest-key 1 md5 cisco
area 1 virtual-link 2.2.2.2 authentication message-digest
```

## 7.6 LSA 的过滤

```
area 1 filter-list prefix          //在区域间过滤 LSA
```

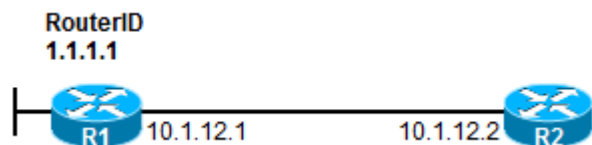
## 7.7 调整管理距离

**distance** AD 值 路由更新源 IP 反掩码 ACL

**distance ospf external ad1 inter-area ad2 intra-area ad3**

第一条命令能根据路由更新源并用 ACL 匹配路由条目进行设置 AD；这个路由更新源 IP 对于 OSPF 来说，指的就是 OSPF 的 routerID。

第二条命令中的 inter-area 针对的是 O IA 路由，intra-area 针对的是本区域内的路由



如上图，R2 能学习到邻居 R1（routerID1.1.1.1）发来的路由，如果希望在 R2 修改来自 R1 的路由的 AD 值使用 distance 222 1.1.1.1 0.0.0.0，可行，使用 distance 222 10.1.12.1 0.0.0.0，无效，因此这个更新源 IP，对于 OSPF 而言，实际上就是 OSPF ROUTER ID

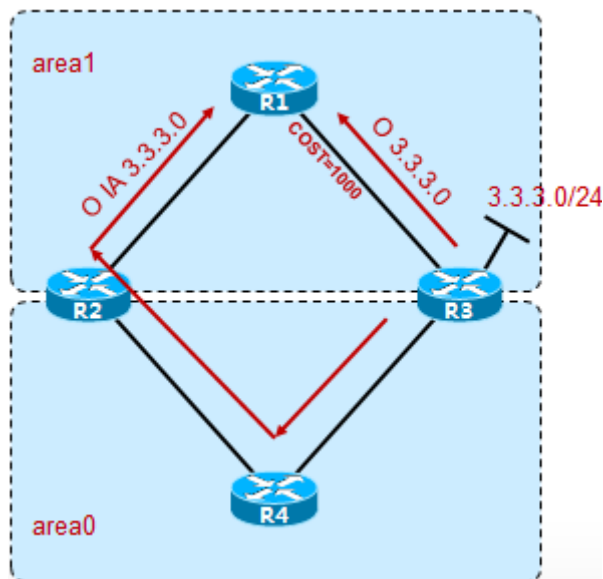
## 8 OSPF 路由优选规则

### 8.1 基本优选规则

1. 直连路由：本路由器发起的 LSA 1、2；
2. 区域内路由：O；LSA 1、2；
3. 区域间路由：O IA；LSA 3；
4. 1 类外部路由：O E1；LSA 5 类型 1；
5. 2 类外部路由：O E2；LSA 5 类型 2；
6. 1 类 NSSA 路由：O N1；LSA 7 类型 1；
7. 2 类 NSSA 路由：O N2；LSA 7 类型 2。

注意：E1 绝对优于 E2，而 E1 之间、E2 之间的比较，则需要严格注意（见本文档 LSA 详解部分章节的讲解）。另外，OE 及 ON 的优选，要看厂商，不同的厂商，优选方式不同。

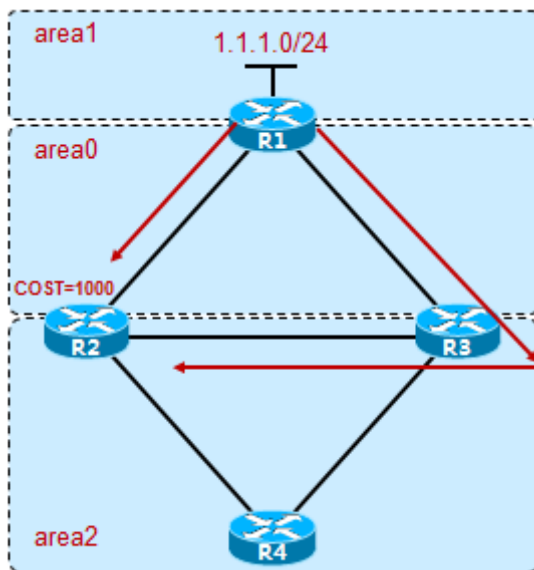
#### ● 验证实验：O 及 O IA 的比较



在这个图中，R1 同时从 R2 及 R3 学习到 3.3.3.0 的路由，一条是 O，一条是 OIA，R1 将绝对优选 R3 作为前往 3.3.3.0 的下一跳，而不看 metric。

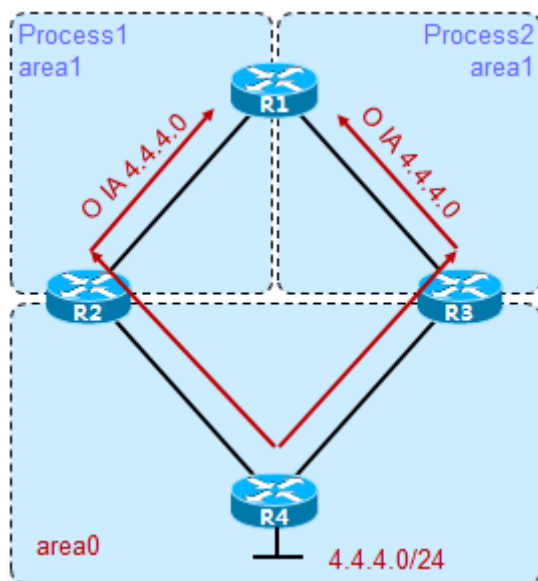
#### ● 验证实验：OIA 之间的比较

一般来说，OIA 之间比较 metric，谁小取谁，但是也有特例，如下：



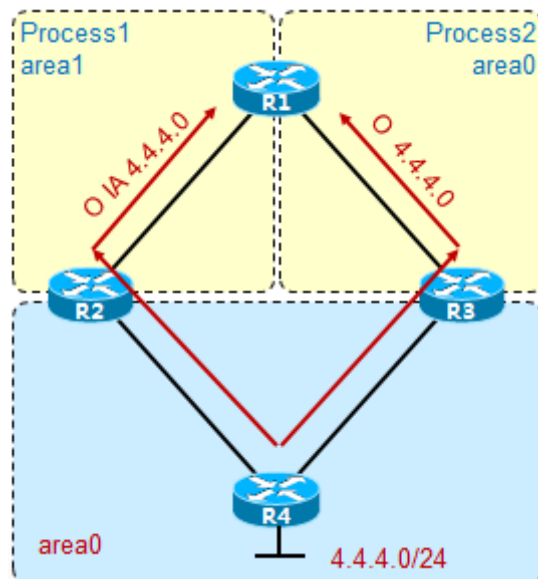
上图中，R2 上，能学习到由 ABR R1 及 R3 发出的关于路由 1.1.1.0 的 3 类 LSA，两条路由都是 O IA，但是 R2 将恒定优选 R1 作为去往 1.1.1.0 的下一跳，而不管这条路由的 metric 多大。

#### ● 验证实验：多进程 OSPF 问题 1



同一台路由器，本地如果跑两个 OSPF 进程，则各自独立，路由信息不会自动相互注入；另外如果同时从两个进程学习到同一条路由，则不管 metric 如何，先到先得，而不会负载均衡。

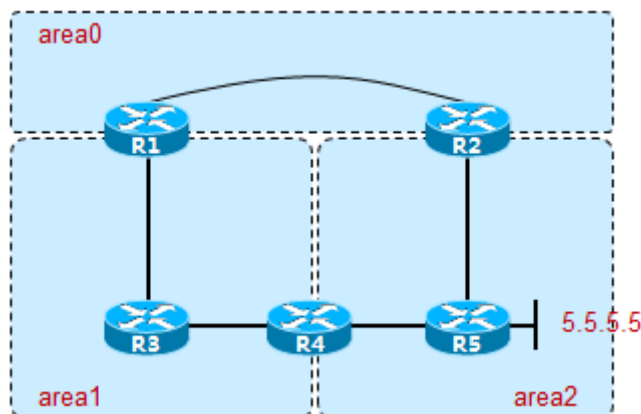
#### ● 验证实验：多进程问题 2



由于是两个 OSPF 进程，因此依然是先到先得，R1 上，如果率先与 R2 建立邻居关系，那么 4.4.4.0 的路由在路由表中，仍然是 O IA 的。

即使通过 R3 过来的是 O，但是由于是不同的进程，因此只看谁先来，不管 metric 也不管什么 O>IA>OE1>OE2

## 8.2 OSPF 的距离矢量特征

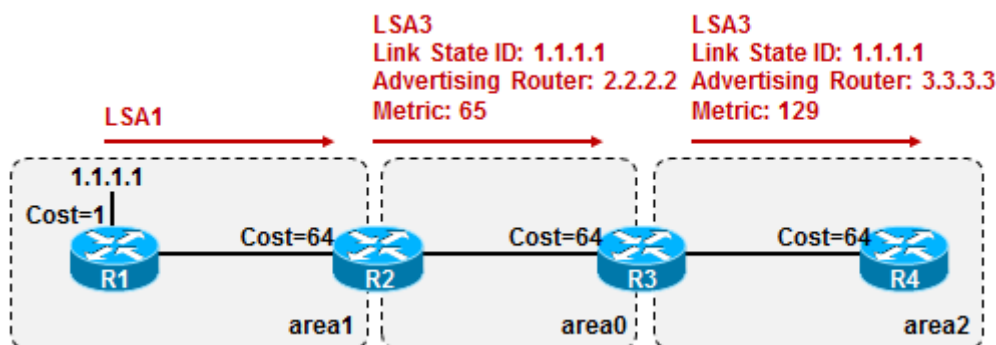


上图中，假设所有的 OSPF 接口 cost=1，那么 R5 更新 5.5.5.0 这条路由，R3 是能学习到的，R3 会怎么走呢？直观的来看，R3 去往 5.5.5.0 走 R4，cost 会更小，然而实际的结果是，R3 去 R5 会走 R1。

因为 R1 作为 ABR 产生的 3 类 LSA，而 R4 不是 ABR（还记得 ABR 的概念么？）无法下发 3 类 LSA，那么 R3 将只会从 R1 收到 3 类 LSA，因此不管从 R4 走 cost 如何小，R3 将只能通过 R1 到 5.5.5.0 网络。

**其实，OSPF 在区域内部，的确是链路状态协议的处理方式，然而在区域之间，OSPF 的对路由的处理却有典型的距离矢量特性。**

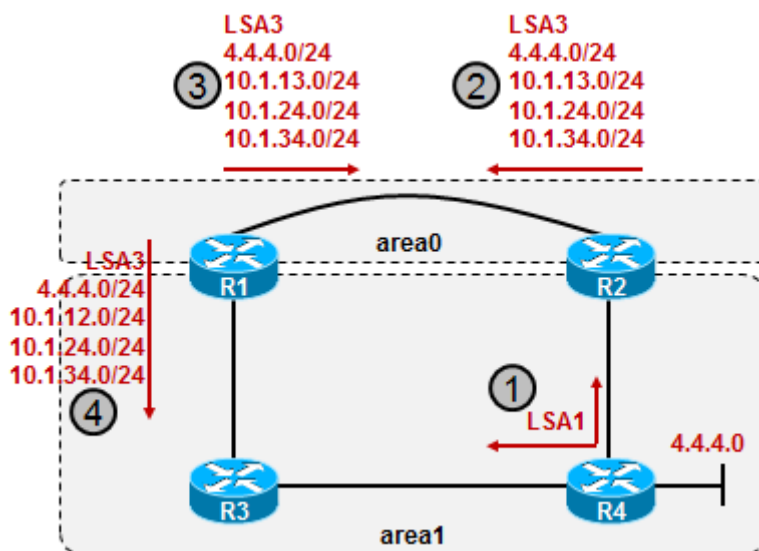




我们看上面的图，R1 宣告 1.1.1.1/32 这条直连链路，R1 将在 area1 内泛洪自己的 LSA1，其中就包含这段直连链路，这是典型的链路状态协议的特征。随后 R2 搜集到 LSA1，并且为 area0 生成 LSA3，注意，这时候 LSA3 的通告路由器就变成了 2.2.2.2 也就是 R2，同时 R2 为路由加上自己的接口 metric， $64+1=65$ 。随后这条 LSA3 被 R3 接收，R3 将信息改写，仍然是这条路由，通告路由器变成了 3.3.3.3 也就是 R3，同时，R3 在自己接收到的 metric 65 的基础上，累加上自己接口的 cost，得到 129，并将这条 LSA3 再注入到 area2 – 上述动作，是典型的距离矢量路由协议的特征，RIP 不就是这么干的么？路由一跳一跳的传递，每跳路由器累加一跳然后发出去。好了那么 RIP 作为距离矢量路由协议，有水平分割等防环机制，**那么 OSPF 呢？OSPF 则要求，所有的非骨干区域的 3 类 LSA 的传递必须经过 area0 来中转，也就是说，所有常规区域必须与骨干区域 area0 直连，以此来达到防止环路的目的。**

这么一来本小节的第一个图就能理解了，既然 3 类 LSA 只能通过 area0 进行中转，且中转的动作必须由 ABR 来完成，那么 R4 这个屌丝自然无法为 area1 或 area2 去生成 3 类 LSA。

接下去，我们来看一个典型的问题：



上图中，编号 1、2、3、4 并非实际的协议运行步骤，只是我们观察点的一个转移而已。R1、R2、R3、R4 都会在 area1 内泛洪自己的 LSA，从而描述自己所有直连的 OSPF 接口（Link），因此 R3 将 4.4.4.0 的路由装载进路由表，这是经过 SPF 算法计算得出的最短路径，路由的类型为 O，区域内的路由。接下去我们将视角切换到

R1、R2，这两个 ABR 都会收到 area1 内路由器的 1LSA 的泛洪，它两会各自将 area1 内的 LSA1 及 LSA2（若有）搜集后归纳成 LSA3，并且在 area0 内进行泛洪，LSA3 的 LINKID 就是其描述的路由的前缀，在 R1 及 R2 上查看 OSPF database 可以看到，在 area0 中，R1 及 R2 都会各自泛洪 4 条 3 类 LSA，分别描述 10.1.13.0、10.1.34.0、10.1.24.0、4.4.4.0 这四个网段，如此一来，在 area0 中，就有共计 8 条 3 类 LSA 在泛洪。接下去我们将视角放到 R1，R1 从 R2 收到这四个网段的 3 类 LSA，会将这些 LSA 的通告路由器改为 R1 自身，并且进一步在 area1 内泛洪，如此一来 R3 将从 R1 收到四条 3 类 LSA，尽管 R3 并不需要这些 LSA，R3 会将忽略它们，甚至不会安装进自己的 LSDB。

OK，那么现在我们在 R3 上定义个 ACL，匹配 4.4.4.0，随后 debug ip routing access-list x 去跟踪这条路由的更新，然后在 R4 上 DOWN 掉 4.4.4.0，这个时候会出现个诡异的现象：

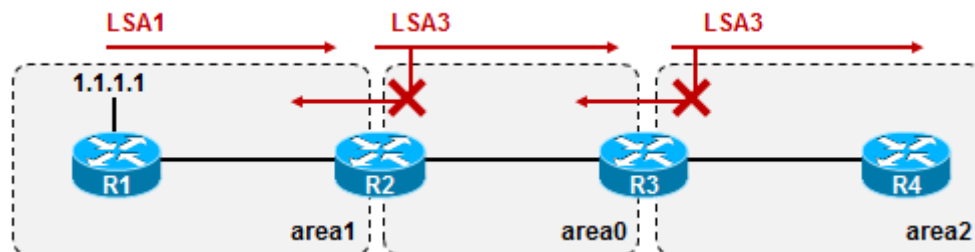
```
*Mar  1 11:08:00.057: RT: del 4.4.4.4/32 via 10.1.34.4, ospf metric [110/65]
*Mar  1 11:08:00.057: RT: delete subnet route to 4.4.4.4/32 // 路由 delete，因为 R4 的告知
*Mar  1 11:08:00.057: RT: NET-RED 4.4.4.4/32
*Mar  1 11:08:00.121: RT: SET_LAST_RDB for 4.4.4.4/32
    NEW rdb: via 10.1.13.1

*Mar  1 11:08:00.121: RT: add 4.4.4.4/32 via 10.1.13.1, ospf metric [110/193]
*Mar  1 11:08:00.125: RT: NET-RED 4.4.4.4/32 //路由竟然又回来了，因为 R1 还没收敛，仍然通告 LSA3
R3#
*Mar  1 11:08:05.125: RT: del 4.4.4.4/32 via 10.1.13.1, ospf metric [110/193]
*Mar  1 11:08:05.125: RT: delete subnet route to 4.4.4.4/32 //路由 delete R1 收敛了
*Mar  1 11:08:05.129: RT: NET-RED 4.4.4.4/32
*Mar  1 11:08:05.129: RT: SET_LAST_RDB for 4.4.4.4/32
    NEW rdb: via 10.1.34.4

*Mar  1 11:08:05.133: RT: add 4.4.4.4/32 via 10.1.34.4, ospf metric [110/321] //又回来了，R4 凌乱了
*Mar  1 11:08:05.133: RT: NET-RED 4.4.4.4/32
*Mar  1 11:08:05.265: RT: del 4.4.4.4/32 via 10.1.34.4, ospf metric [110/321]
*Mar  1 11:08:05.265: RT: delete subnet route to 4.4.4.4/32
*Mar  1 11:08:05.265: RT: NET-RED 4.4.4.4/32 //终于平静了
```

我们可以看到，R3 上关于这条路由经历了上面这个诡异的过程，造成这种现象的原因，还是在于 LSA3，OSPF 处理 LSA3 的行为，是典型的距离矢量的行为，是不是有感觉了？

当然，对于 LSA3，OSPF 也是存在类似水平分割的机制，如下图：



R2 将 area1 内的路由归纳成 LSA3 注入 area0，R3 收到后，将这些 LSA3 的信息修改，通告路由器变成自身，同时累加路由的 metric，然后再传入 area2，那么在这个过程中，R3 是不会将这些 LSA3 再注入回 area0 的。我们刚才那个的双 ABR 的环境比较特殊，因此存在上述的问题。

## 9 Forward Address

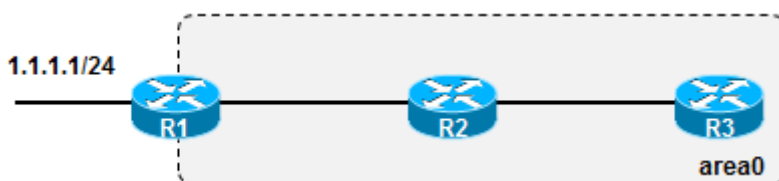
### 9.1 全 0 FA

The forwarding address is set to 0.0.0.0 if the ASBR redistributes routes and OSPF is *not enabled* on the next hop interface for those routes.

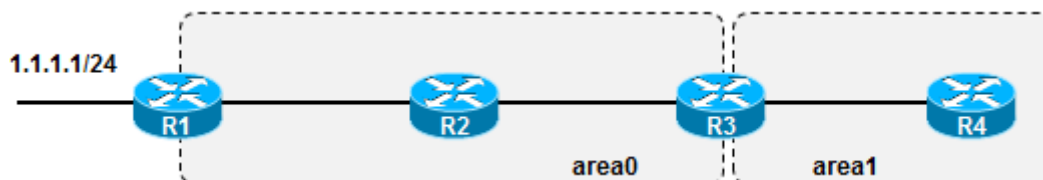
其他路由器在收到 5 类 LSA 时，必须在数据库中存在关于产生该 5 类 LSA 的 ASBR 信息。

ASBR 信息不是指路由表中存在 ASBR Router-id 的路由信息，而是数据库中存在 ASBR 产生的 Router LSA。

若数据库中存在 ASBR 产生的 Router LSA，则该路由器可以加载 5 类 LSA 进入路由表，其下一跳为到达 ASBR 的最近一跳



R2、R3 收到 R1 产生的 5 类 LSA 的 FA 为全 0，并且 R2、R3 都存在 R1 通告的 Router LSA，因此 R2、R3 都可以将 1.1.1.0 的外部路由加载进路由表。



当接收到 5 类 LSA 的路由器 R4 与产生该 5 类 LSA 的 ASBR 不在同一区域内时，由于 R1 (ASBR) 产生的 Router LSA 不会泛洪进 Area 1 (1 类 LSA 只在本区域内泛洪)，R4 的数据库中不存在 R1 (ASBR) 产生的 Router

LSA，因此将无法加载 5 类 LSA 进路由表，因为他不知道 ASBR 在哪。为了解决 ASBR 信息的问题，ABR（本例中的 R3）会向它所连接的其他区域（非 ASBR 所在的区域）通告一条 4 类 LSA，以标识 ASBR 信息（其中包含 ABR 到达该 ASBR 的距离）

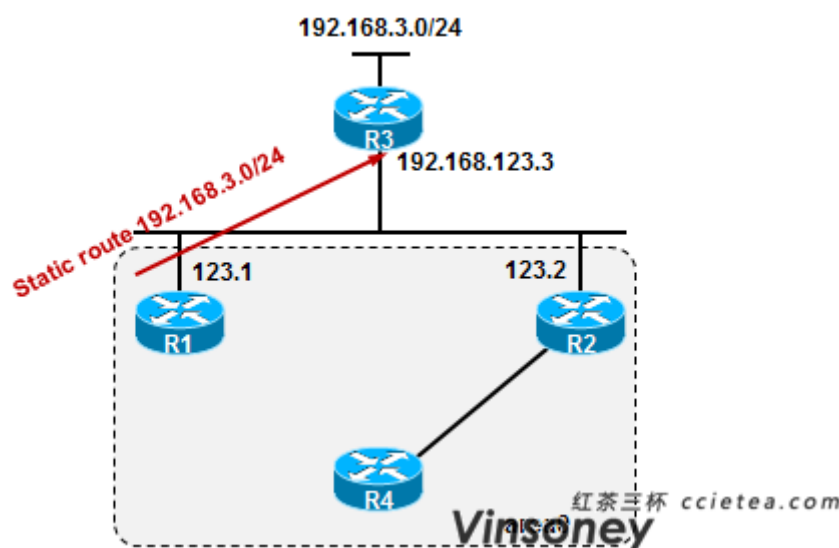
如此一来，R4 既有了关于 1.1.1.0 路由的 5 类 LSA，又有了通告这条外部路由的 ASBR 信息，因此可以将这条外部路由装载进路由表。

## 9.2 非 0 FA(no NSSA)

**非 NSSA 区域 ASBR 产生的 5 类 LSA，其 FA 非 0 的条件（四个条件需全部满足，缺一不可）：**

- 外部路由的下一跳接口启动 OSPF
- 外部路由的下一跳地址在 network 范围内
- 外部路由的下一跳接口没有被设置为被动接口
- 外部路由的下一跳接口的网络类型不是 P2P 或者 P2MP

**当上述条件都满足，ASBR 在产生 5 类时就将 FA 置为外部路由的下一跳。**

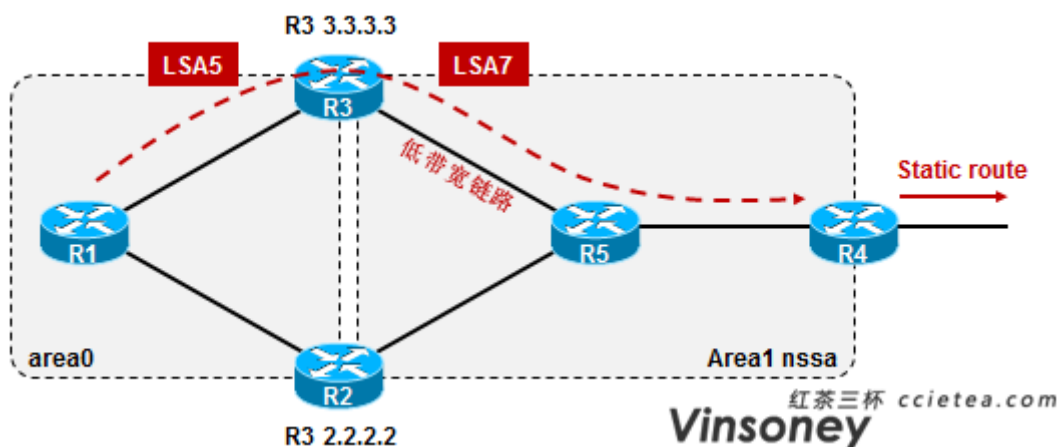


为什么需要 FA 呢？考虑一下如果没有 FA 的情况下回如何：R1 重发布静态路由（去往 3.0，下一跳是 123.3），R2 接收到 R1 产生的 5 类 LSA，其 FA 为 0，因此加载该外部 LSA 进入路由表，下一跳为到达 ASBR（R1）最近的下一跳也就是 192.168.123.1，R4 接收到 R1 产生的 5 类 LSA，其 FA 为 0，因此加载该外部 LSA 进入路由表，下一跳为到达 ASBR 最近的下一跳，结果是 R4 访问 192.168.3.0 是用的次优路径，R2---R1---R3 这样走。

**解决办法：**使用非 0 FA，R1 始发的这条关于外部路由 192.168.3.0 的 5 类 LSA，其 FA 设置为 192.168.123.3，R2 接收到 R1 产生的 5 类 LSA，其 FA 非 0（符合上述的四个条件），FA 地址为 192.168.123.3，该地址通过直连路由可达，因此加载 5 类 LSA 进路由器，其下一跳为到达 FA 的下一跳。如此一来，次优路径的问题就解决了。

## 9.3 非 0 FA (NSSA)

### ● 验证实验 1

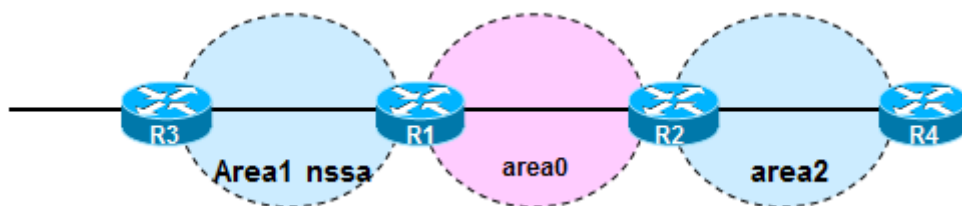


在 NSSA 区域存在多 ABR 时,只有 router-ID 大的 ABR 才会进行 7 转 5 的动作,这个我们已经知道鸟。

上图中, R4 重发布静态路由,只有 R3(Router-ID 大的)进行了 7 转 5,并且在 7 转 5 的过程中,没有产生 Type 4 LSA (描述 ASBR R4,因为右边是 NSSA 区域, R3 虽为 NSSA 的 ABR,但是同时也兼具 ASBR 的属性,这些 7 转 5 后生成的 5 类 LSA,对于 area0 而言,是产生自 R3 的), R1 无法掌握真实的 ASBR( R4 ) 的信息,因此在路由表加载时,选择了一条次优路径, R1-R3-R5-R4。

使用 FA 可以解决这个问题, R4 在重发布进静态路由时, 7 类 LSA 携带 FA =192.168.45.4, 这个 FA 伴随着 R3 的 7 转 5 的动作传递到 R1,而 R1 去往这个外部路由,就看本地前往该 FA 的内部路由(最短路径),因此走 R2、R5、R4。

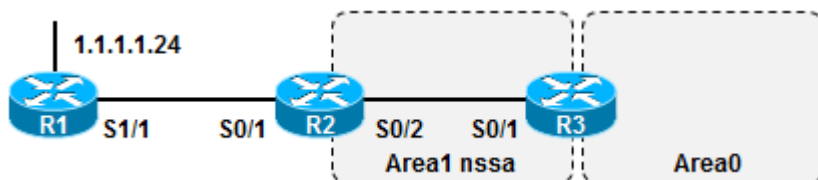
### ● 验证实验 2



由于 r3 在重发布直连时,产生的 7 类 LSA 的 FA 被填充为一个非 0 的 IP 地址 ( 172.16.13.3 ), R1 负责 7 转 5,FA 不变仍为 172.16.13.3, R4 在收到 5 类 LSA 后,发现其 FA 非 0,因此就首先在路由表中查找是否有到达 172.16.13.3 的 OSPF 内部路由,结果查找不到, 5 类 LSA 所指示的网段 192.168.3.0/24 就无法加载进路由表。

**解决办法:** 使用 area 1 nssa translate type7 suppress-fa, R1 进行 7 转 5 时,会将 FA 地址置为全 0。

### ● 总结实验

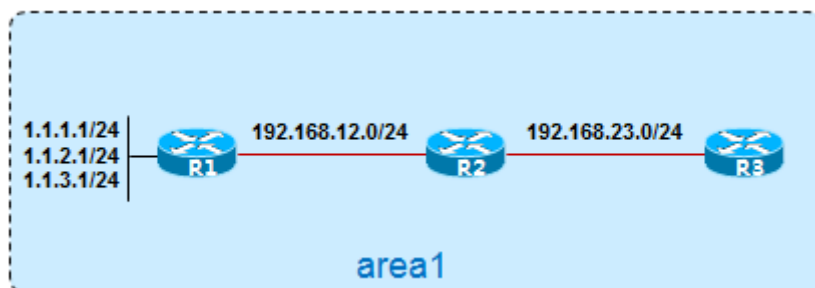


关于 NSSA 相关的 FA，具体填充的是什么地址，需要通过实验验证，并且不同的厂商有所不同。

1. **R2 作为 NSSA 区域的 ASBR，在 R2 上如果 1.1.1.0 是通过其他协议学习到的，比如 rip，EIGRP，无论是内部路由还是外部路由，在重分发进 OSPF 的时候：**
  - 1) 如果 R2 的 S0/1 不宣告进 OSPF，那么 FA 将被设置为 192.168.23.2，即 R2 的 S0/2 的接口地址
  - 2) 如果 R2 的 S0/1 被宣告进 OSPF 且 R2 与 R1 之间链路类型为广播型链路，那么 FA 是 R1 的 S1/1 地址 192.168.12.1
  - 3) 如果 R2 的 S0/1 被宣告进 OSPF 且 R2 与 R1 之间链路类型为点到点型链路，那么 FA 是 R2 的 S0/1 地址 192.168.12.2
2. **如果被重分发进 OSPF 的网络不是通过其他协议学习到的，而是本地直连的**
  - 1) 如果除了 R2 的 S0/2 被宣告进 OSPF 之外，还有其他接口也被宣告进了 OSPF，那么选取其他接口的地址作为 FA，多个接口被宣告，选择最大的宣告 IP。
  - 2) 除了 R2 的 S0/2 没有其他接口被宣告进 OSPF，那么选取 R2 的 S0/2 的地址作为 FA

## 10 CASE 分析

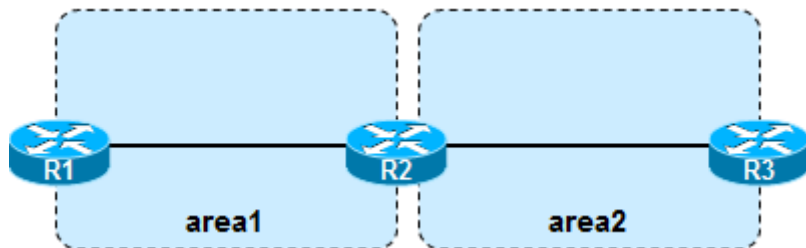
### 1. 单区域非 area0 的情况



当只有一个区域的时候，这个区域就未必一定要是 area0，可以是任意的常规区域，由于单区域的话，LSA1、2 的泛洪都没有问题，且借助 LSA1、2 区域内的路由器都能了解到整个区域的所有链路，因此全网路由都是可达的。

### 2. 关于多区域无 area0 的情况





R1 及 R2, R2 及 R3 都能建立起邻居关系, 但是 R1 无法学习到 area2 的 LSA, R3 同样无法获取 area1 的 LSA, 因为没有 ABR, ABR 必须是与 area0 及非骨干区域直连的路由器, 显然 R2 不是, 既然不是 ABR, 也就没有权利泛洪 type 3 LSA。

R1 宣告直连进 area1, 则 R2 能学习到 ( 单区域内的 LSA1、2 泛洪没有问题 ), R3 无法学习到。

### 3. 关于辅助地址

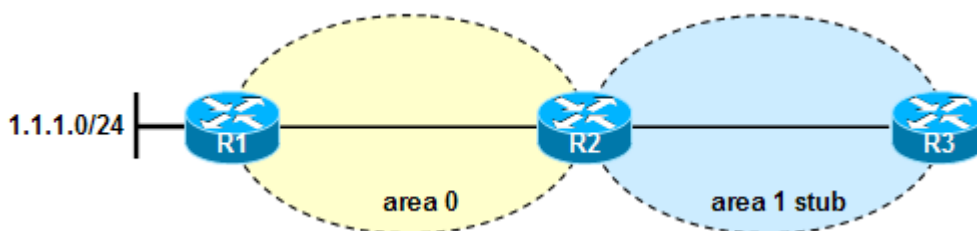
只有在主网络或子网也运行 OSPF 协议的时候, OSPF 才会通告一个辅助的网络或子网。

OSPF 将把辅助地址看作是末梢网络(这些网络上没有 OSPF 邻居),从而不会在这些网络上发送 Hdlo 数据包。因此,在辅助网络上就无法建立邻接关系。



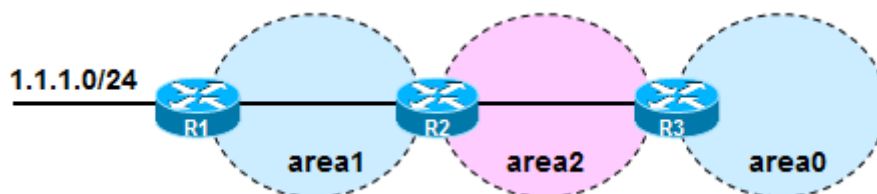
如果需要通过辅助地址与其他路由器建立邻居关系, 如上图, R2 可使用子接口与 R1 建立邻居关系

### 4. 关于 4 类 LSA 的生成



R1 重发布 1.1.1.0 入 OSPF, 此时 R1 为 ASBR, 而 R2 为 ABR, 但是目前的拓扑结构中, R2 又不能向 area1 注入 4 类 LSA, 因为 area1 为 stub, 所以这个时候 R2 的 database 里并没有 4 类 LSA, 倘若 R2 上再挂一个常规区域 area2, 则 R2 会为 area2 生成 LSA3 类, 指向 R1。

### 5. 关于 4 类 LSA 的生成 2



R3 能收到 1.1.1.0 的 5 类 LSA，但是不会生成对应的路由条目，因为 R3 上没有关于 R1 这个 ASBR 的 4 类 LSA。原因是 R2 并不认为自己的 ABR（没有任何一个接口属于 area0），因此 R2 不会产生 4 类 LSA

## 11 参考书目

### 1. BOOKS

TCP/IP 卷一

疑难解析