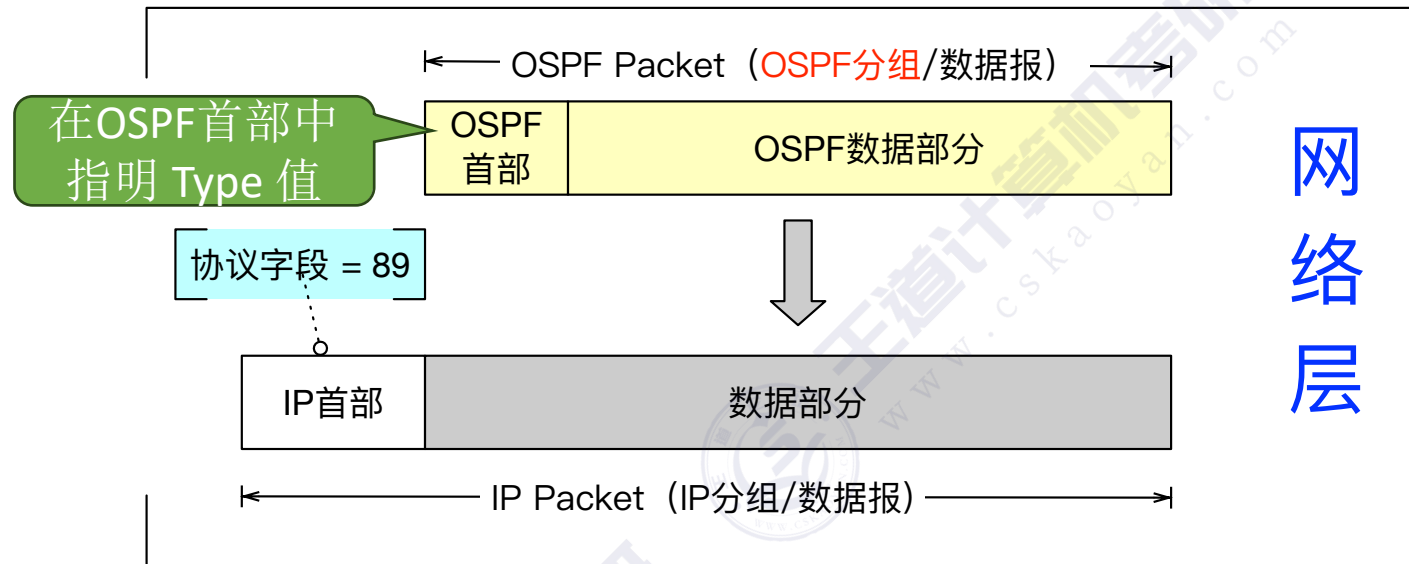


本节内容

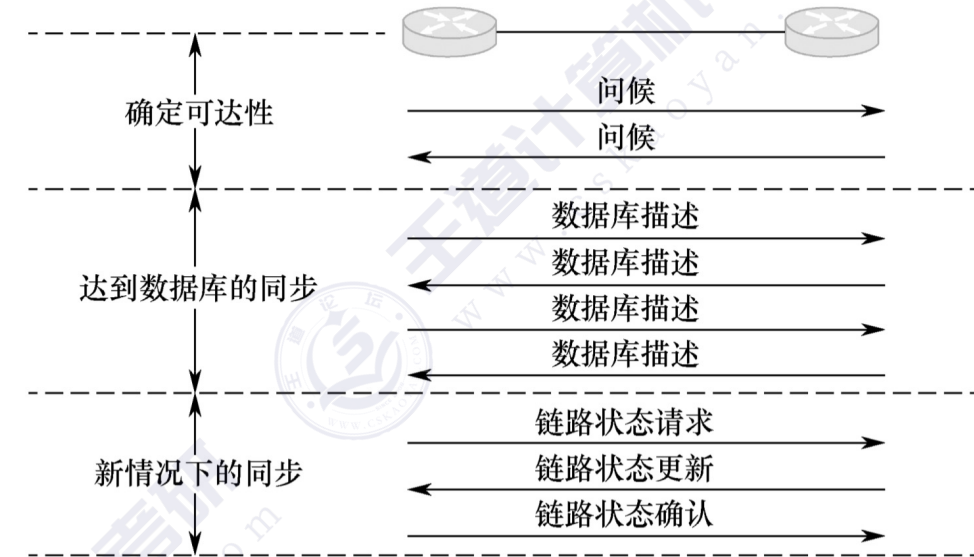
OSPF

的分组类型

OSPF 的分组类型



新建立邻居关系的两台路由器



OSPF协议定义了5种OSPF分组类型（见下表）

Type值	英文缩写	中文译名	作用
1	Hello Packet	问候分组	建立 & 维持邻居关系（每隔10秒发一次，40秒超时）
2	DD Packet	数据库描述分组	邻居建立时，向邻居给出自己的 LSDB 摘要（即LSA的头信息）
3	LSR Packet	链路状态请求分组	向邻居请求缺少的 LSA（指明头信息即可）
4	LSU Packet	链路状态更新分组	向邻居传输具体的 LSA（可能引发全网洪泛）
5	LSAck Packet	链路状态确认分组	收到邻居发来的 LSU后，向邻居确认收到了哪些 LSA（头信息）

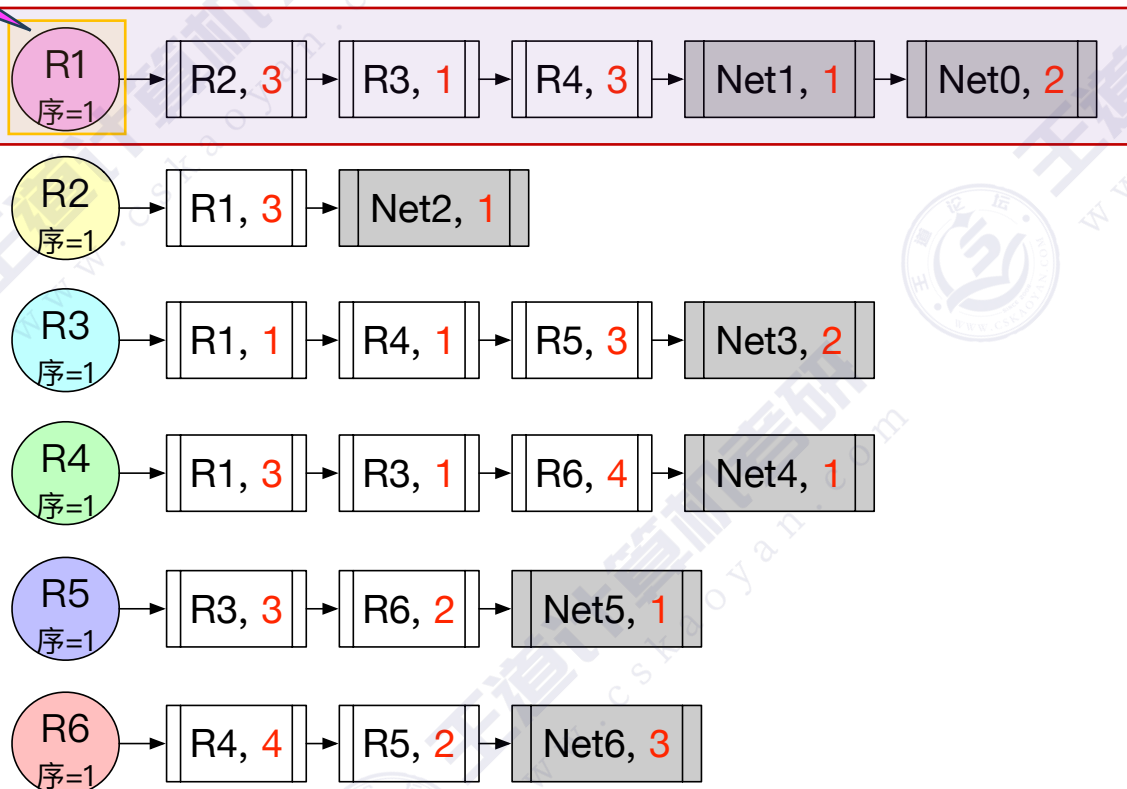
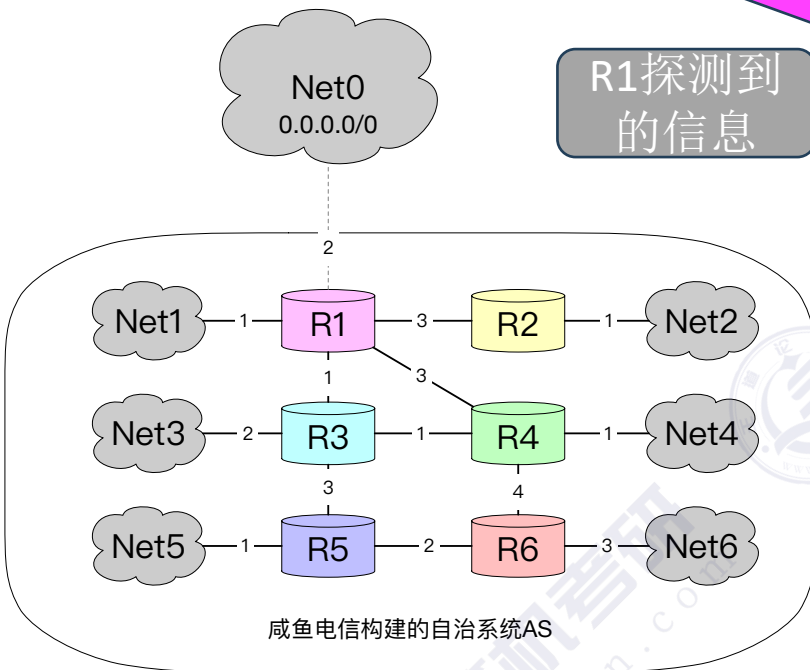
几个术语的关系：LSDB、LSA、LSA 头信息

LSA Header
(LSA 头信息)

OSPF协议中定义
的一种数据结构

R1探测到
的信息

→ LSA 链路状态通告
(Link-State Advertisement)



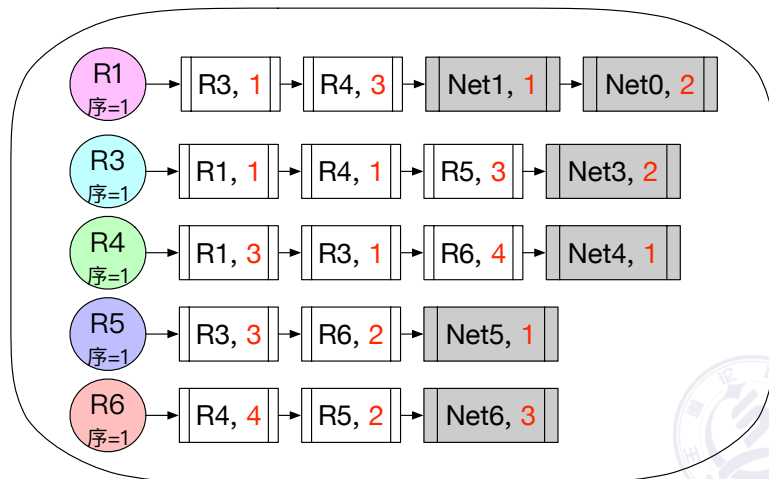
LSDB (Link-State Database) —— 链路状态数据库

每台路由器都要
建立自己的LSDB

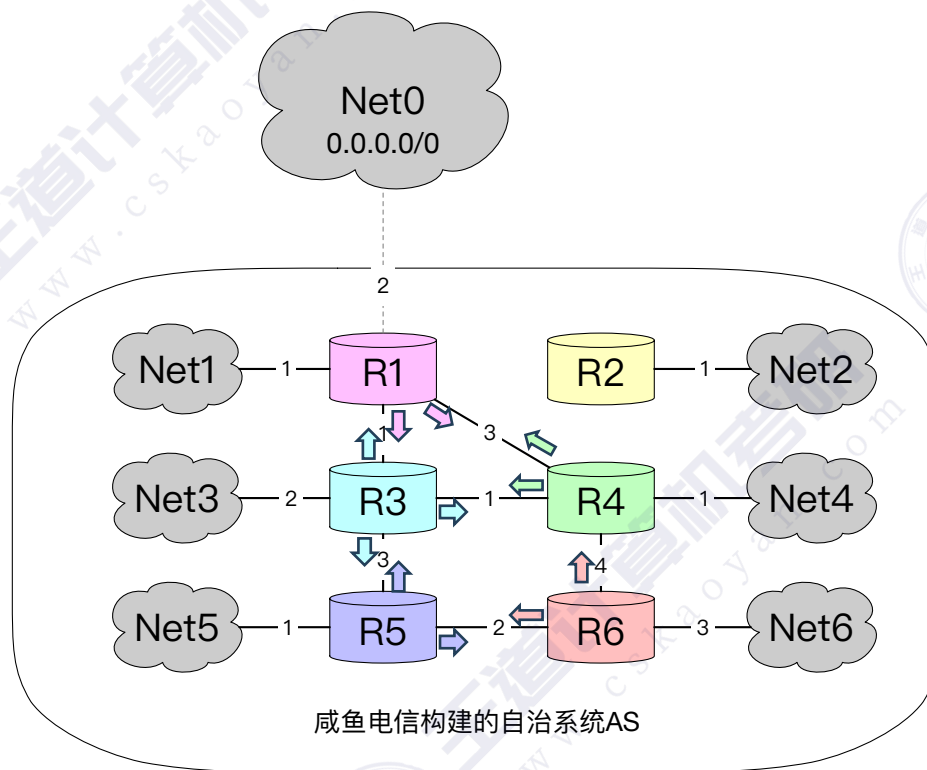
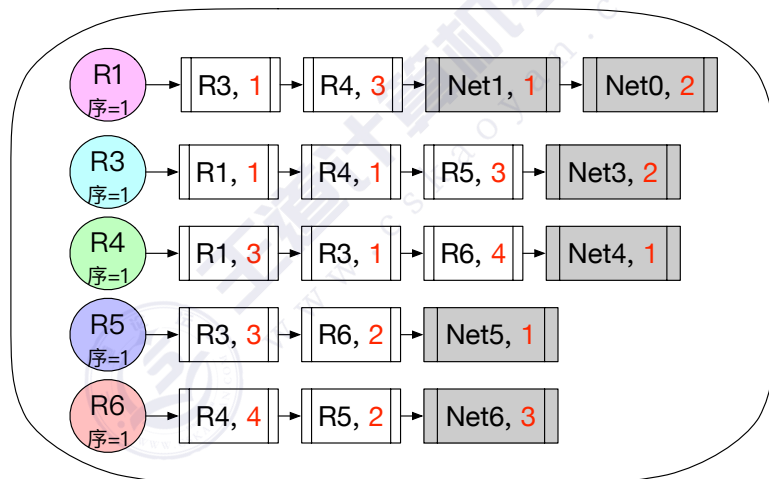
Type 1: Hello分组（问候分组）

Hello分组 = 定期签到打卡

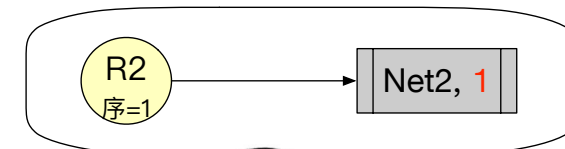
R1 的 LSDB



R3、R4、R5、R6 的 LSDB都一样



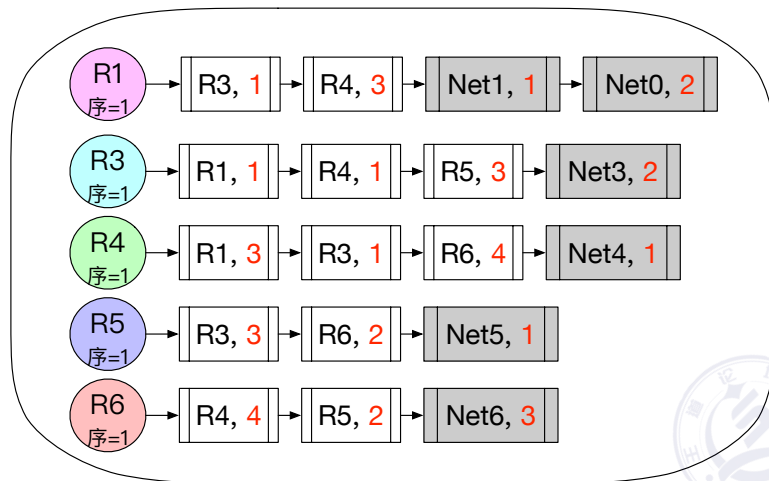
R2 的 LSDB



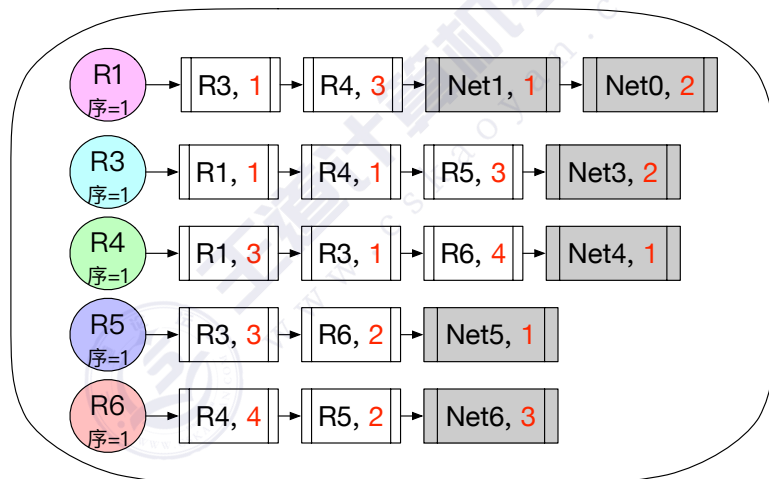
- 各台路由器每隔10秒向直接邻居发送一次“问候分组”（Hello分组）
- 如果超过40秒未收到邻居的问候，就认为该邻居不可达

Type 1: Hello分组（问候分组）

R1 的 LSDB

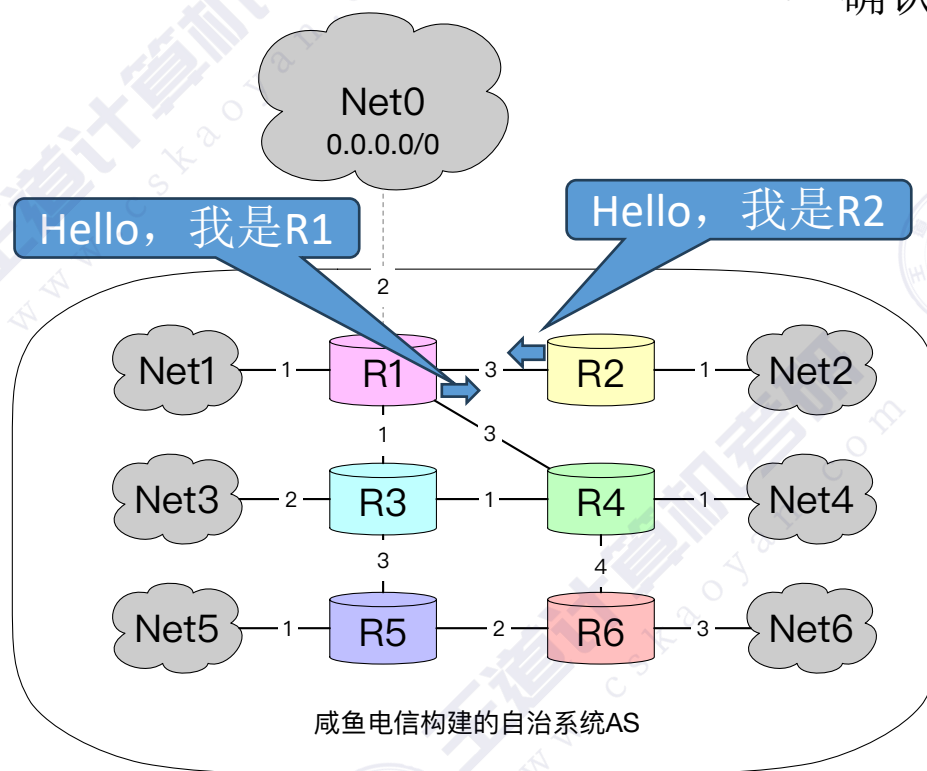


R3、R4、R5、R6 的 LSDB都一样

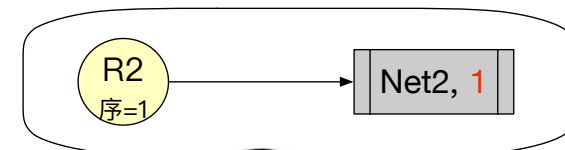


假设：某时刻 R2 与 R1 建立连接

- 向新邻居发送 Hello 分组
- 确认邻居关系



R2 的 LSDB



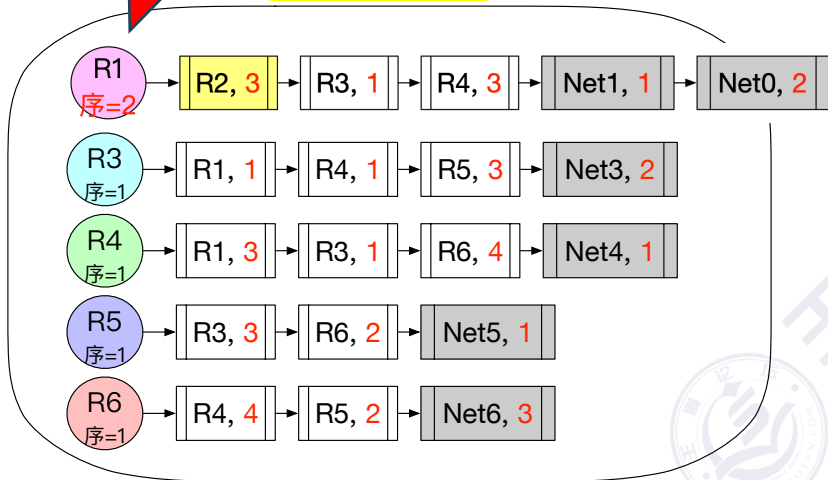
孤苦伶仃 无依无靠

- 各台路由器每隔10秒向直接邻居发送一次“问候分组”（Hello分组）
- 如果超过40秒未收到邻居的问候，就认为该邻居不可达

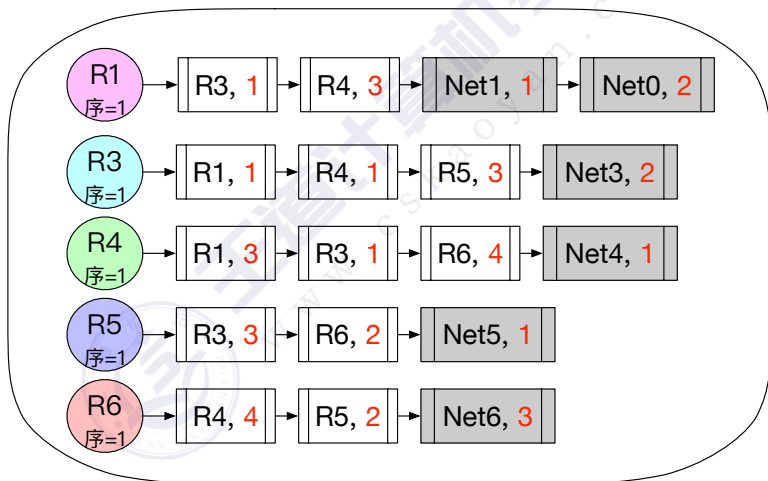
Type 1: Hello分组（问候分组）

R1的LSA变化，序号+1

R1 的 LSDB

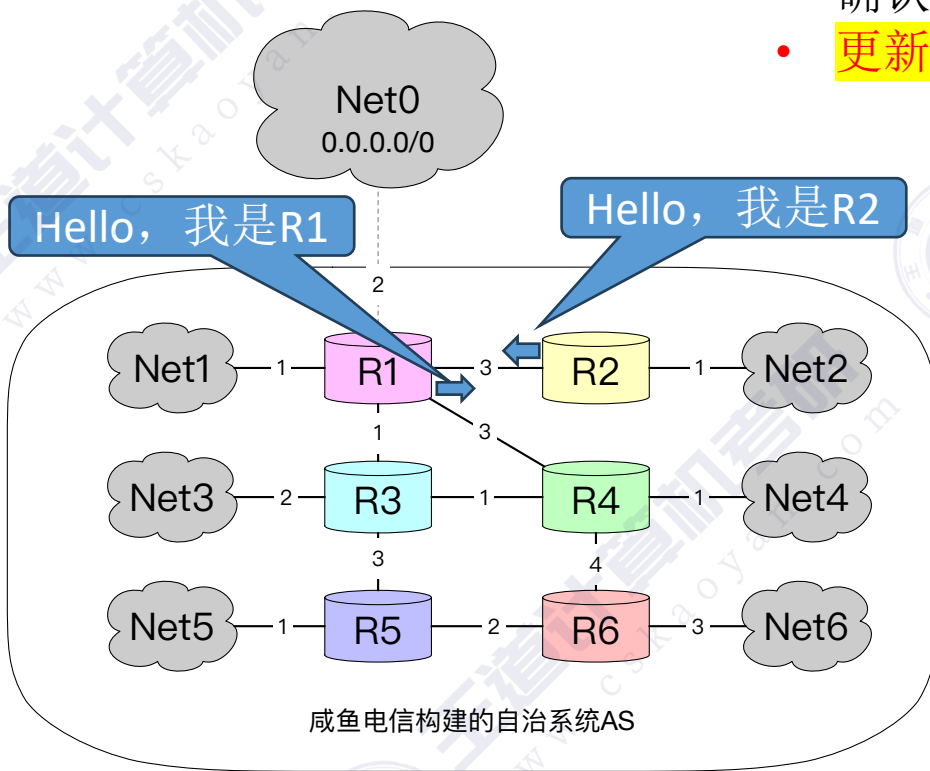


R3、R4、R5、R6 的 LSDB都一样

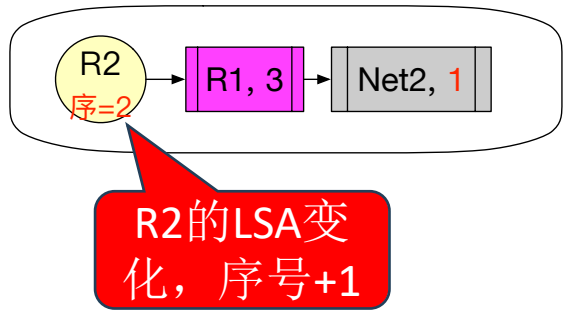


假设：某时刻 R2 与 R1 建立连接

- 向新邻居发送 Hello 分组
- 确认邻居关系
- 更新链路状态信息



R2 的 LSDB

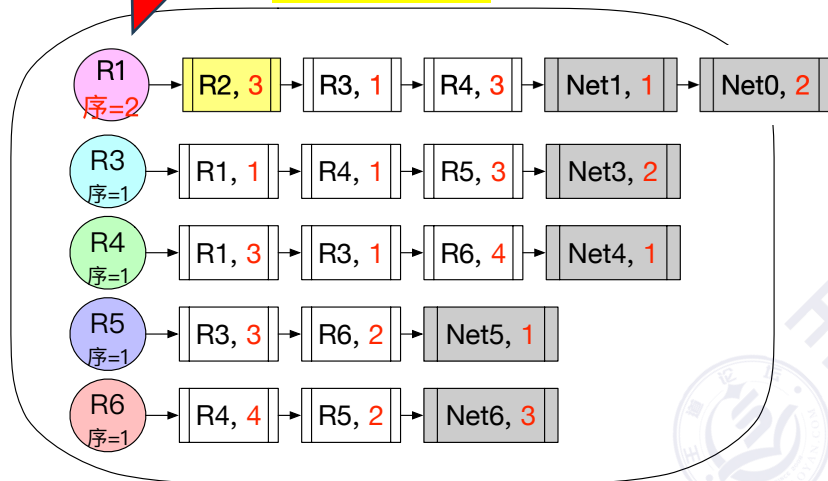


- 各台路由器每隔10秒向直接邻居发送一次“问候分组”（Hello分组）
- 如果超过40秒未收到邻居的问候，就认为该邻居不可达

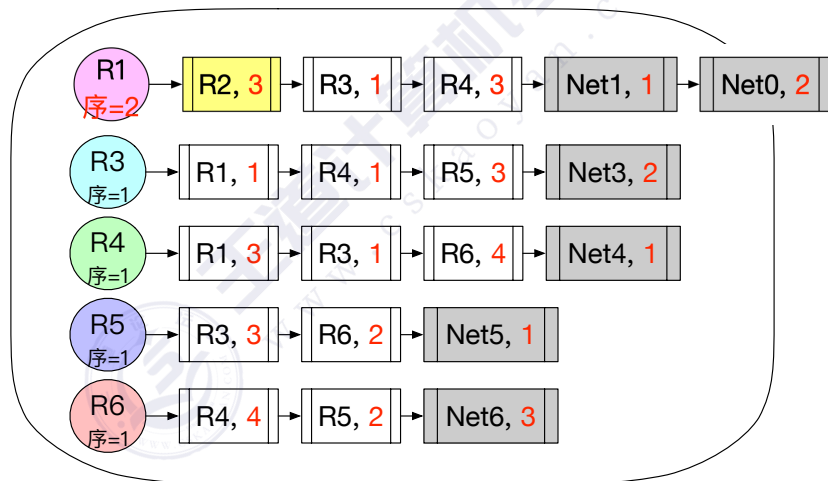
Type 1: Hello分组（问候分组）

R1的LSA变化，序号+1

R1 的 LSDB



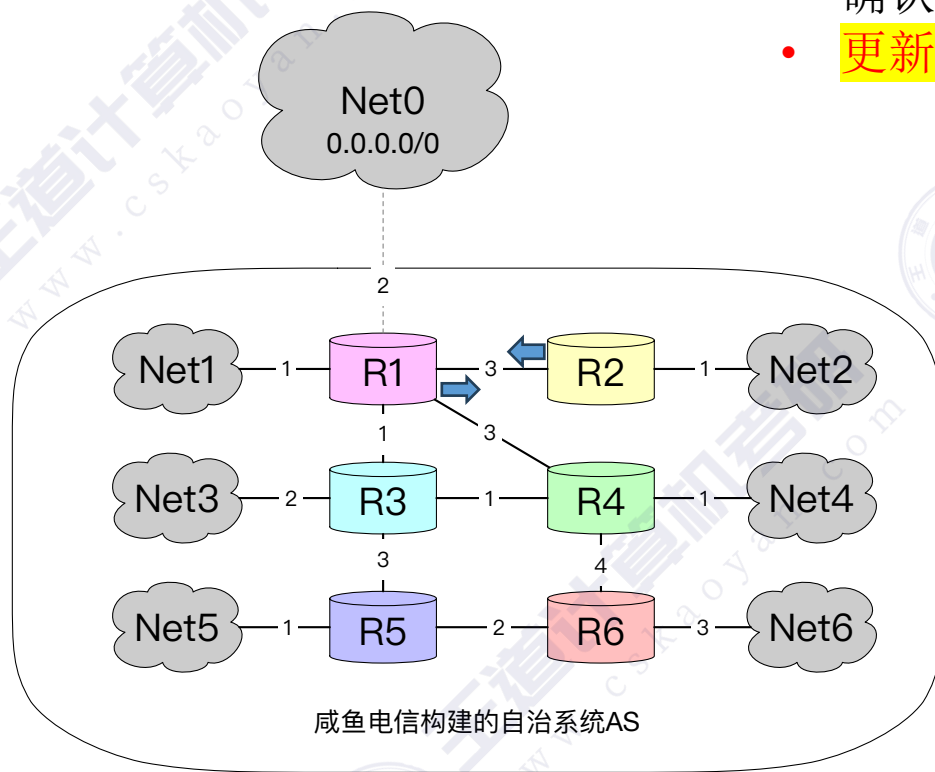
R3、R4、R5、R6 的 LSDB都一样



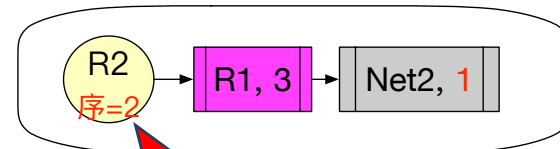
假设：某时刻 R2 与 R1 建立连接

- 向新邻居发送 Hello 分组
- 确认邻居关系
- 更新链路状态信息

注：在现实应用中，一旦链路信息发生变化就要立即洪泛通知全网，为简化理解，此处暂不展开探讨



R2 的 LSDB

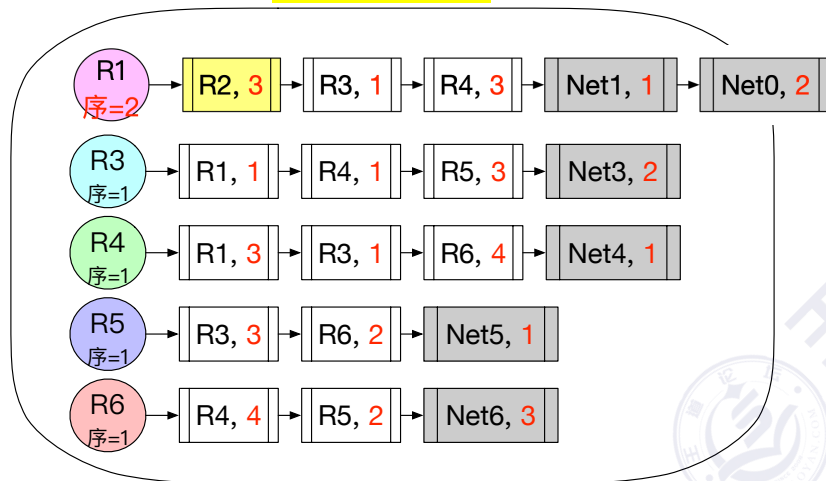


R2的LSA变化，序号+1

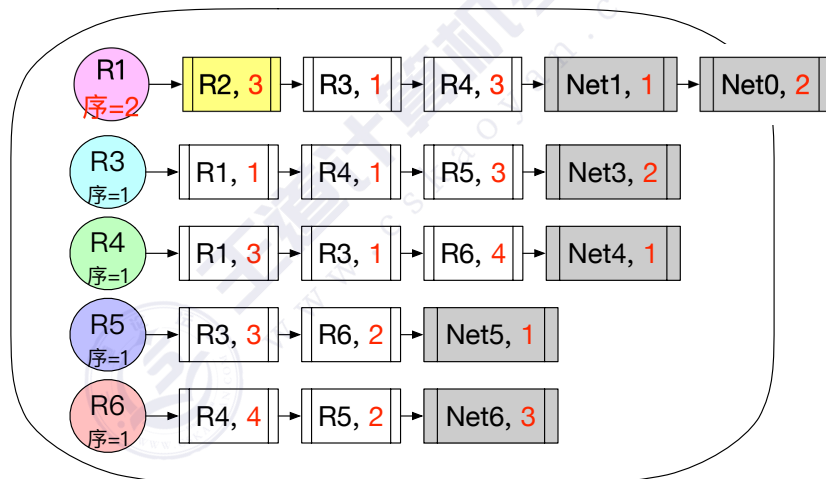
- 各台路由器每隔10秒向直接邻居发送一次“问候分组”（Hello分组）
- 如果超过40秒未收到邻居的问候，就认为该邻居不可达

Type 2: DD 分组（数据库描述分组）

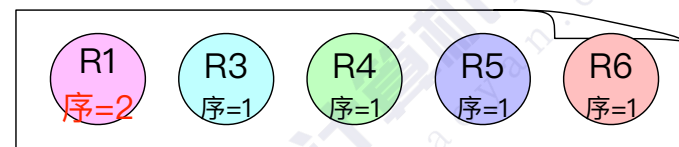
R1 的 LSDB



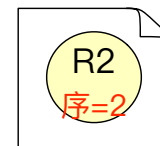
R3、R4、R5、R6 的 LSDB 都一样



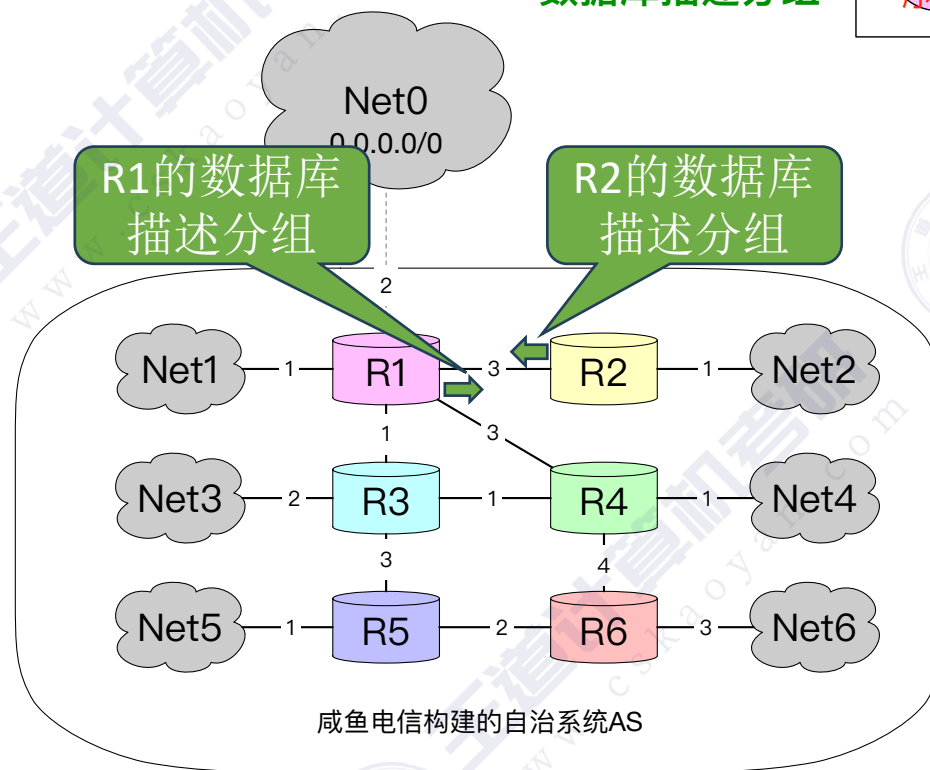
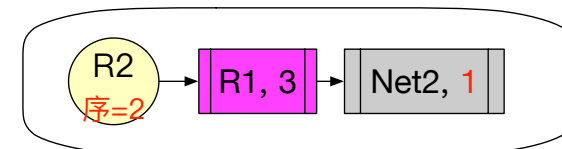
路由器R1构造的
数据库描述分组



路由器R2构造的
数据库描述分组



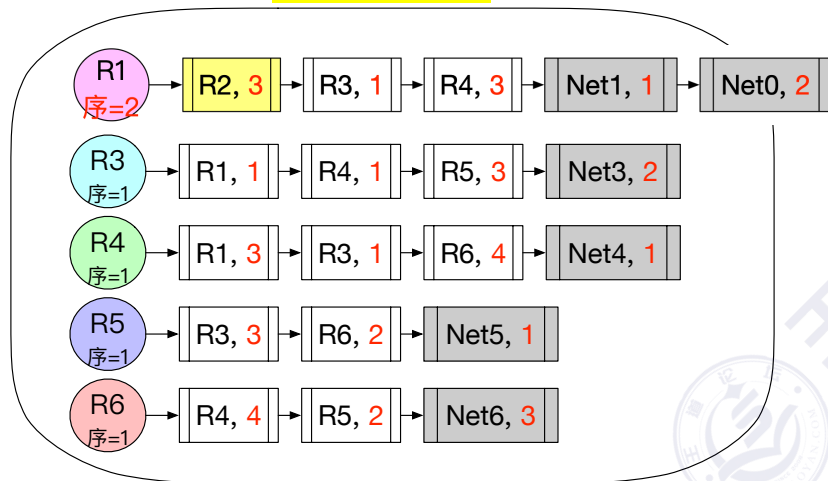
R2 的 LSDB



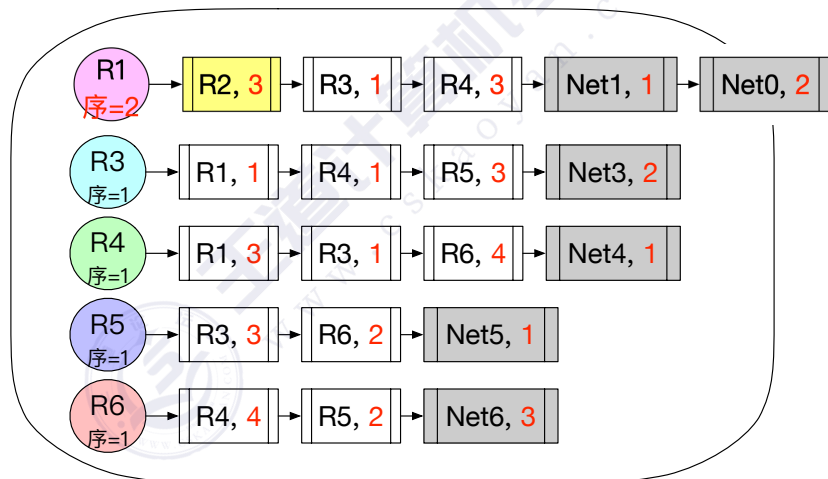
- 两台路由器的邻居关系建立时, 通过数据库描述分组, 向邻居给出自己的 **LSDB 摘要** (即LSA的头信息)
- 如果LSDB摘要信息量很大, 可以拆分成多个数据库描述分组

Type 3: LSR 分组（链路状态请求分组）

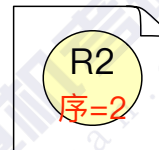
R1 的 LSDB



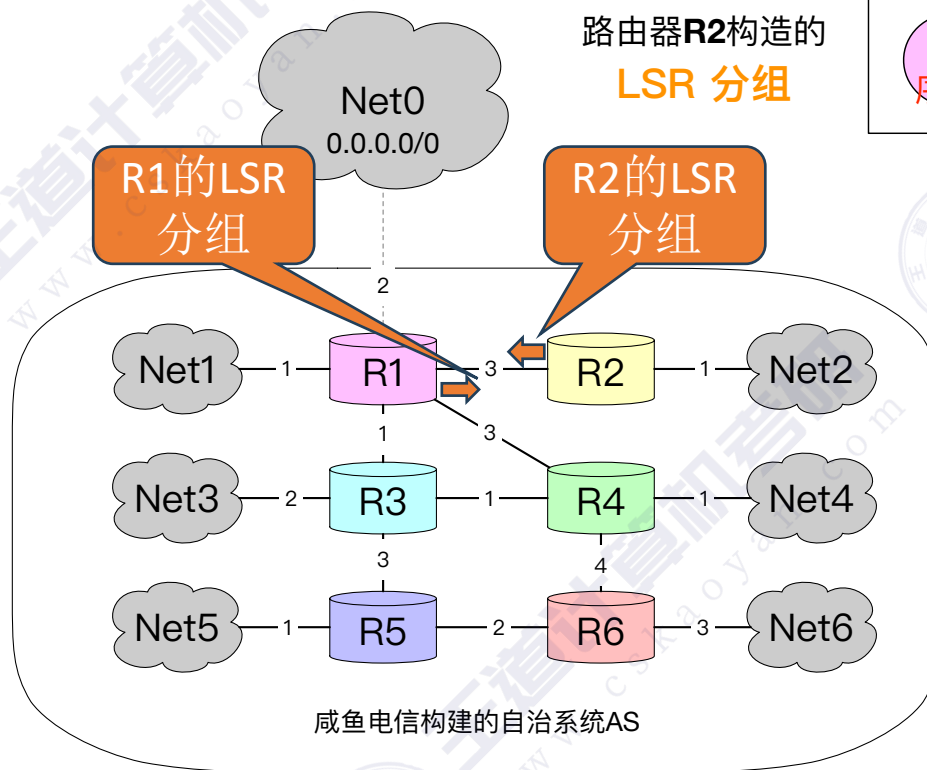
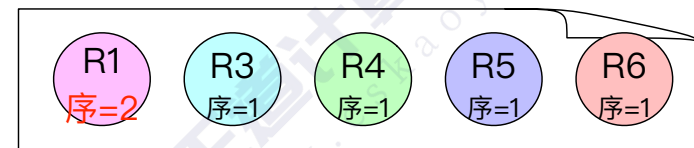
R3、R4、R5、R6 的 LSDB都一样



路由器R1构造的
LSR 分组

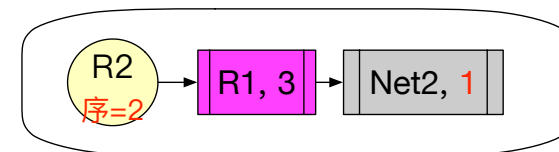


路由器R2构造的
LSR 分组



好朋友作业借我抄抄

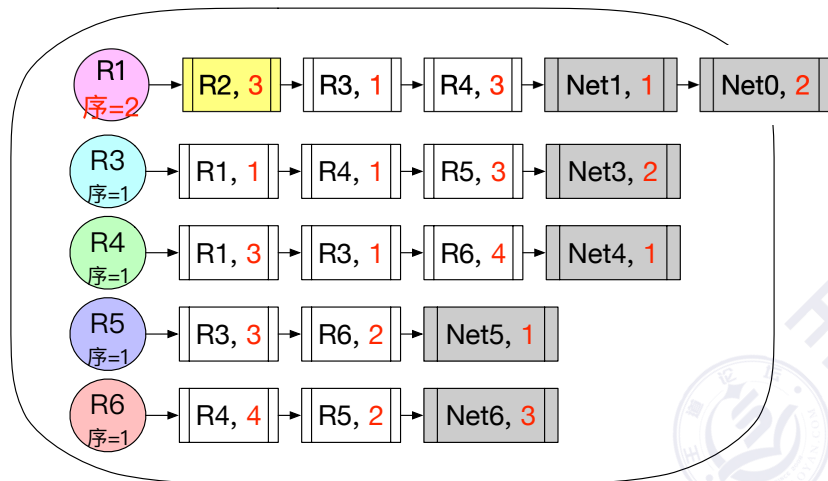
R2 的 LSDB



- 通过LSR分组，向邻居请求缺少的LSA（头信息）

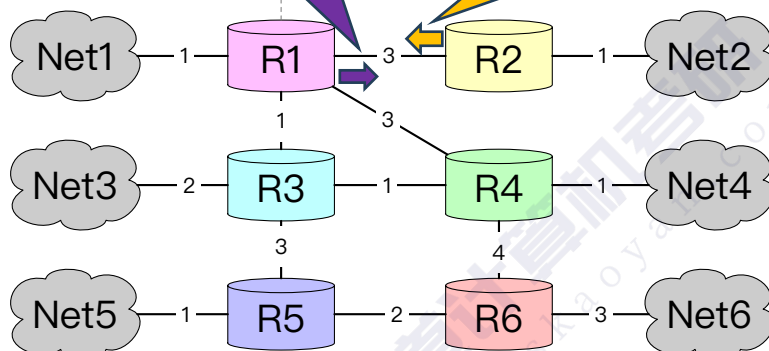
Type 4: LSU分组（链路状态更新分组）

R1 的 LSDB



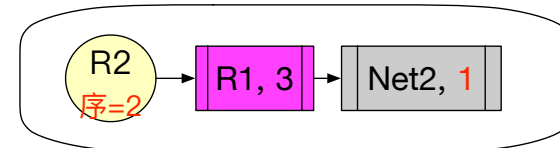
R1的LSU分组，
包含 R2 请求
的所有 LSA 完
整信息

R2的LSU分组，包
含 R1 请求的所有
LSA 完整信息

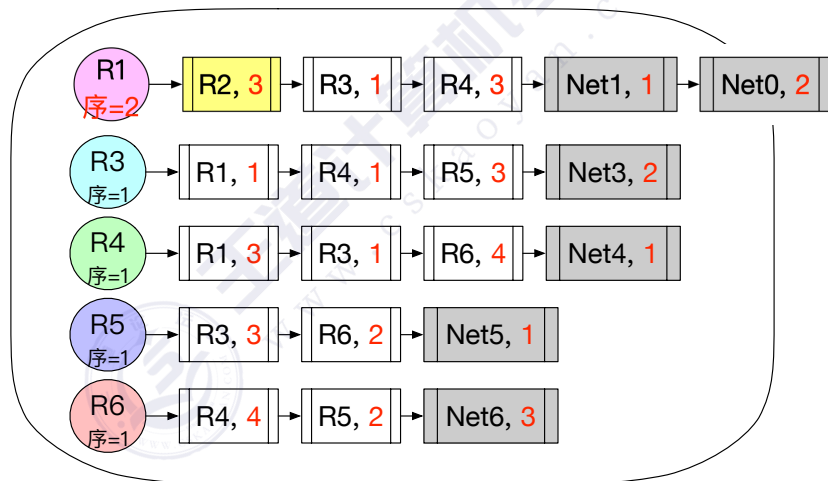


咸鱼电信构建的自治系统AS

R2 的 LSDB



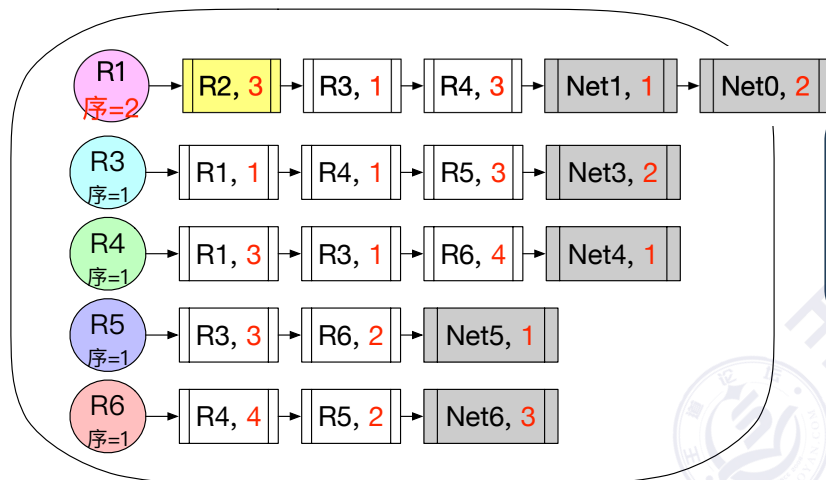
R3、R4、R5、R6 的 LSDB都一样



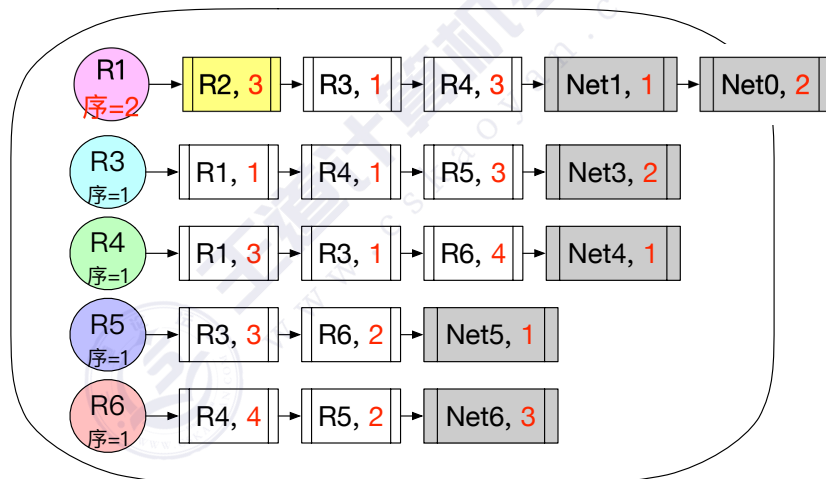
- 路由器通过 LSU 分组，向邻居传输具体的 LSA（可能引发全网洪泛）

Type 4: LSU分组（链路状态更新分组）

R1 的 LSDB

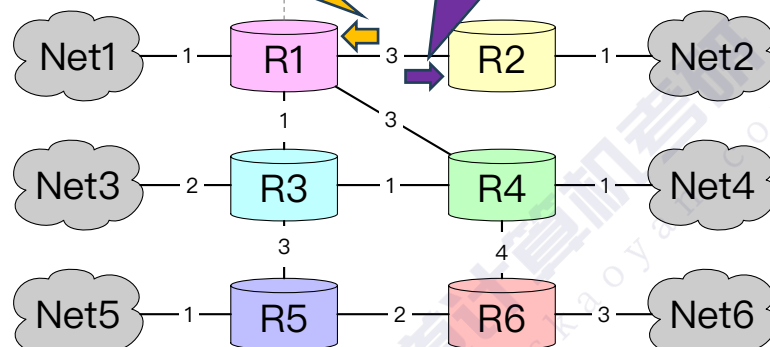


R3、R4、R5、R6 的 LSDB都一样



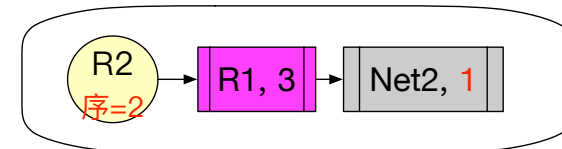
R2的LSU分组，包含 R1 请求的所有 LSA 完整信息

R1的LSU分组，包含 R2 请求的所有 LSA 完整信息



咸鱼电信构建的自治系统AS

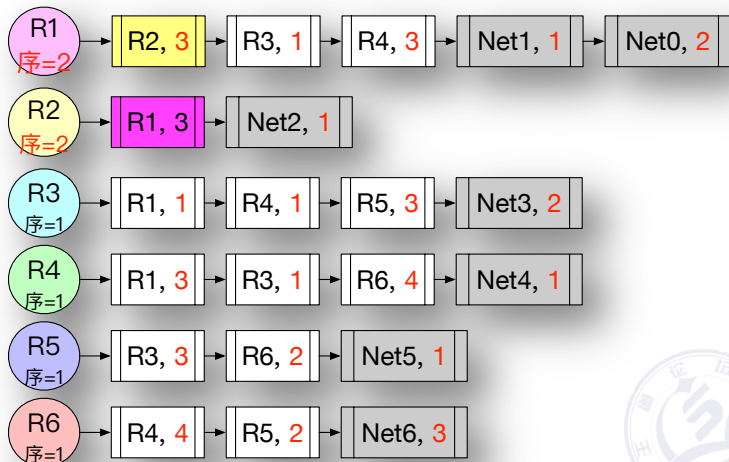
R2 的 LSDB



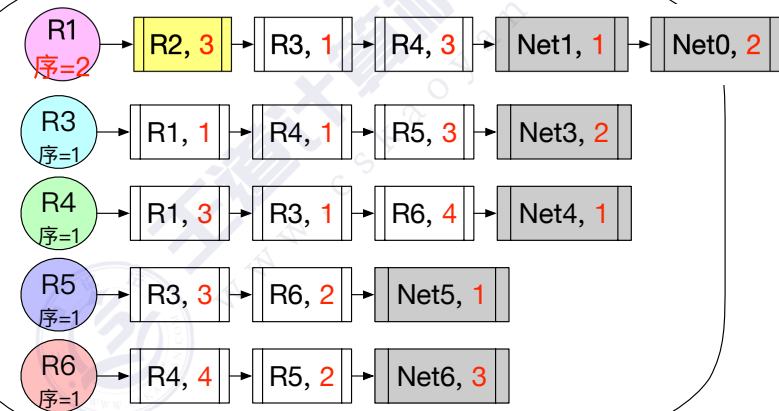
- 路由器通过 LSU 分组，向邻居传输具体的 LSA（可能引发全网洪泛）

Type 4: LSU分组（链路状态更新分组）

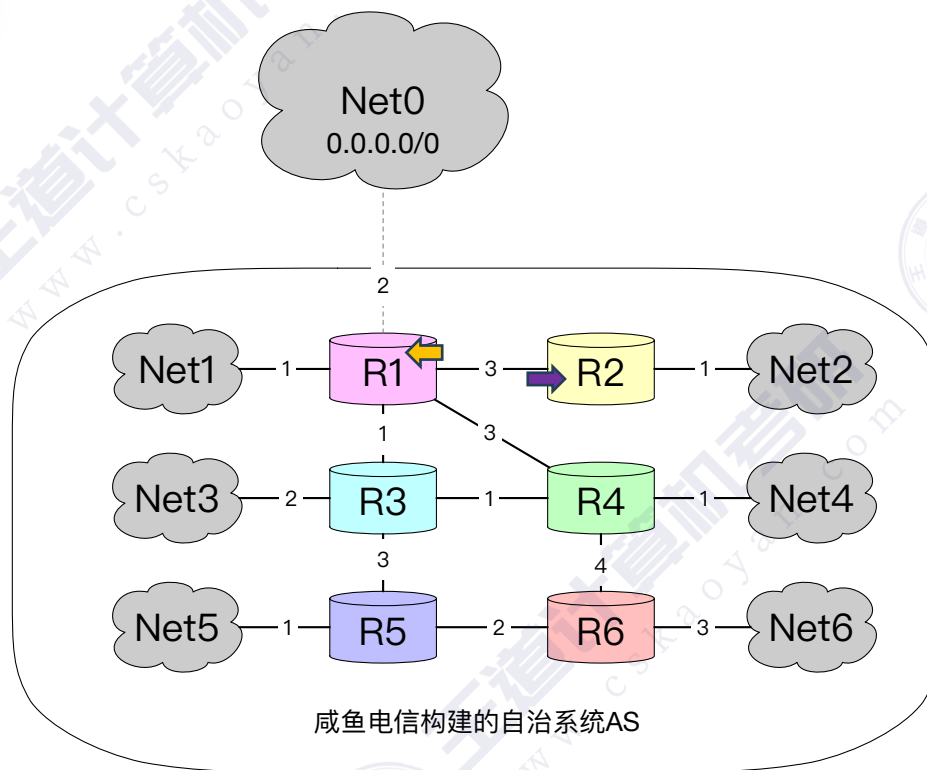
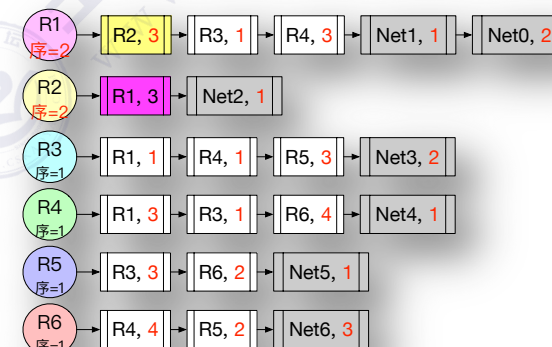
R1 的 LSDB



R3、R4、R5、R6 的 LSDB都一样



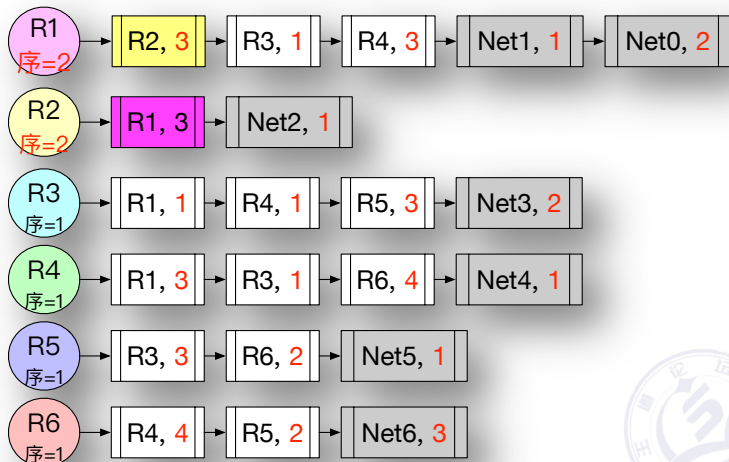
R2 的 LSDB



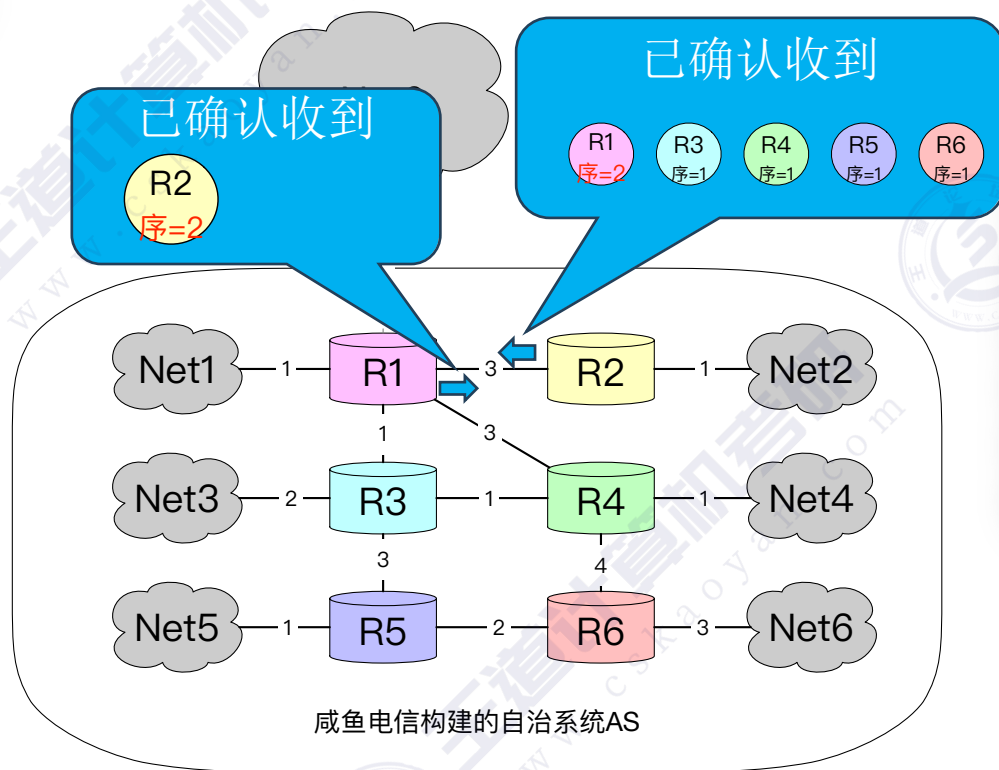
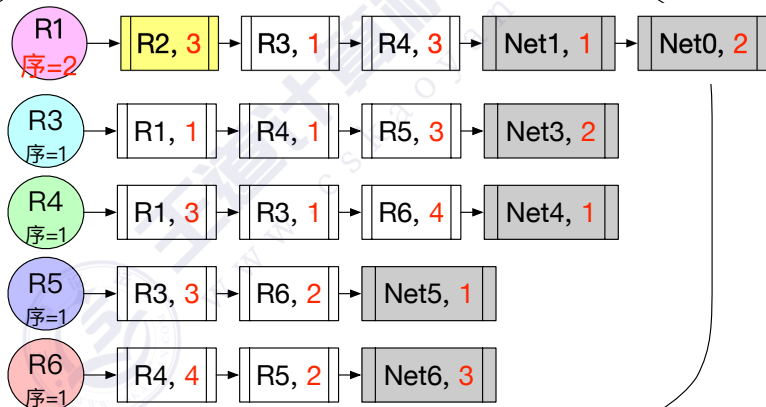
- 路由器通过 LSU 分组，向邻居传输具体的 LSA（可能引发全网洪泛）

Type 5: LSAck 分组（链路状态确认分组）

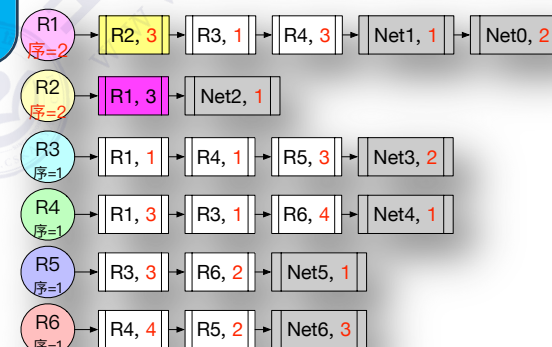
R1 的 LSDB



R3、R4、R5、R6 的 LSDB 都一样

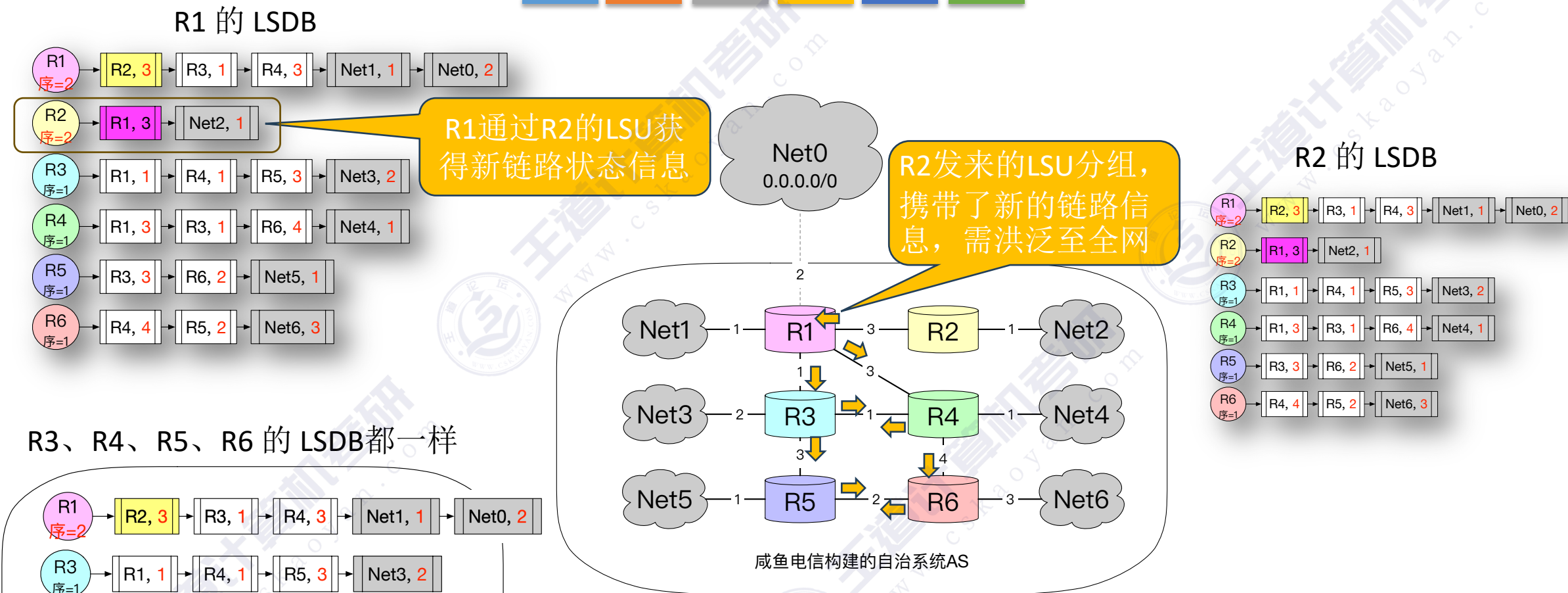


R2 的 LSDB



- 路由器收到邻居发来的 LSU 后，返回 **LSAck 分组**，向邻居确认收到了哪些 LSA（头信息）

LSU分组可能引发全网洪泛



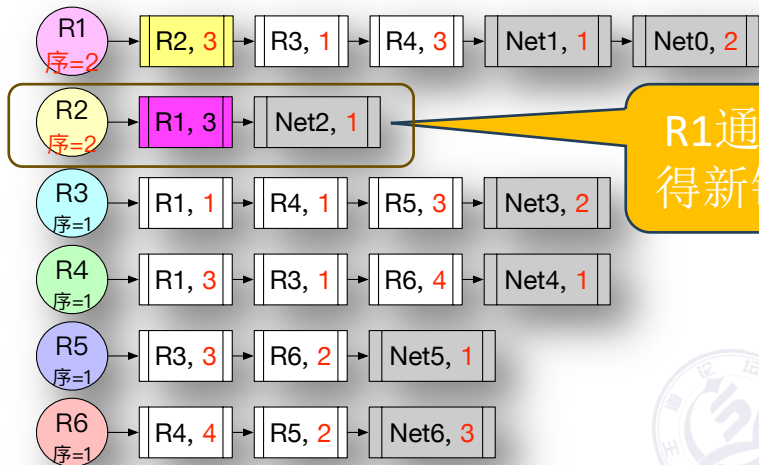
- 路由器通过 LSU 分组，向邻居传输具体的 LSA（可能引发全网洪泛）

注：即便重复收到洪泛的LSU分组，也要给邻居返回 **LSAck**

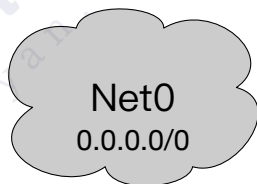
LSU分组可能引发全网洪泛



R1 的 LSDB

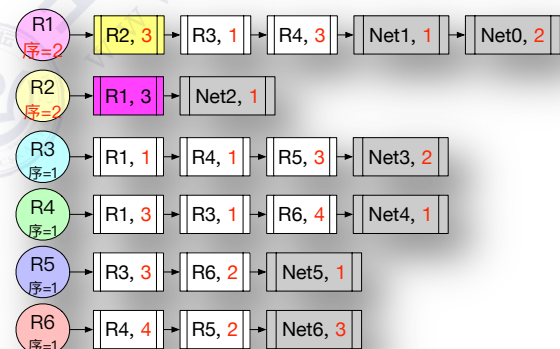


R1通过R2的LSU获得新链路状态信息

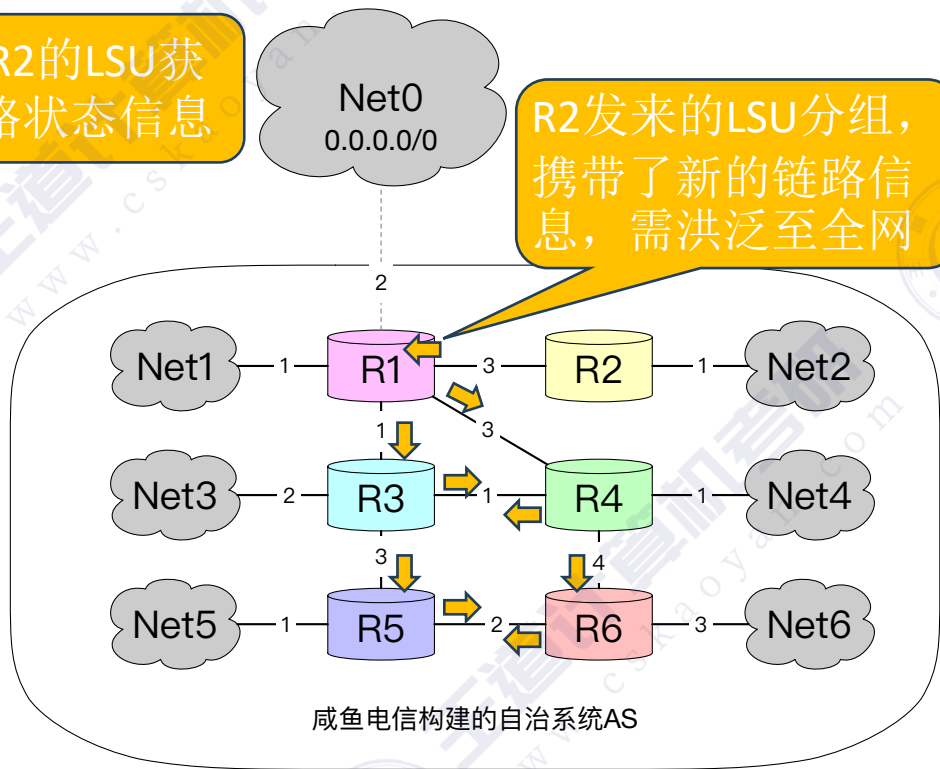
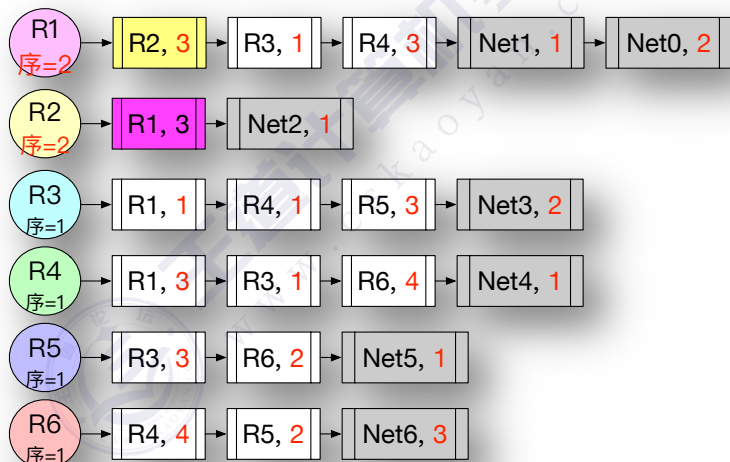


R2发来的LSU分组，携带了新的链路信息，需洪泛至全网

R2 的 LSDB



R3、R4、R5、R6 的 LSDB都一样



咸鱼电信构建的自治系统AS

- 路由器通过 LSU 分组，向邻居传输具体的 LSA（可能引发全网洪泛）

注：即便重复收到洪泛的LSU分组，也要给邻居返回 LSAck