



Ch. 5 IPv6

◆ 5.1 IPv4的问题

- ✧ 互联网正在成为其**自身发展的牺牲品**，几乎每隔20s就有一台主机加入
- ✧ **全球 IPv4地址已经于2011年8月底耗尽**：子网和CIDR不再有效
- ✧ 地址空间的不足，“**三只熊**”问题：
- ✧ 每个地址重 1Gram
 - ☞ A类16777216 : 10^4 Kg ;
 - ☞ B类65536 : 10^2 Kg ;
 - ☞ C类256 : 10^0 Kg



A= 10^4 Kg

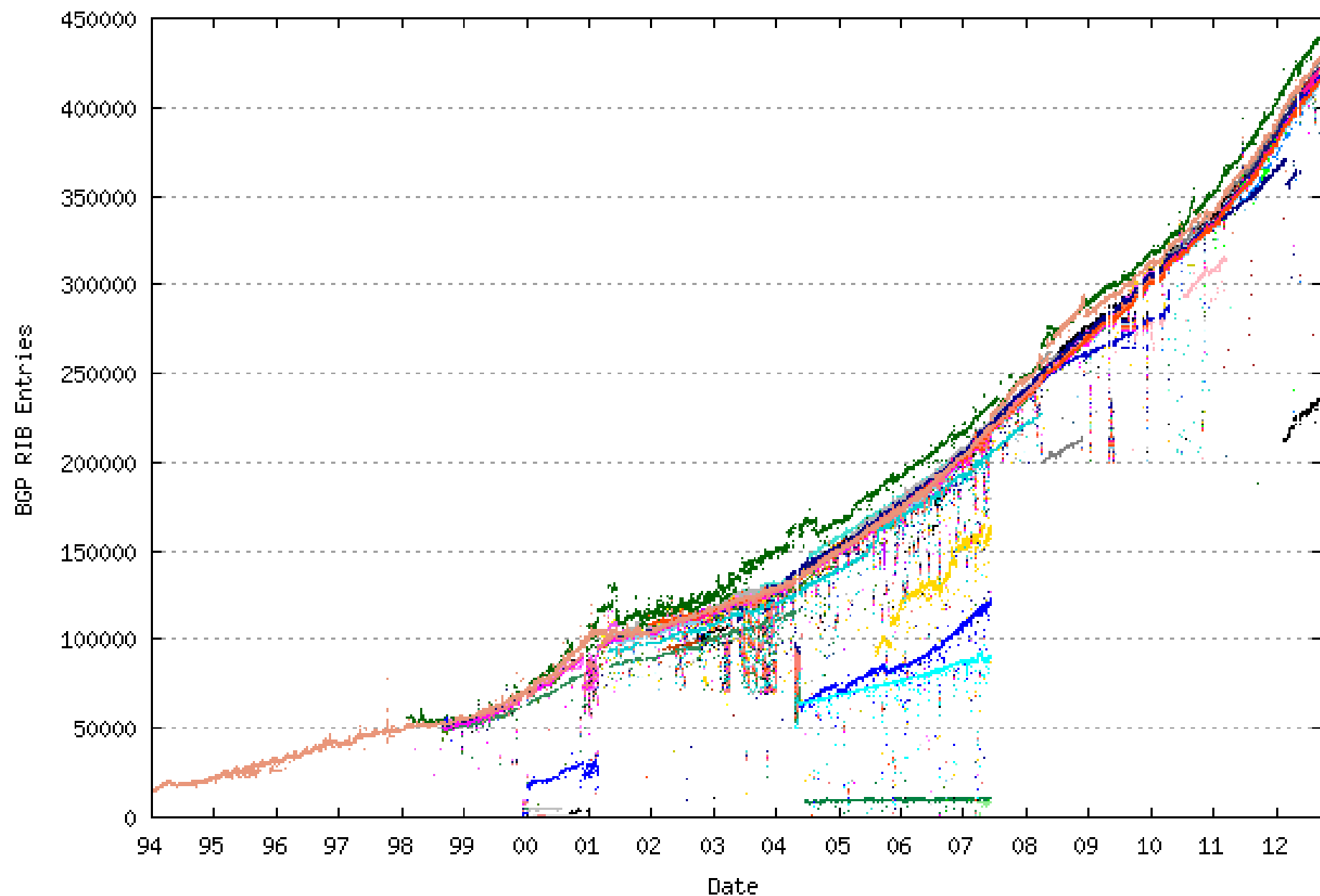



B= 10^2 Kg



C= 10^0 Kg

路由表项剧增Active BGP entries (FIB)



- 
- ◆ 对现有路由技术的支持不够：**路由表爆炸**，IPv4头长度不固定(0-40字节)**不利于ASIC设计**，没有利用包前后的相关性，每个包进行同样处理，MTU导致**分段和逐段**校验，路由处理慢
 - ◆ 无法提供多样的QoS：最大努力最短时间，但不保证是否进行和何时进行，IP尽力而为的FIFO对**实时**多媒体信息的处理会带来延迟、间断，无法满足多媒体传输质量的要求：**QoS**
 - ◆ **移动设备,家电,传感网络**和因特网的连接：HPC/ PDA将占计算机总数的50%
 - ◆ **安全**支持问题

设计目标和主要特征

◆ 扩大地址空间、路由更结构层次化

- ♣ 32bits → 128 bits
- ♣ 全局unicast地址 (2000::/3) 等价IPv4公开地址
- ♣ 网络前缀取代掩码，前缀表示子网号

◆ 报头格式大简化，方便硬处理

- ♣ 基本报头固定40bytes，地址是v4的4倍
- ♣ 简化路由器的操作
- ♣ 引入结构化扩展报头，取消可选项长度限制

◆ 网络管理 更加简单

- ✧ 建立一系列**自动发现和自动配置**功能
- ✧ 最大**单元发现** (MTU discovery)
- ✧ 邻接**节点发现** (neighbor discovery)
- ✧ 路由器通告 (router advertisement)
- ✧ 路由器请求 (router solicitation)
- ✧ 节点**自动配置** (auto-configuration)

◆ 安全性支持

- ✧ IP security ,**提供IP层的安全**
- ✧ 实现**认证头** (Authentication Header)
- ✧ **安全载荷封装** (Encapsulated Security Payload)

◆ QoS能力

- ✱ **流标号** (flow label) ,20比特,发送者可以要求路由器对此流进行特殊处理, R可以鉴别特殊流的所有报文

◆ 多播寻址

- ✱ 在multicast地址中增加了 范围“ scope”字段, 允许将多播路由限定在正确的范围内
- ✱ 另一个“ 标志” 字段允许区分**永久性多播**地址和**临时性**多播地址

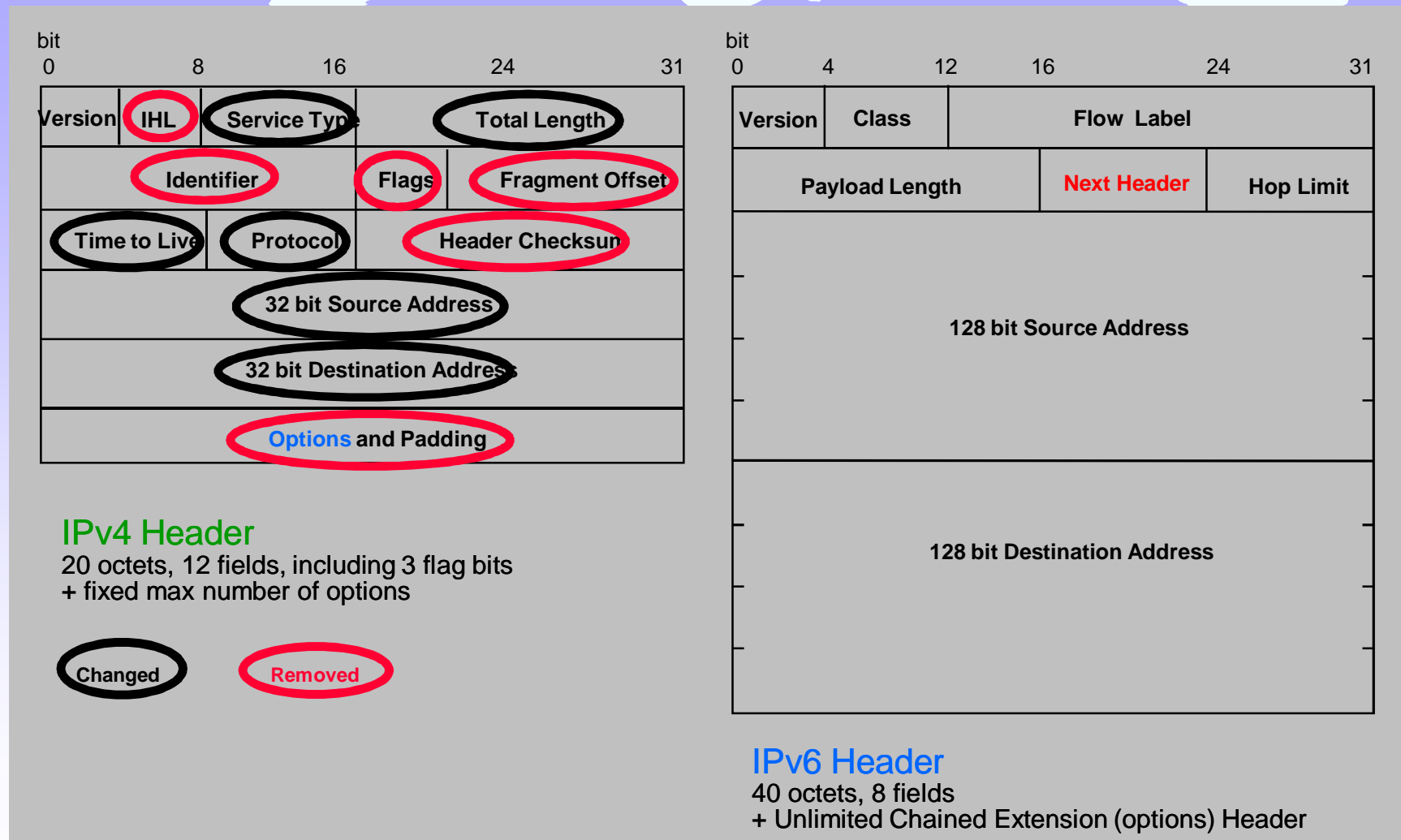
◆ 新的**集群通信**地址方式 - anycast

- ✱ 一点到多点多播时, 把报文传播到组中 (通常最近) 的一个, 从而允许源路由中由节点控制数据报的传送路径

◆ 可移动性

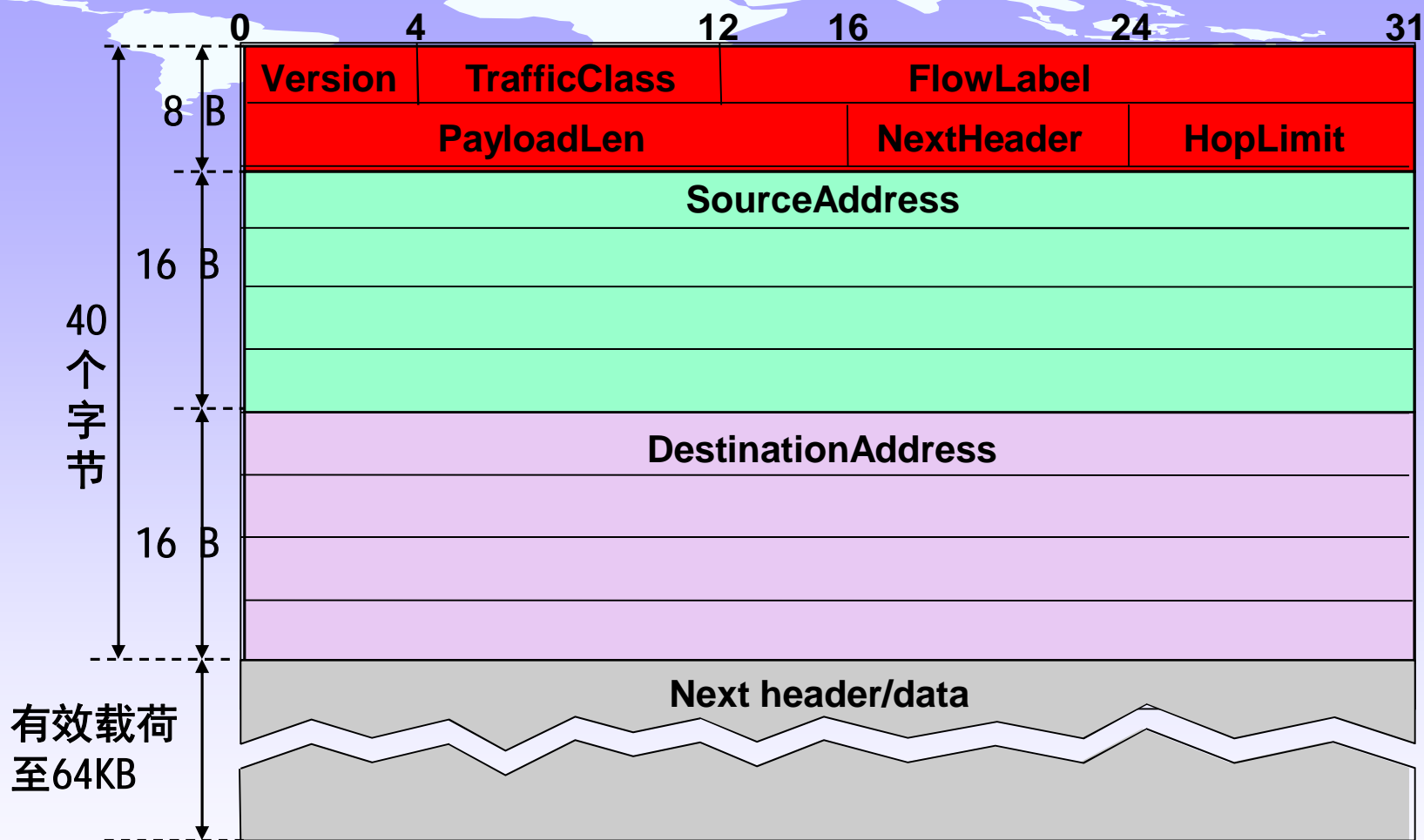
- ✱ 信宿选项报头、路由选项报头、自动配置、安全机制、以及anycast技术, 将QoS同移动节点结合, 从而强化对移动的支持

7.1.2 IPV6的结构



IPV4 = 20个字节 + 选项：13个域（含3个标志位）；IPV6 = 40个字节：8个域

IPv6的报文格式



- ◆ 报头仅40个字节，且固定长度 版本号4位 (v6=6)
- ◆ 优先级8位 (一般0---7，有优先8-15) 流量标识20位，负载长度16位
- ◆ 下一报头号8位，可跳节点数8位 (TTL)



◆ A、主要改变

- ♣ **对齐** (alignment) 已经从32bit的整数倍改为64bit整数倍 ($5 \times 64 \text{ bits} = 40 \times 8 \text{ B} = 320 \text{ bits}$)
- ♣ **取消了报头长度字段，基本报头长度固定40Bytes**
- ♣ **数据报长度地段被有效载荷字段取代**
- ♣ **源目地址字段增加到每个字段16个bytes**
- ♣ **分片信息已经从基本报头的固定字段移到一个扩展报头中**
- ♣ **生存时间TTL改为跳数极限hop-limit字段**
- ♣ **业务类型改为数据流标号flow label 字段**
- ♣ **协议字段改为下一个报头字段，以指明下一个报头类型**

◆ 4 bits - IP协议的版本号 = 6

◆ 8 bits - 通信流类型

- ✱ 相关应用层填充该类型值，默认值是全0
- ✱ 某些节点可对某些比特按特定要求改变其产生、转发和接收，对不能改变的比特，节点忽略
- ✱ 上层协议不能假定信源填充的值不变，宿端收到的值可能与源端不同

◆ 20 bits - 数据流标号

- ✱ 流：一条路径及其上的一些路由器，它保障一定的服务质量；或有相同源目地址的包集合，由信源给出标号
- ✱ 支持新的机制：资源预定（流：一条路径及其上的一些路由器，或有相同源目地址的包集合，它保障一定的服务质量）
- ✱ 允许路由器将每个数据报同一个给定的资源分配相联系
- ✱ 仍在实验中，两个例子：发收视频图象的两个应用程序之间可以建立一个数据流，其带宽和延时可得到保证；；ISP要求用户指明他所希望的QoS，然后指明一个数据流来限制某个计算机或应用程序所发送的流



◆ 8 bits - 跳数极限

- ♣ 对应IPv4的TTL

◆ 128 bits - 源/目地址

- ♣ 源/目地址分别都是 $16 \text{ Bytes} \times 8 = 128 \text{ Bits}$
- ♣ 如果扩展头中出现路由报头，宿地址可能不是最终接收站

◆ 16 bits - 有效载荷长度

- ♣ 因基本报头已固定40bytes，故其长度字段不必要
- ♣ 用16bits表示有效载荷（即不报括基本报头的40bytes，但出现的任何扩展报头都计入有效载荷长度）
- ♣ 故一个IPv6数据报最多可容纳 $2^{16} = 2^6 \times 2^{10} = 64 \text{ Kbytes}$

◆ 8 bits - **下一个报头**(相当V4的**协议字段**或**可选字段**)

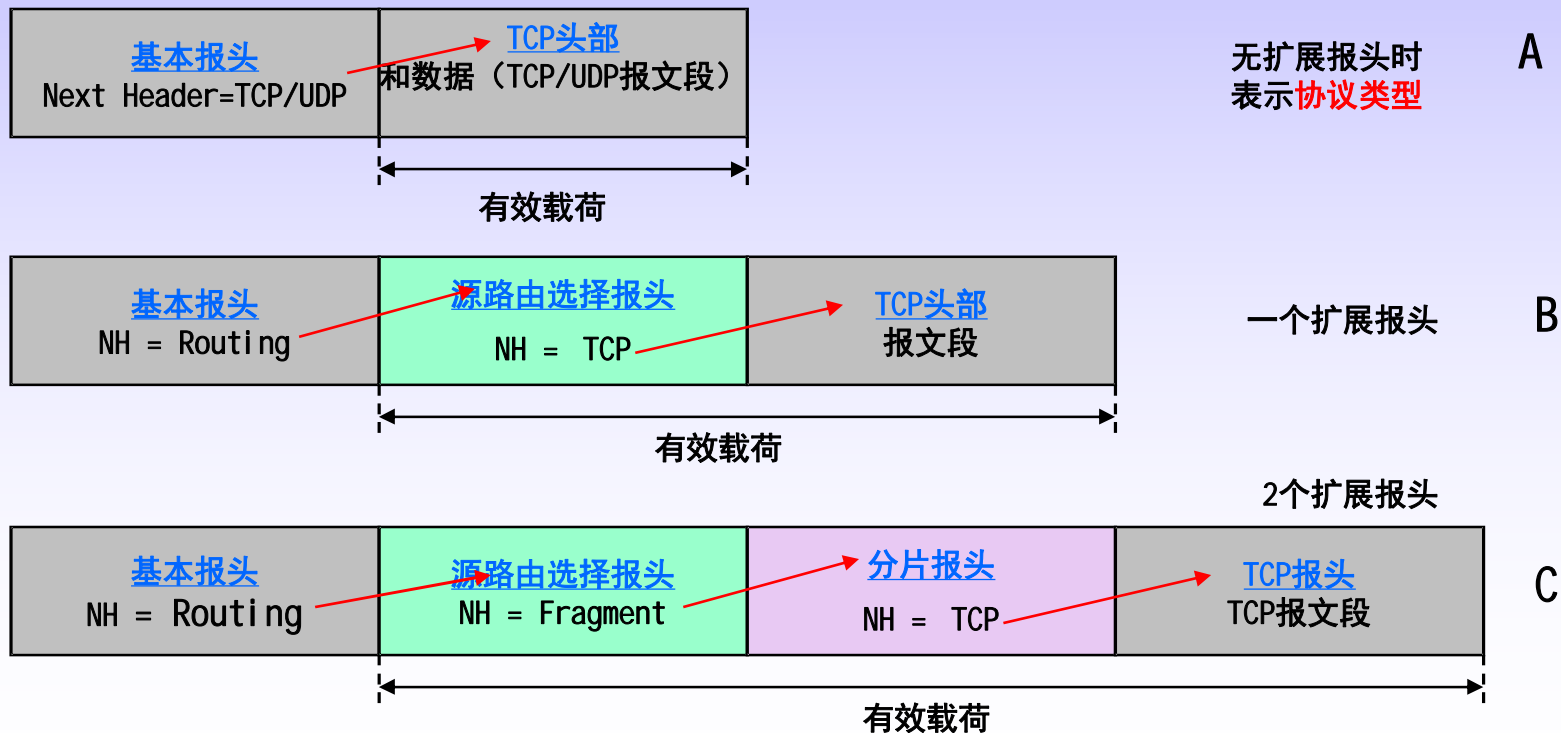
✧ 是IPv6的**重大改进**；

✧ **当无扩展首部时**

☞ 指明基本首部后面的数据应交付给**高层哪一个协议**（如，
6→tcp;17→UDP）

✧ **当有扩展首部时**

☞ 指明标识后面第一个扩展首部的**类型**



报头的扩展

◆ 扩展报头概述(等价V4首部中的选项，6种)

- ✱ V4路由器必须对选项一一检查，降低了处理速度
- ✱ V6把选项等效功能放在**扩展首部**，并将此留给两端主机处理，中间路由器**除逐跳外**（**信息传递给所有路过的路由器**）、**源路由选择**，都不处理，提高了效率

◆ 目的：协议还要支持**分段、安全封装、目的站选项及鉴别**等功能

- ✱ 若把其放在固定报头中将是不合算的：如在局域网内发送数据报时，每个包中无用选项会占很大的比重

◆ 扩展报头放在基本报头和高层报头之间

- ✱ 每一个扩展报头都由上一个报头中的下一个报头值决定
- ✱ 每一个中间路由器和最终目的站要对该值进行分析、决定处理否

7.1.3 IPv6的地址空间

- ◆ 4.3.0 How big is the 128 bits address space ?
- ◆ Increased address space
 - ♣ 128 bits = 340 trillion (10^{12}) trillion trillion addresses
 - ♣ $2^{128} = (2^{10})^{12.8} > (10^3)^{12.8} = 10^{38.4}$, 准确数目是
340,282,366,920,938,463,463,374,607,431,768,211,456 $\geq 3.4 \times 10^{38.4}$
 - ♣ 655,570,793,348,866,943,898,599 addresses per m² of the planet' s surface = $2^{128} / 511,263,971,197,990 \text{ m}^2$ (地球表面积) $\approx 6.02 \times 10^{23}$ 即约每平方米1摩尔地址,阿伏加德罗常数
 - ♣ 而IPv4则只有每平方米4个地址
- ◆ Hierarchical address architecture
 - ♣ Improved address aggregation
 - ♣ Simpler address design

If an IPv4 Address Weighed 1 Gram...

IPv4 = 1/76th weight of Empire State Building



Empire State Bldg. = 365,000 tons*
= 328.5 billion grams

$$\frac{32.85e+10}{2^{32}} = 76.48$$



所以帝国大厦是**IPv4**所有地址重量的**76**倍

If an IPv6 Address Weighed 1 Gram...

IPv6 address space = 56.7 billion 个地球重量



Earth = 6.00e+24 kg*

* <http://www.howstuffworks.com/question30.htm>

$$\frac{2^{128}}{6.00e+27} = 56,713,727,820 > 567 \text{ 亿个地球!}$$

I Pv6的地址结构

◆ A.地址的三种文本表示，以方便怎样阅读、输入和操作

♣ 点分十进制

☞ 104.230.140.100.255.255.255.255.0.0.17.128.150.10.255.255

♣ 冒分16进制，共8个，相同字间距。上面地址为

☞ 68E6:8C64:FFFF:FFFF:0:1180:96A:FFFF

♣ 0压缩::表示，对连续长串0用::代替，一个地址中仅出现一次，如：

☞ 2080:0:0:0:8:800:200C:417A→2080::8: ... ; unicast address

☞ FF01:0:0:0:0:0:0:101→FF01::101 ; multicast address

☞ 0:0:0:0:0:0:0:1 → ::1 ; loopback address

☞ 0:0:0:0:0:0:0:0 → :: ; undefined address

♣ 混合表示，x: x: x: x: x: x: d.d.d.d，x:表示16进制(16 Bits)， d.表示10进制(8 Bits)

☞ 0:0:0:0:0:0:13.1.168.3 或 ::13.1.168.3

☞ 0:0:0:0:0:FFFF:129.144.52.38或 ::FFFF:129.144.52.38

B. 地址结构前缀的表示

◆ 类似CIDR形式

♣ IPv6地址/前缀长度，长度是10进制，表明地址最左端连续比特个数

♣ 正确表示12AB00000000CD3的60bits前缀是

☞ 12AB:0000:0000:CD30:0000:0000:0000:0000/60

☞ 12AB::CD30:0:0:0:0/60

☞ 12AB:0:0:CD30::/60

♣ 不正确的表示为

☞ 12AB:0:0:CD3/60 ; 可能丢失前面的0

☞ 12AB::CD30/60 ; 可理解为12AB:0:0:0:0:0:0:CD30

☞ 12AB::CD3/60 ; 可理解为12AB:0:0:0:0:0:0:0CD3

◆ HUST的IPV6地址 = $2^{81} = 2^{80} \times 2$

♣ 2001:0250:4000::/48;

♣ 2001:0DA8:3000::/48;

IPv6的地址模式

- ◆ **地址分配到接口：**
 - ♣ 这同v4一样，没有变化
 - ♣ 一个接口可有多多个地址
- ◆ **地址有范围之分**
 - ♣ Link Local
 - ♣ Site Local (RFC3879解释了取消站点局部地址)
 - ♣ Global
- ◆ **地址有寿命**
 - ♣ 有效的
 - ♣ 永久的
- ◆ **地址结构**
 - ♣ 前缀 + 接口ID

IPv6的寻址

◆ 地址类型：

- ♣ Unicast: One to One(Global,Link local,Site local, Compatible)
- ♣ Anycast:One to Nearest(Allocated from Unicast)
- ♣ Multicast:One to Many
- ♣ Reserved

◆ 单个接口可能分配有任何单播、近播和组播地址

◆ 广播由组播代替

IPv6的寻址

以太网适配器 本地连接:

```
连接特定的 DNS 后缀 . . . . . :  
IPv6 地址 . . . . . : 2001:250:4000:4400:50f2:e352:c018:e4b5  
IPv6 地址 . . . . . : 2001:250:4000:4406:50f2:e352:c018:e4b5  
临时 IPv6 地址. . . . . : 2001:250:4000:4400:71fe:b99b:507:b54  
临时 IPv6 地址. . . . . : 2001:250:4000:4406:71fe:b99b:507:b54  
本地连接 IPv6 地址. . . . . : fe80::50f2:e352:c018:e4b5%12  
IPv4 地址 . . . . . : 115.156.236.164  
子网掩码 . . . . . : 255.255.255.0  
默认网关. . . . . : fe80::eda:41ff:fe1b:a7fb%12  
115.156.236.254
```

- ◆ 全局IPv6地址（别人可以访问）
- ◆ 临时IPv6地址（访问别人，基于微软rfc3041“ Privacy Extensions for Stateless Address Autoconfiguration in IPv6”）
- ◆ 本例链接IPv6（fe开头访问同一局域网内IP）

IPv6的地址类型

- ◆ **A.地址分类**：1998RFC2460对IPv6的地址类型分为三类
 - ✧ **单播 - unicast**：目的地址指明一个单一的计算机（single interface），可是主机或路由器，发送到unicast的包将选择一条最短的路径到达目的站
 - ✧ **近播 - anycast（集群...）**：目的地址是共享一个地址前缀的计算机集合（a set of interfaces），典型的情况是在**同一物理网络上的不同节点**，发送到anycast地址的包将选择一条最近路径到达该集群（路由度量距离最近的节点）中一个。
 - ✧ **组播-multicast**：目的地址是一组计算机（a set of interfaces），典型情况是属于不同网路的不同节点，发送到一个multicast地址的把将通过硬件或广播投递给组中的每个成员。IPv6中没有广播地址，其功能可由组播取代
- ◆ **所有IPv6地址都是分配给interface而不是node的，所有接口都必须有至少一个link-local unicast，一个单接口可分配任何一种类型的多重地址（uni/any/multicast）或地址范围**
- ◆ **子网前缀仅与一条链路相连（继承v4），但多重子网前缀可分配给同一条链路**

B. 地址类型的表示

- ◆ 一个IPv6地址的具体类型是由其领先的bits决定的
- ◆ 包含这些比特的变长字段称为格式前缀FP (Format Prefix)
- ◆ 其最初的分配表如下

分配	前缀		所占比例
	2进制	16进制	
保留	0000 0000	0::/8	1/256
未分配	0000 0001	100::/8	1/256
为NSAP分配保留	0000 001	200::/7	1/128
为IPX分配保留	0000 010	400::/7	1/128
未分配	0000 011	600::/7	1/128
未分配	0000 1	800::/5	1/32
未分配	0001	1000::/4	1/16
可聚类全局 unicast地址	001	2000::/3	1/8
未分配	010	4000::/3	1/8
未分配	011	6000::/3	1/8
未分配	100	8000::/3	1/8
未分配	101	A000::/3	1/8
未分配	110	C000::/3	1/8
未分配	1110	E000::/4	1/16
未分配	1111 0	F000::/5	1/32
未分配	1111 10	F800::/6	1/64
未分配	1111 110	FC00::/7	1/128
未分配	1111 1110 0	FE00::/19	1/512
Link-local Unicast 地址	1111 1110 10	FE80::/10	1/1024
Site-local Unicast地址	1111 1110 11	FEC0::/10	1/1024
Multicast地址	1111 1111	FF00::/8	1/256

表中内容的几点说明

- ◆ 不要把保留地址和未分配地址混淆,保留地址不等于未分配地址
- ◆ 保留地址有3种, 共占 $2^{128-8}/2^{128} = \text{占} 1/256$, 由前缀 0000 0000 表示,
 - ☞ 全零地址 - 没有规定的地址
 - ☞ Loopback地址 - 回送地址
 - ☞ 嵌入到了IPV4地址的IPV6地址
- ♣ 其它保留地址
 - ☞ 0000 001: 为NSAP (Network Service Access Point) 保留, 占空间 1/128
 - ☞ 0000 010: 为Novell的IPX保留, 占空间 1/128

- ◆ 前缀为001 - 111，除Multicast地址（1111 1111）外，都必须有64 Bits的EUI-64G格式的接口标识符
- ◆ 表上的分配仅仅使用了地址空间的15%，剩余的85%留待今后使用
- ◆ 这种分配方式支持聚类地址、本地使用地址和Multicast地址的直接分配，剩余空间既可支持现有使用扩展（如附加可聚类地址），也可用于新的领域
- ◆ Unicast 和 multicast 的地址靠最高8位来区分，FF表示是multicast地址，其它是Unicast地址
- ◆ Anycast 地址从Unicast中分出，格式上同Unicast没有区分

C. Uni cast 地址(RFC2374)

- ◆ Unicast地址严格聚类，具有连续子网掩码
- ◆ 目前的Unicast地址分配的几种形式是
 - ♣ 全局可聚类地址
 - ♣ NASP地址
 - ♣ IPX层次地址
 - ♣ Link-Local地址
 - ♣ Site-Local地址
 - ♣ IPv4兼容主机地址
 - ♣ 将来可定义的其它地址类型

IPv6地址的结构

- ◆ IPv6地址结构的解释依据节点所扮演的角色（主机或者是路由器），最小情况下，可认为没有任何内在的结构，如图1
- ◆ 稍复杂一点的主机，可能知道与其相连子网的 n 位前缀，如图2

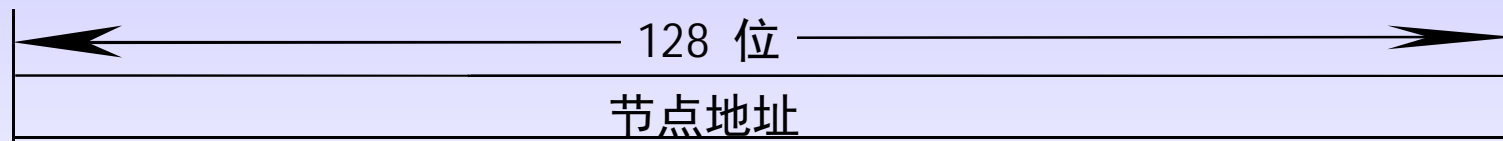


图1. 没有内在结构的Uni cast地址

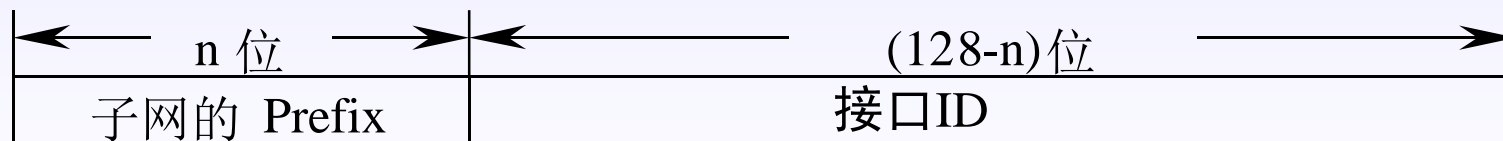
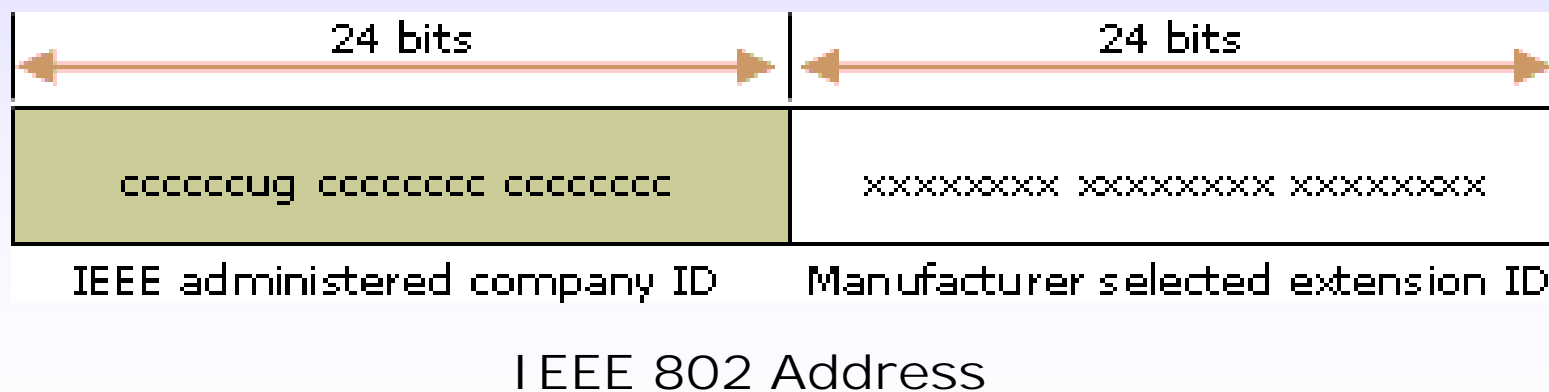


图2. 有子网前缀的Uni cast地址

C.1 接口标识符

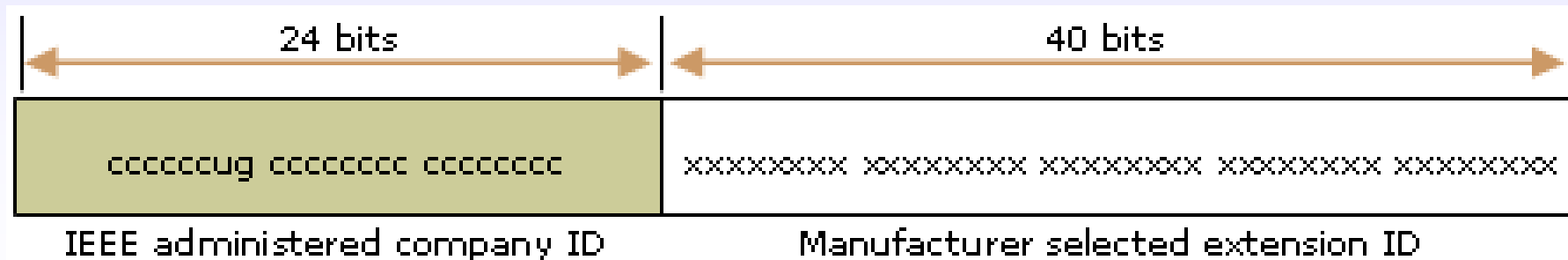
◆ IEEE EUI-Extended Unique Identifier : 扩展唯一标识符

- ♣ RFC 2373 规定所有unicast地址必须有64比特的EUI-64的接口ID
- ♣ EUI 来自IEEE 802 Address, 24位制造商地址, 24位扩展地址—合成为硬件、物理或MAC地址



◆ EUI-64的接口ID

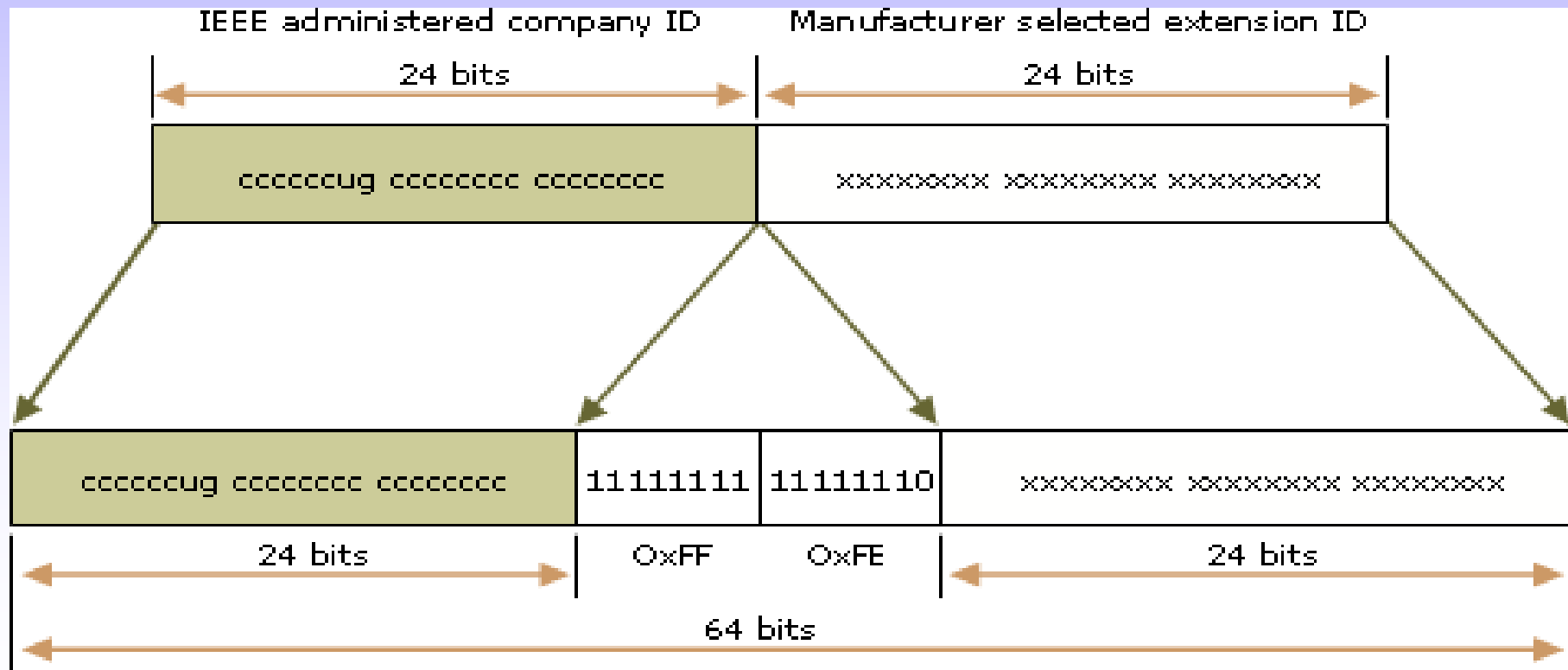
- ♣ The IEEE EUI-64 address 表示网络物理接口寻址的新标准
- ♣ 公司 ID 仍然 24-bits, 扩展ID 40 bits, 给网卡商更大的地址空间
- ♣ EUI-64 地址中的 U/L 和 I/G bits 与 IEEE 802 address的表示意义相同
- ♣ Universal/Local (**U**/L)是第7位, 0: 表示全局ID; 1: 表示局部ID; Individual/**G**roup (I/G)是第8位, 0: 表示单播地址; 1: 组播地址
- ♣ **C**是制造商的标识符



IEEE EUI-64 Address

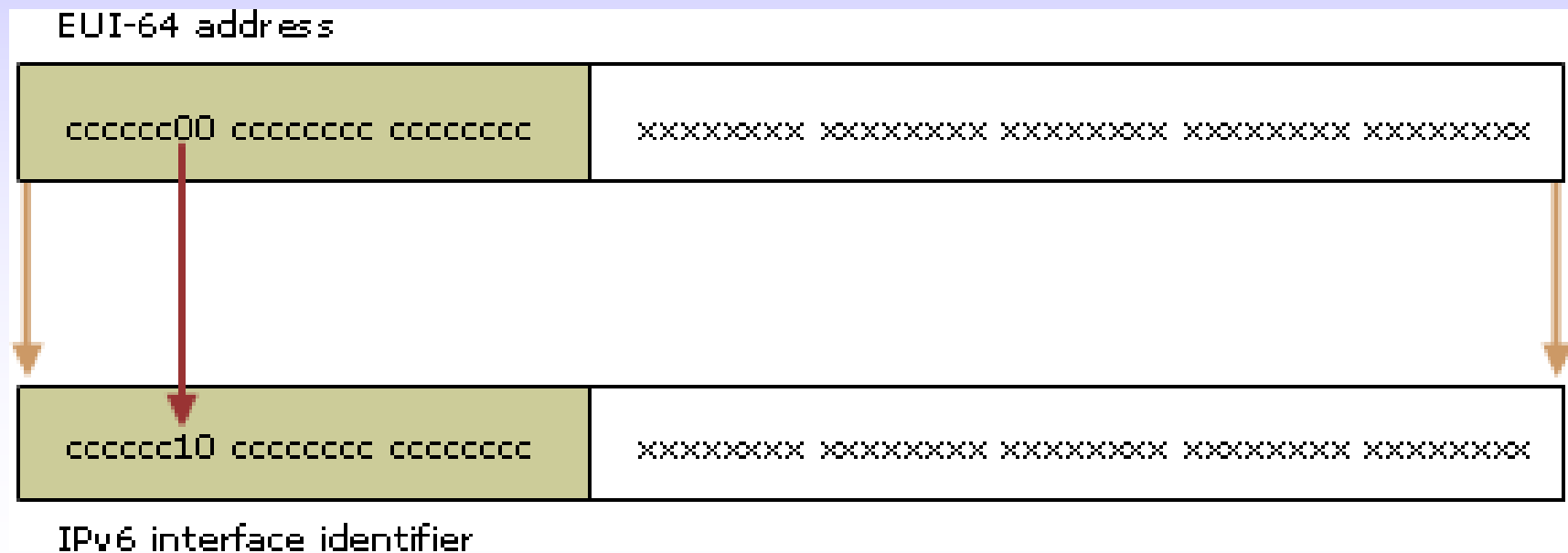
Mapping IEEE 802 to EUI-64 addresses

- ◆ To create an EUI-64 address from an IEEE 802 address, the 16 bits of **11111111 11111110** (0x**FFFE**) are inserted into the IEEE 802 address between the company ID and the extension ID.



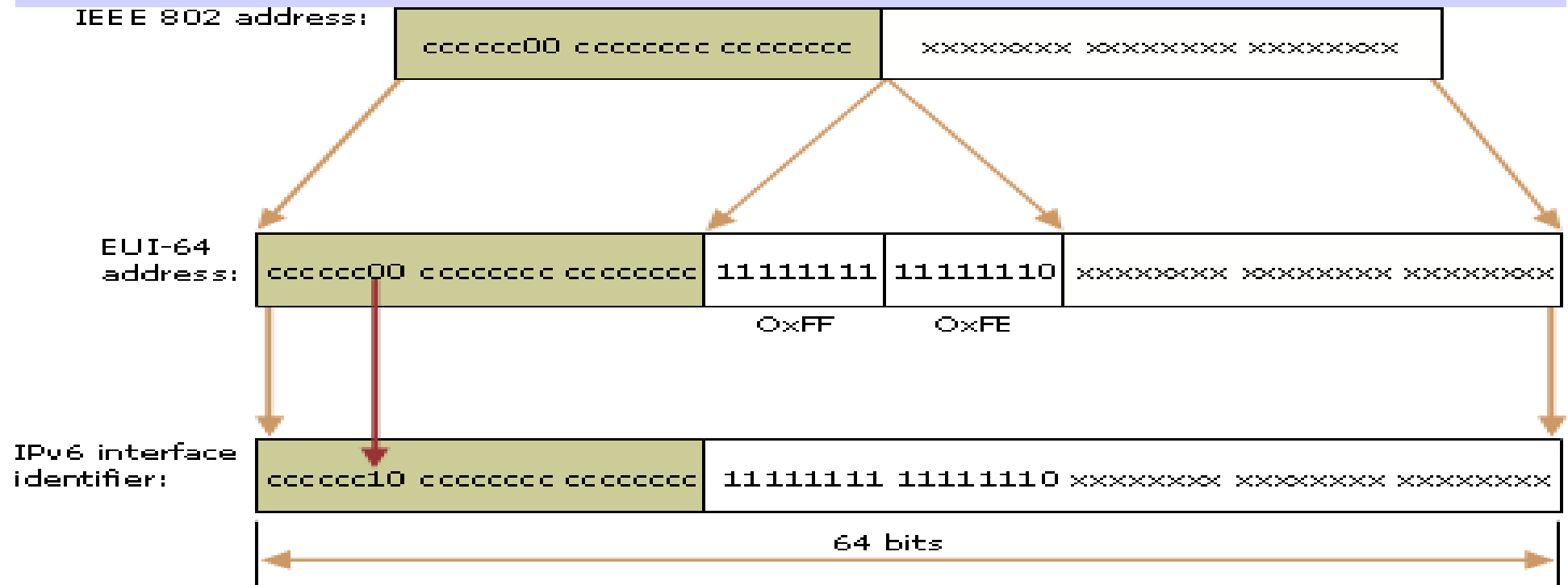
Mapping EUI -64 addresses to IPv6 ID

- ◆ To obtain the 64-bit interface identifier for IPv6 unicast addresses, the U/L bit in the EUI -64 address is complemented (取补) if it is a 1, it is set to 0; and if it is a 0, it is set to 1).



Mapping IEEE 802 to IPv6 ID

- ◆ To obtain an IPv6 interface identifier from an IEEE 802 address, you must first map the IEEE 802 address to an EUI-64 address, and then complement the U/L bit. The following illustration shows the conversion process for a universally administered, unicast IEEE 802 address.



C.2 三种保留地址

- ◆ 未规定的地址 - 全零地址，0:0:0:0:0:0:0:0；不分配给任何节点，表示一个缺失地址。应用例之一是，还未分配IP地址的主机初始化中，要发送IPV6包时，用全零地址作为自己的**暂时源地址**。它不能作为信宿地址
- ◆ Loopback - 自环地址，0:0:0:0:0:0:0:1；可被任何节点用于向自身发送IPV6包，它不能分配给任何物理接口，不能作为任何包的信源地址，以此为信宿的包永远不能发出该节点，永远不能被路由器转发
- ◆ 包含IPv4地址的IPv6地址：RFC 1993定义了IPv6 over IPv4 tunnel 机制。应用这种技术的IPv6节点被分配给一种特殊的IPv6 unicast 地址，其最低32位是IPv4地址

◆ 这种支持IPV4的地址又分2种

- ✱ 既支持V4，也支持V6：如图1，是与V4兼容的V6地址，
- ✱ 仅支持V4，不支持V6：如图2，对不支持V6的R/H，应屏蔽最低32位以上的部分，V6还规定33--48位全部为F，高位全0时其为V4地址

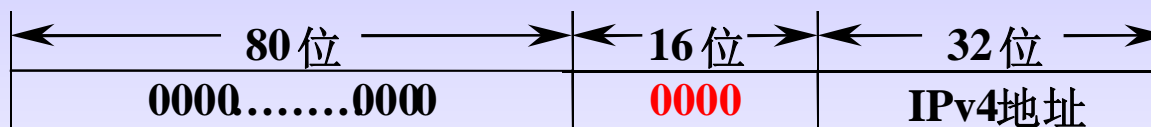


图. IPV4 兼容的IPV6地址

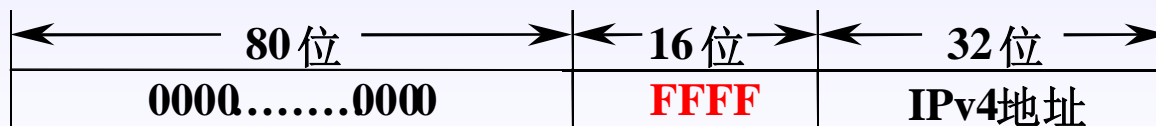


图2. IPV4 映射的IPV6地址



其它保留地址

- ◆ 前缀 = 0000 001 ; 为NSAP (Network Service Access Point) 分配保留的地址, 占1/128
- ◆ 前缀 = 0000 010 ; 为Novell IPX分配保留的地址, 1/128

C.3 可聚类全局Unicast地址

- ◆ FP为**001**都是全局可聚类Unicast地址，由RFC 2374 给出
- ◆ 基本假设：路由系统基于连续bit边界的“最长前缀匹配”算法来选择转发路径
- ◆ 特点：
 - 与IPv4的CIDR不同，IPv6强制规定，地址中的前64个bit才能作为网络地址。
 - 在理想情况下，一个核心主干网路由器只须维护不超过8192个表项
 - IPv6改变了地址的分配方式，从用户拥有变成了ISP拥有。全球网络号由因特网地址分配机构（IANA）分配给ISP，用户的全球网络地址是ISP地址空间的子集。每当用户改变ISP时，全球网络地址必须更新为新ISP提供的地址。这样ISP能有效地控制路由信息，避免路由爆炸现象的出现。

C.3 可聚类全局Uni cast地址

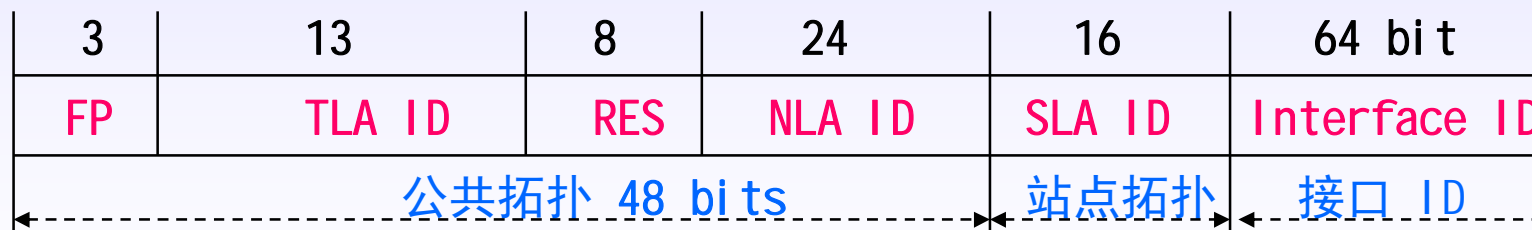
◆ 特点：（继续）

- IPv4中，一旦用户从某机构处申请到一段地址空间，他就永远使用该地址空间，而不管他是从哪个因特网服务提供者（ISP）处获得服务。
- IPv6既支持现有基于提供者（provider based）聚类，还支持基于交换者（exchanges）聚类方式
- IPv6从而既能为与网络提供者直接相连的节点提供有效的路由聚类，也能为与交换者直接相连的节点提供有效的路由聚类

IPv6全局可聚类Unicast地址格式

◆ 共分3级，6个段

- ♣ 001:全球可聚合Unicast地址
- ♣ TLA ID (Top Level Aggregator) : 顶级聚合标识符，分配给大型ISP，从IANA直接获得。
- ♣ RES:留做将来使用 - Reserved for future use
- ♣ NLA ID (Next Level Aggregator) : 次级聚合标识符，中型ISP从TLA获取。
- ♣ SLA ID (Site Level Aggregator) : 站点级聚合标识符，小型ISP从NLA获得
- ♣ 接口ID : 接口标识符Interface Identifier



顶级聚类标识符 – TLA ID

- ◆ 位于路由层次的顶层
- ◆ 共支持 $2^{13} = 8192$ 个 TLA ID , 扩展方式:
 - ✧ 将 TLA 字段扩展到保留字段中去
 - ✧ 将这种格式应用于两外的格式前缀
- ◆ **保留字段 - Reserved** 2^8
 - ✧ 现在必须设置成 0
 - ✧ 可为 TLA NLA 提供增长的可能

次级聚类标识符—NLA ID

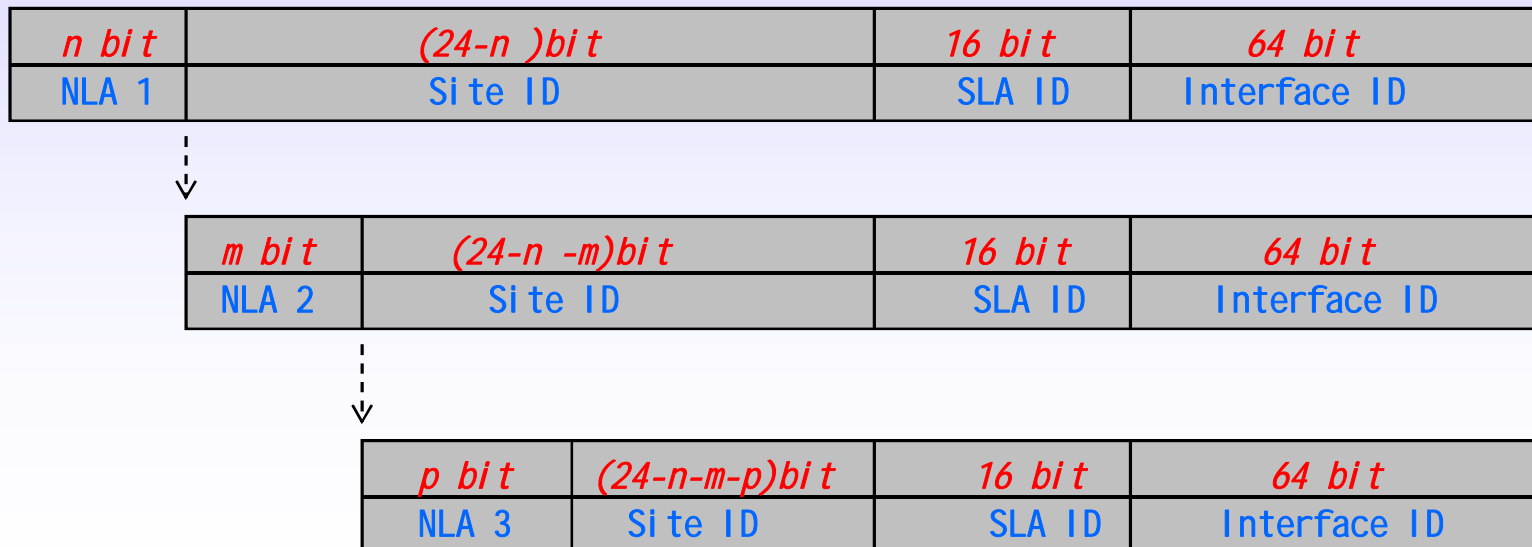
- ◆ 为已得到TLA ID的机构创建的地址层次和标识站点
- ◆ 每1个TLA ID空间允许该机构为大致相当于目前IPv4 internet所支持网络总数的其它结构提供服务
- ◆ 下面是一个NLA ID的可能结构

<i>n bit</i>	<i>(24-n) bit</i>	<i>16 bit</i>	<i>64 bit</i>
NLA 1	Site ID	SLA ID	Interface ID

用字段中的其它比特来标识
将要提供服务的站点

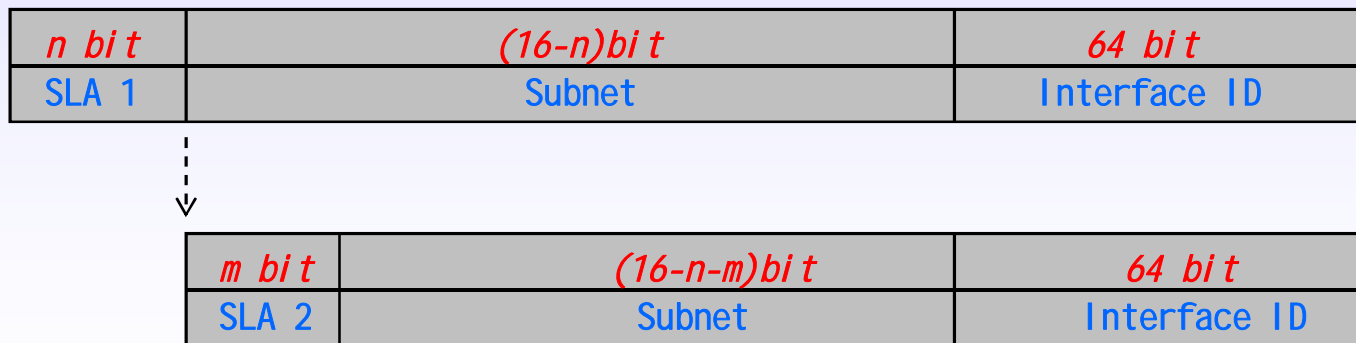
多层NLA结构

- ◆ 得到TLA ID的机构可以在自身的NLA ID的空间中支持多级NLA ID
- ◆ 上前一级NLA ID机构负责管理下一级的NLA ID空间的bit
- ◆ NLA ID的分级是路由聚类效率和灵活性的折中，层次越多，则聚类可能越大，同时减少了路由表的规模；而平坦（flat）的NLA ID分配简单且灵活，同时也增加了路由表的规模



地点聚类标识符 – SLA ID

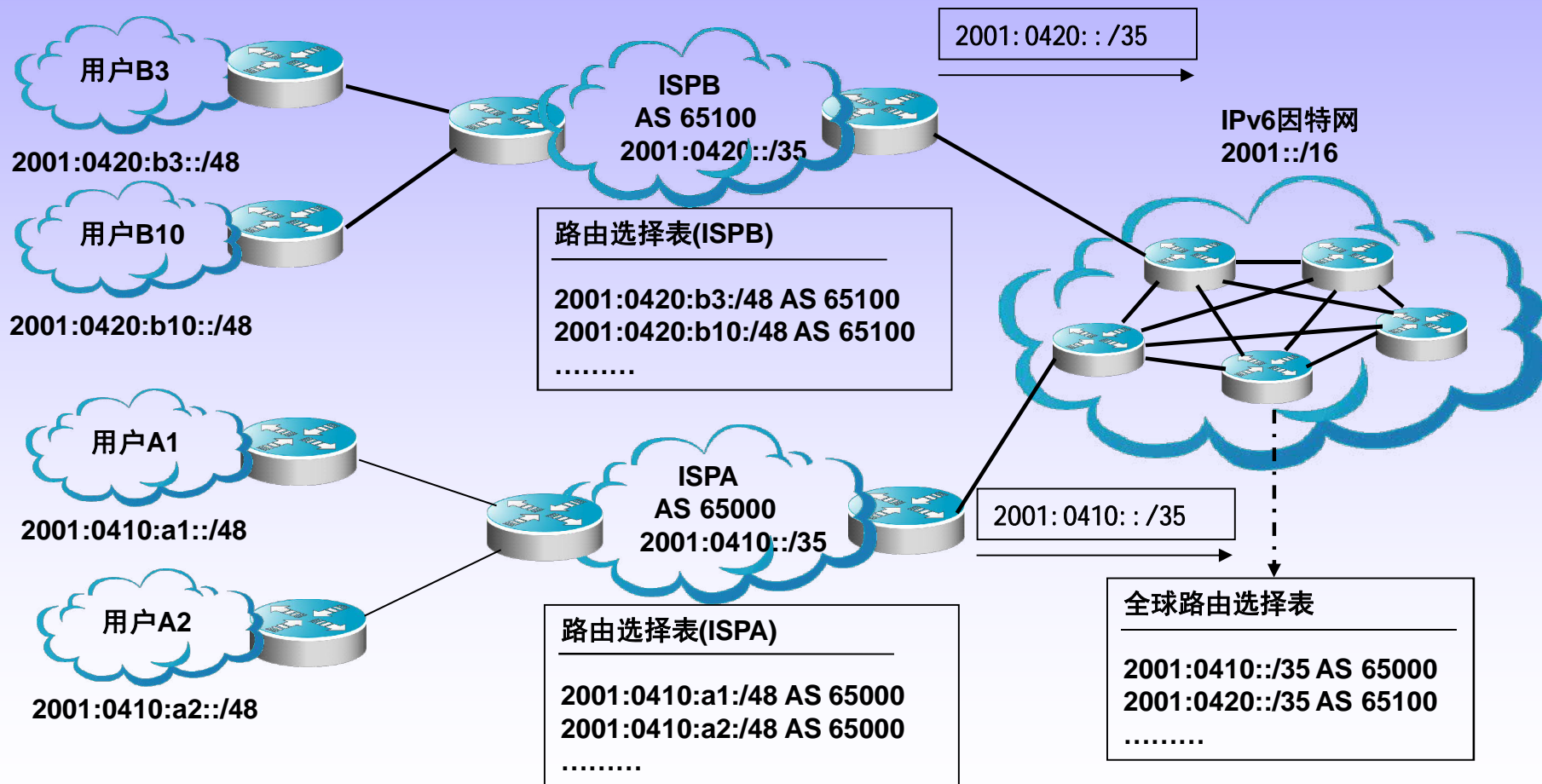
- ◆ SLA ID字段由单独机构用决定来创建其内部的地址层次并标识子网（这类似v4的子网划分），SLA ID有 $2^{16} = 65535$ 个子网
- ◆ 可选择将SLA ID设置成flat路由（SLA内无任何逻辑关系，但增加路由表规模）；也可在SLA ID字段中创建更多的层次和级别（将减少路由表规模）
- ◆ 如果需要额外的子网，可向internet上级服务机构申请更多的站点标识符来创建子网



可能的SLA结构

有效的、分级的寻址和路由结构

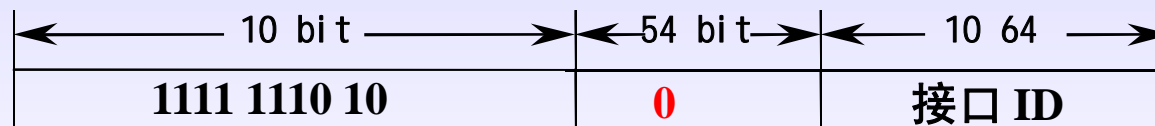
下面以一个实例来说明IPv6路由的聚合：




提供商聚合客户的前缀并公告他们的前缀到IPv6因特网

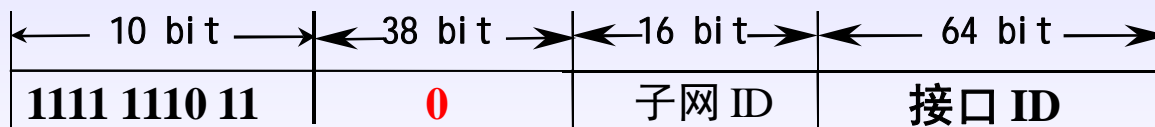
C.4 本地使用的2类Uni cast地址

- ◆ Link-Local address : **前缀PF** = 1111 1110 10 (FE80::/10)
- ◆ **用于单条链路上的地址分配，例如**
 - ♣ 自动地址配置 - auto-address configuration
 - ♣ 邻站发现 - neighbor discovery
 - ♣ 没有路由器时可以使用
- ◆ **路由器不能转发任何以Link-Local为源目的包到其它链路**



Link local 的地址格式

- 
- ◆ Site-Local address : 前缀 PF = 1111 1110 11
 - ◆ 用于一个单独的站点，站点内不需要全局前缀的地址分配
 - ◆ 路由器不能转发任何以Site-Local为源目的的包到其它链路



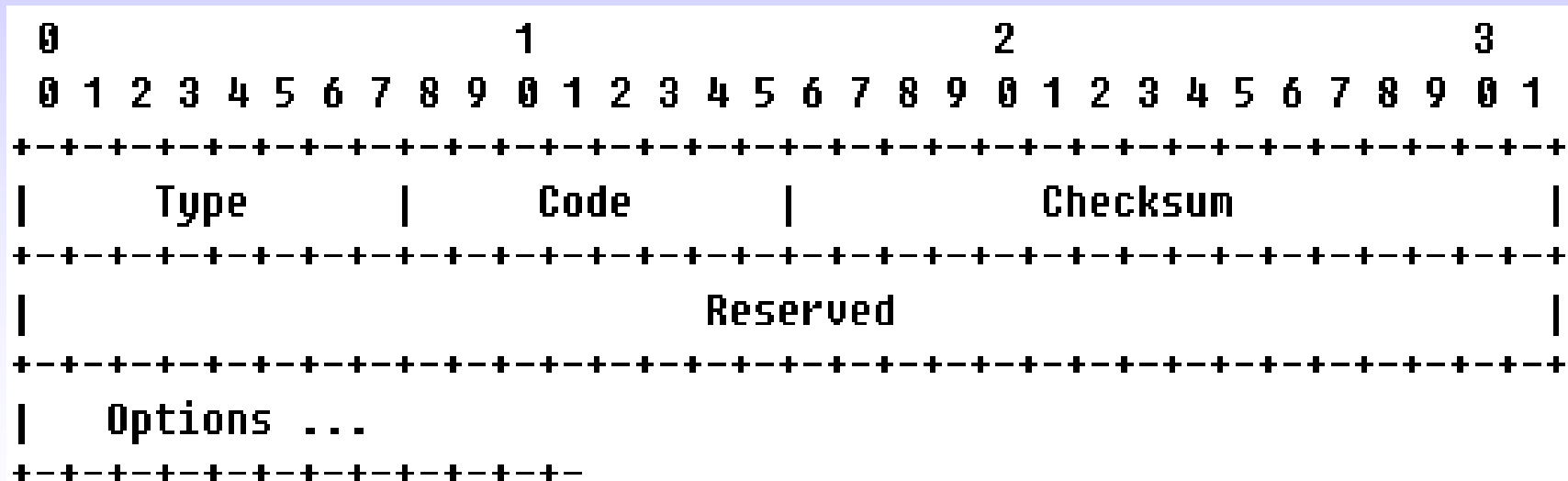
Site local 的地址格式

邻居发现协议报文

- ◆ **基于ICMPv6报文实现其功能 (RFC 4861)**
- ◆ **路由器请求 (Router Solicitation)**
- ◆ **路由器通告 (Router Advertisement)**
- ◆ **邻居请求 (Neighbor Solicitation)**
- ◆ **邻居通告 (Neighbor Advertisement)**

Router Solicitation报文

- ◆ RS是主机发送的报文，触发路由器迅速产生路由器通告。
- ◆ 回应报文为RA报文



Router Solicitation报文结构

◆ IP 部分

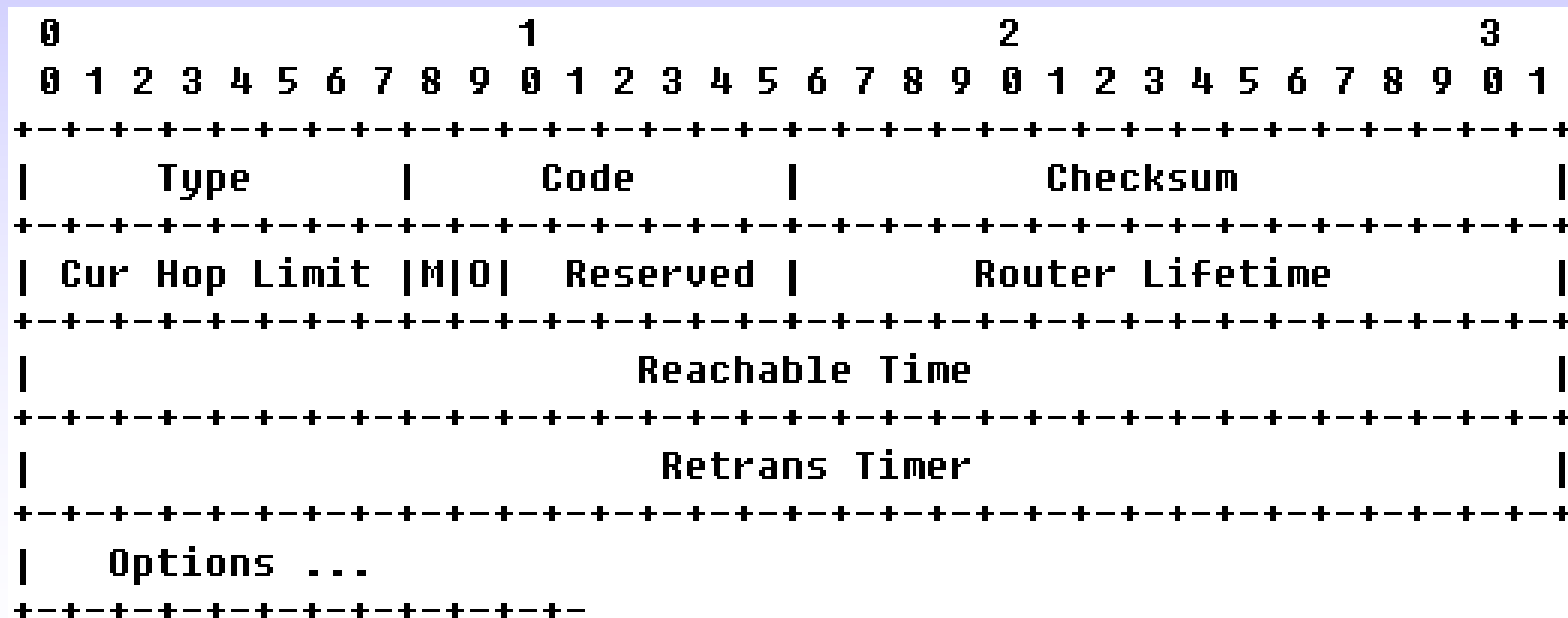
- ♣ **源地址**：接口（link-local）的地址或者 unspecified（全0）。
- ♣ **目的地址**：全部路由器组播地址FF::02
- ♣ **跳数**：255

◆ ICMP部分

- ♣ Type=133
- ♣ Code=0
- ♣ **选项部分包含了发送者的link-layer地址**

Router Advertisement报文

- ◆ 由路由器发出
- ◆ 路由器周期性地发送路由器通告消息，或者对路由器请求作出响应
- ◆ 报文结构如下：



Router Advertisement报文结构

◆ IP部分

- ♣ 源地址：发送者Link-local地址
- ♣ 目的地址：全部节点组播地址FF02::1或发送RS的主机单播地址
- ♣ 跳数：255

◆ ICMP部分

- ♣ Type=134
- ♣ Code=0
- ♣ Cur hop limit=主机发送IPv6报文，hop limit的值是多少
- ♣ 选项部分包含了发送者的link-layer地址
- ♣ 选项部分包含了MTU、地址前缀

Router Advertisement报文结构（续）

◆ ICMP部分

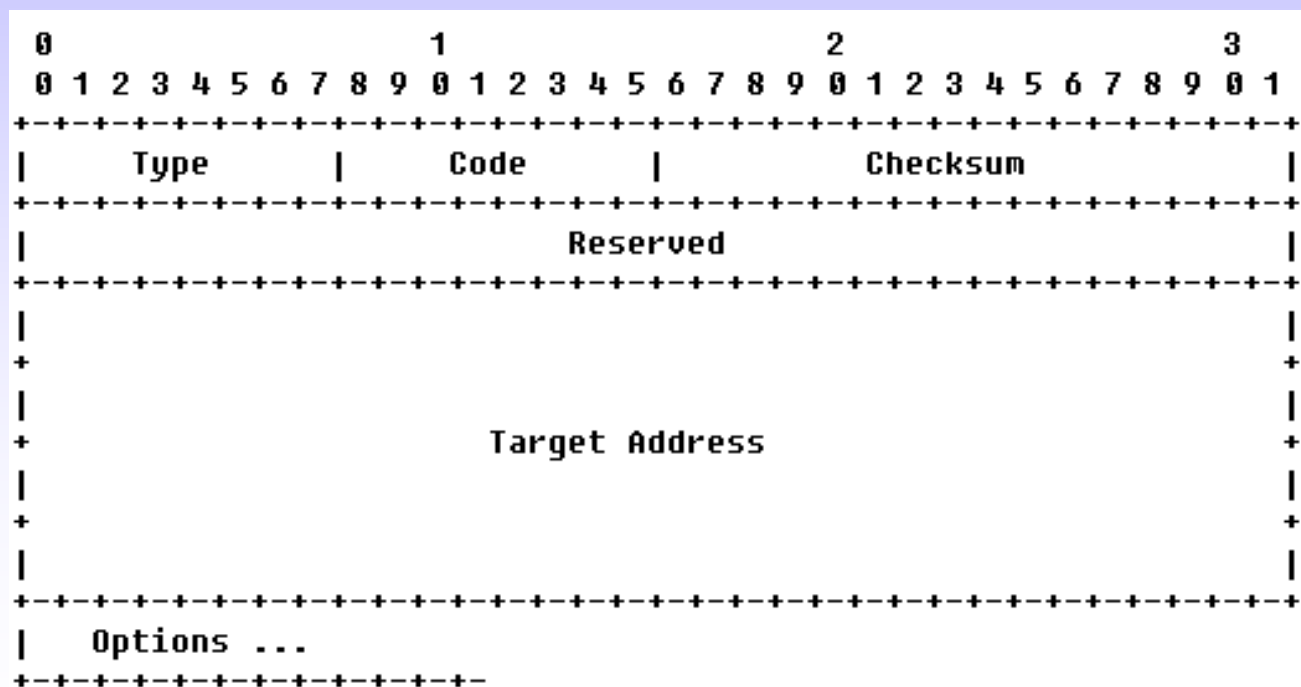
- ♣ M=0 , 表示使用stateless 地址自动配置
- ♣ M=1 , 表示使用stateful 地址自动配置(DHCPv6)
- ♣ O bit , other information , such as DNS
- ♣ Router Lifetime , 表示存在于主机default router缓存中的时间
- ♣ Reachable Time , 表示存在于主机邻居缓存中的时间
- ♣ Retrans Timer , 表示进行邻居检测时的重新发送间隔

Neighbor Solicitation报文

◆ 用途：

- ♣ 链路层地址解析
- ♣ 地址重复检测 (DAD)

◆ 报文结构如下：



Neighbor Solicitation报文结构

◆ IP部分

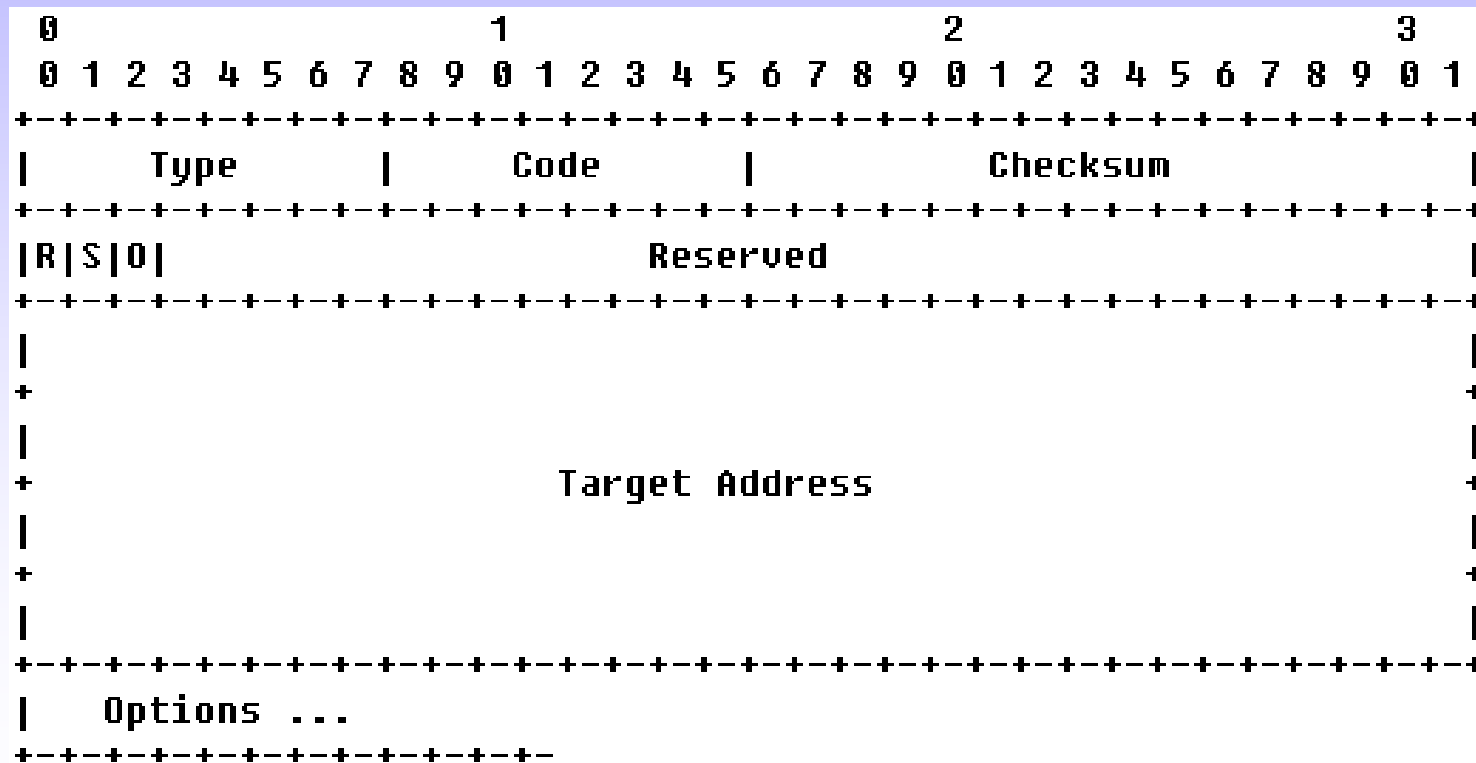
- ♣ **源地址：发送者IPv6地址（地址解析用）或 unspecified地址（DAD重复地址检测）**
- ♣ **目的地址：请求节点组播地址（DAD用）或单播地址（地址解析用）**
- ♣ **跳数：255**

◆ ICMP部分

- ♣ **Type=135**
- ♣ **Code=0**
- ♣ **Target address=发送者IPv6地址**
- ♣ **选项部分包含发送者链路层地址**

Neighbor Advertisement报文

- ◆ 回复NS报文
- ◆ 报文结构如下：



Neighbor Advertisement报文结构

◆ IP部分

- ♣ 源地址：发送者IPv6地址
- ♣ 目的地址：全部节点组播地址FF02::1（DAD用）或发送NS的主机单播地址（地址解析用）
- ♣ 跳数：255

◆ ICMP部分

- ♣ Type=136
- ♣ Code=0
- ♣ 选项部分包含了发送者链路层地址

邻居发现协议—地址解析

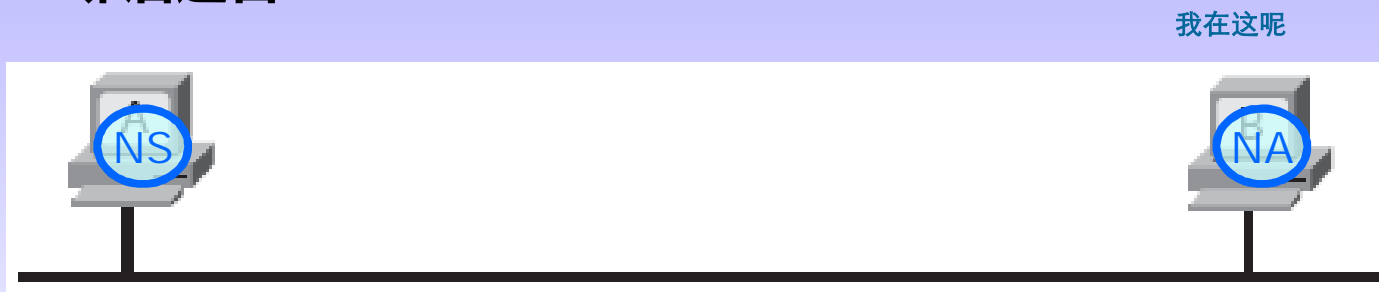
- ◆ 地址解析在三层完成，不同的二层介质可以采用相同的地址解析协议
- ◆ 可以使用三层的安全机制（例如IPSec）避免地址解析攻击
- ◆ 使用组播方式发送请求报文，减少了二层网络的性能压力

邻居发现协议—地址解析

◆ 使用两种ICMPv6报文完成交互过程

✧ 邻居请求NS

✧ 邻居通告NA



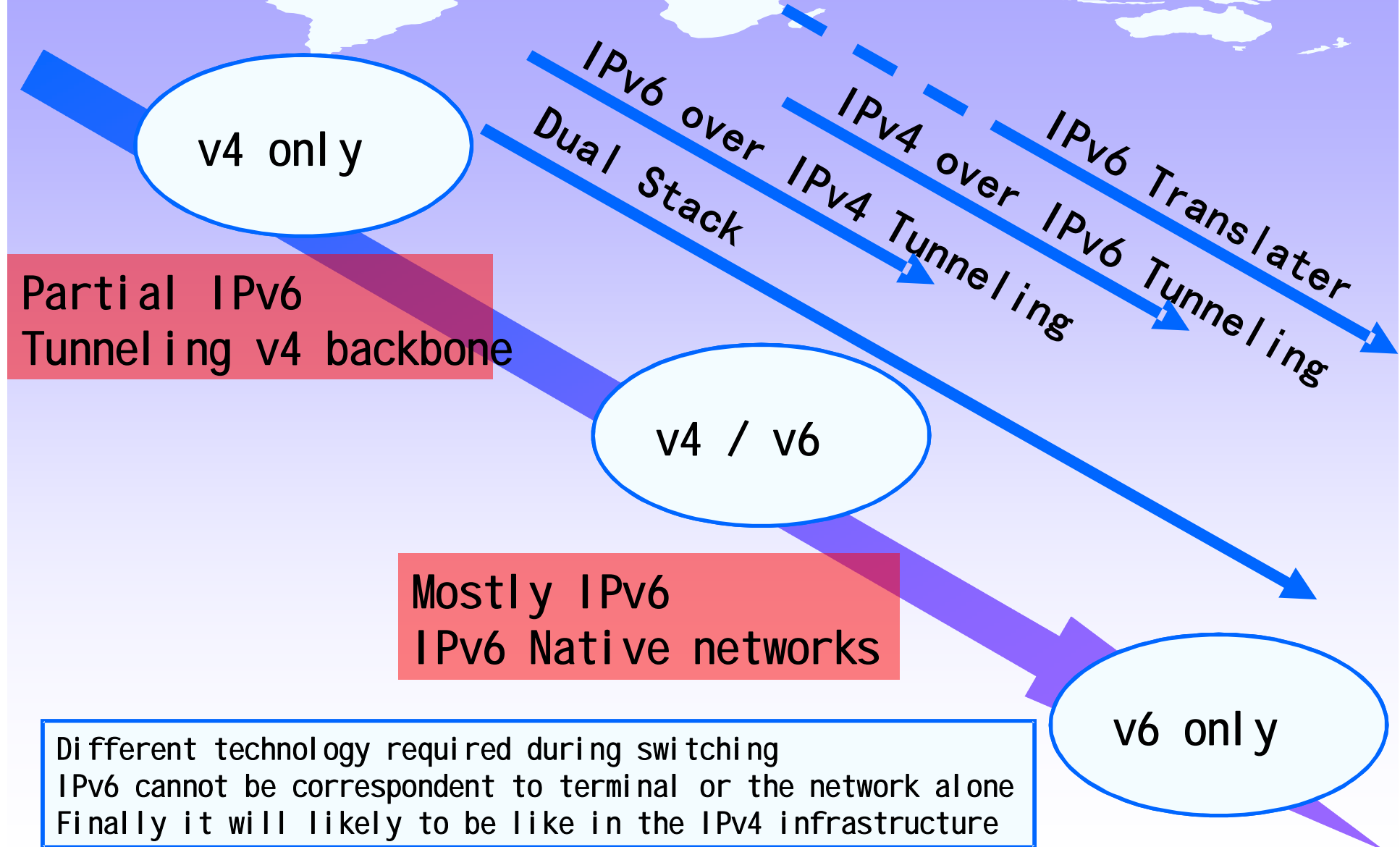
- 以太网报头
 - 目的MAC：多播MAC地址
- IPv6报头
 - 源地址：A
 - 目的地址：B的请求节点多播地址
- ICMP类型135
- NS报文头
 - 目标地址：B
- NS选项
 - A的MAC地址

- 以太网报头
 - 目的MAC：A的MAC地址
- IPv6报头
 - 源地址：B
 - 目的地址：A
- ICMP类型136
- NA报文头
 - 目标地址：B
- NA选项
 - B的MAC地址

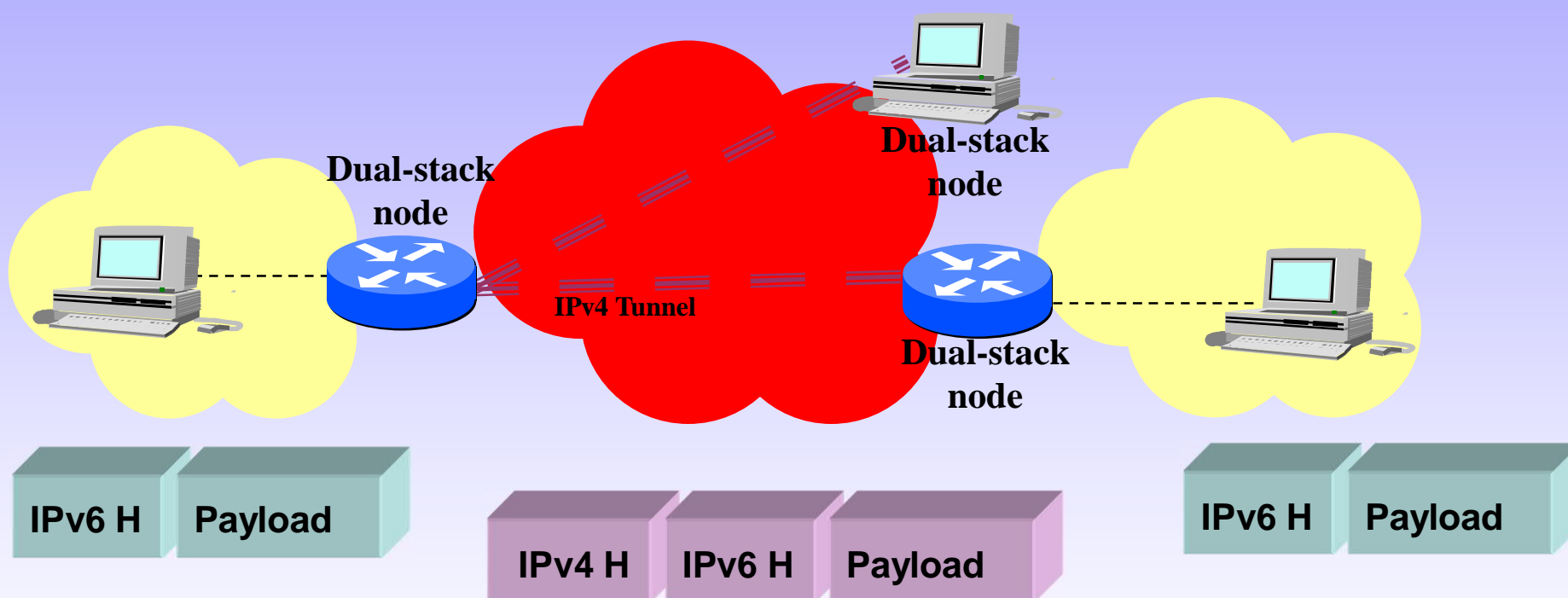
邻居发现协议—重复地址检测 (DAD)

- ◆ 重复地址检测确保网络中无两个相同的单播地址
- ◆ 所有地址都需要做DAD
- ◆ 使用NS和NA完成DAD交互过程
- ◆ 若发现有地址重复
 - ♣ 全局单播地址：不安排给接口
 - ♣ 链路本地地址：将接口置于不可用状态

7.1.4 From IPv4 to IPv6

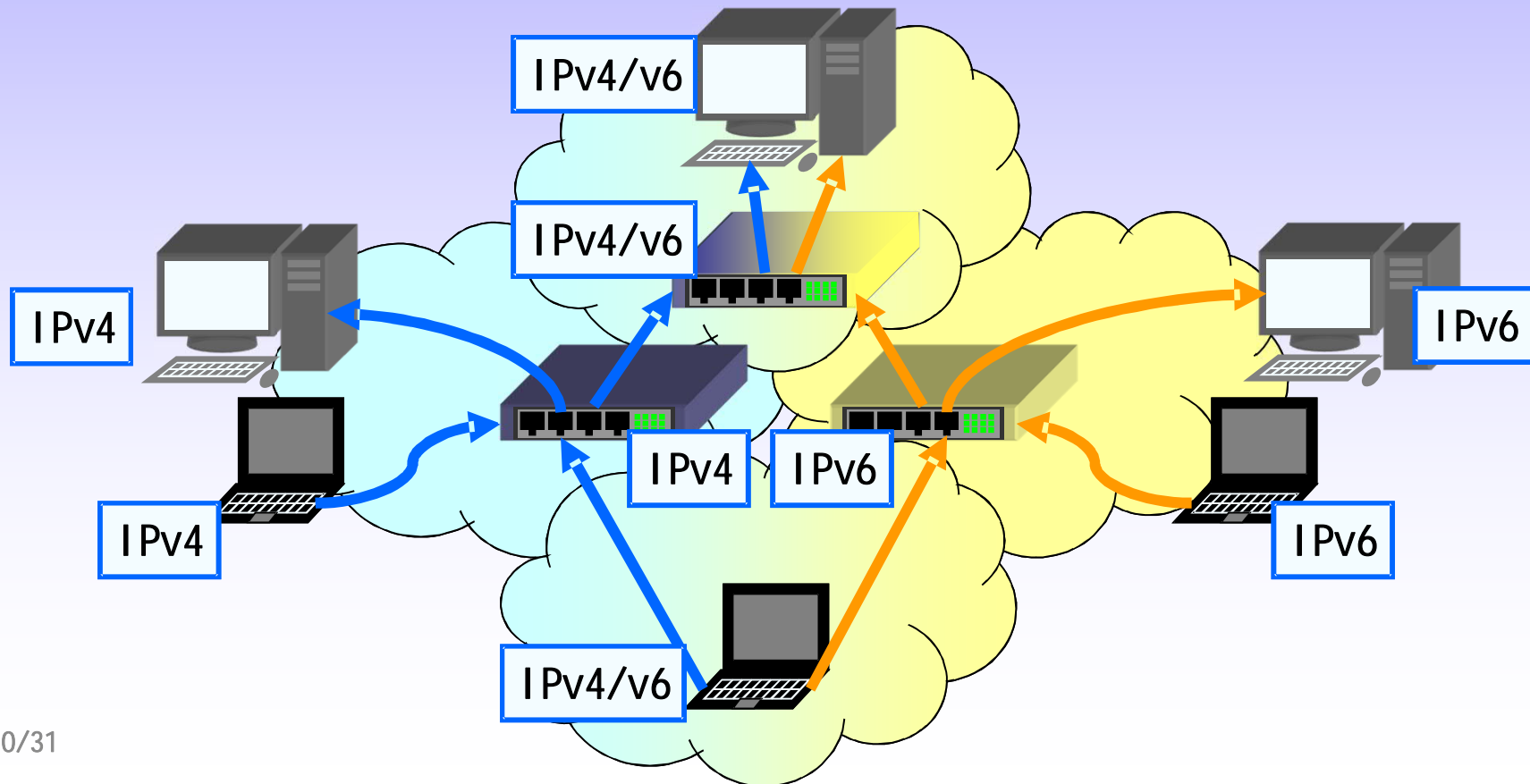


隧道过渡方式的示意图



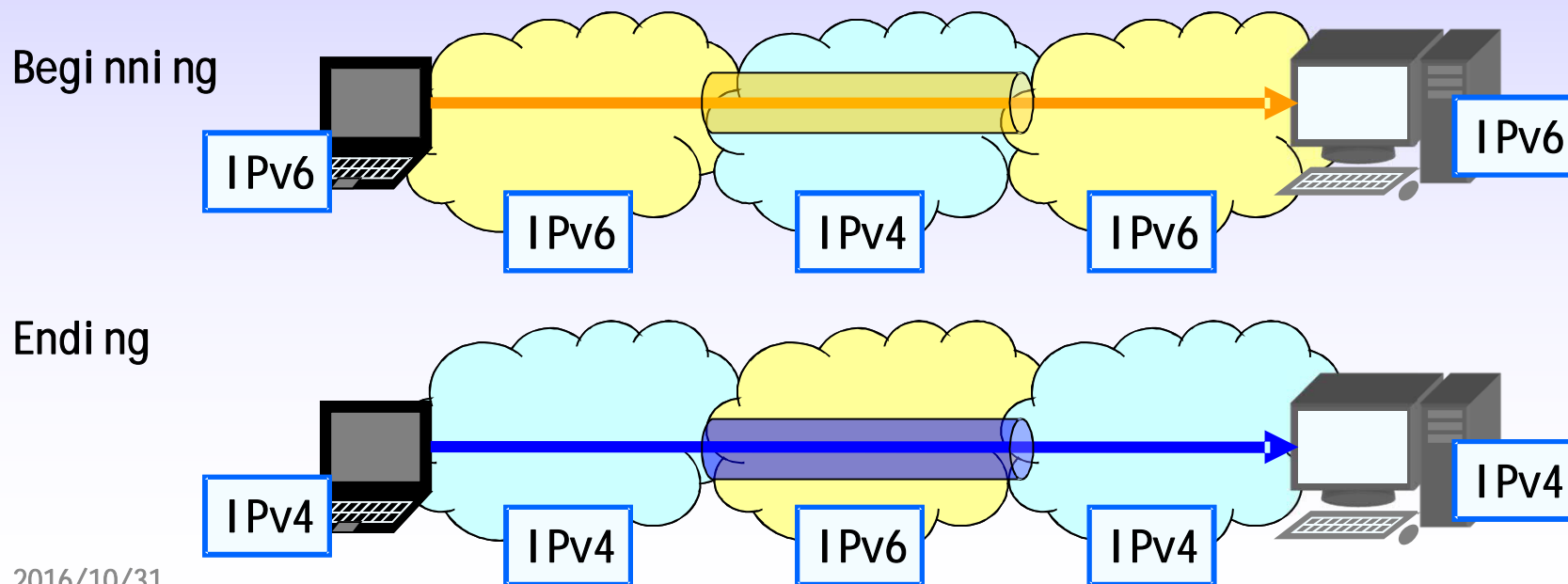
双栈 (Dual Stack)

- ◆ IPv4/IPv6 can be used
- ◆ Server/Router/Client
- ◆ Until No IPv4 nodes are available



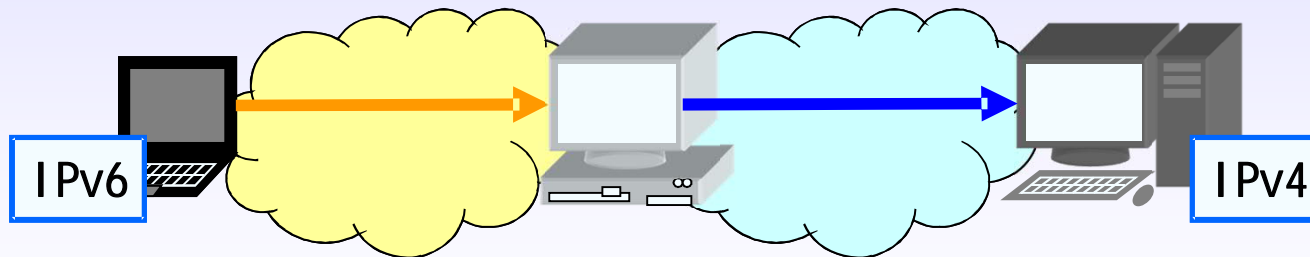
隧道 (Tunneling)

- ◆ IPv6 network tunnels through IPv4 network (IPv4 network tunnels through IPv6 network)
- ◆ Encapsulation mechanism



转换器 (Translator)

- ◆ To communicate IPv4 only supported host to IPv6 only supported host
- ◆ NAT, SOCKS, Layer realization



问题？

- ◆ 已知华中科技大学(总面积4517542平方米=约450万平方米=4.5平方公里)分配到的IPv6地址是
 - ♣ 2001 : 0250 : 4000::/48
- ◆ 请问这是一个何种类型的 Unicast / Multicast地址？其相应FP/TLA ID/RES/NLA ID/SLA ID/接口ID分别是多少？
- ◆ HUST是一个TLA ID或 NLA ID或SLA ID机构？
- ◆ HUST所分得的地址空间相当于v4的多好个A类地址？该地址占整个IPv6地址空间的比例是多大？解答
 - ♣ 0010 0000 0000 0001 : 0000 0010 0101 0000 : 0100 0000 0000 0000 ::/48
 - ♣ $2^{128-48} = 2^{80} \approx (2^{10})^8 = (10^3)^8 = 10^{24}/450\text{万m}^2 = 2.22 \times 10^{17}/\text{m}^2$
 - ♣ $2.22 \times 10^{17} / 6.02 \times 10^{23} \approx 0.368 \times 10^{-6} = \text{百万分之一的3分之一摩尔数}$
 - ♣ $2^{80} / 2^{24} = 2^{56} \approx (10^3)^{5.6} = 10^{16.8} \text{个A类}$
 - ♣ $2^{80} / 2^{128} = 1/2^{48} \approx 1/10^{14.4} = \text{百万亿分之一}$