学习完了RIP协议，趁热打铁，继续来学习OSPF协议。废话不多说，走起~

一、OSPF基本工作原理

OSPF英文全称为Open Shortest Path First，即开放最短路径优先，是为克服RIP的缺点在1989年开发出来的。

- “开放”表明OSPF协议不是受某一家厂家控制，而是公开发表的。
- “最短路径优先”是因为使用了Dijkstra提出的最短路径算法SPF。

OSPF采用SPF算法计算路由，从算法上保证了不会产生路由环路。OSPF相对于RIP，不限制网络规模，更新效率高、收敛速度快。

OSPF是基于链路状态的，而不是像RIP是基于距离向量的。

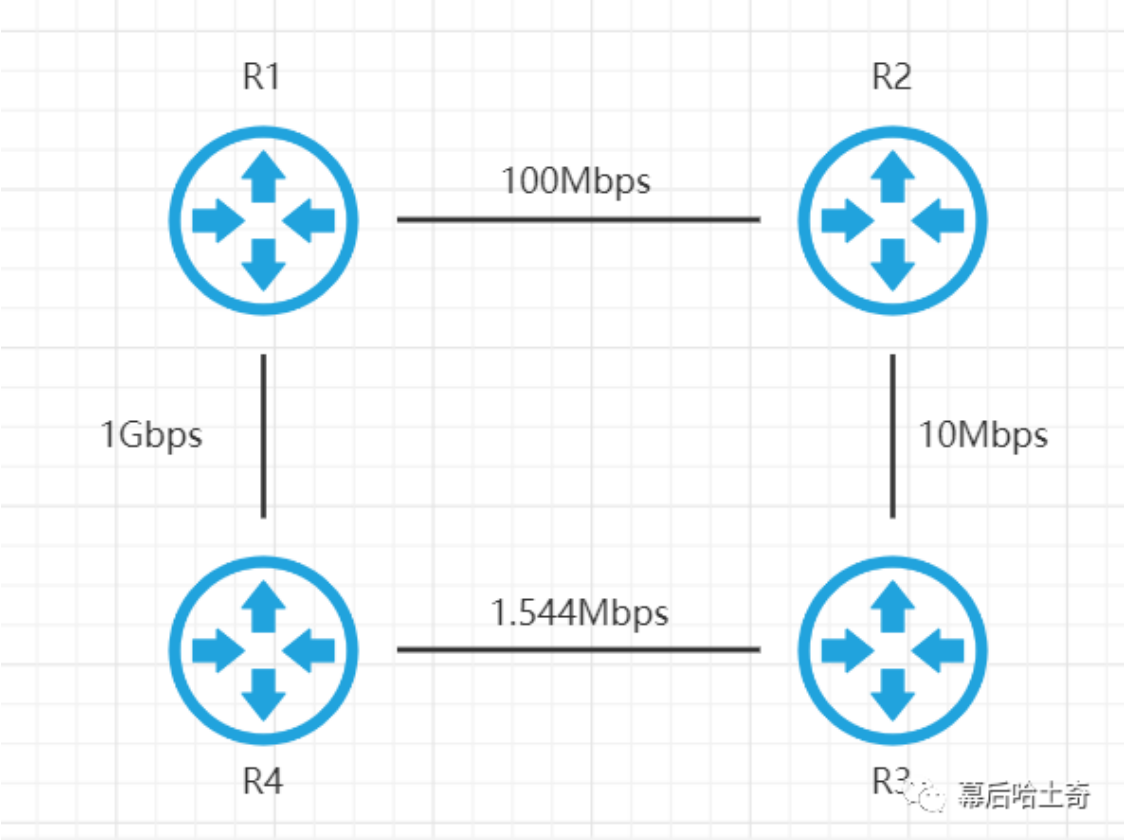
这里说的链路状态是指本路由器都和哪些路由器相邻，以及相应链路的“代价”（cost）。

代价用来表示费用、距离、时延、带宽等等，这些都由网络管理人员来决定的。

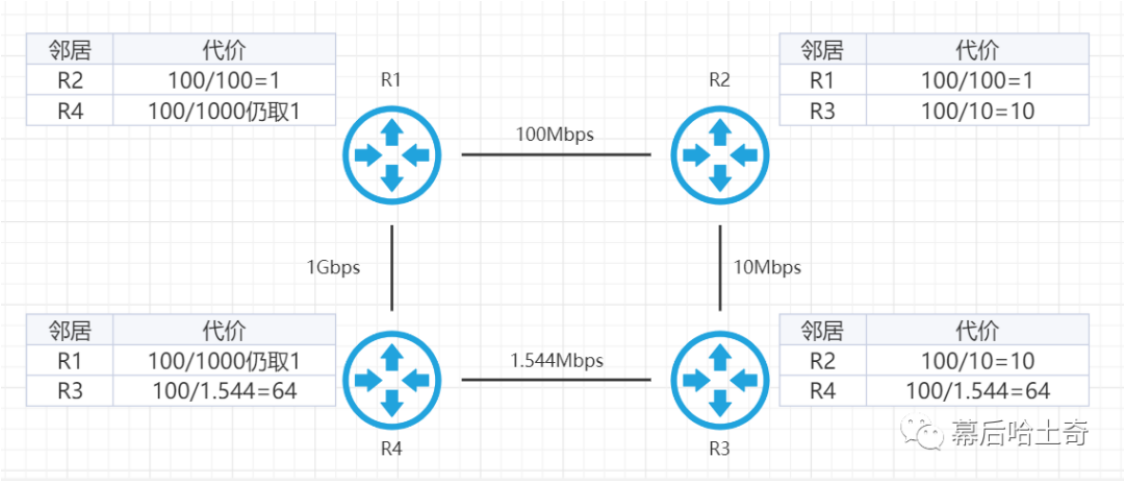
说起来很抽象，我们来举个例子，比如在思科路由器中OSPF计算代价的方法是：用100Mbps / 链路带宽。

- 计算结果小于1的值仍然记为1；
- 计算结果大于1的值且有小数的，舍去小数。

显然考虑的是带宽因素，我们假设有下面这么一个网络：



下面来看看其链路状态是多少。



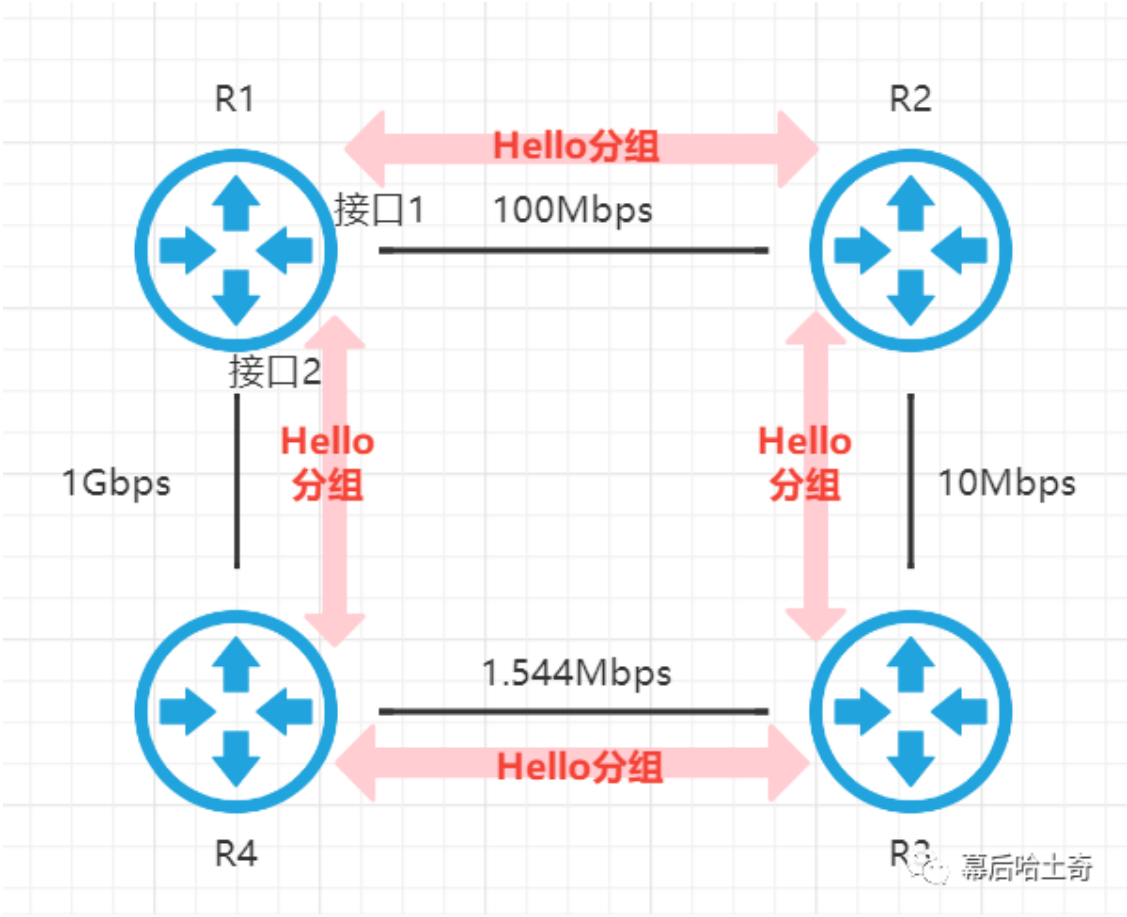
链路状态将是后续相邻路由器之间主要交互的信息。

二、OSPF问候分组

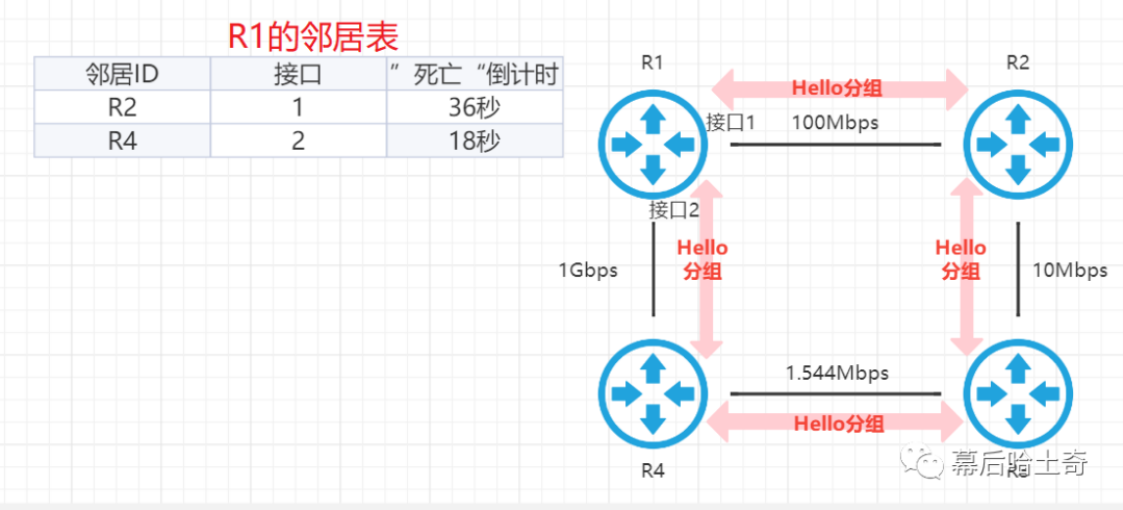
问候分组即Hello分组，Hello分组封装在IP数据报中，发往组播地址224.0.0.5，该数据报头部结构如下：

IP数据报首部中的协议号字段的取值应为89，来表明IP数据报的数据载荷为OSPF分组。

问候分组的发送周期为10秒，若40秒内未收到来自邻居路由器的问候分组，则认为该邻居路由器不可达，因此每个路由器都会建立一张邻居表，里面有一个值是死亡倒计时，就是从40秒开始倒计时，如果40秒内一直收不到来自某个邻居路由器的问候分组，则认为此邻居路由器已经死亡，即不可达，我们来看个实例：



假设这是R1路由器的一张邻居表，R2是R1的一个邻居路由器，R1中的邻居ID就用R2来标识（这里只是简单举例说明），R2连接到R1的接口1上，此时“死亡”倒计时还剩余36秒，若在死亡倒计时到0之前收到了来自R2的问候分组，则重新将R2对应的“死亡“倒计时置为40秒重新倒计时。



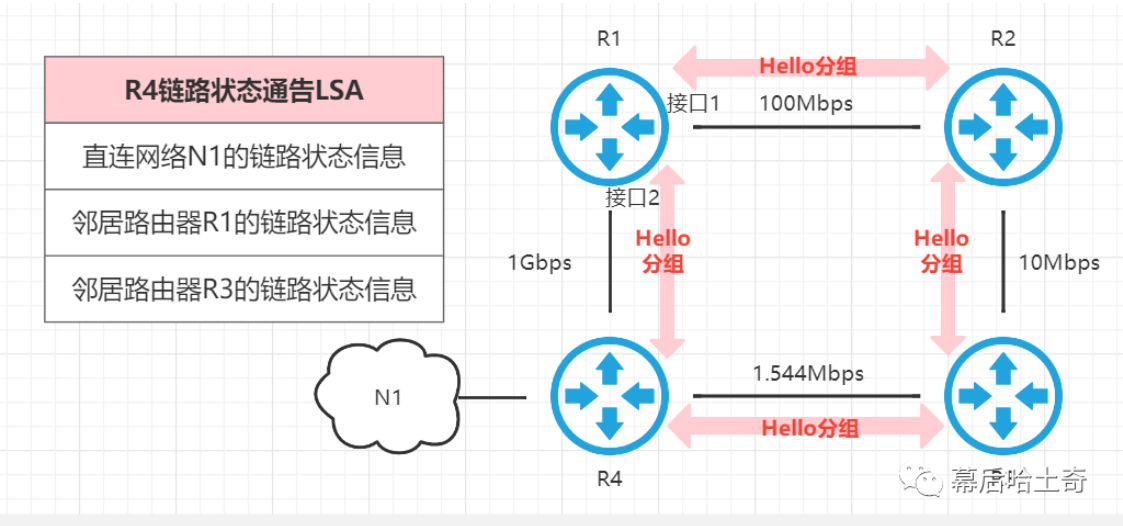
否则，当死亡倒计时到达0时，则判定该邻居路由器不可达。

三、OSPF链路状态通告LSA

使用OSPF的每个路由器都会产生链路状态通告LSA（Link State Advertisement），LSA中包含以下内容：

- 直连网络的链路状态信息
- 邻居路由器的链路状态信息

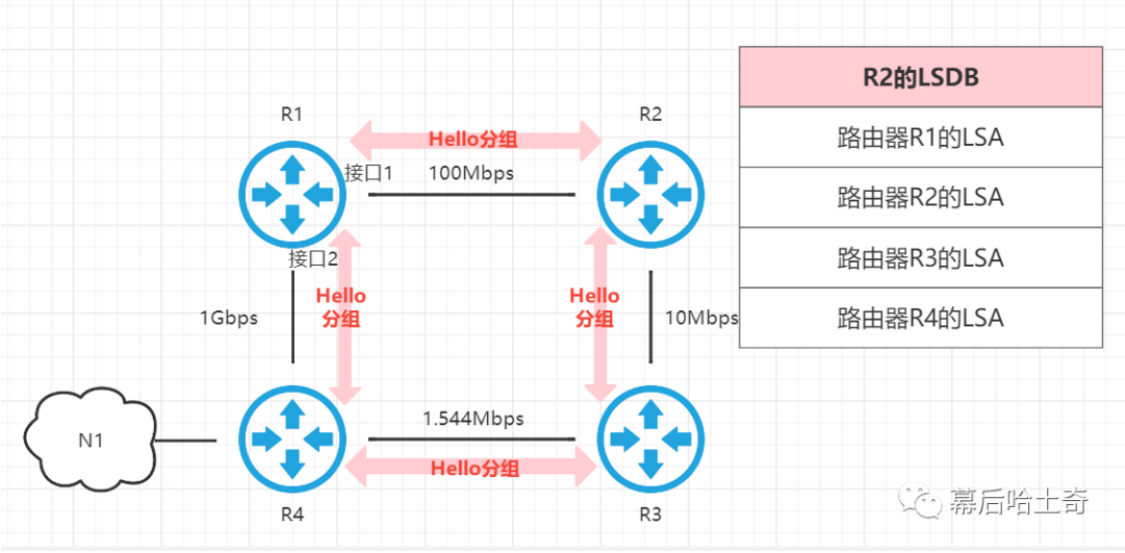
以R4为例，R4的链路状态通告应包含R4与该直连网络的链路状态信息、其邻居路由器R1和邻居路由器R3的链路状态信息。



LSA被封装在链路状态更新分组LSU中，采用洪泛法发送，收到链路状态更新分组的路由器，将从自己所有其他接口转发此分组，也就是进行洪泛转发。

这样，自治系统内每个路由器所发送的封装有LSA的LSU会传递给系统内其他所有路由器。

使用OSPF的每个路由器都有一个链路状态数据库LSDB，用于存储LSA。



通告各路由器洪泛发送封装有自己LSA的LSU分组，各路由器的LSDB状态最终将达到一致。

使用OSPF的各路由器，基于LSDB进行最短路径优先SPF计算，构建出各自到达其他各路由器的最短路径，即构建各自的路由表。

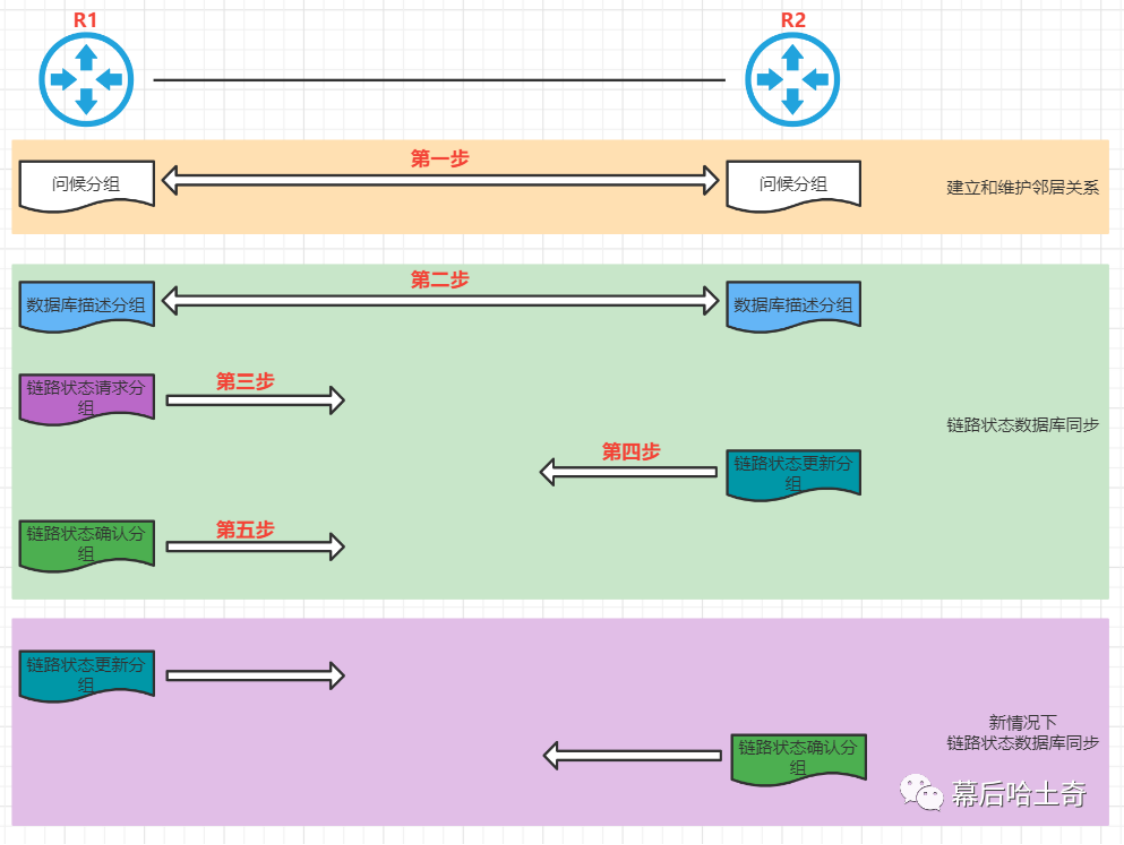
关于SPF如何构建最短路径的过程不是这里讨论的范围，将不展开说明。

四、OSPF基本工作过程

在说明基本工作过程之前，需要说明，OSPF拥有以下五种分组类型：

- 类型1，问候（hello）分组：用来发现和维护邻居路由器的可达性；
- 类型2，数据库描述（Database Description）分组：向邻居路由器给出自己的链路状态数据库中的所有链路状态项目的摘要信息；
- 类型3，链路状态请求（Link State Request）分组：向邻居路由器请求发送某些链路状态项目的详细信息；
- 类型4，链路状态更新（Link State update）分组：路由器使用这种分组将其链路状态进行洪泛发送，即用洪泛法对全网更新链路状态；
- 类型5，链路状态确认（Link State Acknowledgment）分组：这是对链路状态更新分组的确认分组。

是不是除了类型1，其他有点懵，下面结合以上五种类型说明工作过程即可明白为什么用到五种分组类型、每种分组类型的作用是什么。



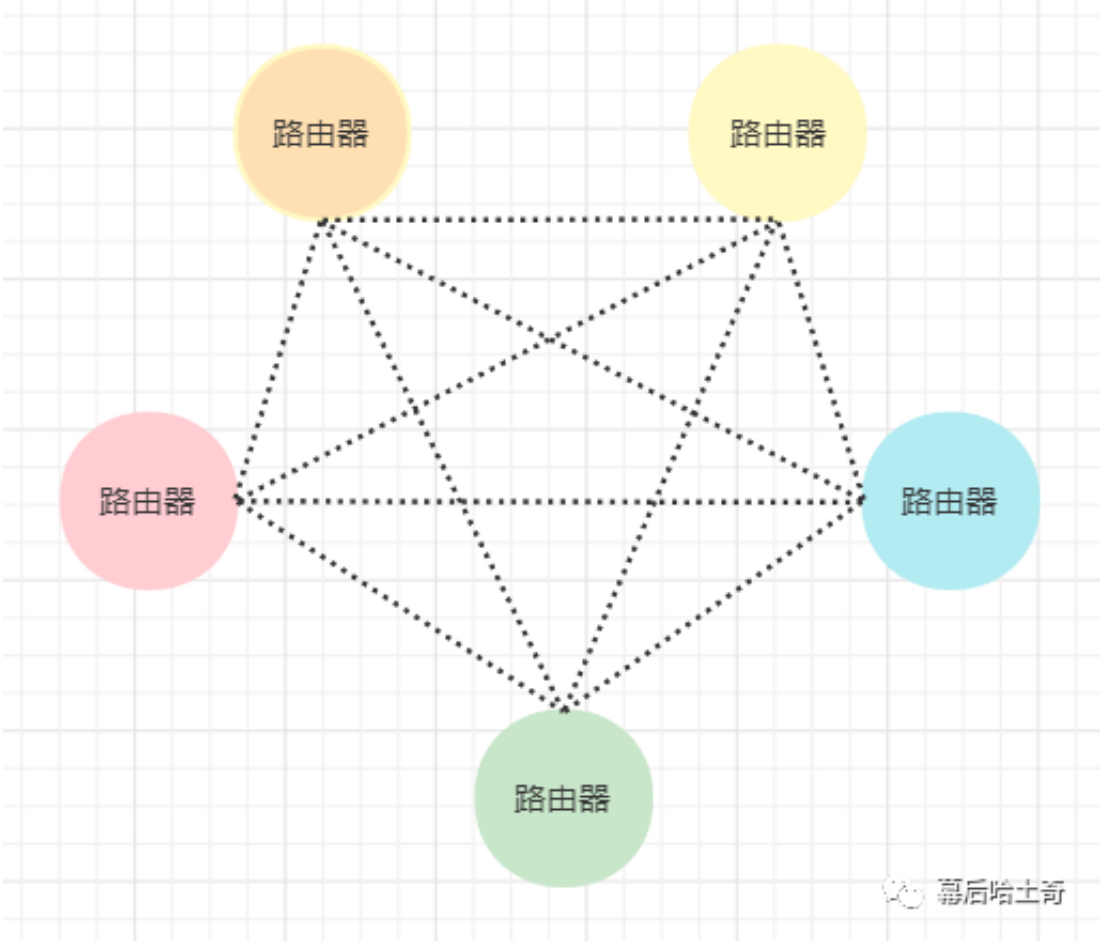
- **第一步**：相邻路由器之间周期性发送问候分组，以便建立和维护邻居关系。
- **第二步**：建立邻居关系后，给邻居路由器发送数据库描述分组，也就是将自己的链路状态数据库中的所有链路状态项目的摘要信息发送给邻居路由器。（由于OSPF是增量更新，那么如何判断增量呢？无非是先跟别人要一个全量的简单信息，我看缺少哪些，缺哪些就走到第三步）
- **第三步**：例如R1收到R2的数据库描述分组后，发现自己缺少其中的某些链路状态项目，则向R2发送链路状态请求分组。
- **第四步**：R2收到来自R1的链路状态请求分组后，将R1所缺少的链路状态项目的详细信息封装在链路状态更新分组LSU中发送给R1。
- **第五步**：R1收到后将这些所缺少的链路状态项目的详细信息添加到自己的链路状态数据中，并给R2发送链路状态确认分组。

反之，R2也可以向R1发送链路状态请求分组，过程一样，最终R1和R2的链路状态数据库将达到一致，这就是链路状态数据库同步的过程。

每30分钟或链路状态发送变化时，路由器都会发送链路状态更新分组，收到该分组的其他路由器将洪泛转发该分组，并给该路由器发回链路状态确认分组，这又被称为新情况下链路状态数据库同步。

五、OSPF-DR和BDR

我们之前说过，OSPF路由器会周期性地发送问候分组以建立和维护邻居关系，是往多播地址224.0.0.5发送的，**那么当OSPF路由器在多点接入网络中建立邻居关系时，如果不采用其他机制，将会产生大量的多播分组。**

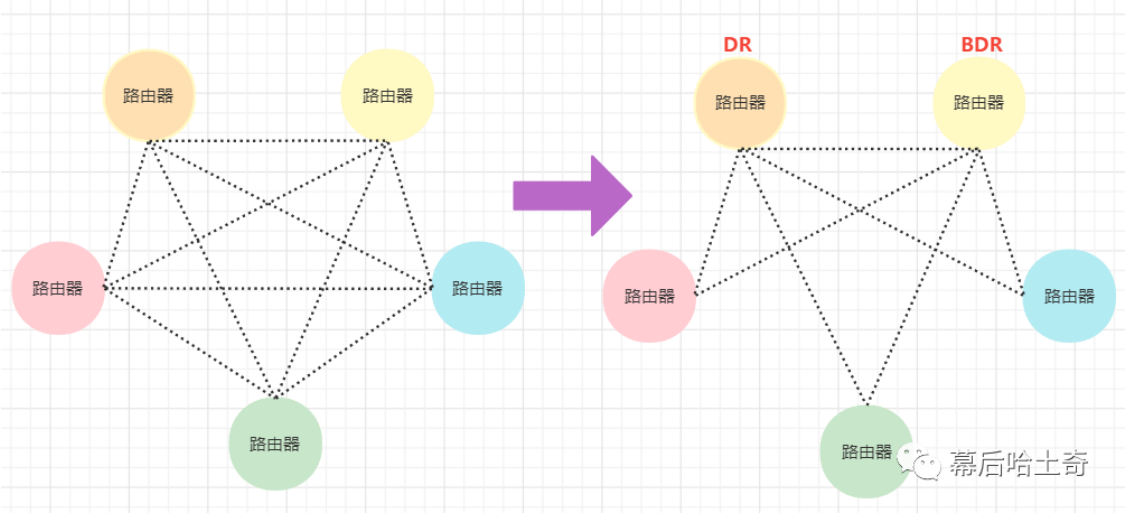


如上图，每个路由器之间互为邻居路由器，那么每个路由器都要向其他（N-1）个路由器发送问候分组和链路状态更新分组。

如何解决这个问题呢?

我们会选择指定路由器DR（designated router）和备用的指定路由器BDR（backup designated router）。

假设下面这两个路由器被选举为DR和BDR，那么所有非DR/BDR的路由器只与DR/BDR路由器建立邻居关系，非DR/BDR之间不能直接交互信息。



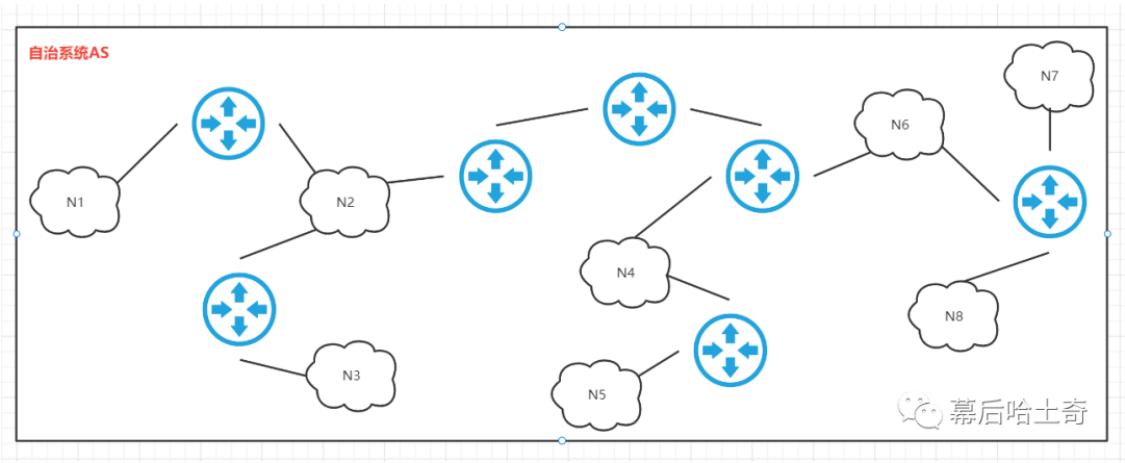
这样邻居关系数量大大减少。若DR出现问题，则有BDR顶替，实现DR和BDR的选举并不复杂，无非就是各路由器之间交互一些选举参数，例如路由器优先级、路由器ID、接口IP地址等，然后根据选举规则选出DR和BDR。

> 笔者注：学到这里，其实无论是问候分组机制，还是链路状态通告机制，还是选择DR机制，其实这些在分布式系统中，都是差不多的解决思路，只是说法或侧重点有略微不同而已，当然，也是造成了理解上复杂性的重要原因，不过没有办法，引入分布式就是为了大型系统高性能的需求，必然就需要面对这些问题，深入理解分布式原理，也是后端程序员的一道职业护城河。

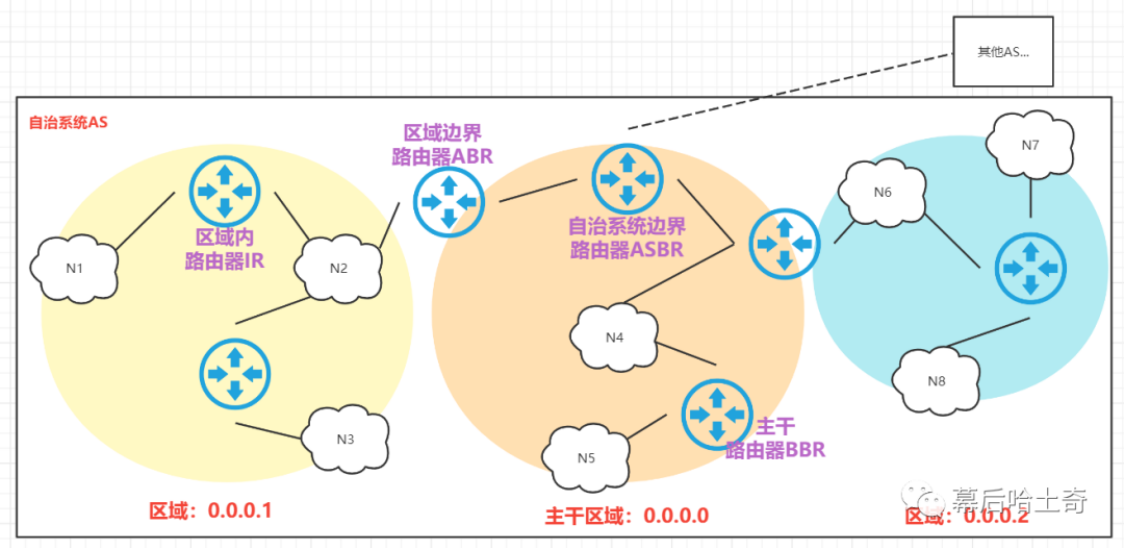
六、OSPF再划分-区域

为了使OSPF能够用于规模很大的网络，OSPF把一个自治系统再划分为若干个更小的范围，称为区域。

原本是一个大的自治系统AS：



OSPF将其划分为三个更小的区域：



每个区域都有一个32比特的区域标识符，可以用点分十进制表示，例如主干区域标识符必须为0，可以表示为点分十进制0.0.0.0。

主干区域用于连通其他区域，其他区域标识符不能为0且互不相同，每个区域的规模不应太大，一般所包含的路由器不应超过200个。

这样划分区域的好处是，把利用洪泛法交互链路状态信息的范围局限于每个区域而不是整个自治系统，这样就减少了整个网络的通信量。

如果路由器的所有接口都在一个区域内，这类路由器称为区域内路由器。

为了本区域可以和自治系统内其他区域连通，每个区域都会有一个区域边界路由器，它的一个接口连接自己所在区域，另一个接口用于连接主干区域。

可见区域边界路由器是一个城门，以区域0和区域1之间的区域边界路由器为例，它向主干区域发送0.0.0.1的链路状态通告LSA信息，并且向0.0.0.1区域发送主干区域0.0.0.0和0.0.0.2的链路状态通告LSA信息。

主干区域内的路由器称为**主干路由器**，我们可以将区域边界路由器和主干区域内的路由器都看作是主干路由器。

此外，主干区域内还有一个特殊的路由器：专门和本自治系统外的其他自治系统交换路由信息，这样的路由器称为**自治系统边界路由器**。

上述几类路由器可归类如下：

- 区域内路由器IR（internal router）
- 区域边界路由器ABR（area border router）
- 主干路由器BBR（backbone router）
- 自治系统边界路由器ASBR（AS border ruter）

这种区域划分的方式，虽然使得OSPF更加复杂，但是大大降低了整个自治系统的通信量，因而使得OSPF可以适用于规模很大的自治系统中。好了，下一节来学习跨AS的边界网关协议BGP。