IP VERSION 6 (IPV6)

Mario Baldi
www.baldi.info

Copyright Notice

Outline

大纲

- → A new version of IP: why?
- → Addresses
- → Modified protocols
- → Socket programming interface
- → Packet header format
- → Neighbor discovery
- → Transition to IPv6(?)

A NEW VERSION OF IP: WHY?

Why a new IP?

Only one true answer

A larger address space

Other answers

→ More efficient on LANs
→ Multicast and anycast
→ Security
→ Policy routing
→ Plug and play
→ Traffic differentiation
→ Mobility
→ Quality of service support

Ported
to IPv4

A long way to IPv6 adoption

- Long time for defining IPv6 and migrating to it 长时间定义IPv6 和迁移
- Problems needed an interim solution in IPv4 临时
- When IPv6 reached "production" stage, many IPv4 solutions were acceptable

Why are IPv4 addresses scarce?

32 bit long

About 4 billion addresses!!!

however ...

Only part of the addresses are assigned to stations

→ Class A, B and C

→ Addresses beginning by bx111 are used for multicast and else

Hence, “just” 3.5 billion addresses can be used!!!

They are used hierarchically

- →The prefix used in a physical network cannot be used in a different one
- →Lots of unused addresses

Interim (IPv4) solutions to the saturation of address space

对于地址空间饱和，临时的解决方法

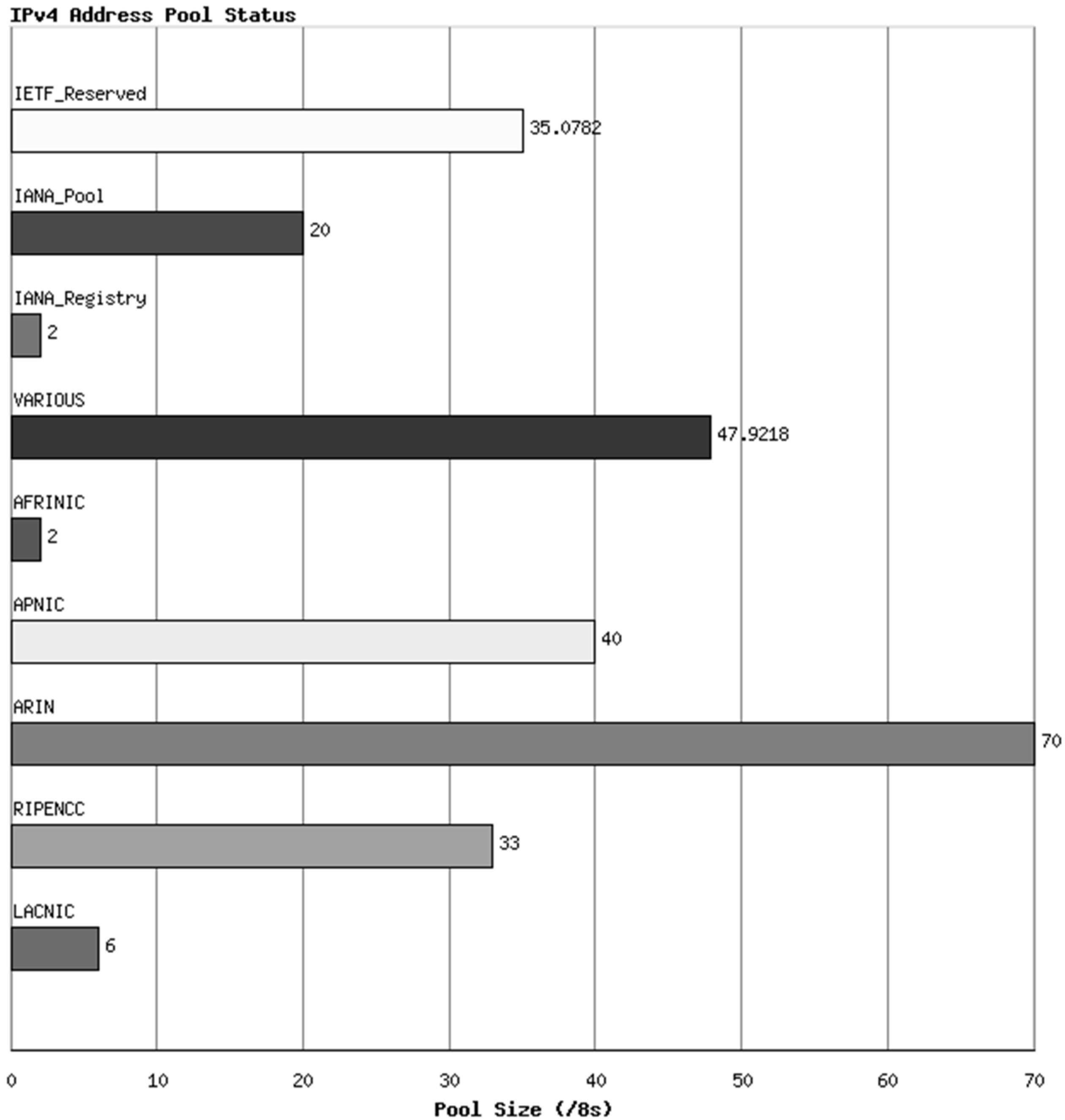
- Introduction of network with "taylored" size
  - Netmask 掩码
- Private addresses 私有地址
  - Intranet, RFC 1918 内联网
  - Not enough to solve the problem NAT：网络地址转换 ALG：应用层网关
    - It should be used in conjunction with NAT or ALG
- Network Address Translator (NAT)
  - Extremely popular 特别流行
  - Proposal for RSIP (Realm Specific IP) 特定区域IP协议
- ALG (Application Level Gateway)

Has all of this been effective?

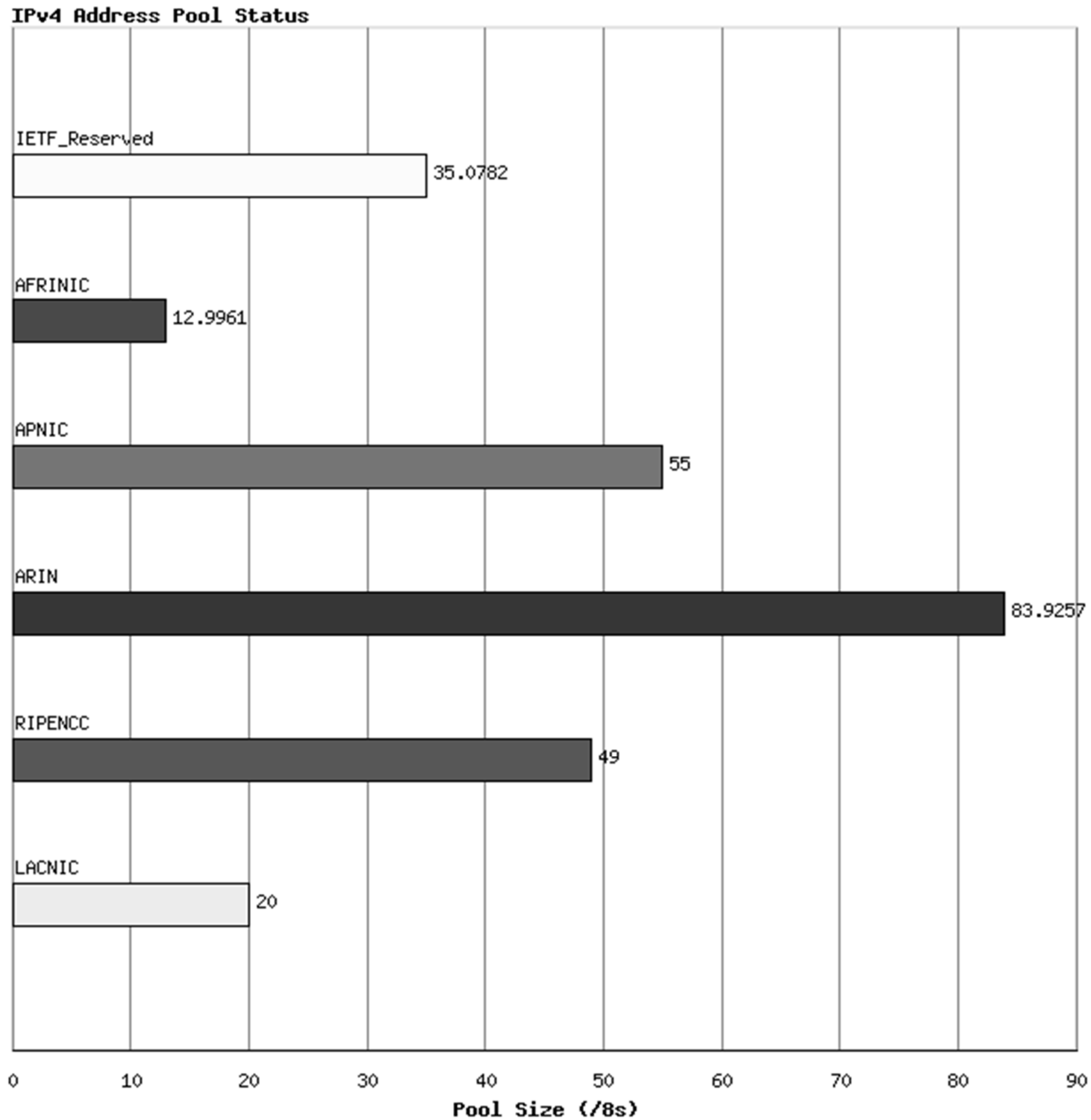
Agencies assigning addresses

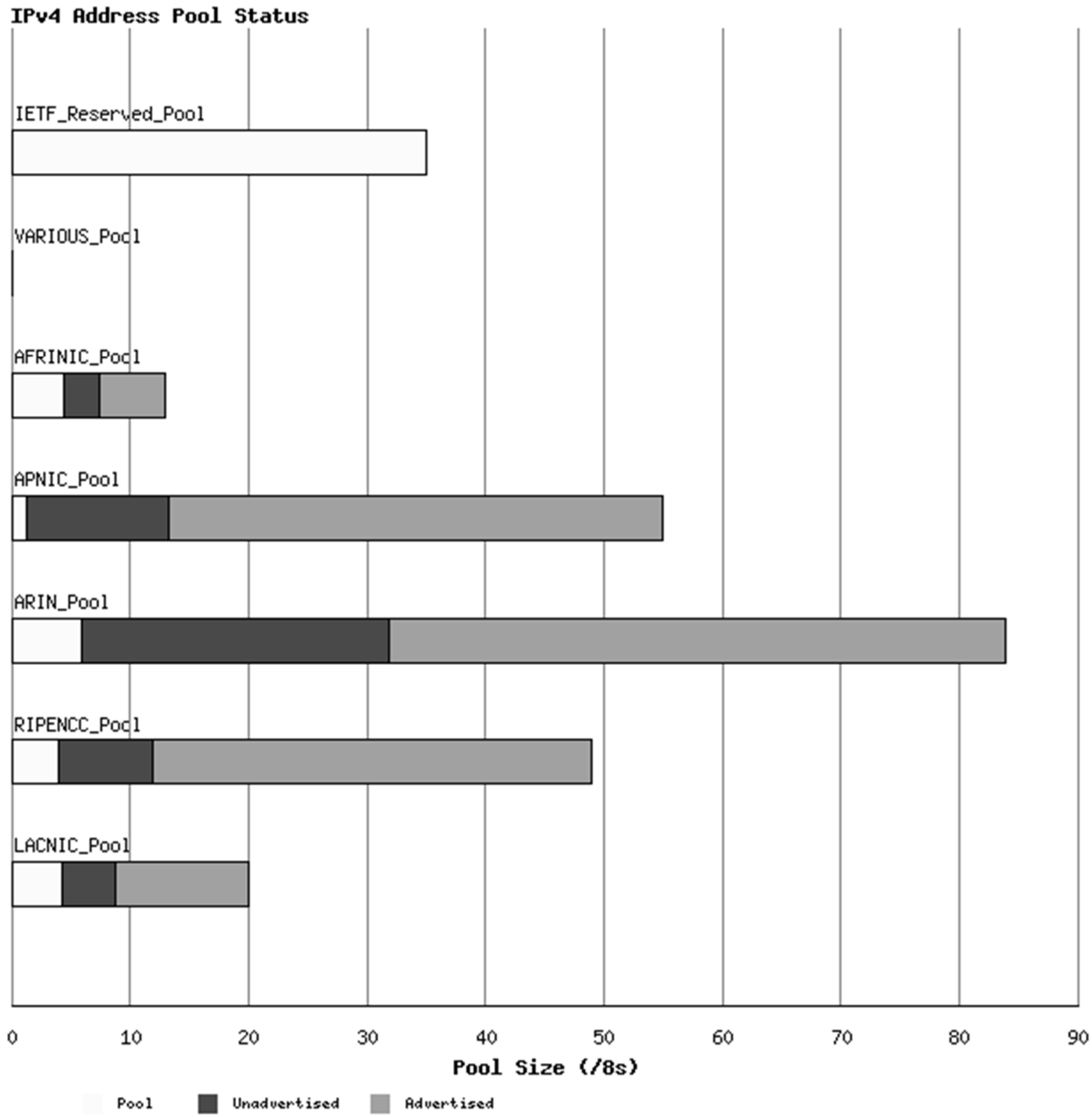
IANA distributes (better: distributed) /8 IPv4 network prefixes to regional registries

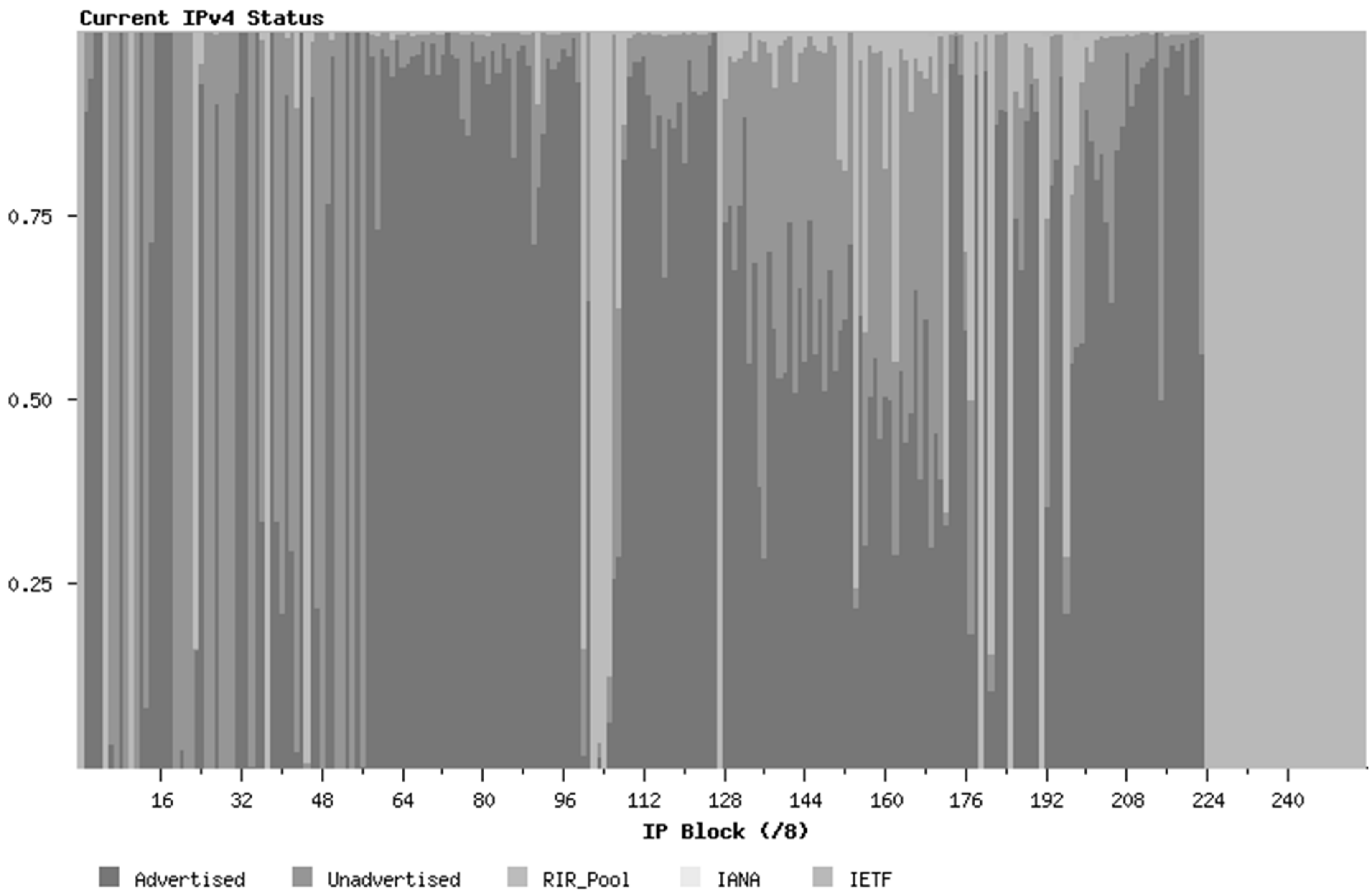
Situation (2010)



Situation (2011)




IPv4 Address Pool Status
IETF_Reserved_Pool
VARIOUS_Pool
AFRINIC_Pool
APNIC_Pool
ARIN_Pool
RIPENCC_Pool
LACNIC_Pool
0
10
20
30
40
50
60
70
80
90
Pool Size (/8s)
Pool
Unadvertised
Advertised



Current IPv4 Status
0.75
0.50
0.25
16
32
48
64
80
96
112
128
144
160
176
192
208
224
240
IP Block (/8)
Advertised
Unadvertised
RIR_Pool
IANA
IETF

路由扩展性问题

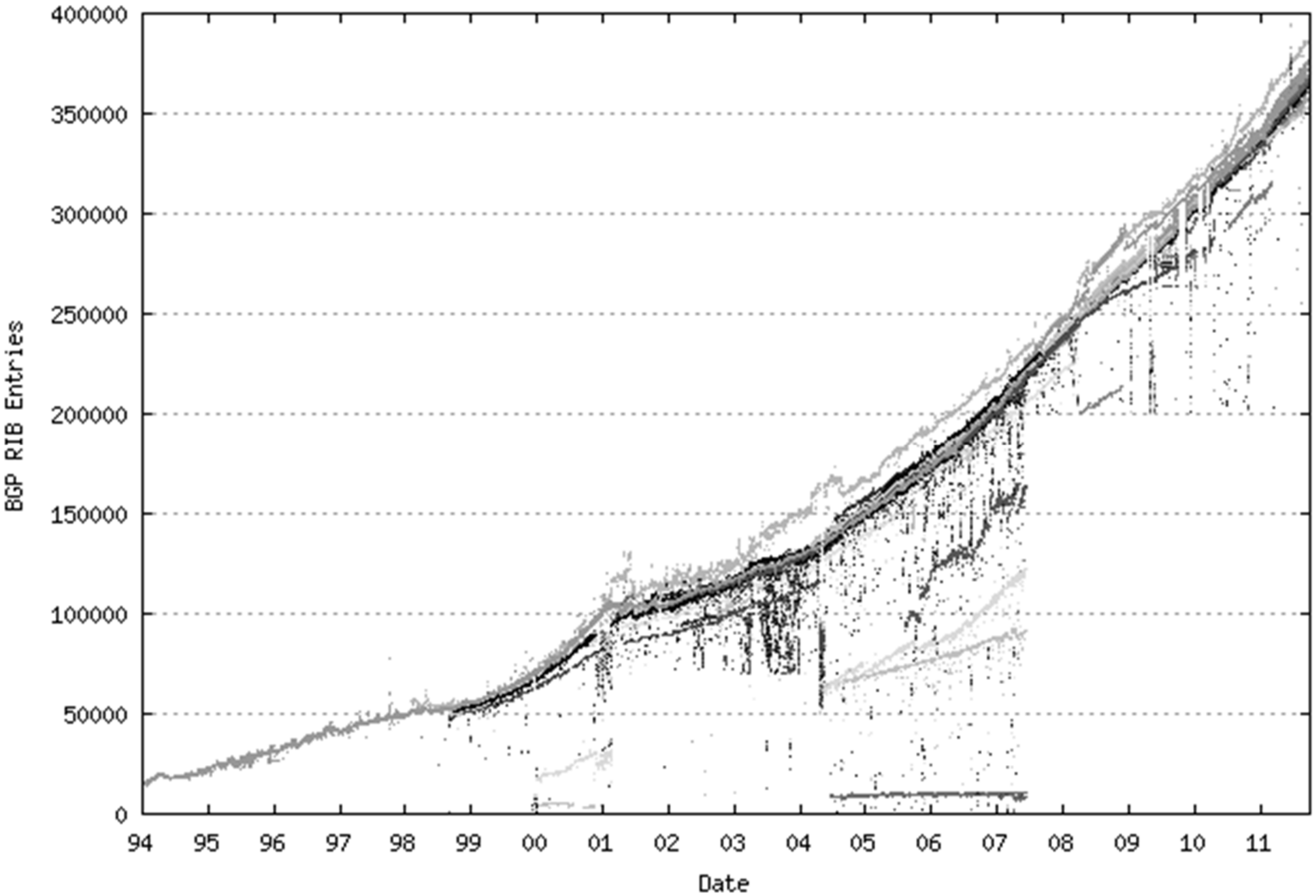
Routing scalability issues

- Routing table size 路由表大小
  - Internet size
  - Each subnetwork must be advertised
- Problems 问题
  - Router resource limitations 路由器资源限制
    - Too much information to manage 太多的信息去管理
  - Routing protocol limitations 路由协议的限制
    - High probability of route changes 路由改变的可能性非常高
  - Mainly affecting backbone routers

主要影响主干路由器

Routing scalability issues

http://bgp.potaroo.net/



Isn’t there a solution with IPv4?

聚集多个线路为一个

- Aggregate multiple routes in one
  - Shorter prefix including others 更短的前置，用来包括其他
  - 1.2.1.0/24, 1.2.2.0/24 …
  - 1.2.0.0/16
- CIDR (Classless Inter-Domain Routing)

无类别域类间路由

- Limited by non-rational assignment of IP prefixes

Interim (IPv4) solutions to routing scalability

- CIDR
  - Classless Inter-Domain Routing
- Limiting the assignment of IP addresses
  - Regional Internet Registry: assign address blocks only to big players 只分配给大的玩家地址块
  - E.g., minimum /20 (4096 addresses) network
- Scalability of routing protocols 路由协议的可扩展性
  - With no solution, at present
- Problem not completely solved
  - It is the major problem that IPv6 wanted to solve that it is still open

Birth of IPv6

- IETF Boston Meeting (1992), “Call for proposals”
  - Appointment of dedicated Working Groups
- Several proposals
  - TUBA: adopting OSI CLNP as new IP
  - CATNIP: integration of different network (IP, CLNP, IPX) and transport (TP4, SPX, TCP, UDP) protocols
  - SIPP: incremental over IPv4
    - Fix some drawbacks
    - Simple: increasing the address field and eliminating unused ones
- Winning proposal: SIPP with 128 bit addresses

ADDRESSES

So, how many addresses should IPv6 have?

→ A scientific approach

→ Addressing efficiency

$$H = \frac{\log_{10}(\text{number of addresses})}{\text{number of bits}}$$

Addressing Efficiency

- → In existing networks
  - → H varies between O.22 and O.26
- → Assuming one million billion networked stations
  - → 68 bits in the minimum efficiency case

Melius abundare quam deficere

128 bits
(16 bytes)

655.57O.793.348.866.943.898.599 IPv6 addresses per sqm of Hearth surface

Notation

8 hexadecimal numbers separated by
“:”

Groups of 2 bytes

FEDC:BA98:0876:45FA:0562:CDAF:3DAF:BB01

1080:0000:0000:0007:0200:A00C:3423:A089

Shortcuts

Leading Os in each digit group can be omitted

→ 1O8O:O:O:O:7:2OO:AOOC:3423

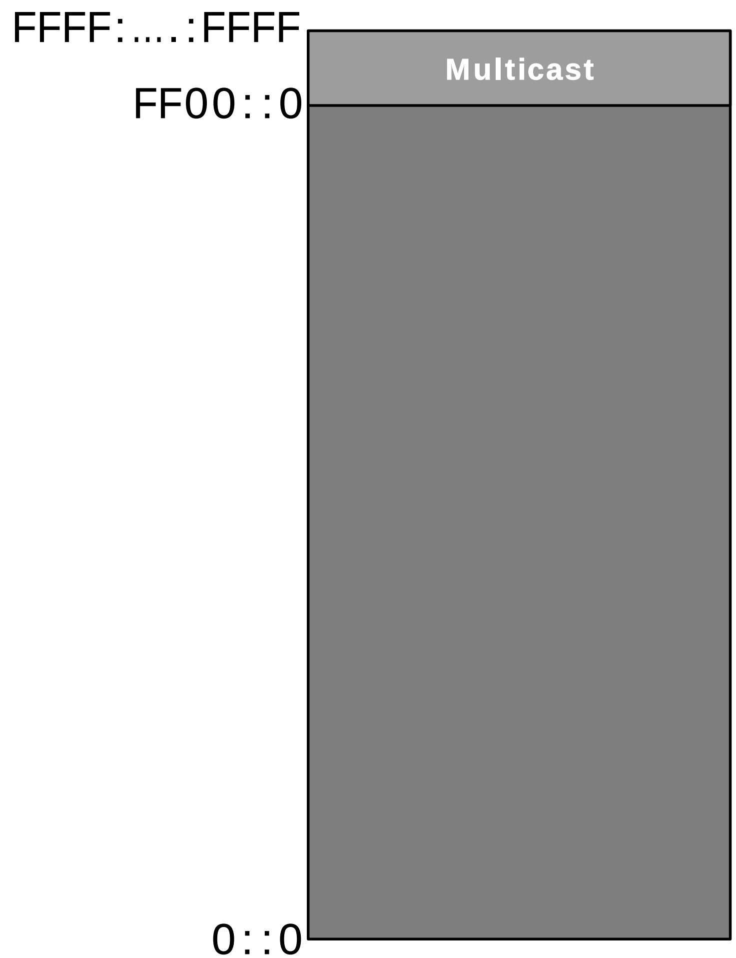
Groups of Os can be substituted by “::”

→ 1O8O::7:2OO:AOOC:3423

→ Not more than once

Addressing Space Organization

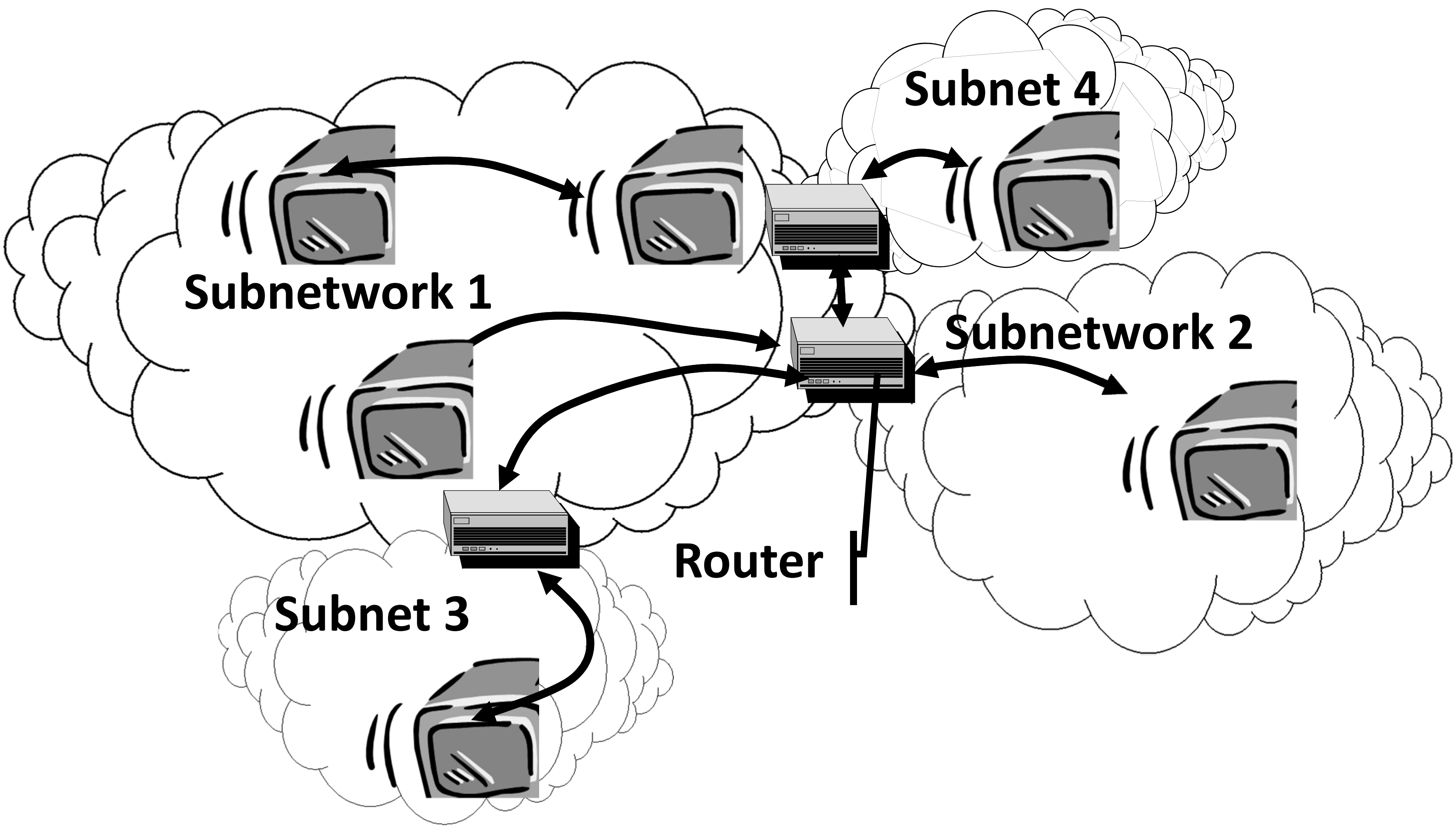
- → Multicast
  - → 1111 1111
  - → FF00::/8
    - →FFxx:xx…



HOST ADDRESSES

Routing and Addressing Principles

Same routing principles as IPv4



Address Structure

n = 64

Same Address Assignment Principles as IPv4
(different terminology)

不同的术语

- → Sub network: set of hosts with same prefix
- → Link: physical network

Subnetwork ≡ link

on=link hosts 拥有相同的前缀

- On-link hosts have same prefix
  - Communicate directly

直接交流

- Off-link stations have different prefix
  - Communicate through a router

Prefix

Address/netmask pair
is substituted by a “Prefix”

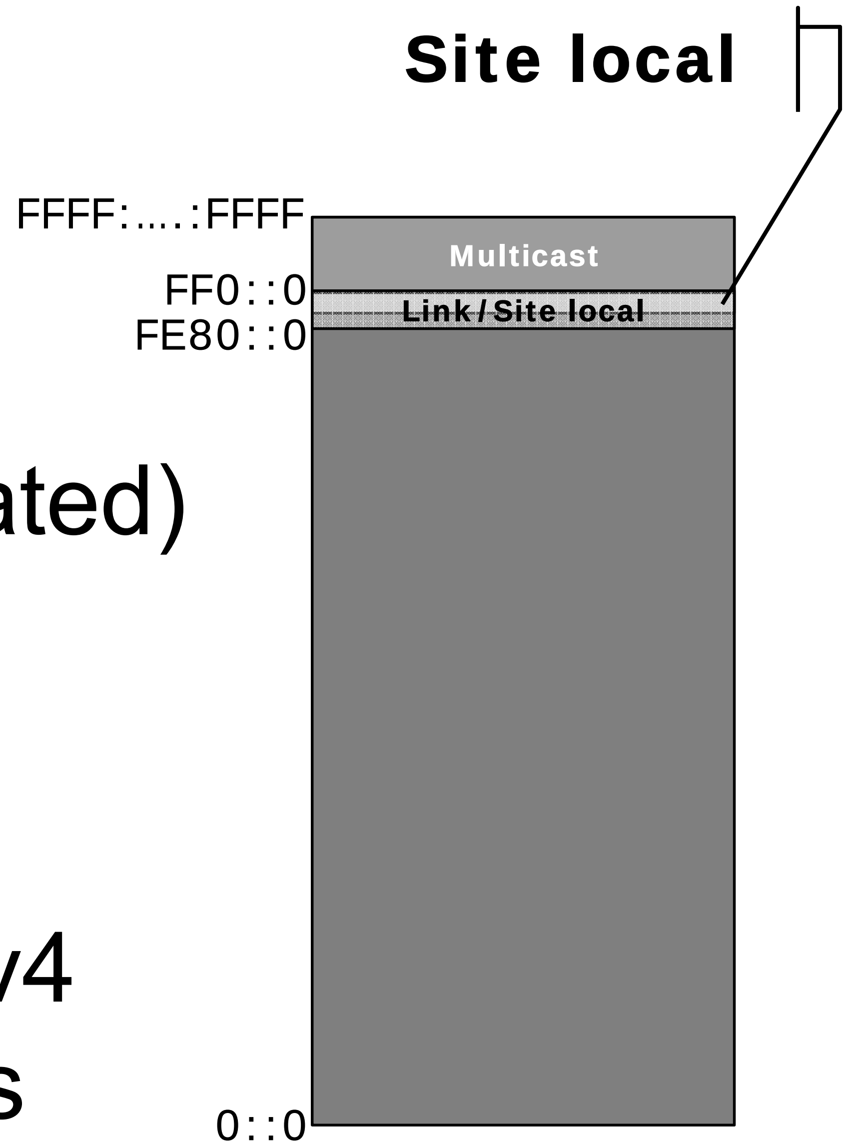
Address/N, where N is the
prefix length [bit]

→ FEDC:0123:8700::/36

→ 1111111011011100
000000010010000111000

No address classes

- Link local/site local
  - 1111 111O 1
- Link local
  - 1111 111O 1O
  - FE8O::/64
- Site local (deprecated)
  - 1111 111O 11
  - FEC0::/1O
  - FE[C-F]…
  - Equivalent to IPv4 private addresses



弃用

Why Deprecated?

- Overlapping private address spaces
- Not a problem in principle, but in practice

  理论上没有问题，但实际应用中有问题

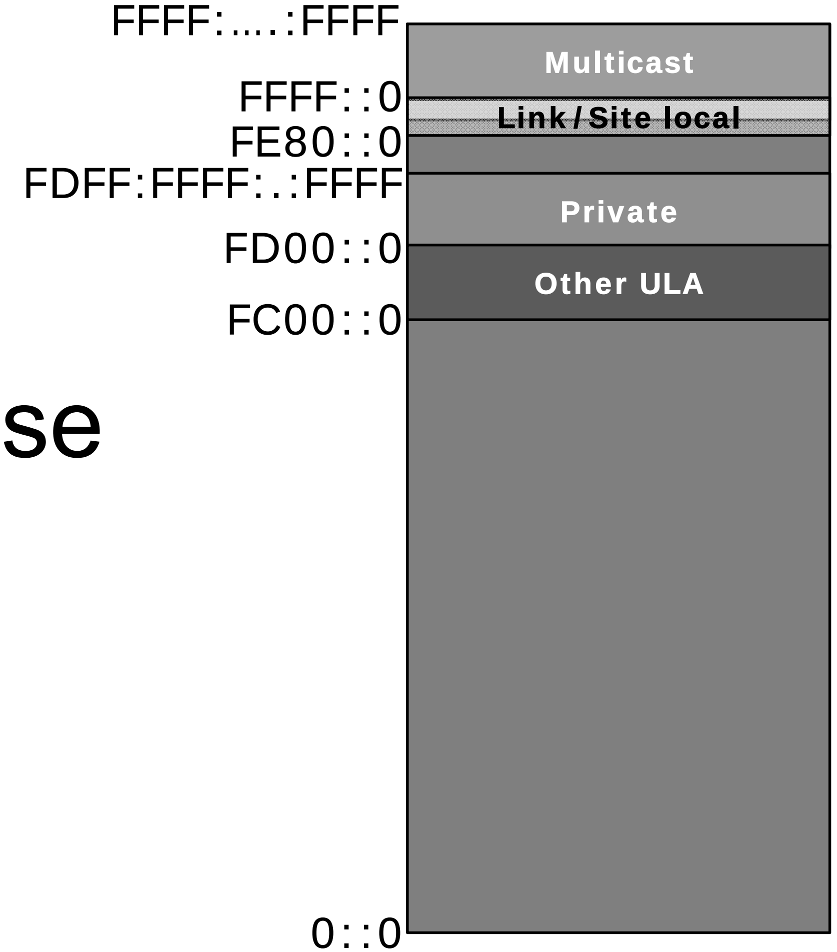
  - Extranets

    外联网

  - Mergers and acquisitions

Private Addresses

- → Unique Local Addresses (ULA)
  - → FC00::/7
  - → 1111 110
- → FC00::/8 for future use
- → Private addresses
  - → 1111 1101
  - → FD00::/8



Local Unicast Addresses

private

8	40	16	64
1111-1101	**randomly generated**	**Subnet ID**	**Interface ID**
FD			

site local

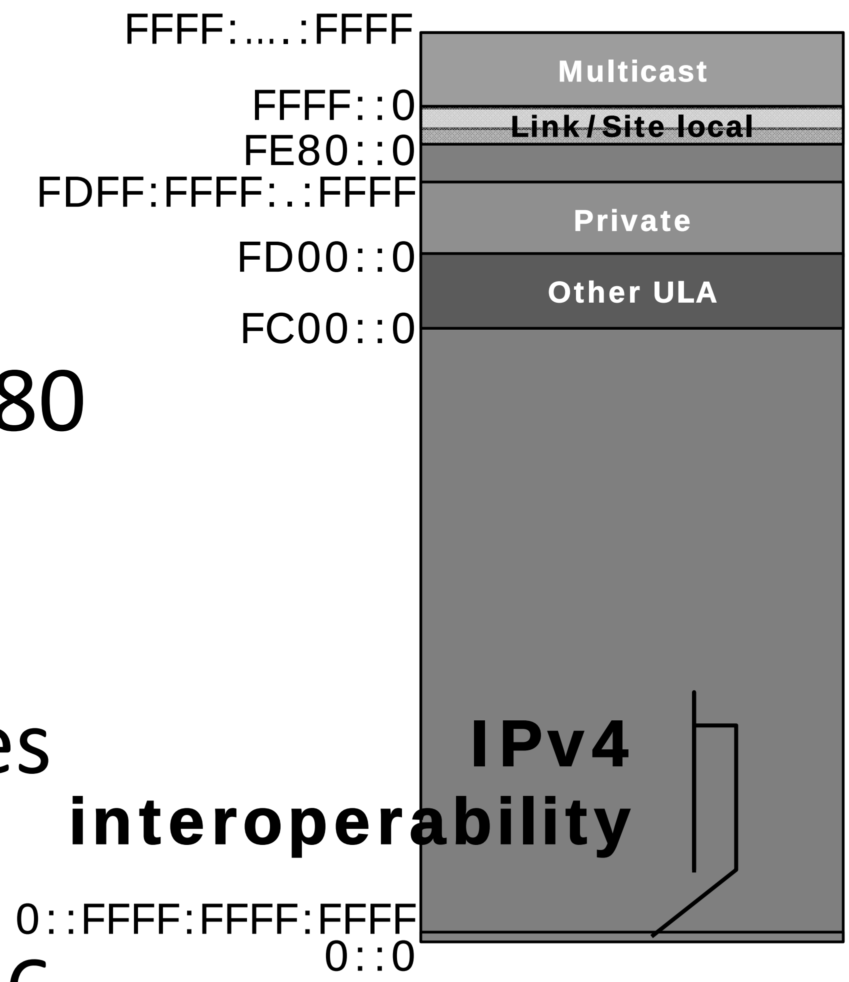
10	38	64
1111-1110-11	**any**	**Interface ID**
FE[C-F]x		

link local

10	54	64
1111-1110-10	**0**	**Interface ID**
FE80		

Remaining addresses
Global Unicast

Global Unicast Addressing Space Organization



互用性

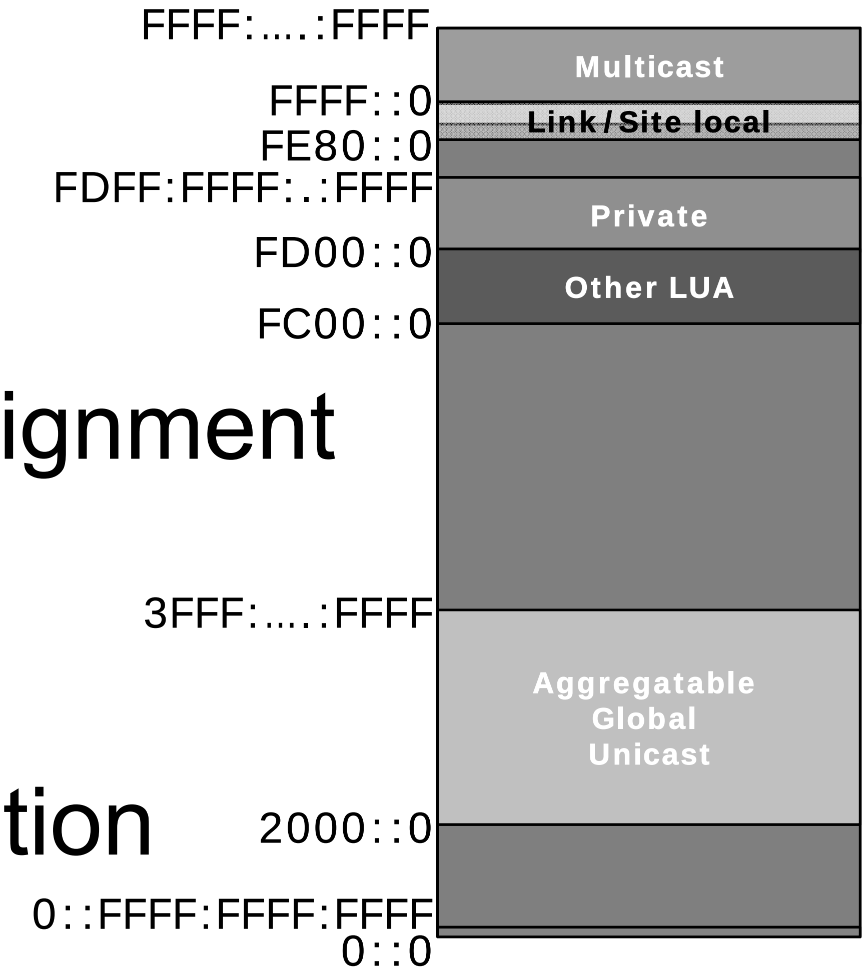
- → IPv4 interoperability
  - → 0...0 (80 bit) → 0::/80
  - → To be used during transition phase
- → IPv4-mapped addresses
  - → 16 bits set to 1
    - → 0:0:0:0:0:FFFF::/96

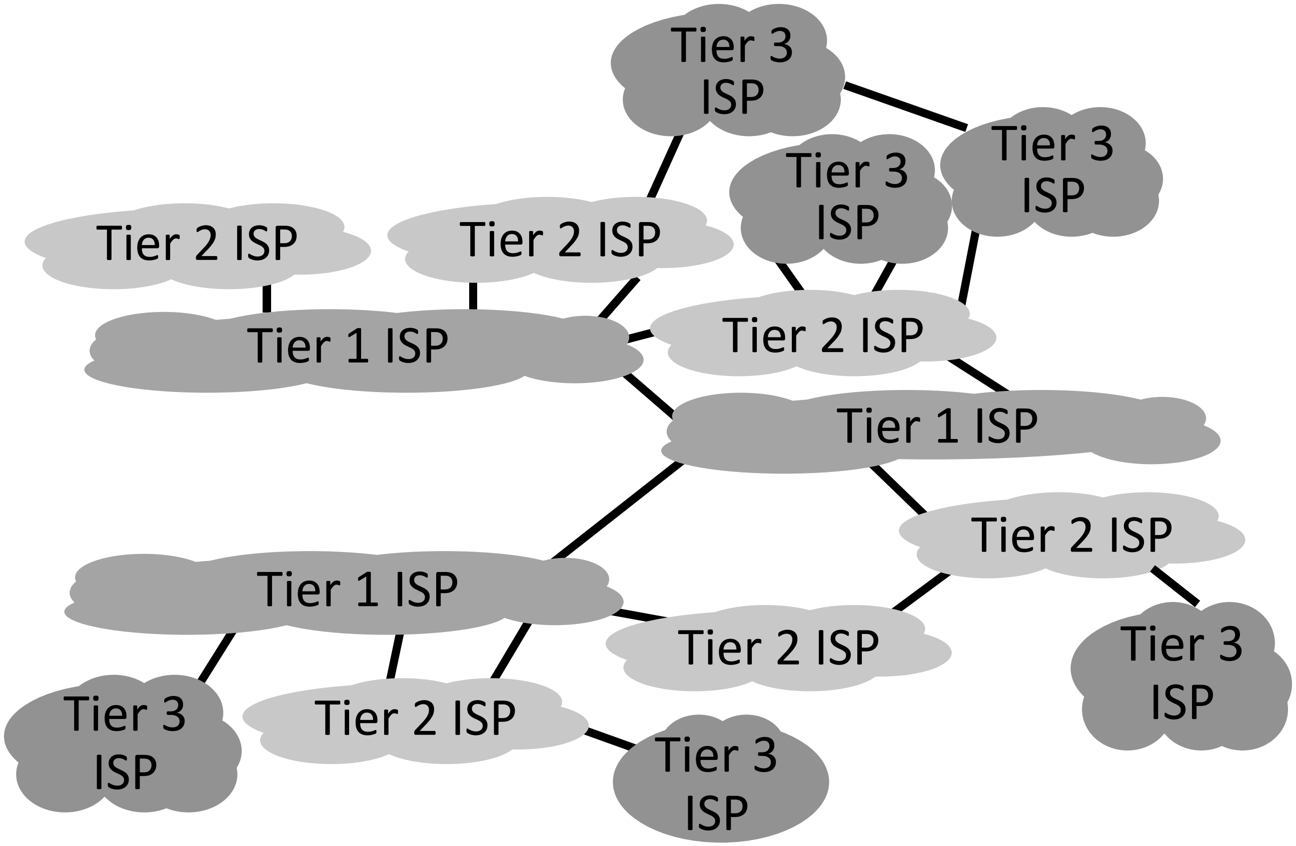
- → IPv4-compatible
  - → Another 16 bits to 0 → 0::/96

    E.g. 0:0:0:0:0:0:A00:1
- → Compact notation
  - → ::A00:1
- → Special notation
  - → ::10.0.0.1

Aggregatable Global Unicast

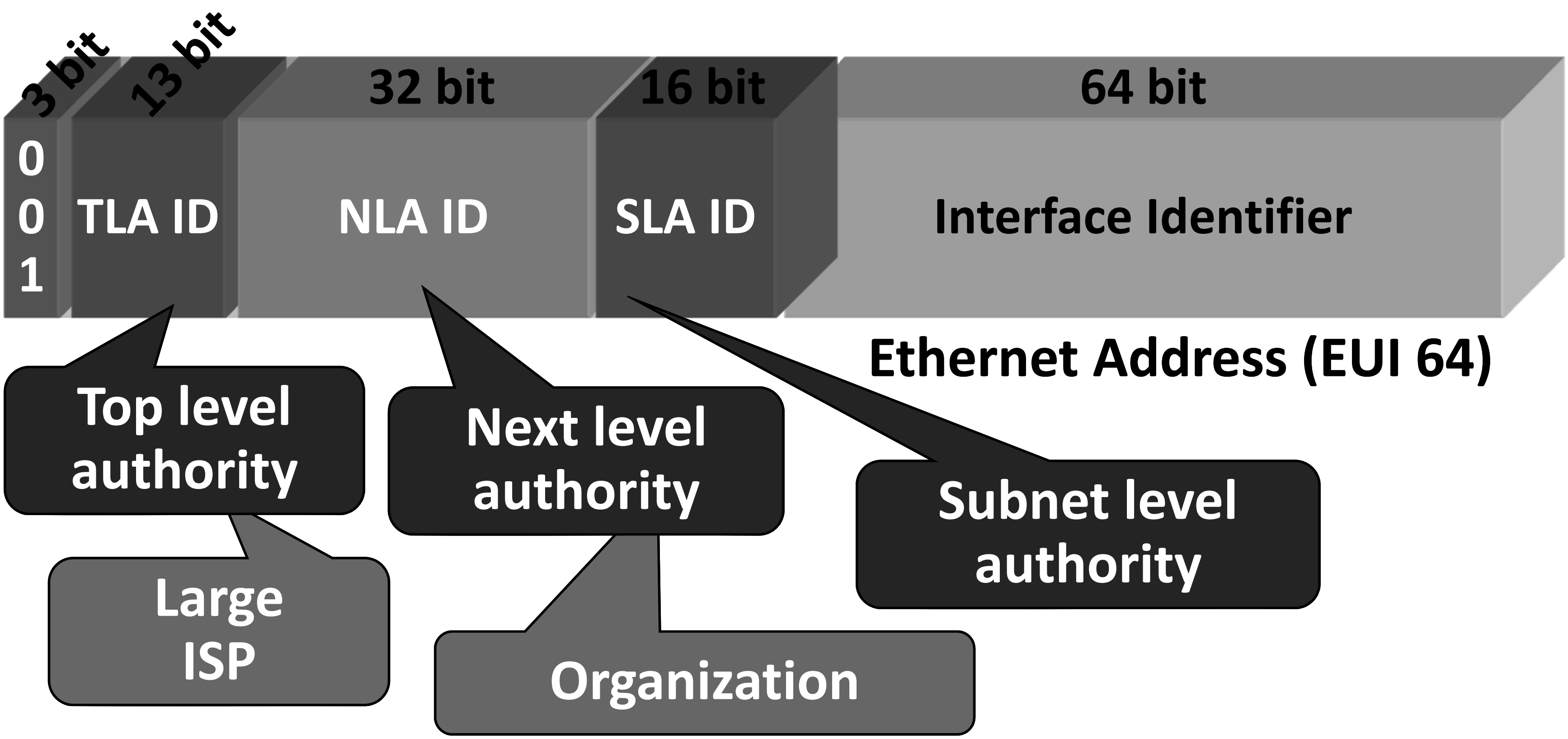
- → Begin with bxOO1
  - → [2-3]…
- → Topology-based assignment
  - → Service provider hierarchy
  - → Effective aggregation





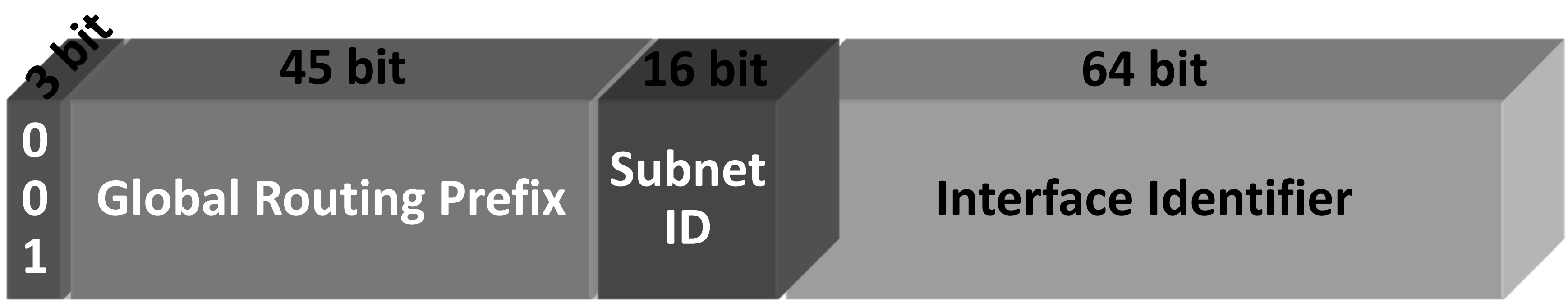
Different assignment criterion for other addresses

Address Assignment



Automatic renumbering

More Flexibility



2001::/21 – 2001::/23 delegated to registries

Plug and Play

Scenarios 场景

→ Dentist Office

→ Thousand computers on the dock

Solution: autoconfiguration

→ Stateless: no server needed

→ Statefull: DHCP server

Special Addresses

- ➔ Loopback address ::1
- ➔ All nodes (multicast) address FF02::1
- ➔ All routers (multicast) address FF02::2
- ➔ Unspecified address ::

MODIFIED PROTOCOLS

What changes in the protocol architecture?

- → IP
- → ICMP
- → ARP
  - → Integrated in ICMP
- → IGMP
  - → Integrated in ICMP

Upgraded But Not Changed

- → DNS (type AAAA record)
- → RIP and OSPF
- → BGP and IDRP
- → TCP and UDP
- → Socket interface

What about layer independence?

SOCKET PROGRAMMING INTERFACE

What is it?

→ Programming interface for TCP/IP services

→ Used in application implementation

  →UDP messages

  →Bytes on TCP connections

Underlying Principles

- → Originated in Unix Environment
  - → I/O as file access
- → Socket descriptor equivalent to a file descriptor for network use

Socket

- → Point of access to network services
- → Associated to TCP connection or UDP session

Socket Operations

- → Wait for connection requests on a port
  - → Server
  - → `listen()`
- → Accept requests (server)

- Connect to a port of a remote server
  - Client
  - Requires specifying address and port

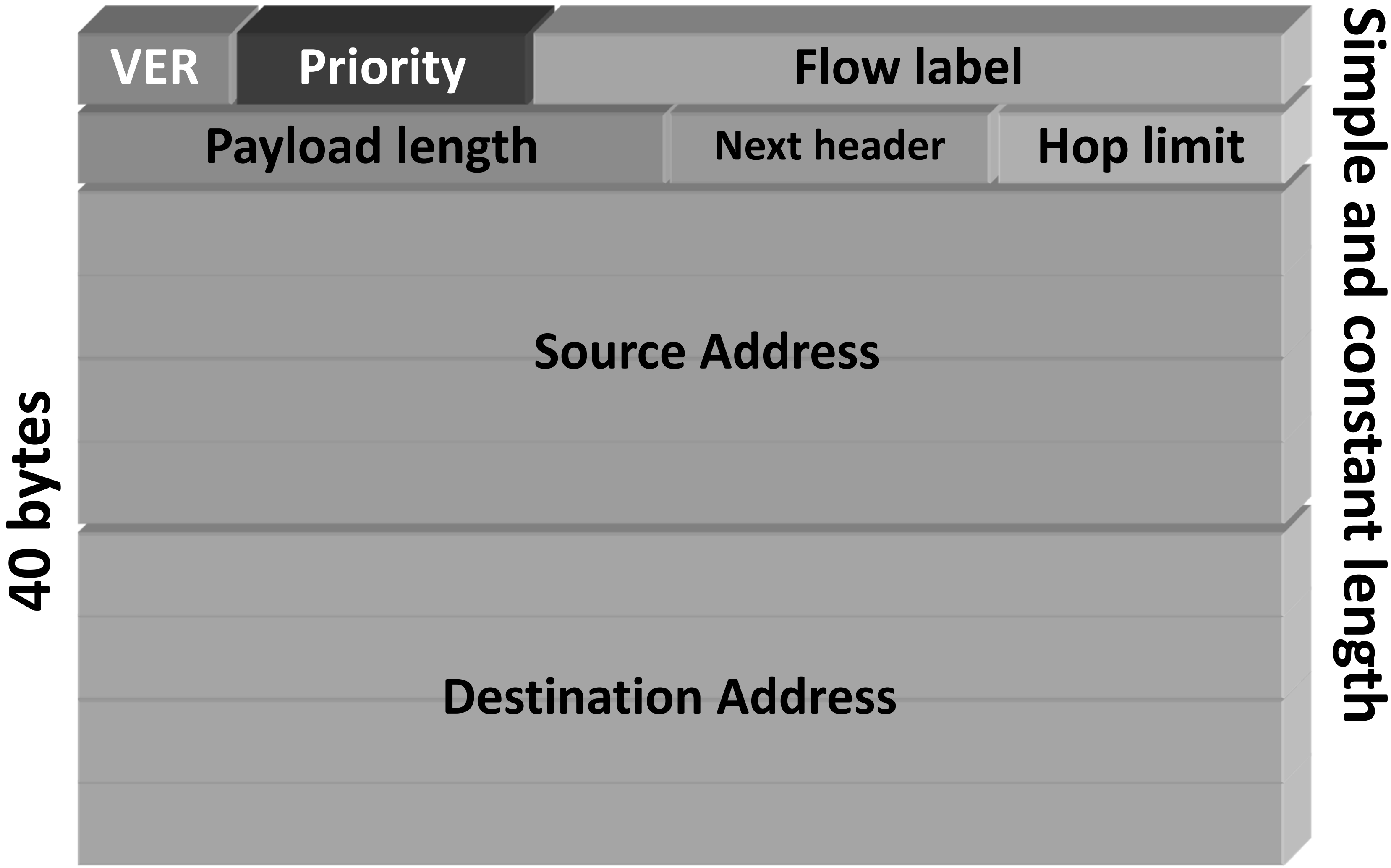
指定，说明

- Send and receive data

PACKET HEADER FORMAT

Do You Remember the IPv4 Header?

Here is the IPv6 One



Field Removal

- → Not very useful
  - → Checksum
- → Not used in each packet
  - → Fragmentation
- → No longer needed
  - → Header length

Extension Headers

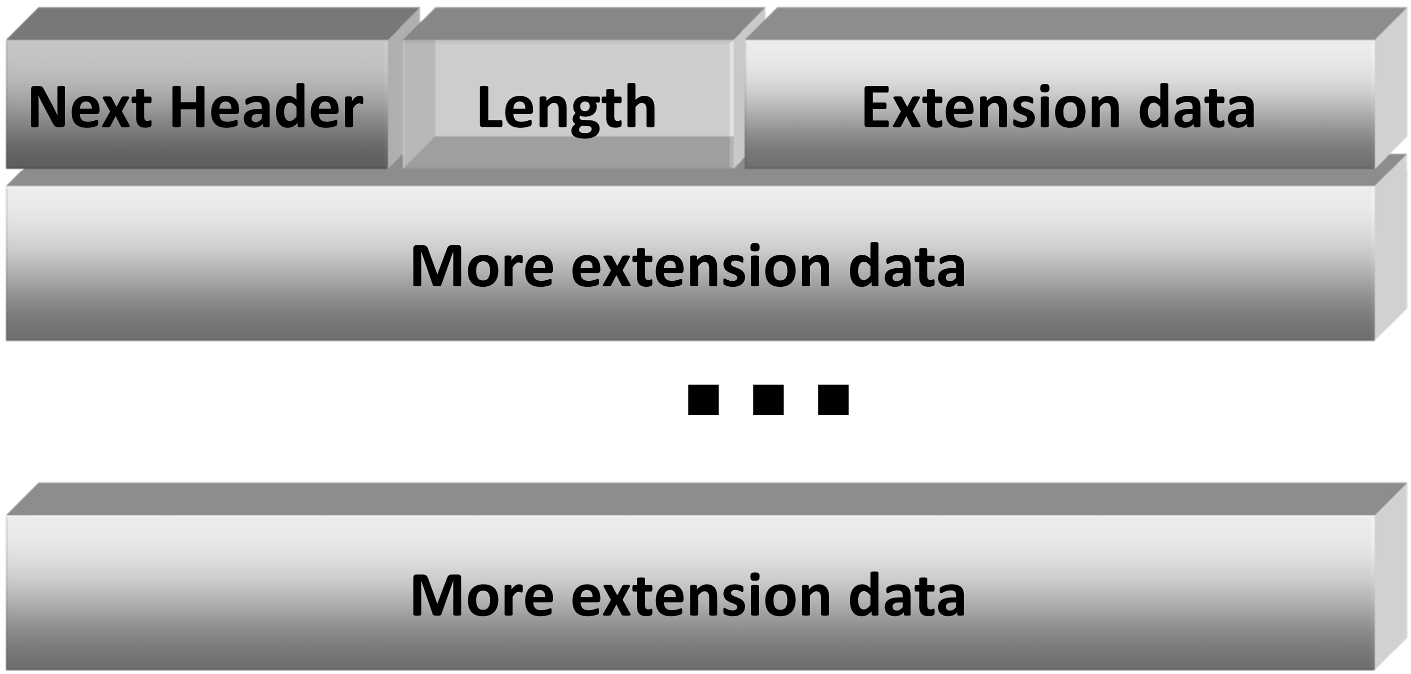
→ Added when useful

→ Not needlessly processed in each packet

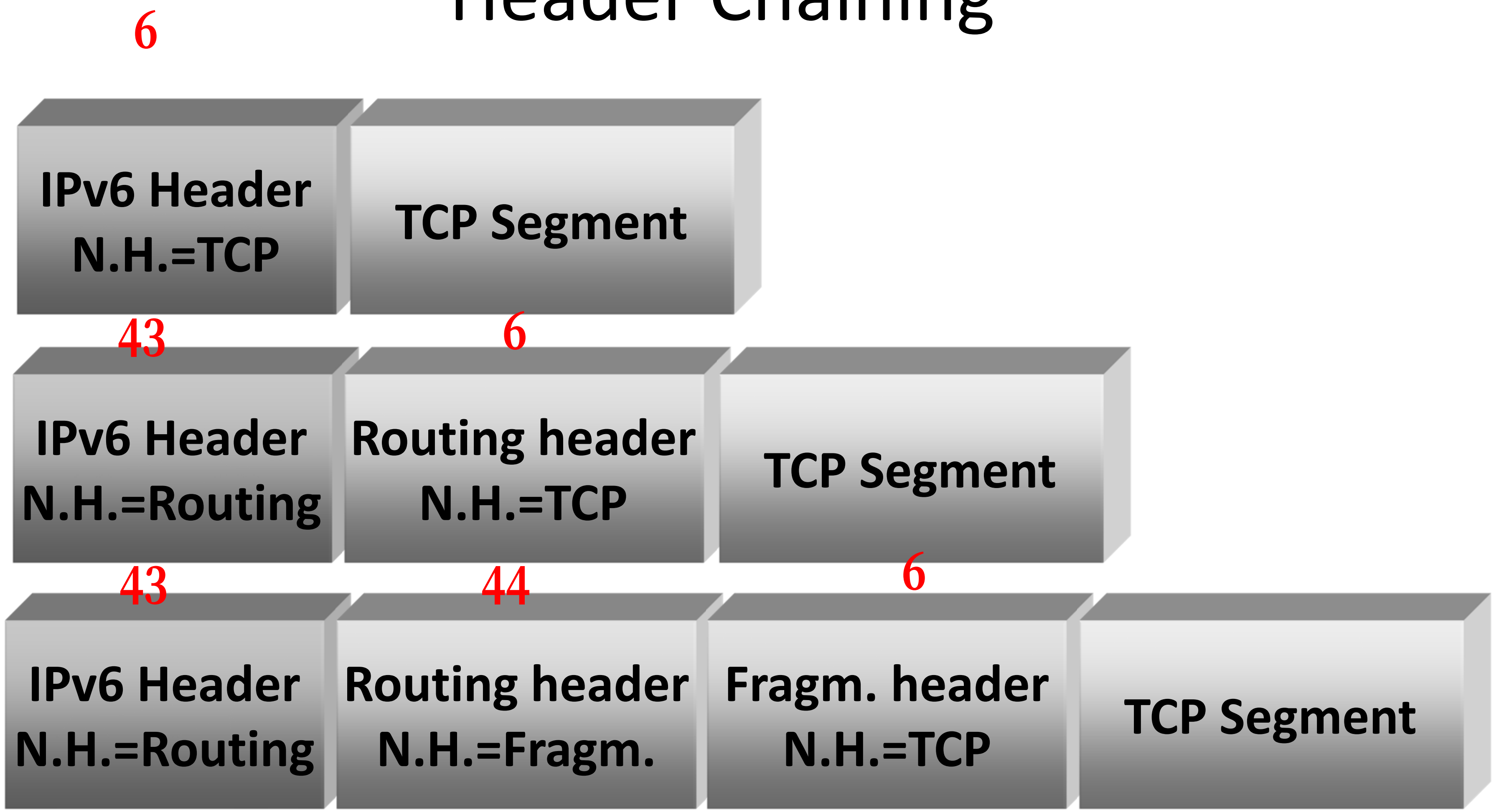
Extension Headers

- → Hop By Hop Option
- → Routing
- → Fragment
- → Authentication
- → Encrypted Security Payload
- → Destination Option

Extension Header Format

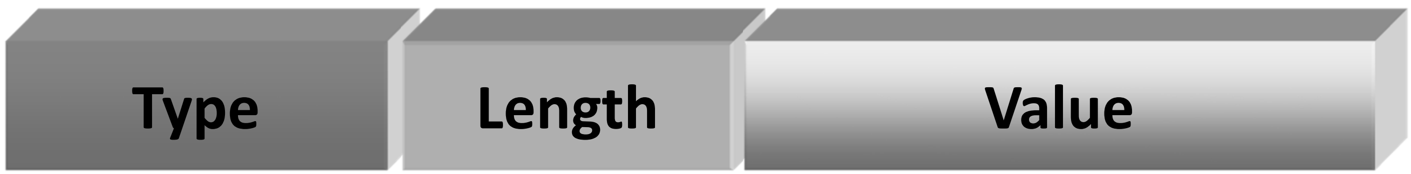


Header Chaining

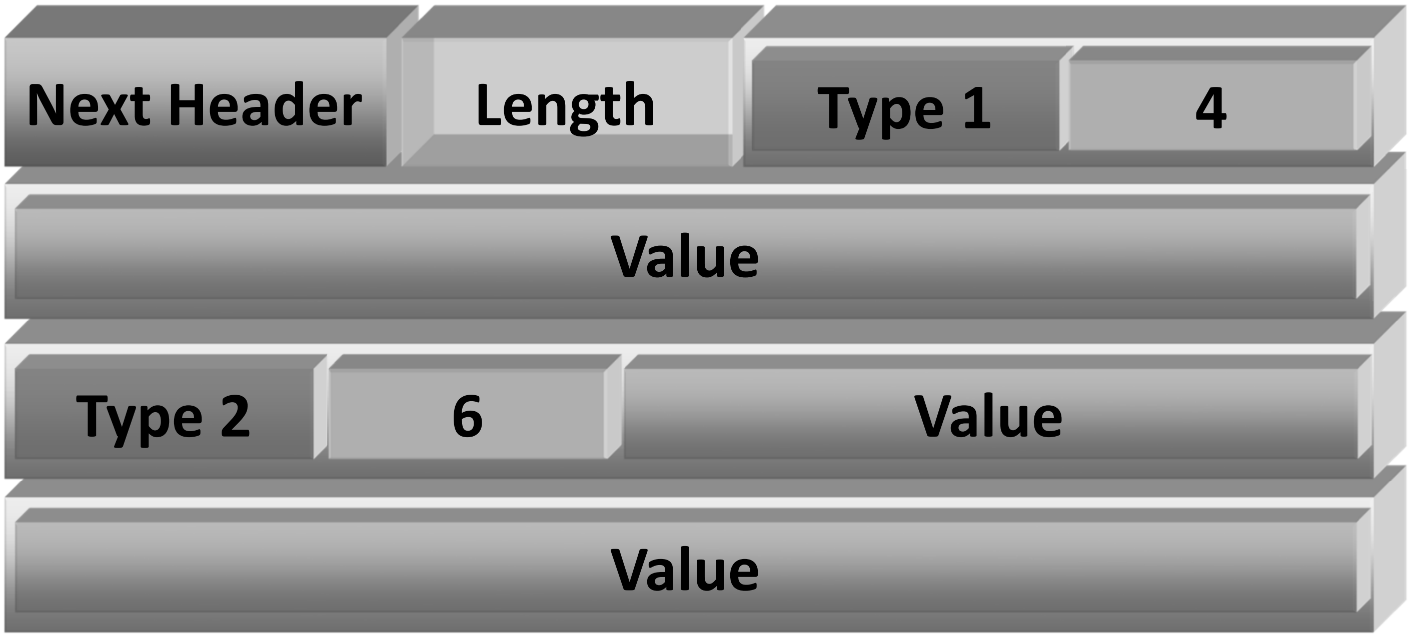


Options

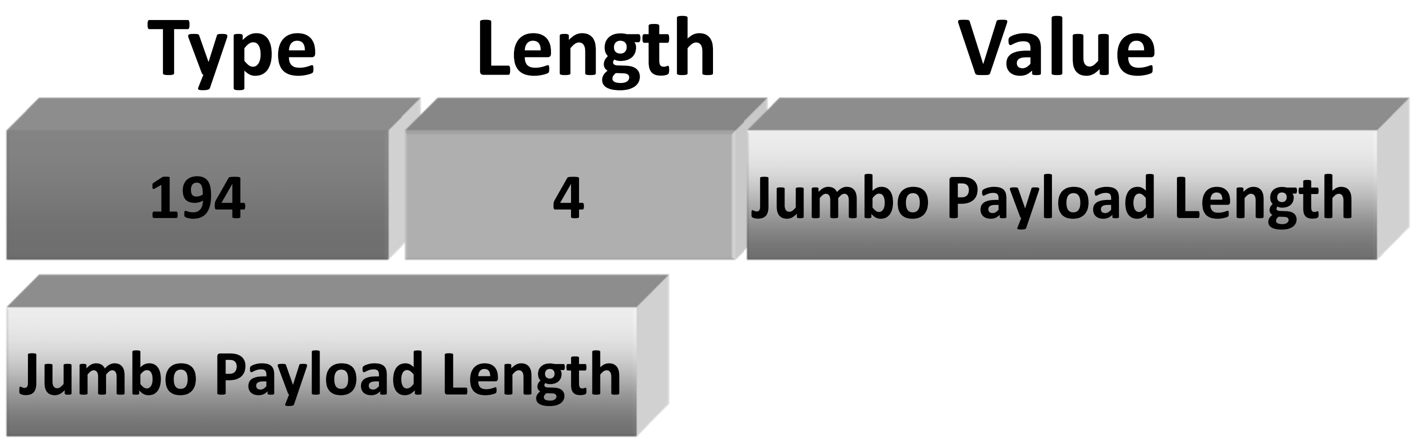
- ➔ To be used in Hop-by-hop and Destination Option Extension Headers
- ➔ TLV format
  - ➔ Type - Length – Value



Sample Option Usage

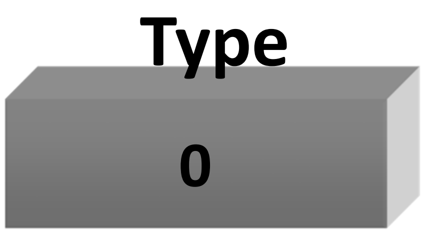


Sample Option: Jumbo Payload

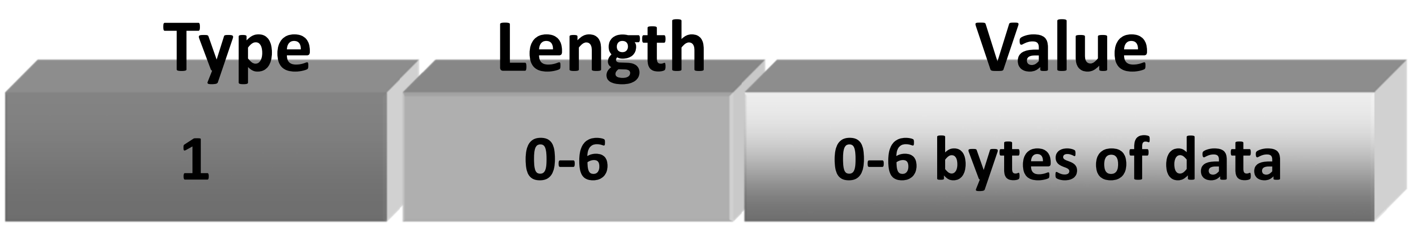


Padding Options

- ➔ Extension Headers must be 64 bit aligned
- ➔ Pad1 Option



- ➔ PadN Option

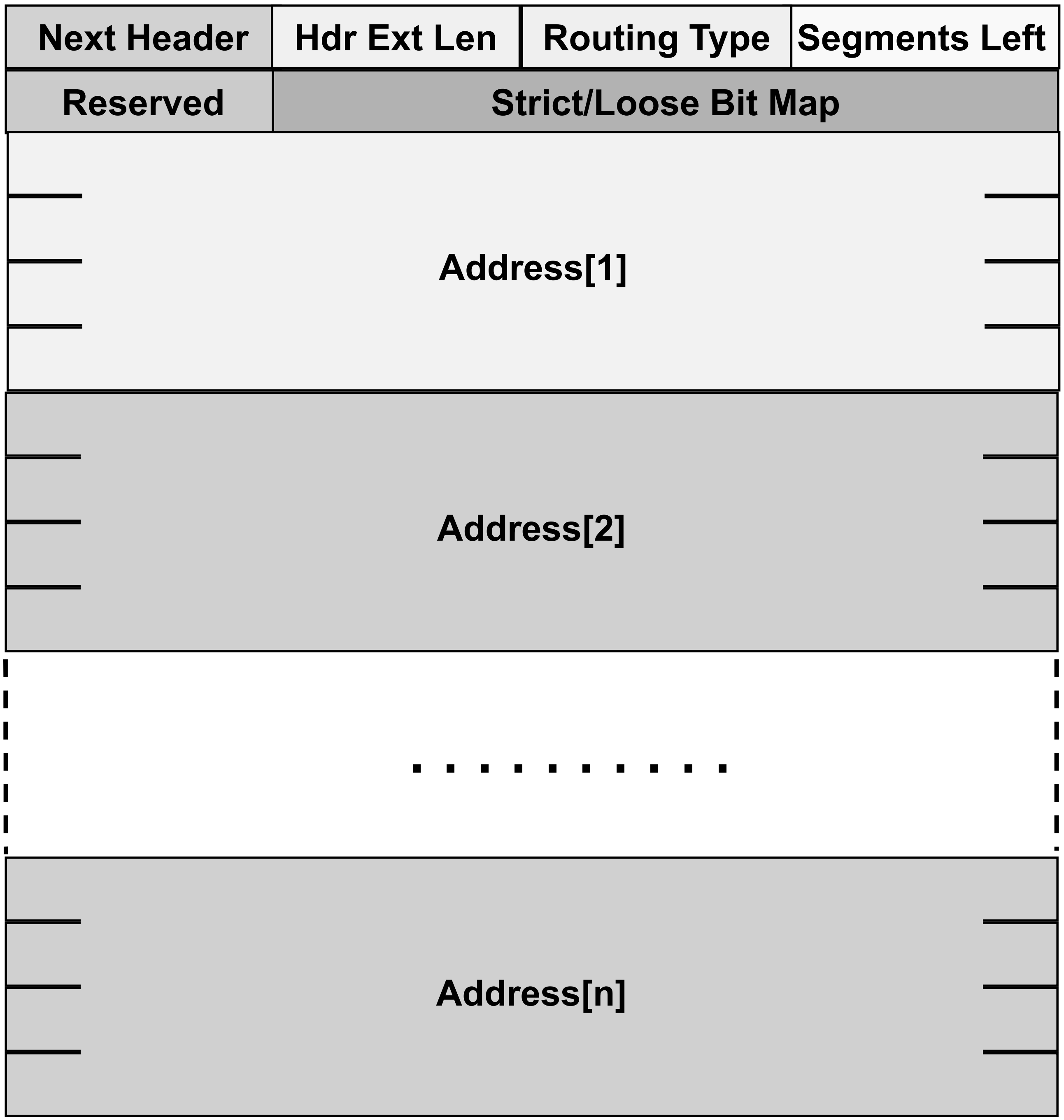


Type Field: first three bits

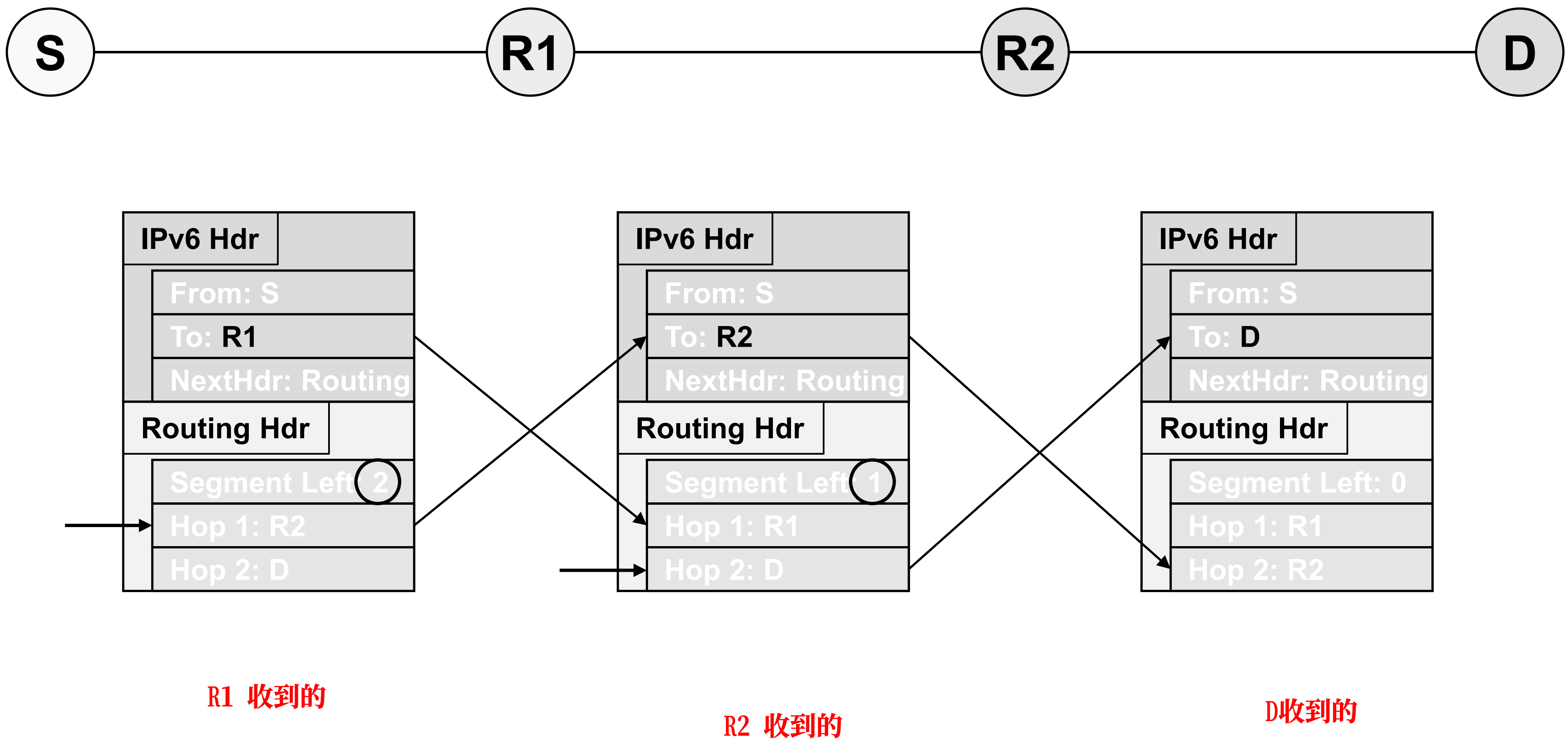
First 2 bits: action in case the option is not recognized	
Code	**Meaning**
00	The current option can ignored. It is possible to proceed with the next one
01	The packet must be discarded
10	The packet must be discarded and an ICMPv6 Parameter Problem message generated
11	The packet must be discarded and an ICMPv6 Parameter Problem must be generated, unless the destination address is a multicast one
Third bit: indicates if the option can be modified on-the-fly	
Code	**Meaning**
0	The option cannot be changed on-the-fly
1	The option can be changed on-the-fly

Routing Header

- ➔ Used by an IPv6 source to list nodes to traverse on the path to the destination
  - ➔ It can be a loose route
- ➔ Segment Left Field shows the number of the remaining path segments
  - ➔ Points to the next router to reach


Next Header
Hdr Ext Len
Routing Type
Segments Left
Reserved
Strict/Loose Bit Map
Address[1]
Address[2]
Address[n]

Routing Header: example



INTERFACING WITH THE LOWER LAYER

Encapsulation

- ➔ Encapsulated in layer 2 frames
- ➔ EtherType: 86DD
- ➔ As a new protocol
  - ➔ Enables dual stack approach
    - ➔ Keep running IPv4 as-is

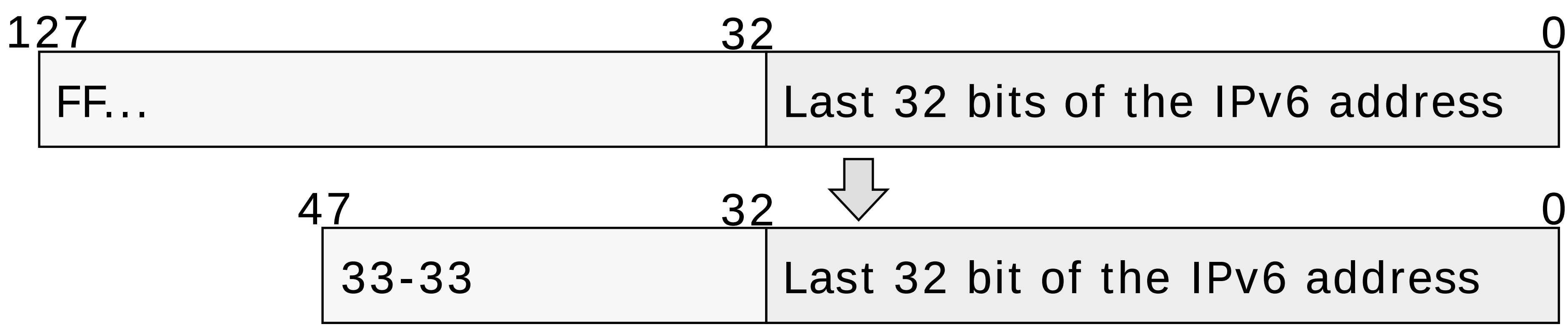
Address Mapping

What is the destination MAC address?

- ➔ IP unicast address
  - ➔ Procedural (protocol-based) discovery
    - ➔ Neighbor Discovery
- ➔ IP multicast address
  - ➔ Algorithmic mapping

IPv6 Multicast Transmission

- → Based on MAC multicast
- → IPv6 multicast address mapped to MAC address
- → 33-33 | 4 least significant bytes of IPv6 address



Multicast Address Mapping Example

- When sending a packet to the IP multicast address FFOC::89:AABB:CCDD
- Encapsulate in MAC frame to 33:33:AA:BB:CC:DD MAC 组播地址

NEIGHBOR DISCOVERY

New Function in ICMP

- → It substitutes ARP
- → Based on multicast
  - → Most likely only one station gets involved

Solicited Node Multicast Address

- → Subscribed by all hosts
- → FFO2::1:FF/1O4 | 24 least significant bits of IP address
- → Likely 1 host per group

Address Resolution

- → ICMP Neighbor Solicitation
  - → To Solicited Node Multicast Address of target IPv6 address
- → ICMP Neighbor Advertisement
  - → To requester address

Resolution Example

- → To find the MAC address of host 2OO1::ABCD:EF98
- → ICMP Neigh Sol to Sol Node Mult Add: FFO2::1:FFCD:EF98
- → Encapsulate in MAC frame to 33:33:FF:CD:EF:98

Host Cache

- → Mapping between IPv6 and MAC address
- → Equivalent to ARP cache

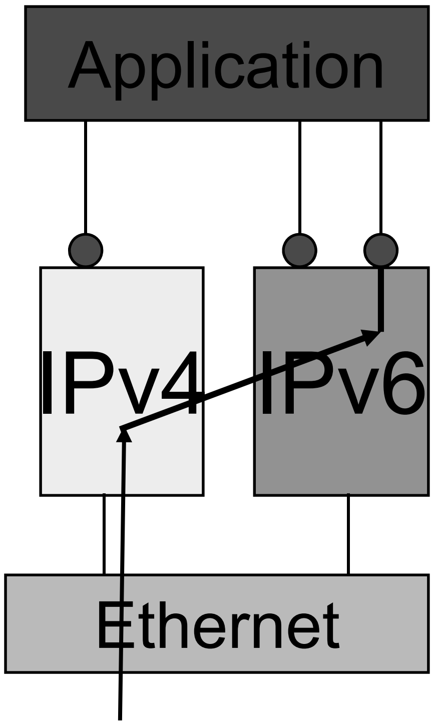
TRANSITION TO IPv6 (?)

IPv4 to IPv6 Transition

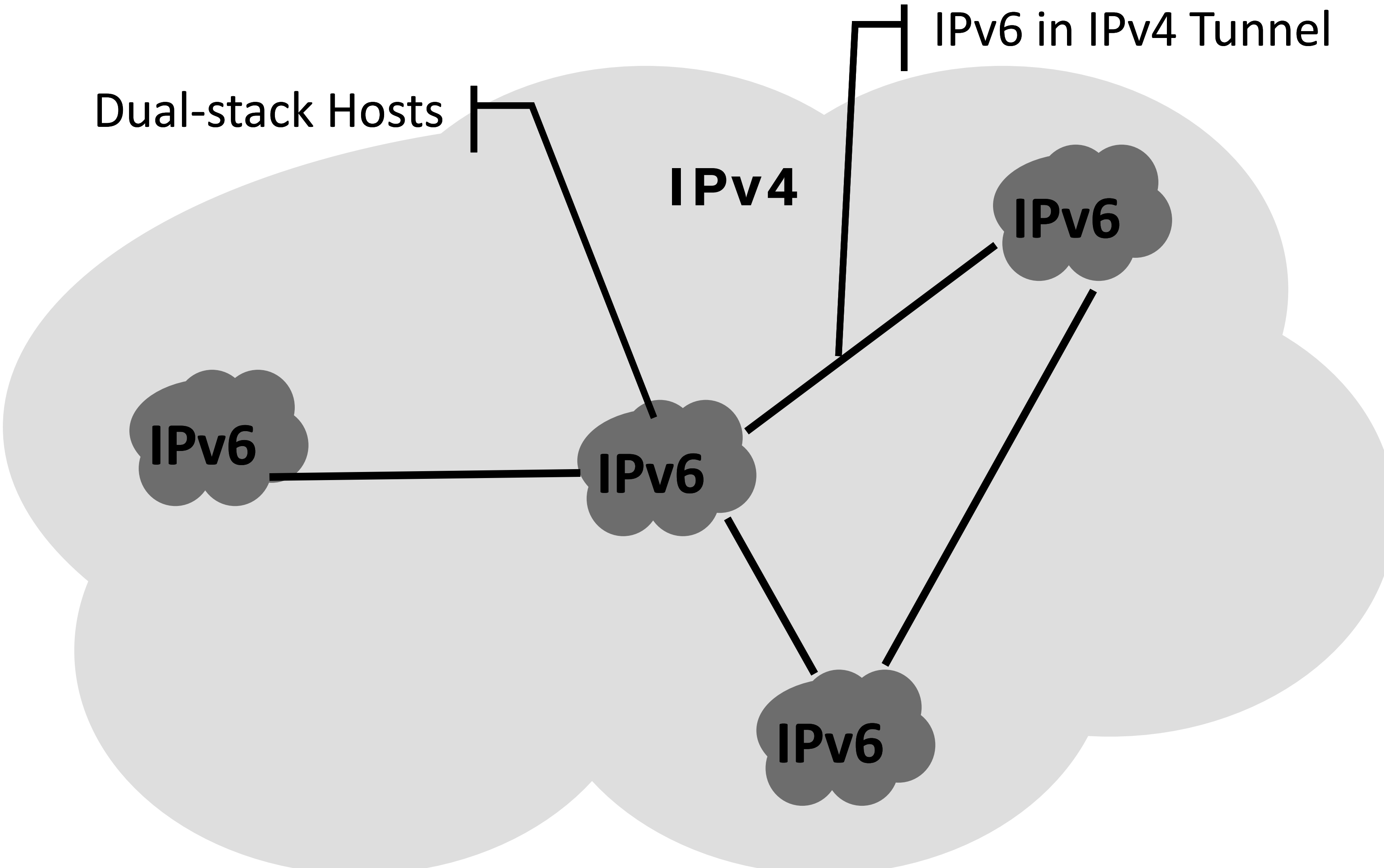
- → Incremental
- → Seamless 无缝衔接
- → Smooth

How can we enable this?

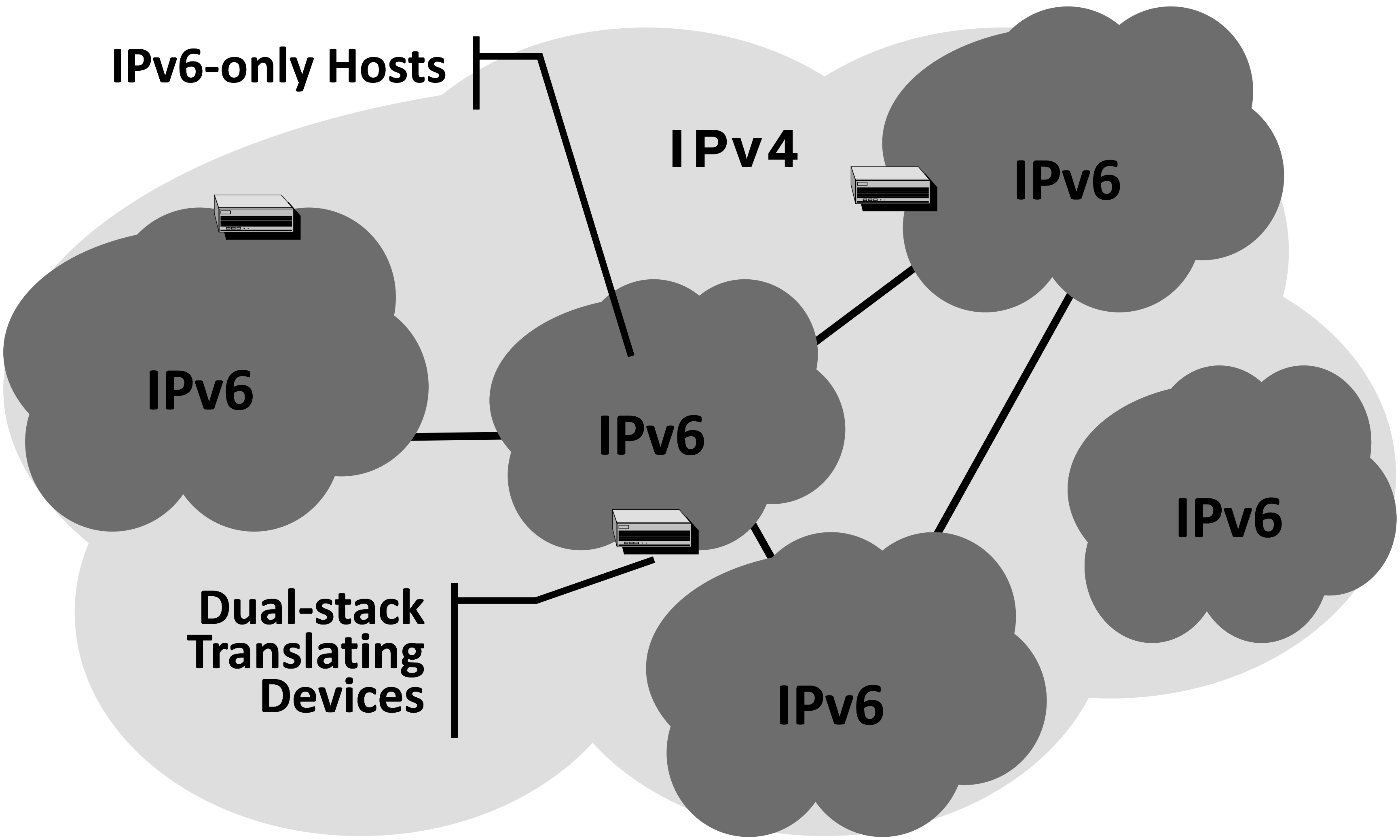
- → Dual-stack approach
  - → IPv6 as a new protocol
  - → Generate/receive v6 or v4 packets as needed
- → Address mapping
- → Tunneling
- → Translation mechanisms



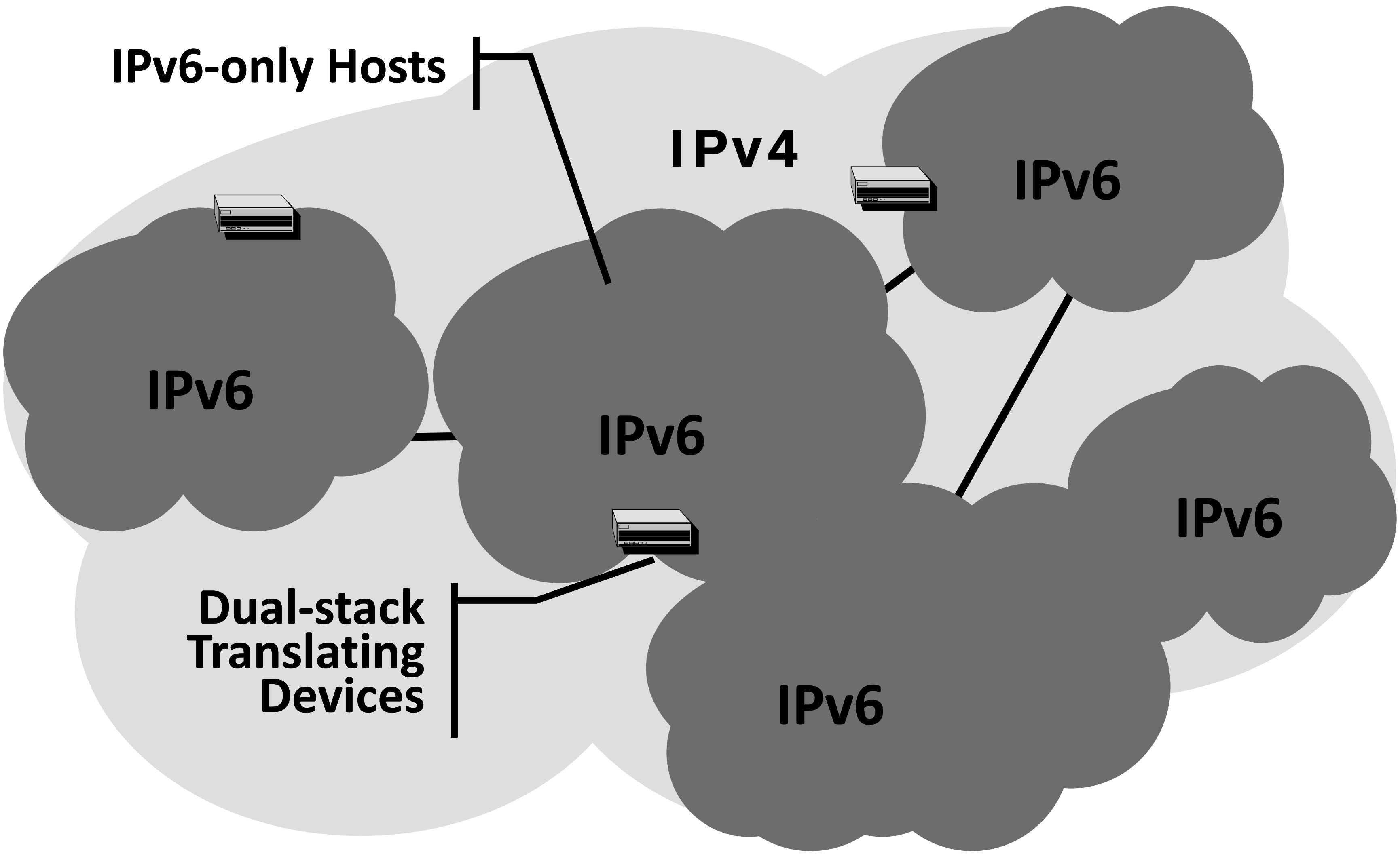
Isolated IPv6 Networks



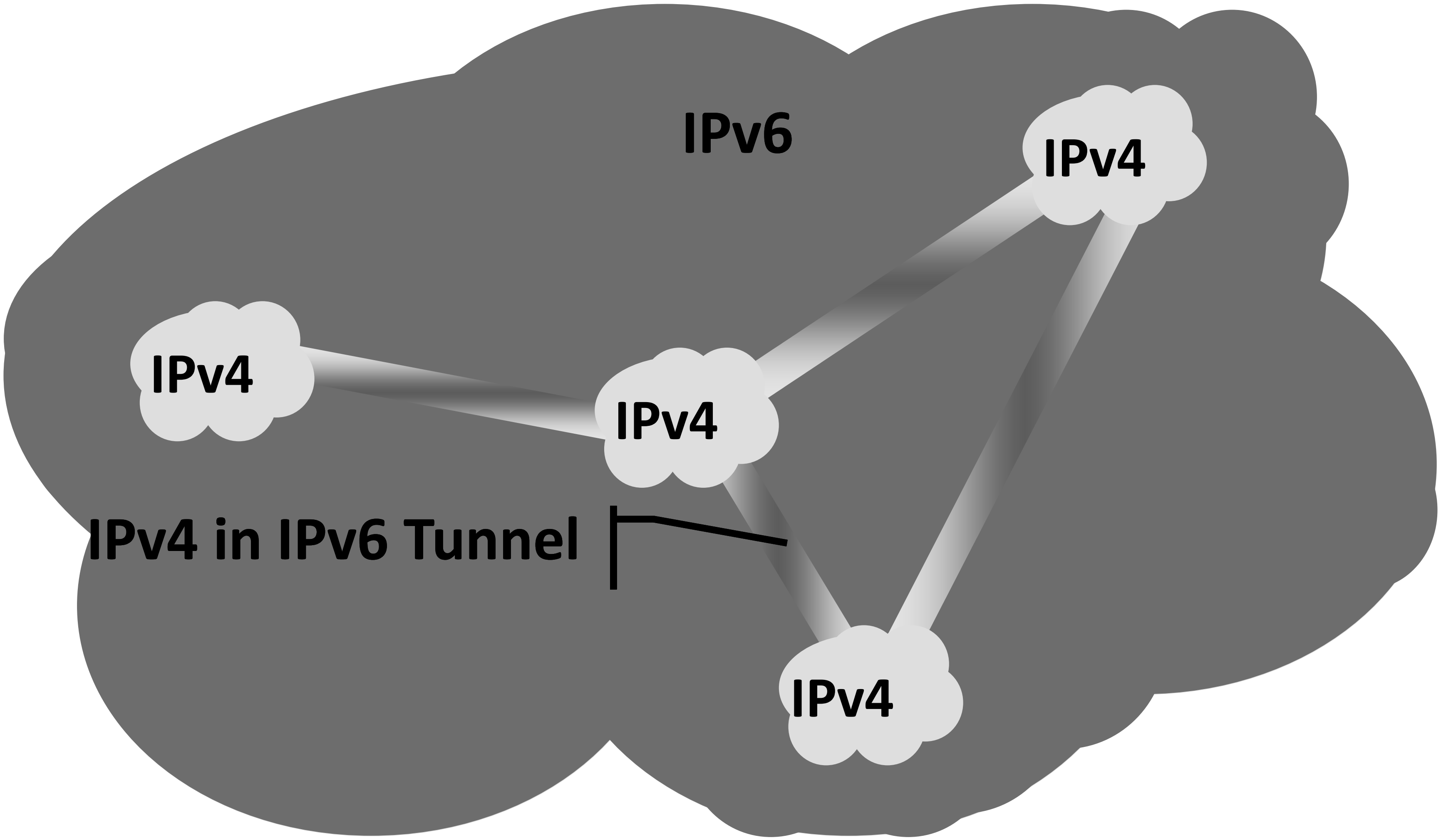
IPv6 Islands Grow



Native IPv6 Connectivity



All the Way to the Doomsday



Are we ready?

- → All protocols specified
  - → For a while: since 1996!!

- → Implemented on routers
  - → Even if less stable than IPv4
  - → Possibly not all functionalities
  - → Some hardware implementations (Layer 3 switch)

- → Implemented in end systems
  - → Windows since 2OOO and XP
  - → Unix, FreeBSD, Linux
- → Quite a few applications
  - → Possibly with a few bugs

When will it happen?

→ Large IPv4 install base

→ Only one true motivation:

Address space depletion 枯竭

- The issue has been mitigated
  - Provident address assignment
  - Extensive use of private addressing
  - NAT and proxying 代理

So, don’t we need IPv6?

→ NAT not suitable for all applications

- → Problematic with security mechanisms

→ User traceability

→ Not practical with servers

→ Not many → pubblic addresses

Acceptable limitations so far

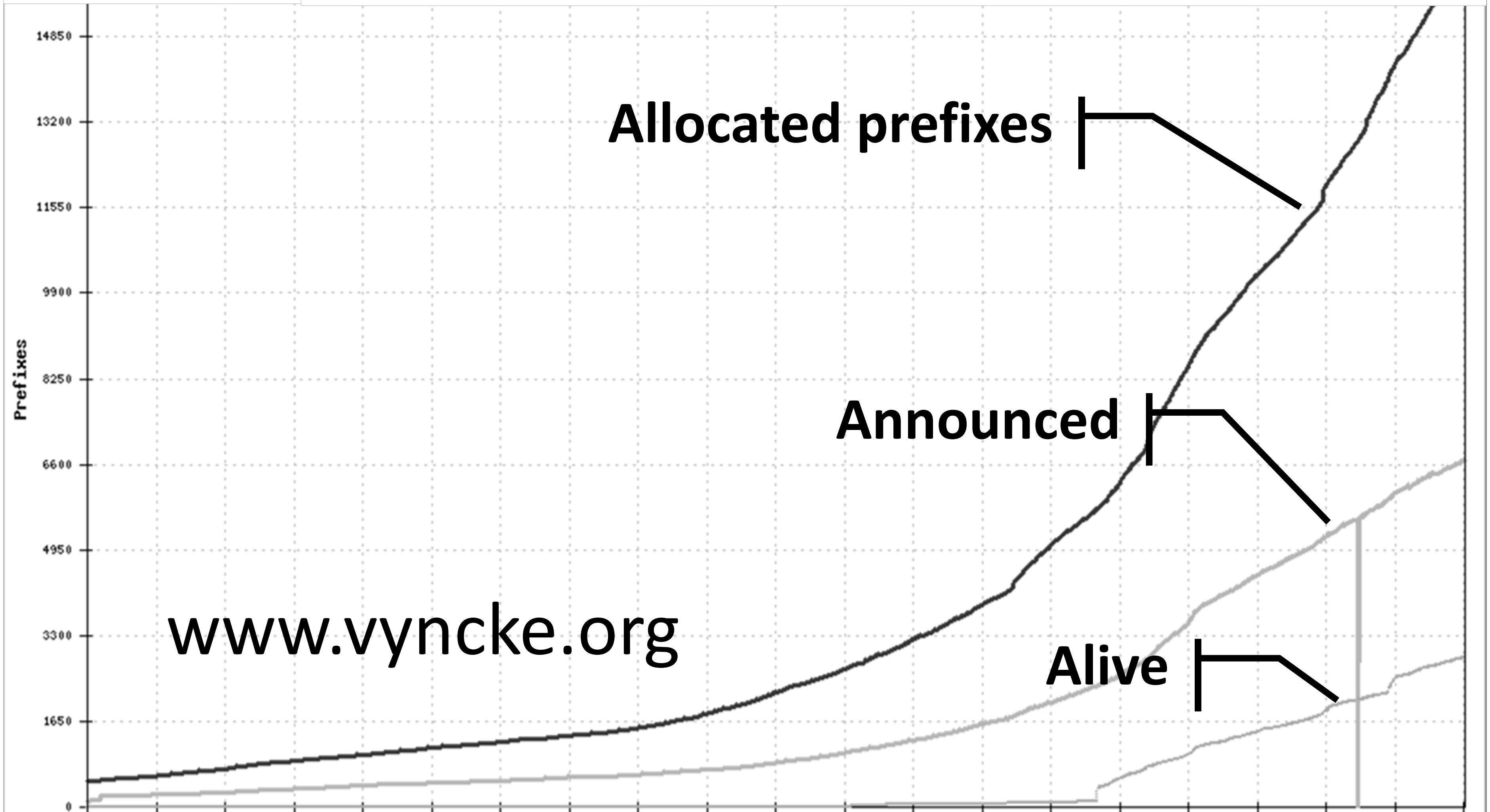
Just Plain Address Space Exhaustion

- → Especially in the Asia-Pacific region
- → IANA ran out of class A prefixes in Feb 2O11
- → RIPE by end 2O11

Possibly legislation

165OO

Allocated Prefixes



June 2003 June 2013

Current IPv6 web deployment

www.vyncke.org