# 第五章 网络层（3）

袁华：hyuan@scut.edu.cn

华南理工大学计算机科学与工程学院

广东省计算机网络重点实验室

# 本节的主要内容（5.2.5节，P288）

- 链路状态路由算法
  - 实例：开放的最短路径优先（OSPF）
- 边界网关协议（BGP）

# 为什么DV逐渐让位于LS？

- DV
  - 站的不高，看得不远
  - 完全相信邻居
- LS
  - 想办法站得高，看更
  - 多高、多远？
  - 怎么做？

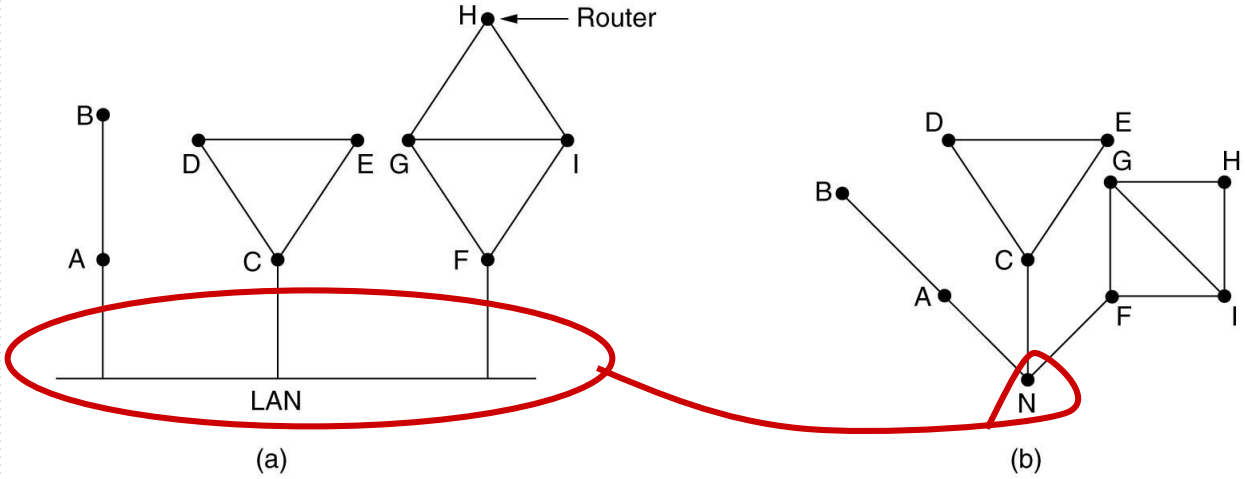

# 链路状态路由（Link State） P288

- 在1979年前，ARPANET采用DV路由（RIP）协议，此后，采用了链路状态路由选择协议
- 目前，链路状态路由算法得到了广泛的应用
- 链路状态路由的主要思想包括如下5个部分：
  - 发现它的邻居节点们，了解它们的网络地址
  - 设置到它的每个邻居的成本度量
  - 构造一个分组，包含它所了解到的所有信息
  - 发送这个分组给所有其他的路由器
  - 计算到每个路由器的最短路径

生活中的例子

# 发现邻居节点 P288

- 当一个路由器启动的时候，在每个点到点的线路发送一个特别的**HELLO分组**
- 收到**HELLO**分组的路由器应该回送一个应答，应答中有它自己的名字（采用一个全球唯一的名字 **globally unique name**）
- 当两个或更多的路由器被一个**LAN**连接起来，这个**LAN**被看作一个节点



# 设置链路成本 P289

- 为了决定线路的开销，路由器发送一个特别的ECHO 分组，另一端立刻回送一个应答
- 通过测量往返时间（round-trip time），发送路由器可以获得一个合理的延迟估计值
  - 为了得到更好的结果，可多次测量，取均值
- 一种常用的选择
  - 与链路带宽成反比

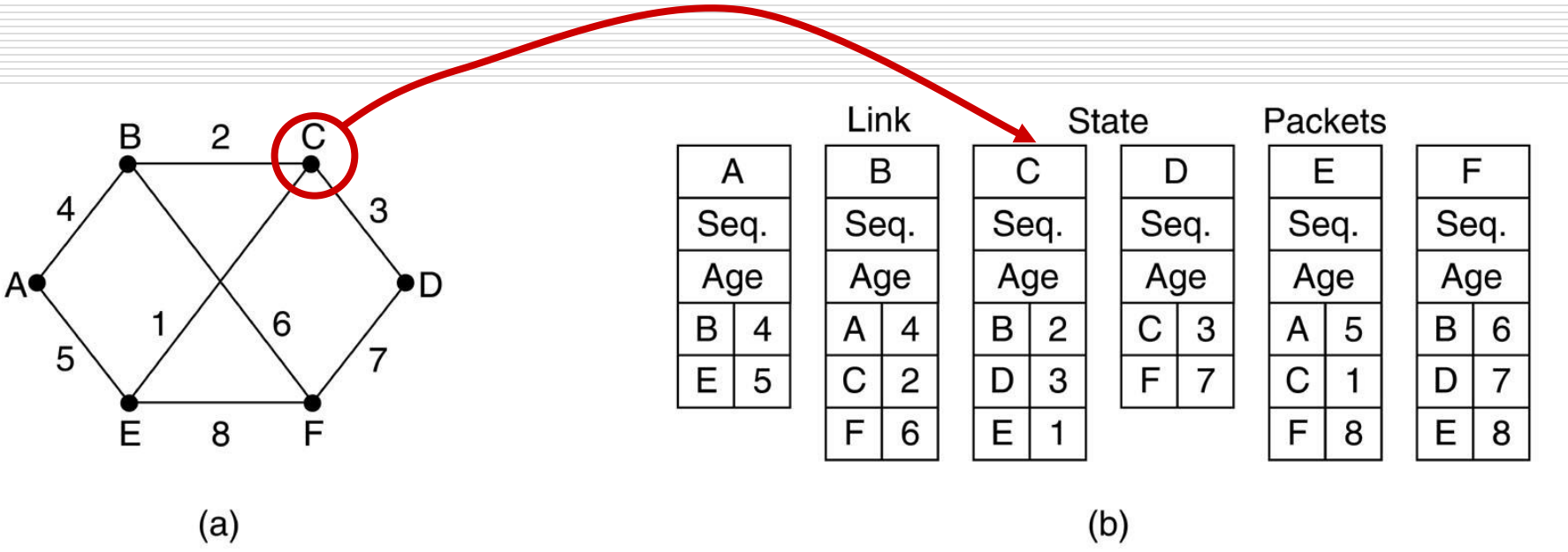
# 构造链路状态分组P289

- 链路状态分组构造后被发送给其他的路由器，分组中包含这些信息：
  - 发送方的标识（ID of the sender）
  - 序列号（sequence number ）
  - 年龄（age ）
  - 邻居列表（list of neighbors ）
  - 到邻居的成本/量度（delay to each neighbor ）

链路状态信息

- 应该什么时候构造分组?
  - 周期性地构造和发送，或者有特别的事件发生时构造，比如某条线路或邻居down掉了

# 构造链路状态分组（续）




**(a) A subnet. (b) The link state packets for this subnet.**

# 发布链路状态分组 P289

- 基本算法：
  - 每个分组都包含一个序列号，序列号随着新分组产生而递增
  - 路由器记录下他看见的所有 (源路由器，序列号)对
  - 当一个的新的分组到达时，路由器根据它的记录：
    - 如果该分组是新的，就被从除了来线路外的所有其他线路转发出去 ( flooding，泛洪)
    - 如果是重复分组，即被丢弃(喜新厌旧)
    - 如果该分组的序列号比对应的源路由器发送的到过此地的分组的最大序列号还小，则该分组被当作过时的信息而被拒绝

# 发布链路状态分组（续）

- 基本算法遇到的问题：
  - 序列号回转，引起新老分组识别混淆
    - 解决办法：使用 32-bit 的序列号，即使每秒产生一个分组，也需要137年才发生号码回转
  - 如果一台路由器崩溃，那么他将丢失自己的序列号记录，如果他再从0开始，新分组将被当作旧分组被拒绝
  - 如果一个序列号被破坏了，比如发送方的序列号是4，但是由于产生了1位错误，序列号被看作65540，那么，序列号为 5 – 65540的分组都被当作过时分组被拒绝

00000000000000010000000000000100

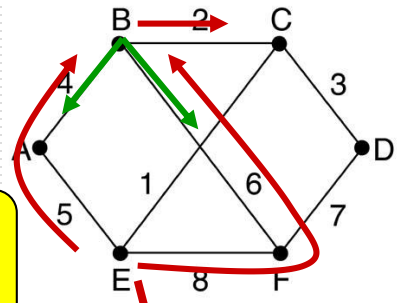
# 发布链路状态分组（续）

- 解决上述的路由器崩溃和序列号损坏的方法是：每个分组的序列号之后是年龄(age)，并且每秒钟年龄减1
- 当年龄为零 ( zero )时，来自该路由器的信息被丢弃
- 通常地，每隔一段时间，比如10秒钟，一个新分组就会到来，所以，只有路由器down机才可能导致超时（ 或者，连续6个间隔因为丢失，没有收到新的分组）

# 发布链路状态分组（续P290）

- 一些改进让基本算法更加健壮：
  - 当一个链路状态分组到达某个路由器时，它首先被放到一个保留区中等待一段时间
  - 如果来自相同路由器的另一个分组到达了，这两个分组的序列号会被比较：
    - 如果相等，是重复分组，丢弃
    - 如果不相等，旧的那个被丢弃
  - 为了防止路由器到路由器的线路发生错误，所有的链路状态分组都要被确认
  - 当一条线路空闲的时候，路由器扫描保留区，以便选择一个分组或确认，并将其发送出去

# 发布链路状态分组LSP （续）




| A | | B | | C | | D | | E | | F | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Seq. | | Seq. | | Seq. | | Seq. | | Seq. | | Seq. | |
| Age | | Age | | Age | | Age | | Age | | Age | |
| B | 4 | A | 4 | B | 2 | C | 3 | A | 5 | B | 6 |
| E | 5 | C | 2 | D | 3 | F | 7 | C | 1 | D | 7 |
| | | F | 6 | E | 1 | | | F | 8 | E | 8 |

Link　　State　　Packets

B的保留区

E's LSP arriv again from C.

| Source | Seq. | Age | Send flags A | Send flags C | Send flags F | ACK flags A | ACK flags C | ACK flags F | Data |
|---|---|---|---|---|---|---|---|---|---|
| A | 21 | 60 | 0 | 1 | 1 | 1 | 0 | 0 | |
| F | 21 | 60 | 1 | 1 | 0 | 0 | 0 | 1 | |
| E | 21 | 59 | 0 | 1 | 0 | 1 | 0 | 1 | |
| C | 20 | 60 | 1 | 0 | 1 | 0 | 1 | 0 | |
| D | 21 | 59 | 1 | 0 | 0 | 0 | 1 | 1 | |

0 0 0 1 1 1

# 计算新的路由路径 P291

- 一旦一个路由器获得了全部的链路状态分组就可以构造出全网络图来了（**Graph**）
- 现在，可以使用 最短路径算法来计算路由器之间的最短路径了
- 计算结果是一棵树，会形成相应的路由，安装在路由表中，引导数据分组的转发

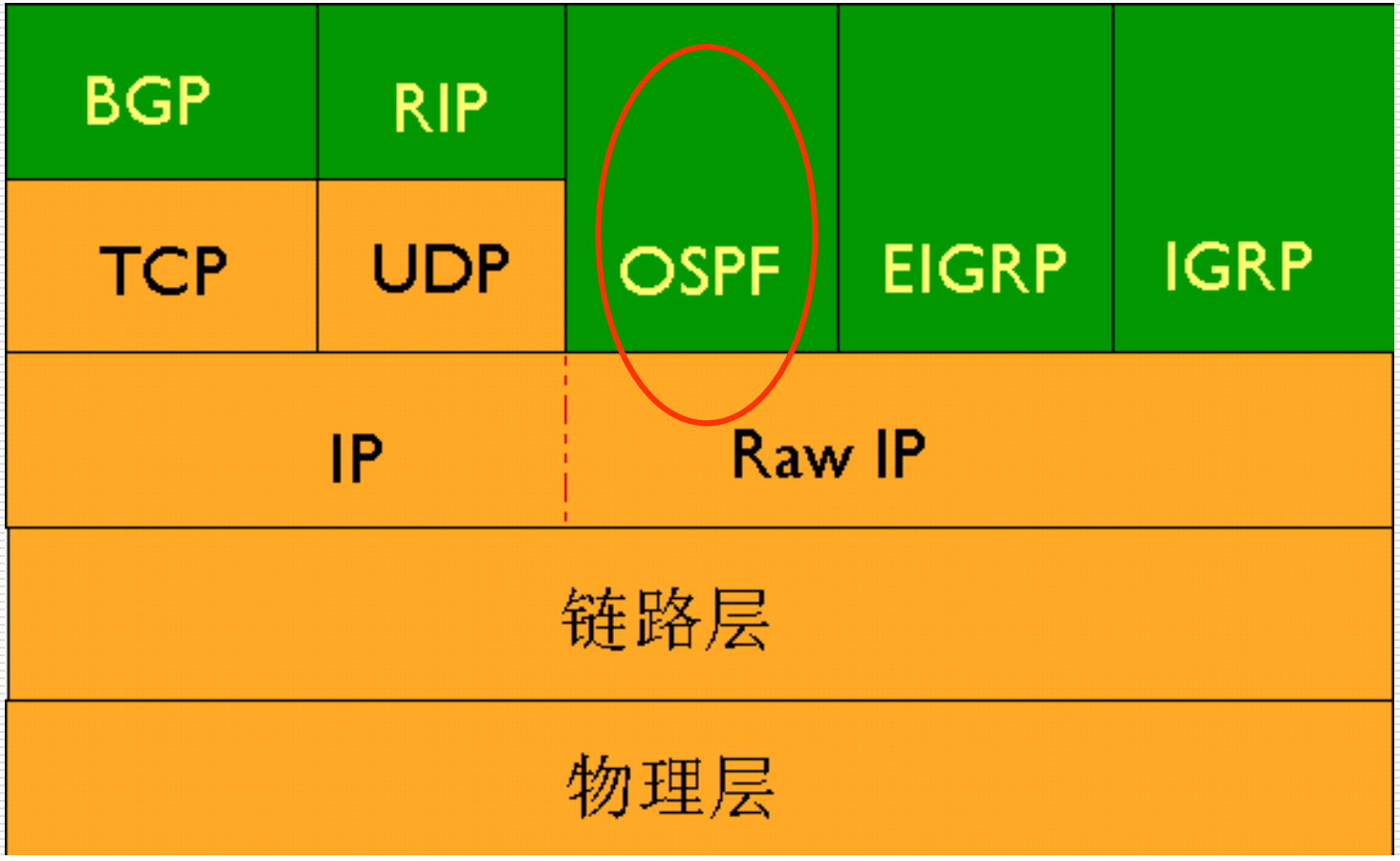
# L-S 路由算法的特点

- 优点
  - 每个路由器的认识一致
  - 收敛快
  - 适合在大型网络里使用
- 缺点
  - 每个路由器需要较大的存储空间
  - 计算负担很大

# L-S路由协议的实例—OSPF P365（5.6.6）

- 开放的路径优先（**Open shortest path first**）
- 使用图（**graph**）来表述真实的网络
  - 每个路由器/Lan都是一个节点
  - 测量代价/量度（**metric**）
- 计算最短路径

# OSPF在参考模型中的地位



# OSPF概述

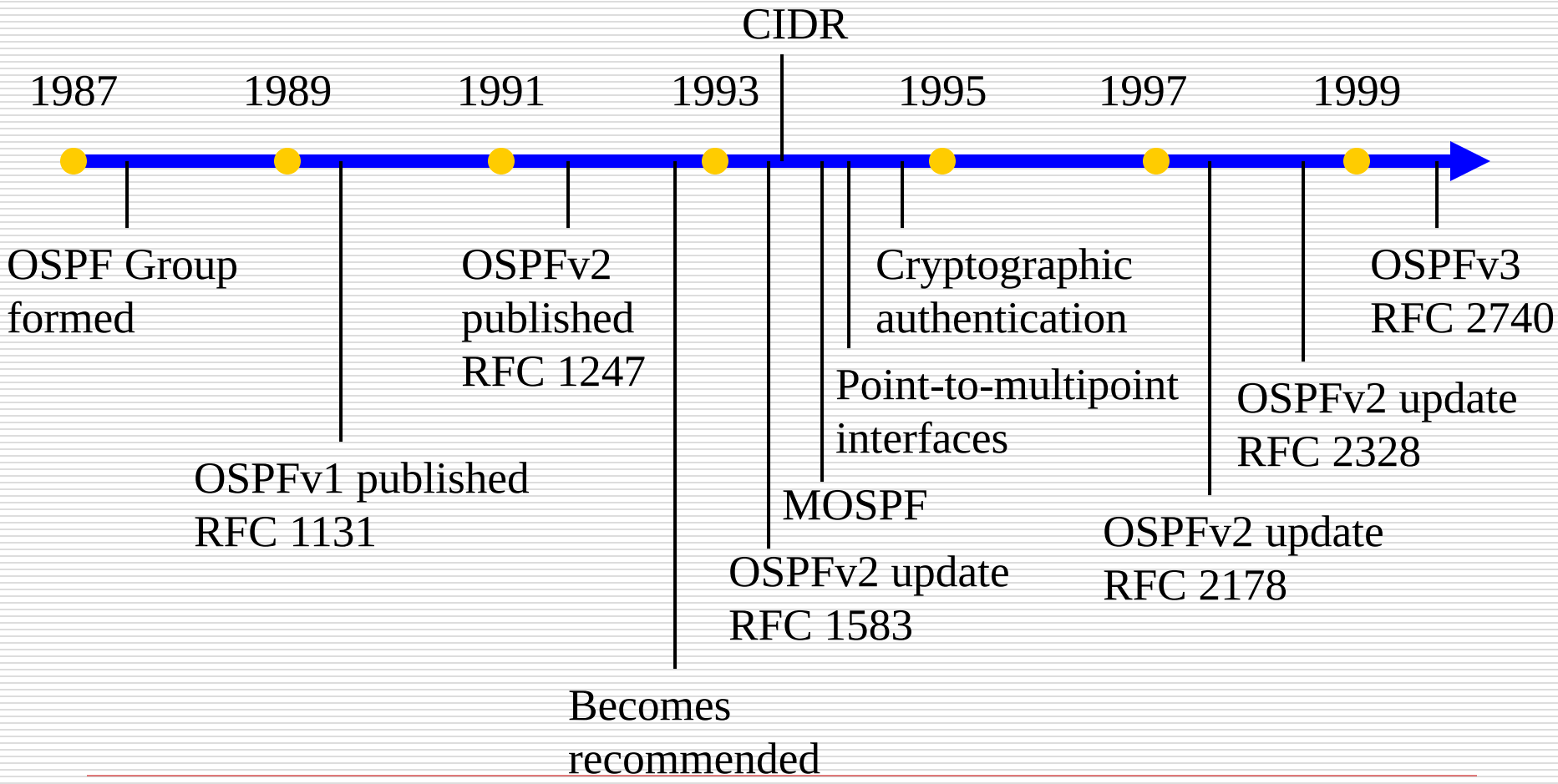
- OSPF是一种基于开放标准的链路状态路由协议，是目前IGP中应用最广、性能最优的一个协议
- OSPF可以在大型网络中使用
- 无路由自环
- OSPF支持VLSM
- 使用带宽作为度量值（$10^8$/BW）
- 收敛速度快
- 通过分区实现高效的网络管理
- 。。。。。

# 单域OSPF的基本概念

- 必须划分区域
  - 为什么?

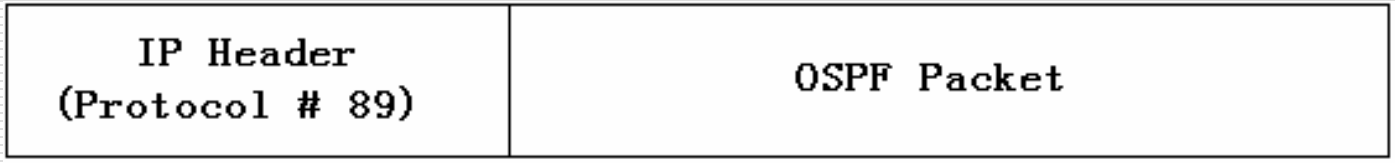
- **Area 0（区域0），骨干区域（ Backbone area ）**
  - 所有子区域必须连接到区域 0上

# OSPF的发展历程


CIDR
1987
1989
1991
1993
1995
1997
1999
OSPF Group formed
OSPFv1 published RFC 1131
OSPFv2 published RFC 1247
Becomes recommended
OSPFv2 update RFC 1583
MOSPF
Point-to-multipoint interfaces
Cryptographic authentication
OSPFv2 update RFC 2178
OSPFv2 update RFC 2328
OSPFv3 RFC 2740

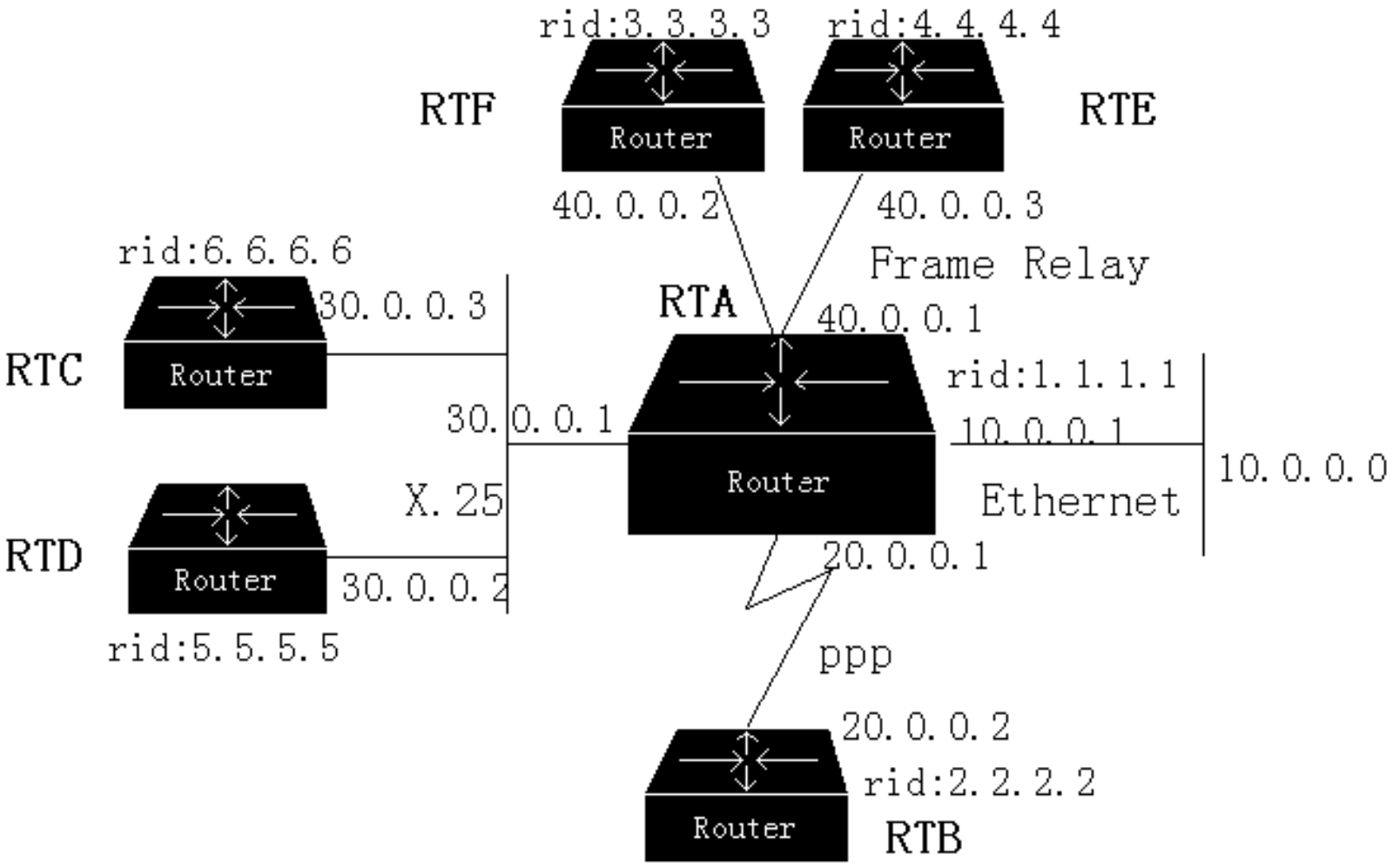
# 单区域OSPF的一些基本情况

- **RouterID：一个32位的无符号整数，是一台路由器的唯一标识，在整个自治系统内唯一**
- **协议号：IP头中代表OSPF报文的协议号是89**

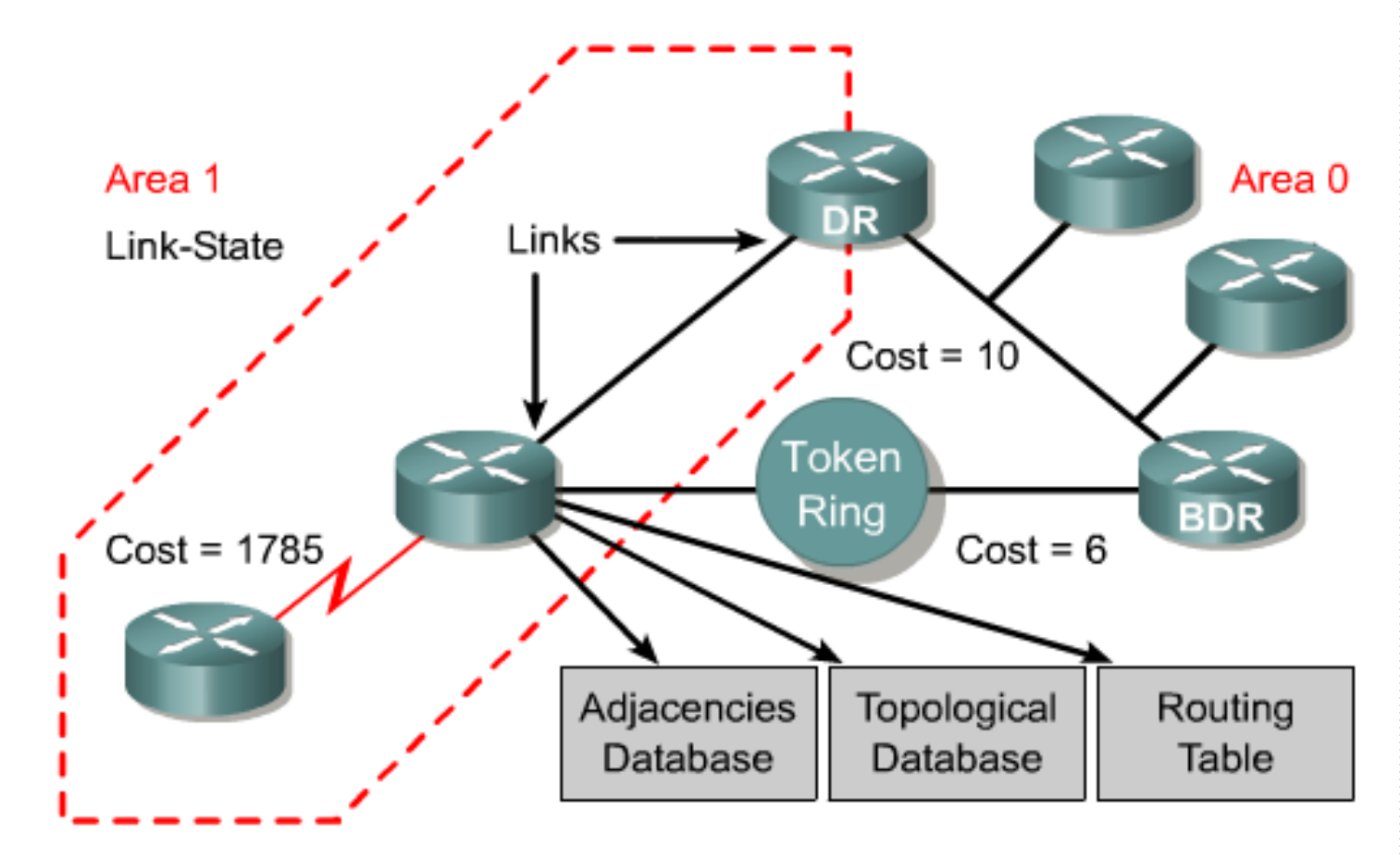



- **TTL=1：通常OSPF报文不转发，只被传递一条，即在IP报头的TTL值被设为1，但虚联接除外**

# OSPF的网络类型 P366



# OSPF术语

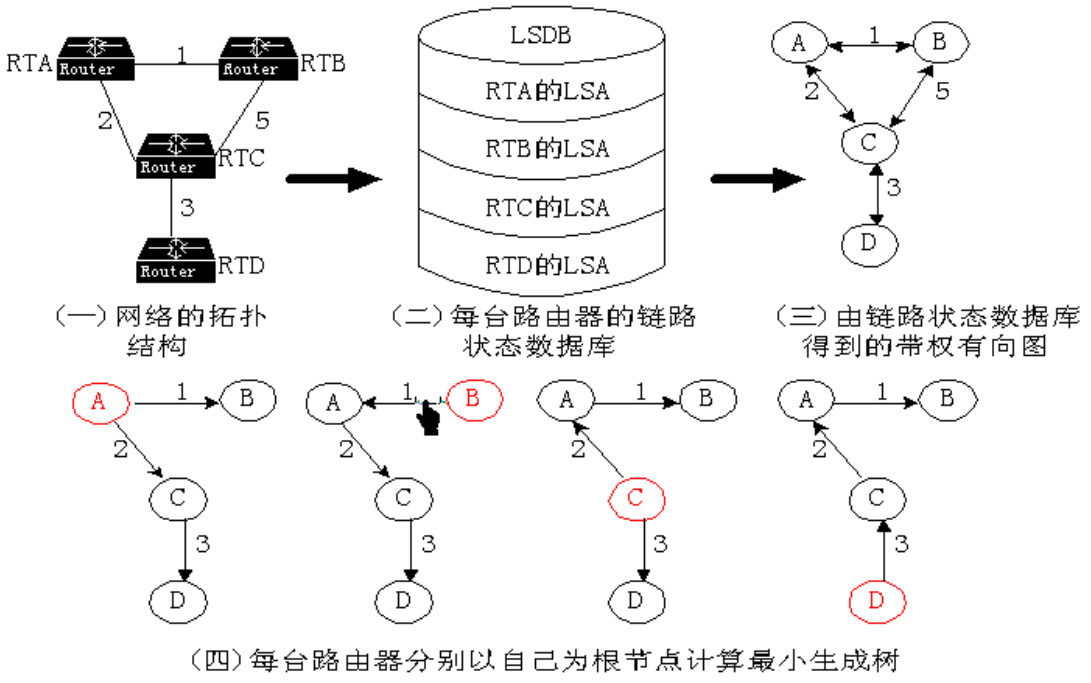

Area 1
Link-State
Links
DR
Area 0
Cost = 10
Token
Ring
BDR
Cost = 1785
Cost = 6
Adjacencies
Database
Topological
Database
Routing
Table

# OSPF分组(packet)类型 P368

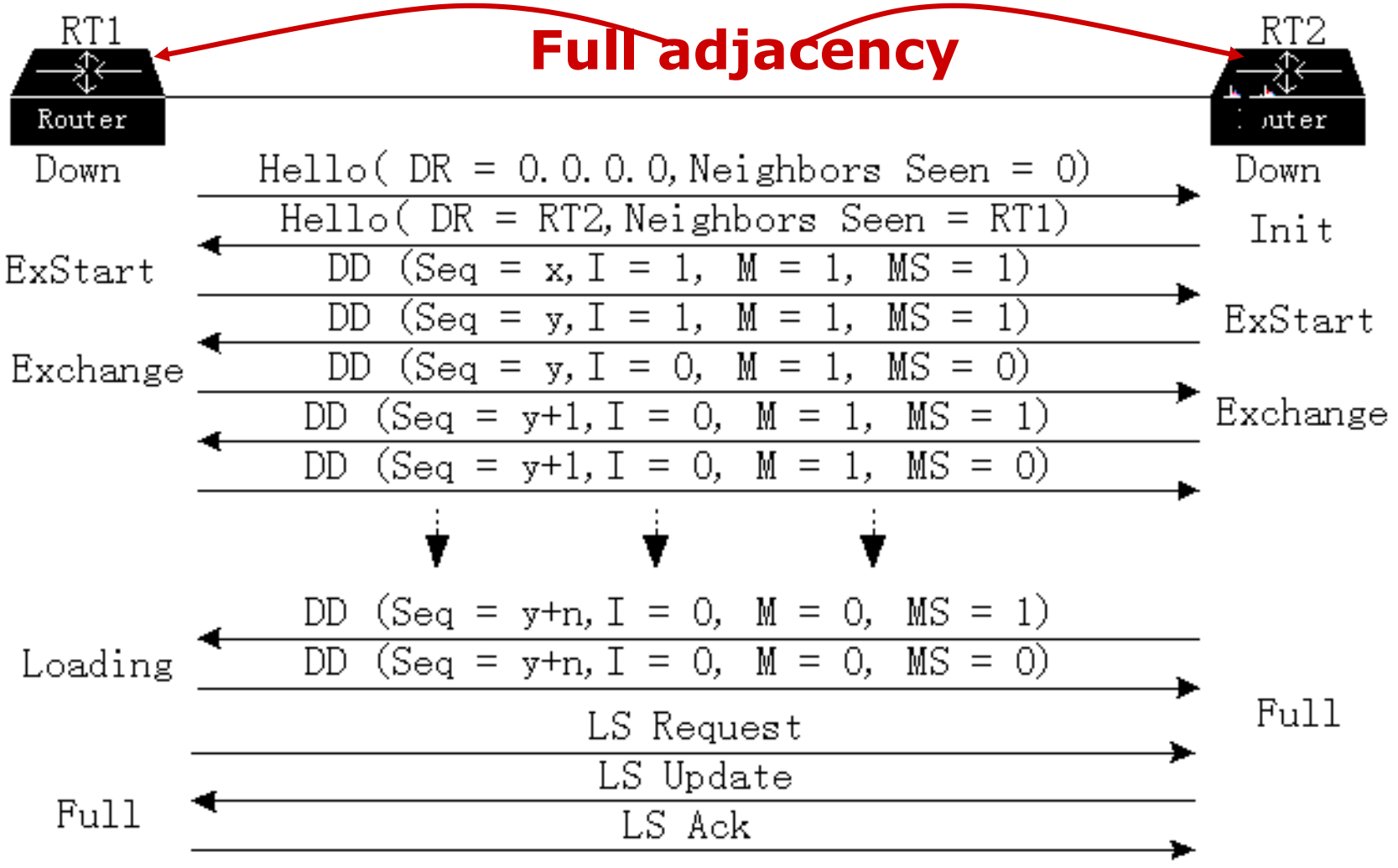
| OSPF数据包类型 | 描述 |
| --- | --- |
| Type 1－Hello | 与邻居建立和维护毗邻关系。 |
| Type 2－数据库描述包（DD） | 描述一个OSPF路由器的链路状态数据库内容。 |
| Type 3－链路状态请求（LSR） | 请求相邻路由器发送其链路状态数据库中的具体条目 |
| Type 4－链路状态更新（LSU） | 向邻居路由器发送链路状态通告 |
| Type 5－链路状态确认（LSA） | 确认收到了邻居路由器的LSU |

# OSPF的运行步骤

- 建立路由器毗邻关系
- 选举DR和BDR
- 发现路由
- 选择最佳路由
- 维护路由信息



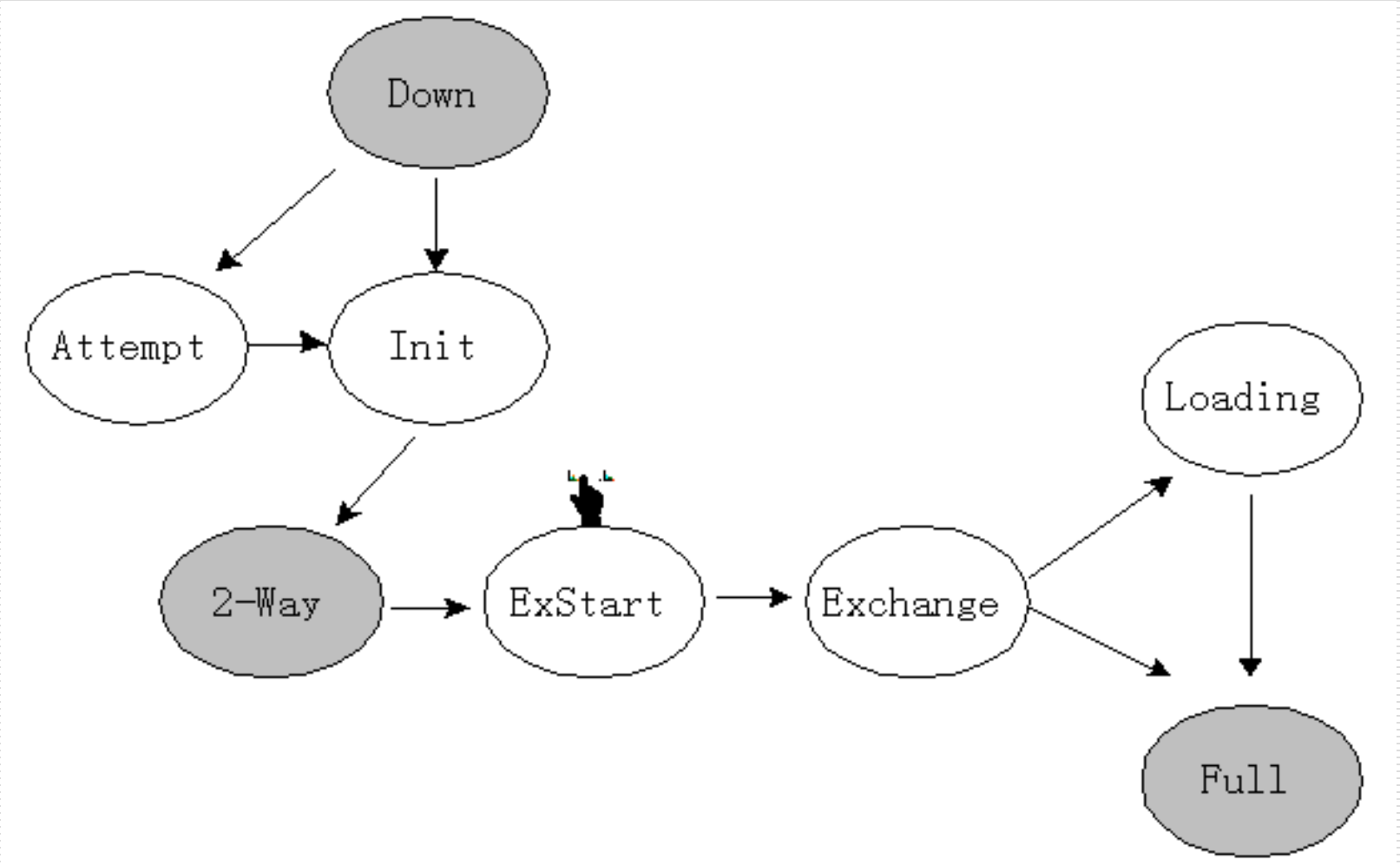
# 建立路由器毗邻关系


RT1
Full adjacency
RT2
Router
Down
Hello( DR = 0.0.0.0, Neighbors Seen = 0)
Down
Hello( DR = RT2, Neighbors Seen = RT1)
Init
ExStart
DD (Seq = x, I = 1, M = 1, MS = 1)
DD (Seq = y, I = 1, M = 1, MS = 1)
ExStart
Exchange
DD (Seq = y, I = 0, M = 1, MS = 0)
DD (Seq = y+1, I = 0, M = 1, MS = 1)
Exchange
DD (Seq = y+1, I = 0, M = 1, MS = 0)
DD (Seq = y+n, I = 0, M = 0, MS = 1)
Loading
DD (Seq = y+n, I = 0, M = 0, MS = 0)
Full
LS Request
LS Update
Full
LS Ack

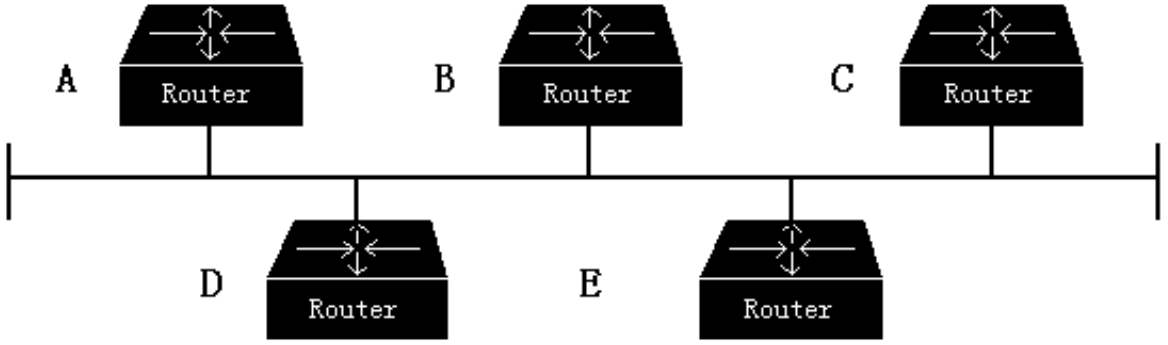
# OSPF 状态

- Down
- Init（初始）
- Two-way（双向）
- ExStart（准启动）
- Exchange（交换）
- Loading（加载）
- Full adjacency（全毗邻）

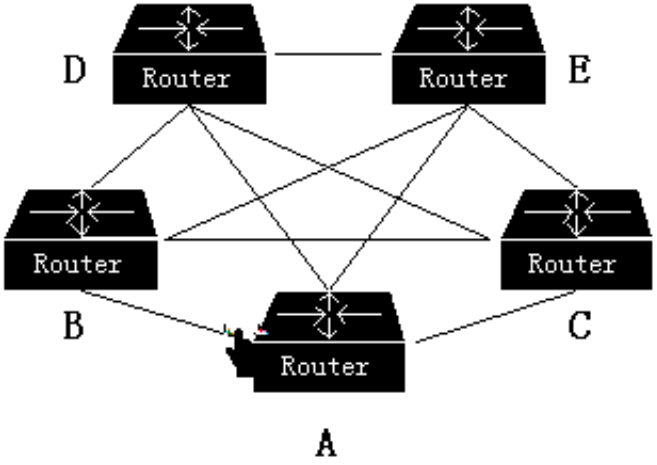
# 运行OSPF的路由器状态图



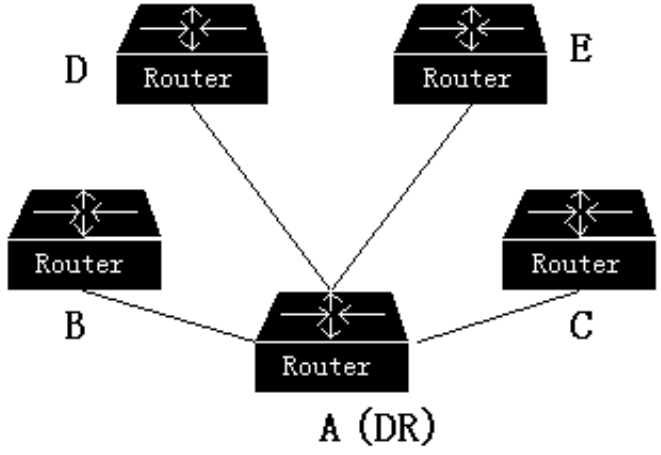
# 为什么要选举DR和BDR？




图一、网络的拓扑结构




图二、没有选举DR时的邻接关系




图三、选举DR后时的邻接关系

# DR（村长）选举过程

- 登记选民

  本网段内的OSPF路由器;

  本村内的18岁以上公民。

- 登记候选人

  本网段内的priority>0的OSPF路由器;

  本村内的30岁以上公民，且在本村居住3年以上。

- 竞选演说

  所有的priority>0的OSPF路由器都认为自己是DR。

  所有的候选人都自认为应该当村长;

- 投票

  选priority值最大的，若priority值相等，选Router ID最大的;

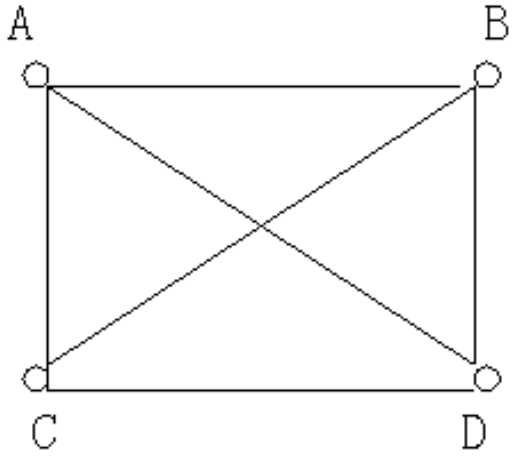
  选年纪最大，若年龄相等，按姓氏笔划排序;

# DR选举中的指导思想

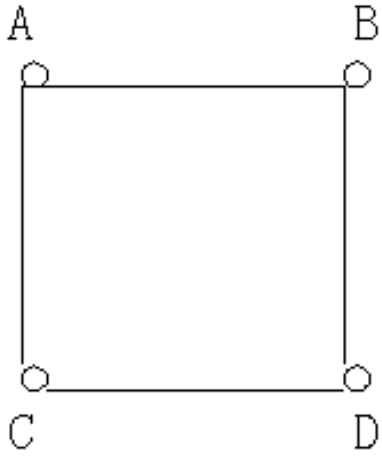
- 选举制
  - DR是路由器选出来的，而非人工指定的
- 终身制
  - DR一旦当选，除非路由器故障，否则不会更换
- 世袭制
  - DR选出的同时，也选出BDR，DR故障后，由BDR接替DR成为新的DR

# DR可能带来的问题

- 非全连通网络（full mesh），如PTMP网络




NBMA-任意两点都直接可达




PTMP--不满足任意两点都直接可达，AD,BC不能直接可达

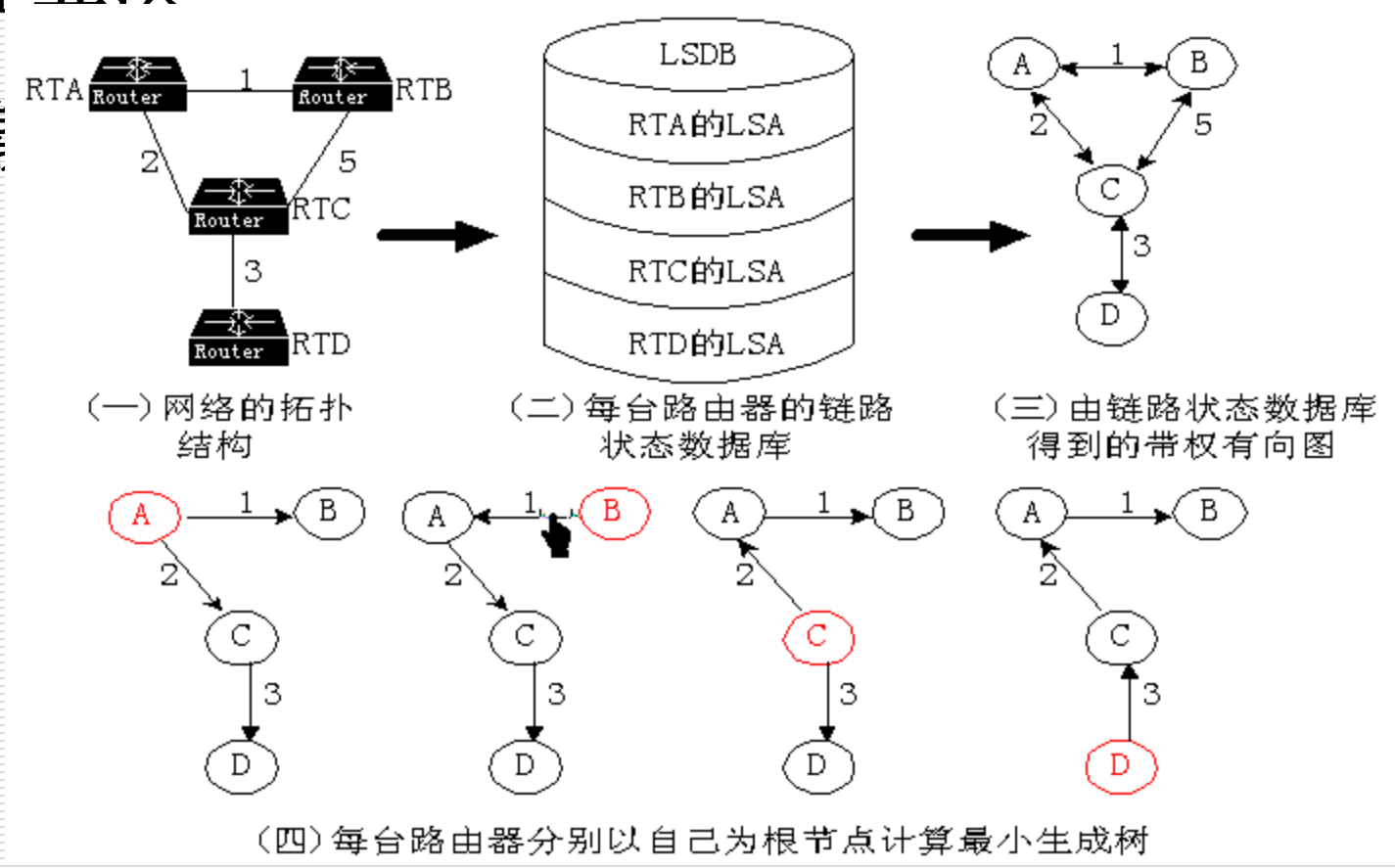
- 由管理员配置成PTMP，不选举DR

# DR带来的变化

- 同步的次数减少了（O（n）），减少了带宽的利用
- 路由器的角色：DR、BDR、DROther
- 路由器间的关系：Unknown、Neighbor、Adjacent

# 选择最佳路由

- SPF算法
- 负载




（一）网络的拓扑结构　（二）每台路由器的链路状态数据库　（三）由链路状态数据库得到的带权有向图

（四）每台路由器分别以自己为根节点计算最小生成树

# 维护路由信息

- 触发更新，LSU
- Hello分组发送的时间间隔：缺省10秒
- Hello分组的失效间隔：缺省40秒
- 即使没有拓扑变化，LSA在条目过期（缺省30分钟）后，发送LSU，通告链路存活

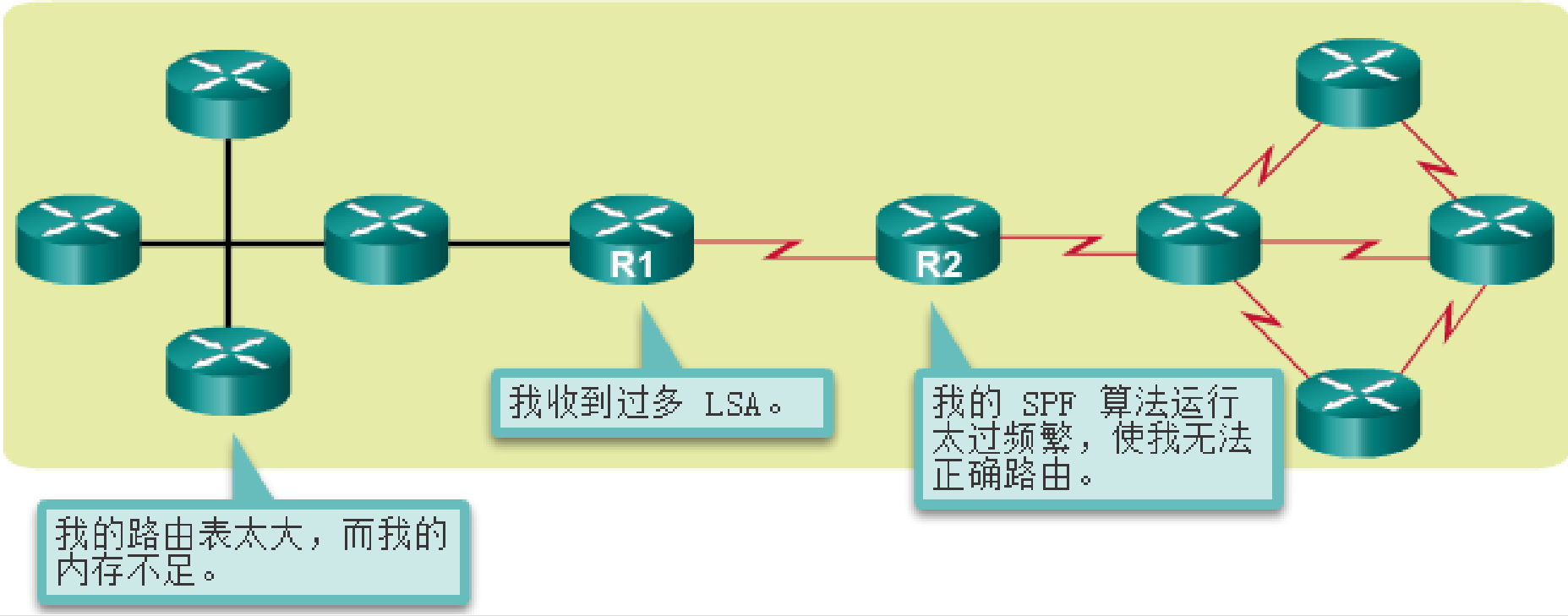
# 为什么说OSPF克服了了路由自环？

- 每一条LSA都标记了生成者（用生成该LSA的路由器的RouterID标记），其他路由器只负责传输，这样不会在传输的过程中发生对该信息的改变和错误理解。
- 路由计算的算法是SPF，计算的结果是一棵树，路由是树上的叶子节点，从根节点到叶子节点是单向不可回复的路径。
- 区域之间通过规定骨干区域避免

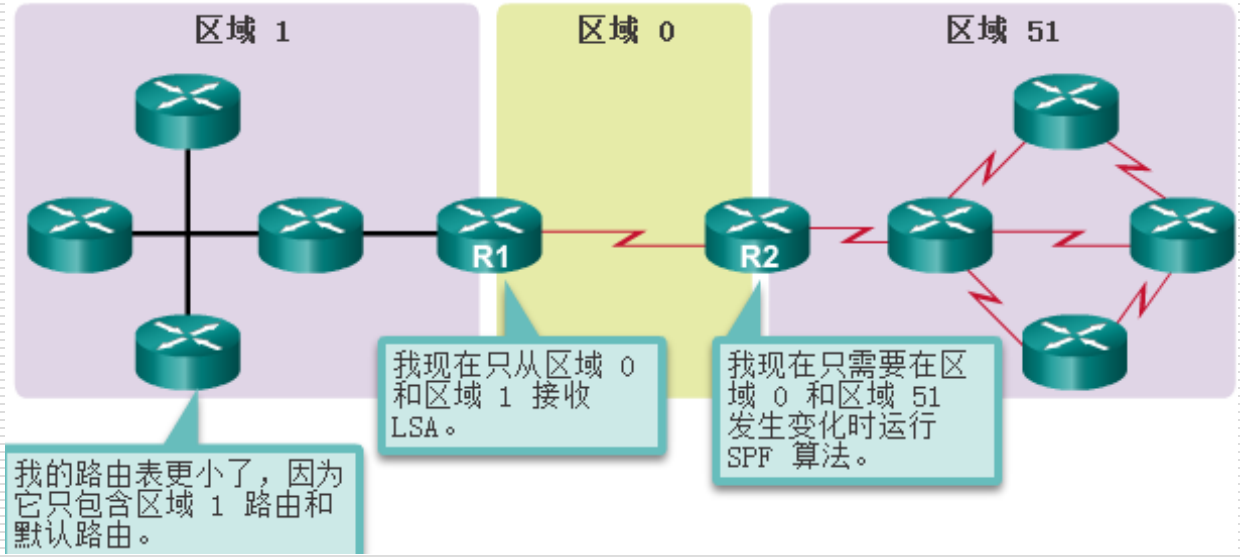
# OSPF在大型网络中可能遇到的问题

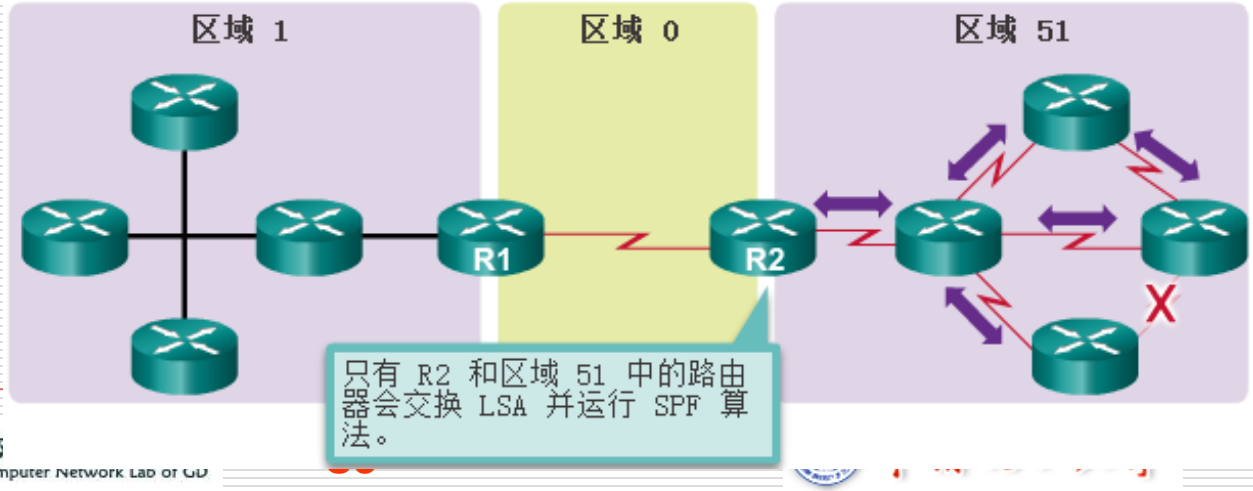
- LSDB非常庞大，占用大量存储空间
- 计算最小生成树耗时增加，CPU负担很重
  - 一点变化都会引发从头重新计算
- 网络拓扑结构经常发生变化，网络经常处于“动荡”之中
  - 接口up down
  - 路由器的增加删除
  - 好比湖水，一个小小的石子都会引发阵阵涟漪

# OSPF在大型网络中可能遇到的问题续

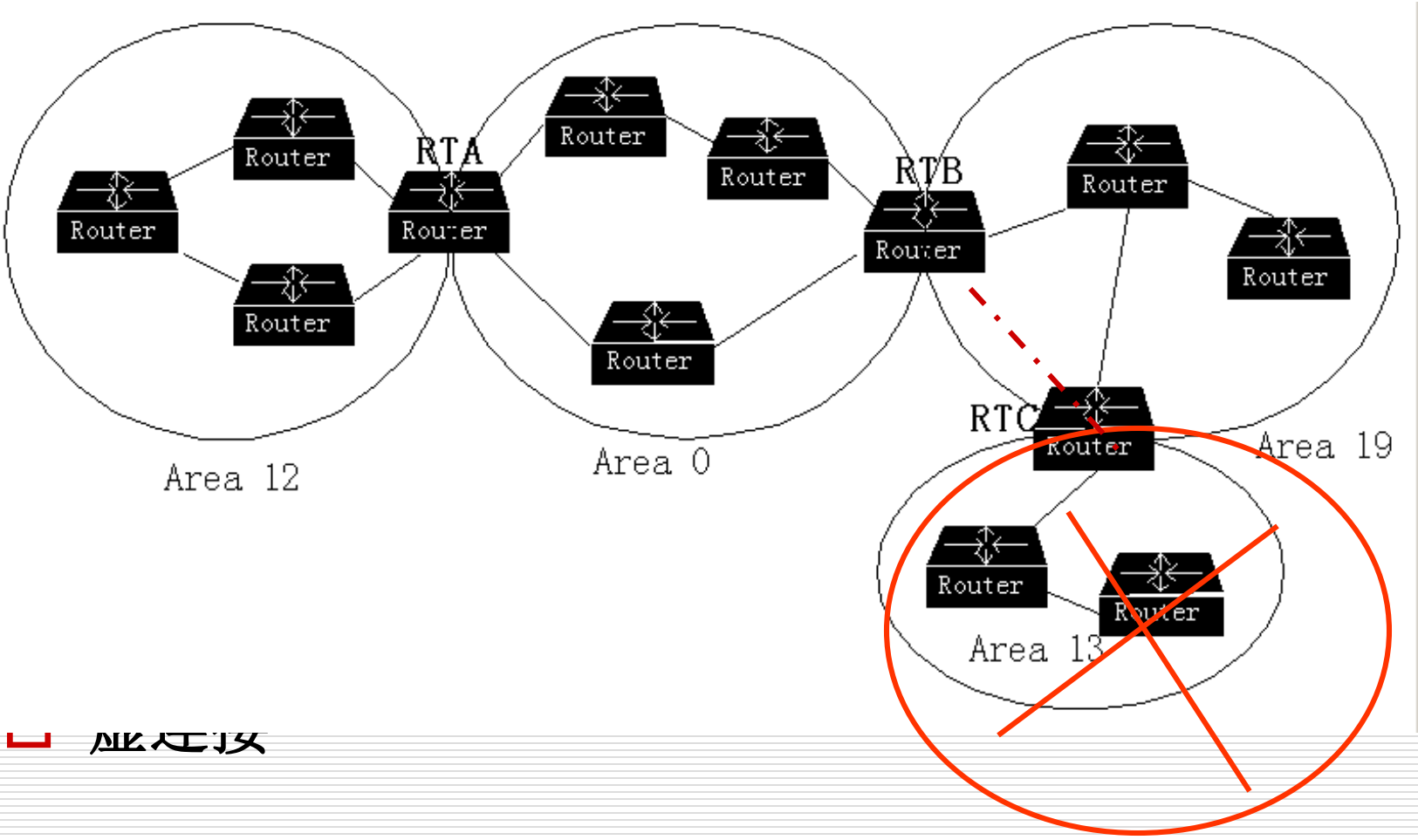


# 分而治之，解决之

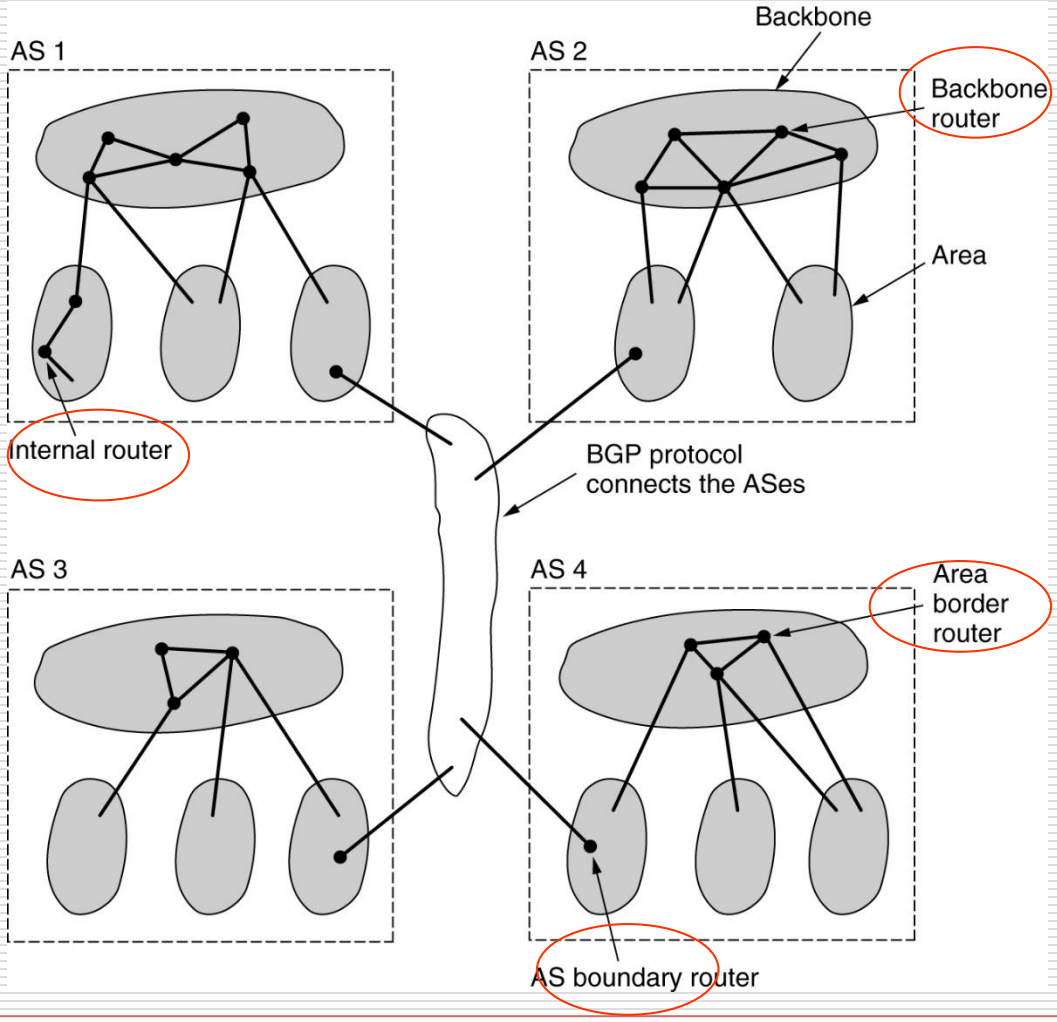

区域 1
区域 0
区域 51
R1
R2
我现在只从区域 0 和区域 1 接收 LSA。
我现在只需要在区域 0 和区域 51 发生变化时运行 SPF 算法。
我的路由表更小了，因为它只包含区域 1 路由和默认路由。


区域 1
区域 0
区域 51
R1
R2
X
只有 R2 和区域 51 中的路由器会交换 LSA 并运行 SPF 算法。

# 如何避免路由环路？


Router
RTA
Router
Router
Router
Router
Router
RTB
Router
Router
Router
RTC
Router
Router
Router
Area 12
Area 0
Area 19
Area 13

# OSPF路由器种类 （1/2）P366



# OSPF路由器种类（2/2）

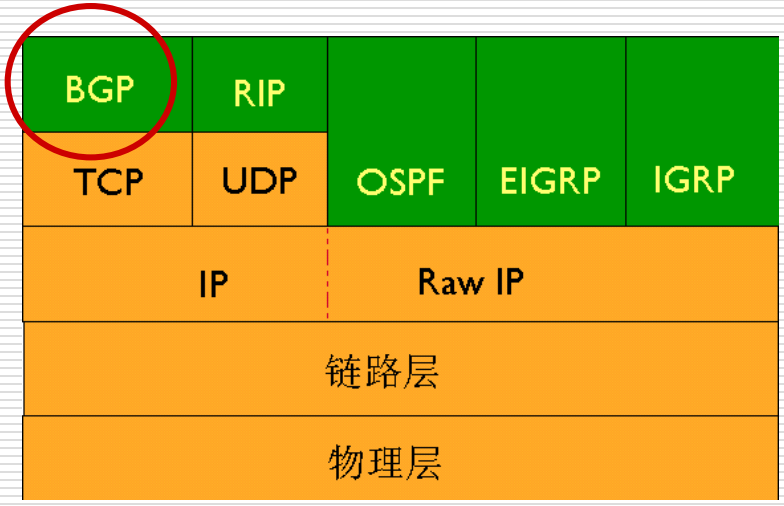
- 内部路由器 --- 路由器所有接口都在一个区
- 主干路由器 --- 所有接口都在主干区域的路由器
- 区域边界路由器(ABR) ---路由器接口分属不同区域
- 自治域边界路由器 (ASBR) --- 路由器至少有一个接口不属于本自治域/OSPF.

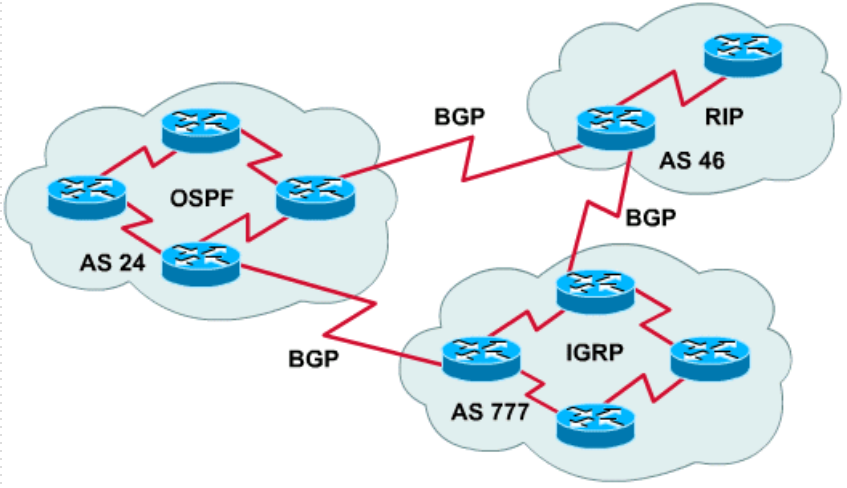
# 比较 DV 和 LS

| 距离矢量路由 | 链路状态路由 |
| --- | --- |
| 从邻居看网络 | 整个网络的拓扑 |
| 在路由器间累加距离 | 计算最短路径 |
| 频繁、周期更新：慢收敛 | 事件触发更新：快收敛 |
| 在路由器间传递路由表的拷贝 | 在路由器间传递链路状态更新 |


# BGP（border gateway protocol）（边界网关协议）P369

- 不同的协议 - **BGP** (Border Gateway Protocol)运行在AS之间
- BGP 定义在 RFCs 1771 到 1774中

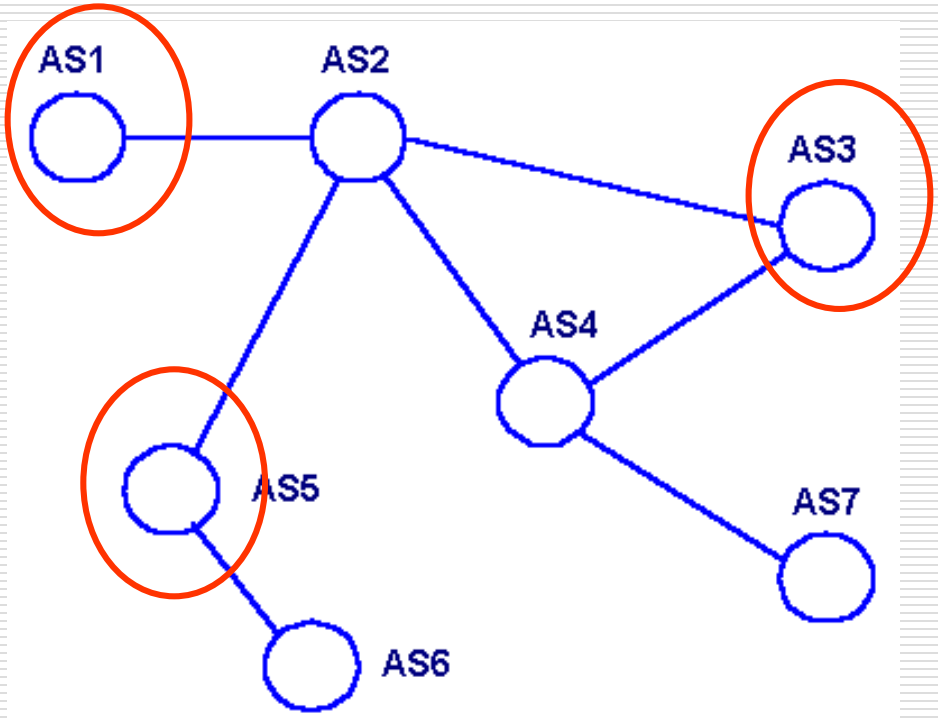


# BGP 的工作原理（1/2，P369）

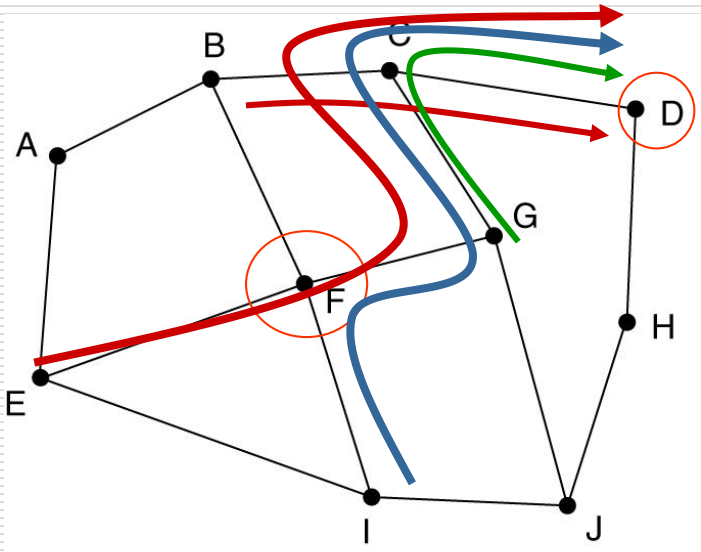
- 外部网关路由器的典型路由策略涉及政治 **political**, 安全**security**, 或经济方面 **economic** 的考虑
- 根据BGP对于中转流量的兴趣，网络被分成三类：
  - stub 自治系统
  - 多连接自治系统
  - 穿越自治系统

# 三种类型的AS网络类型图示



# BGP原理（2/2）P369

- BGP 路由器对之间通过TCP连接来相互通信
- 从根本上来说，BGP 是一个DV路由协议，但是它又不同于一般的DV协议，比如 RIP
  - BGP 路由器记录下全路径信息，而不仅仅是路径代价（ keeps track of the exact path）




(a)

Information F receives from its neighbors about D

From B: "I use BCD"
From G: "I use GCD"
From I: "I use IFGCD"
From E: "I use EFGCD"

(b)

# 本节小结

- 链路状态算法
  - 5步
  - 问题和解决的办法
- OSPF
  - 5种消息类型
  - DR 选举
  - OSPF 运作(OSPF路由器的状态变化)
- BGP

# Thank you all !


中环东路
内环东路
中环西路