IPv6 Basics

AfriNIC IPv6 Training

Accra, Ghana – 24th March 2009

Lagos, Nigeria – 26th March 2009

César Olvera (cesar.olvera@consulintel.es)
Jordi Palet (jordi.palet@consulintel.es)
Alvaro Vives (alvaro.vives@consulintel.es)

IPv6 Tutorial

1. Introduction to IPv6

Why a New IP?

Only *compelling* reason: more addresses!

- for billions of new devices,
  e.g., cell phones, PDAs, appliances, cars, etc.
- for billions of new users,
  e.g., in China, India, etc.
- for “always-on” access technologies,
  e.g., xDSL, cable, ethernet-to-the-home, etc.

But Isn't There Still Lots of IPv4 Address Space Left?

- ~ Half the IPv4 space is unallocated
  - if size of Internet is doubling each year, does this mean only one year's worth?!
- No, because today we deny unique IPv4 addresses to most new hosts
  - we make them use methods like NAT, PPP, etc. to share addresses
- But new types of applications and new types of access need unique addresses!

Why Are NAT's Not Adequate?

- They won't work for large numbers of "servers", i.e., devices that are "called" by others (e.g., IP phones)
- They inhibit deployment of new applications and services
- They compromise the performance, robustness, security, and manageability of the Internet

Incidental Benefits of Bigger Addresses

- Easy address auto-configuration
- Easier address management/delegation
- Room for more levels of hierarchy, for route aggregation
- Ability to do end-to-end IPsec (because NATs not needed)

Incidental Benefits of New Deployment

- Chance to eliminate some complexity, e.g., in IP header
- Chance to upgrade functionality, e.g., multicast, QoS, mobility
- Chance to include new enabling features, e.g., binding updates

Summary of Main IPv6 Benefits

- Expanded addressing capabilities
- Server-less autoconfiguration (“plug-n-play”) and reconfiguration
- More efficient and robust mobility mechanisms
- Built-in, strong IP-layer encryption and authentication
- Streamlined header format and flow identification
- Improved support for options / extensions

Why Was 128 Bits Chosen as the IPv6 Address Size?

- Some wanted fixed-length, 64-bit addresses
  - easily good for 10^{12} sites, 10^{15} nodes, at .0001 allocation efficiency (3 orders of mag. more than IPng requirement)
  - minimizes growth of per-packet header overhead
  - efficient for software processing
- Some wanted variable-length, up to 160 bits
  - compatible with OSI NSAP addressing plans
  - big enough for autoconfiguration using IEEE 802 addresses
  - could start with addresses shorter than 64 bits & grow later
- Settled on fixed-length, 128-bit addresses
  - (340,282,366,920,938,463,463,374,607,431,768,211,456 in all!)

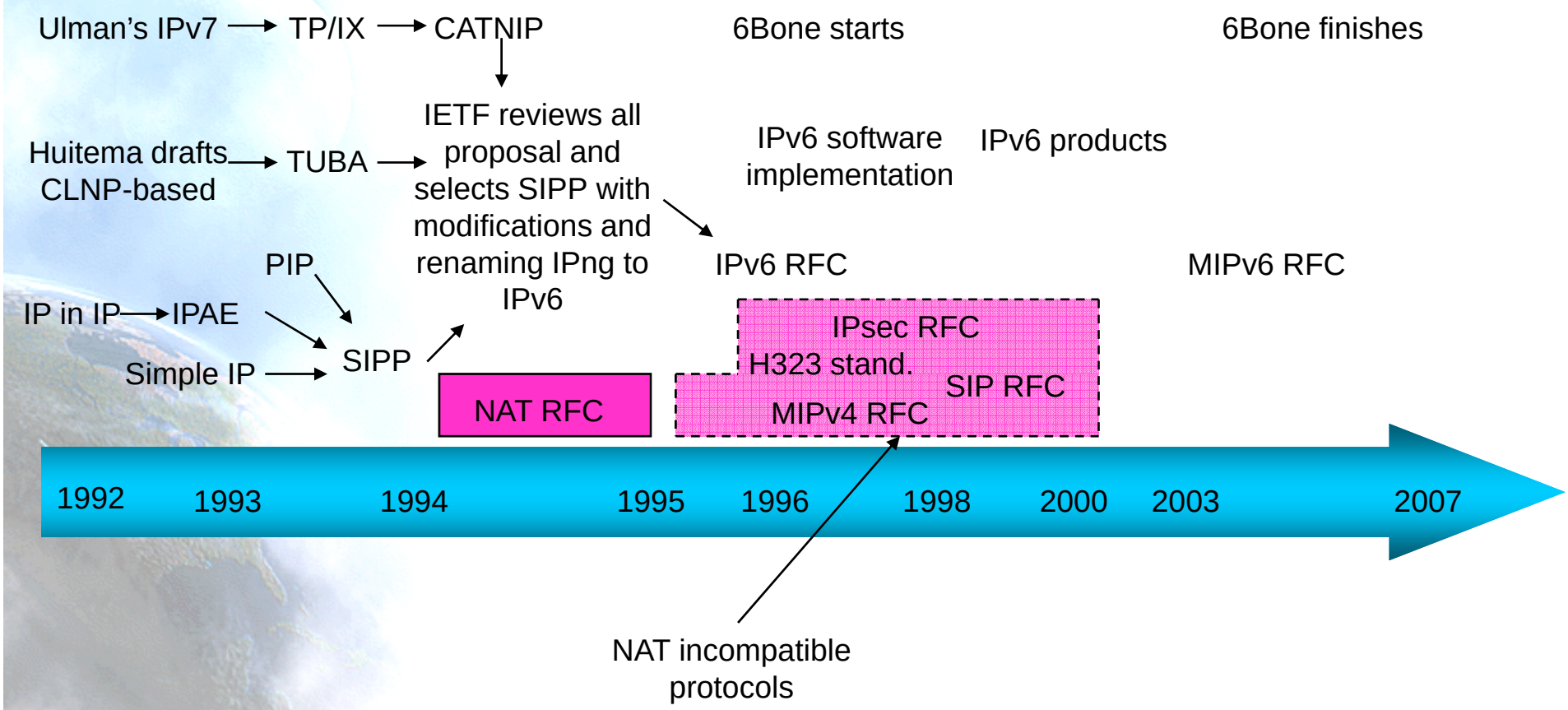
What Ever Happened to IPv5?

0–3		unassigned
4	IPv4	(today's widespread version of IP)
5	ST	(Stream Protocol, not a new IP)
6	IPv6	(formerly SIP, SIPP)
7	CATNIP	(formerly IPv7, TP/IX; deprecated)
8	PIP	(deprecated)
9	TUBA	(deprecated)
10-15		unassigned

Candidates for IPng

- Creation and selection of the new protocol is under the IETF umbrella
- Between 1992 and 1994 there were seven candidatures, but by spring 1994 only three remained
  - CATNIP (Common Architecture for the Internet)
    - Designed as a "convergence protocol," integrating IP, Novell's IPX, and the network layer protocol of the OSI suite.
  - SIPP (Simple Internet Protocol Plus)
    - An evolution from the current IP (IPv4) and interoperable with it
  - TUBA (TCP and UDP with Bigger Addresses)
    - A proposal to adopt the OSI network layer (CLNP) as the new Internet network layer
- By July, 1994 the IETF selected SIPP as protocol that should become IPng
  - The SIPP documents were the basis for working on IPng
  - The SIPP working group disappeared to be integrated into the IPng working group
- Key aspects of SIPP to be chosen
  - Transition aspects from IPv4 to IPng
    - Long period with both IPv4 and IPng protocols coexisting
    - Some nodes never will upgrade to IPng
    - New IPng nodes could used old IPv4-only network to transport IPng packets (tunneling)
    - No need for a flag-day
- Later on, the IPng working group was officially renamed as IPv6 working group

IPng Time-line



IPv6 Tutorial

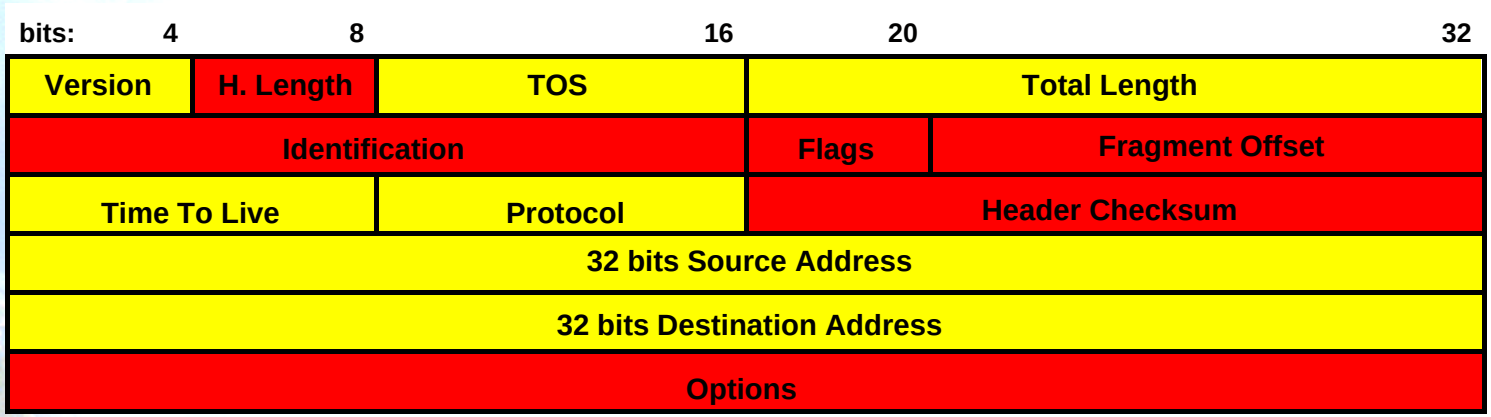
2. Header Formats

RFC2460

- Internet Protocol, Version 6: Specification
- Changes from IPv4 to IPv6:
  - Expanded Addressing Capabilities
  - Header Format Simplification
  - Improved Support for Extensions and Options
  - Flow Labeling Capability
  - Authentication and Privacy Capabilities

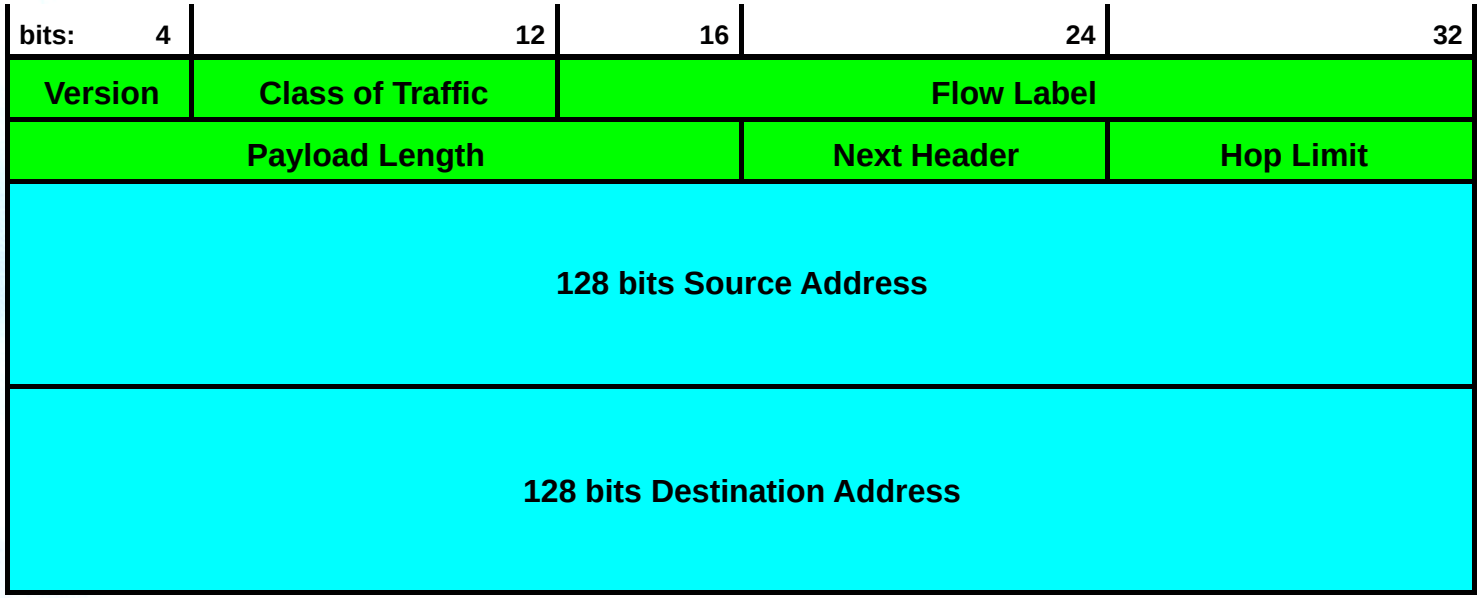
IPv4 Header Format

- 20 Bytes + Options



IPv6 Header Format

- From 12 to 8 Fields (40 bytes)



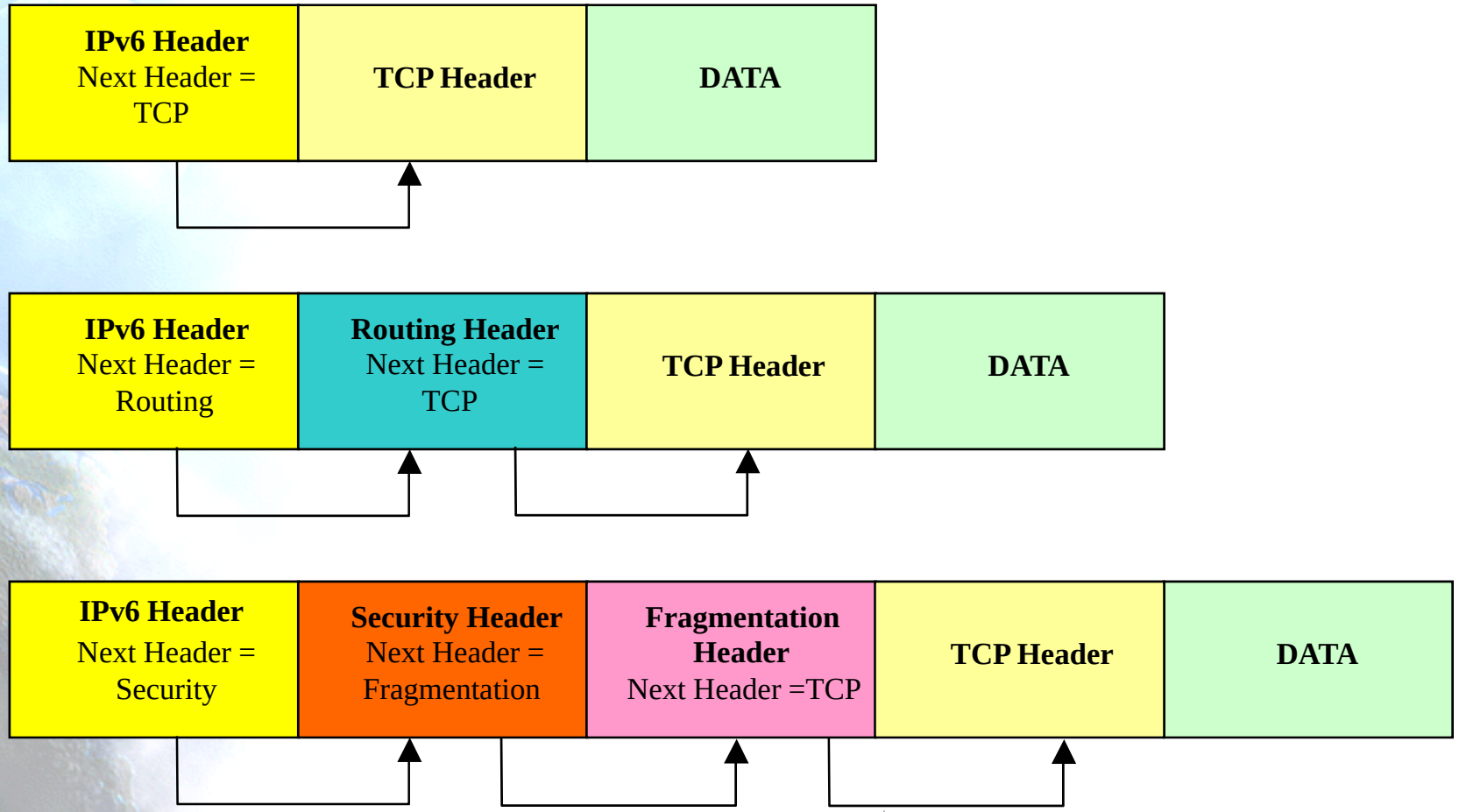
  - Avoid checksum redundancy
  - Fragmentation end to end

Summary of Header Changes

- 40 bytes
- Address increased from 32 to 128 bits
- Fragmentation and options fields removed from base header
- Header checksum removed
- Header length is only payload (because fixed length header)
  - Include length count of present extension headers
- New Flow Label field
- TOS -> Traffic Class
- Protocol -> Next Header (extension headers)
- Time To Live -> Hop Limit
- Alignment changed to 64 bits

Extension Headers

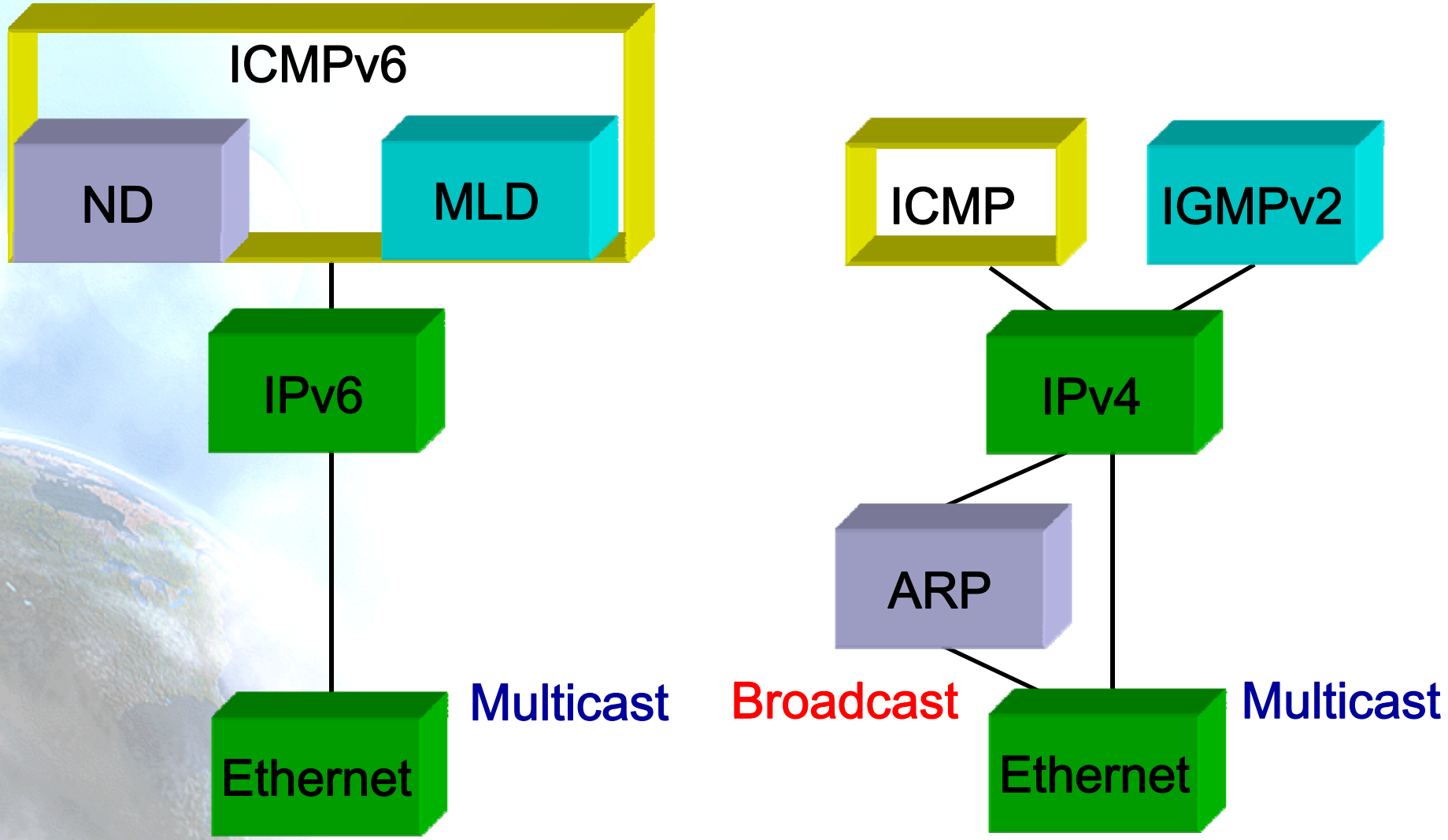
- "Next Header" Field



Extension Headers Goodies

- Processed Only by Destination Node
  - Exception: Hop-by-Hop Options Header
- No more “40 byte limit” on options (IPv4)
- Extension Headers defined currently (to be used in the following order):
  - Hop-by-Hop Options (0)
  - Destination Options (60) / Routing (43)
  - Fragment (44)
  - Authentication (RFC4302, next header = 51)
  - Encapsulating Security Payload (RFC4303, next header = 50)
  - Destination Options (60)
  - Mobility Header (135)
  - No next header (59)
    - TCP (6), UDP (17), ICMPv6 (58)

Control Plane IPv4 vs. IPv6



IPv6 Tutorial

3. Addressing

Text Representation of Addresses

“Preferred” form:
2001:0DB8:00FF:0000:0008:0007:200C:417A

Compressed form: FF01:0:0:0:0:0:0:43
becomes FF01::43

IPv4-compatible: ::13.1.68.3 (deprecated)

IPv4-mapped: ::FFFF:13.1.68.3

Prefixes (CIDR): 2001:DB8::1/48

URL: http://[FF01::43]:80/index.html

Address Types

Unicast (one-to-one)
- global
- link-local
- site-local (deprecated)
- Unique Local (ULA)
- IPv4-compatible (deprecated)
- IPv6-mapped

Multicast (one-to-many)

Anycast (one-to-nearest)

Reserved

Address Type Prefixes

Address Type	Binary Prefix	IPv6 Notation
Unspecified	00…0 (128 bits)	::/128
Loopback	00…1 (128 bits)	::1/128
Multicast	1111 1111	FF00::/8
Link-Local Unicast	1111 1110 10	FE80::/10
ULA	1111 110	FC00::/7
Global Unicast	(everything else)	
IPv4-mapped	00…0:1111 1111:IPv4	::FFFF:IPv4/128
Site-Local Unicast (deprecated)	1111 1110 11	FEF0::/10
IPv4-compatible (deprecated)	00…0 (96 bits)	::/96

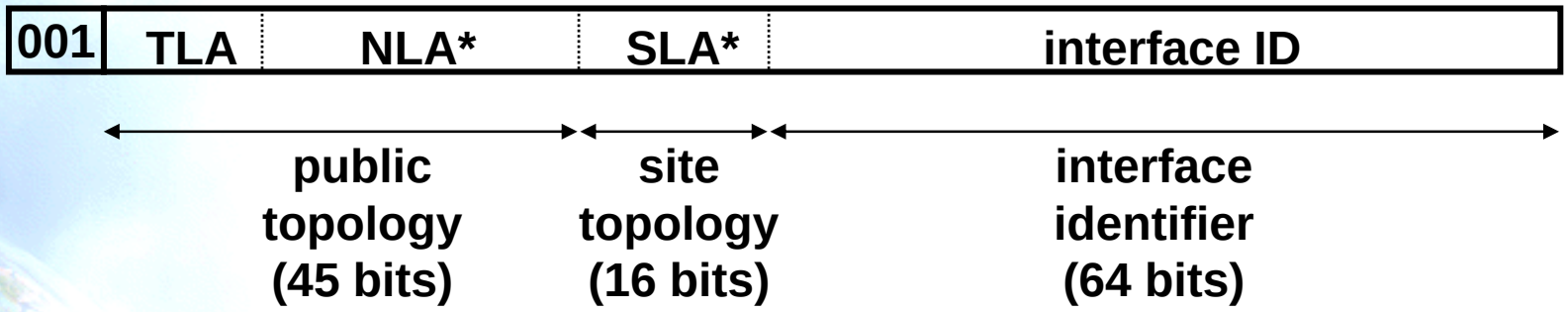
- **Anycast** addresses allocated from unicast prefixes

Global Unicast Prefixes

Address Type	Binary Prefix
IPv4-compatible	0000...0 (96 zero bits) (deprecated)
IPv4-mapped	00…0FFFF (80 zero+ 16 one bits)
Global unicast	001
ULA	1111 110x (1= Locally assigned) (0=Centrally assigned)

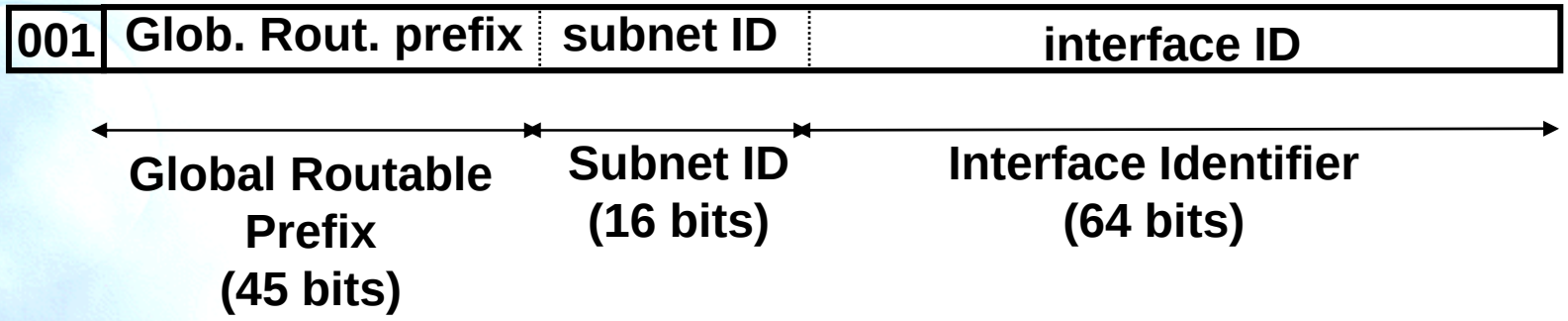
- **2000::/3** prefix is being allocated for Global Unicast, all other prefixes reserved (approx. 7/8ths of total)

Aggregatable Global Unicast Addresses (RFC2374) (Deprecated)



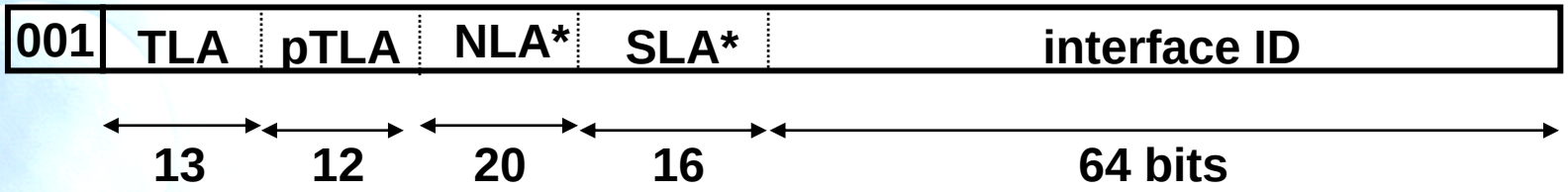
- TLA = Top-Level Aggregator
  NLA* = Next-Level Aggregator(s)
  SLA* = Site-Level Aggregator(s)
- all subfields variable-length, non-self-encoding (like CIDR)
- TLAs may be assigned to providers or exchanges

Global Unicast Addresses (RFC3587)



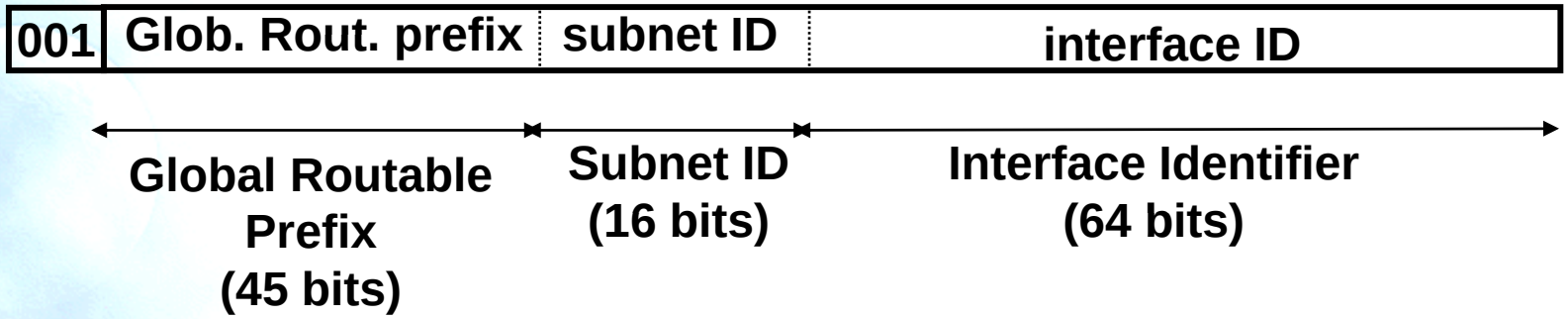
- The global routing prefix is a value assigned to a zone (site, a set of subnetworks/links)
  - It has been designed as an hierarchical structure from the Global Routing perspective
- The subnetwork ID, identifies a subnetwork within a site
  - Has been designed to be an hierarchical structure from the site administrator perspective
- The Interface ID is build following the EUI-64 format

Global Unicast Addresses for the 6Bone (until 6/6/6)



- 6Bone: experimental IPv6 network used for testing only
- TLA 1FFE (hex) assigned to the 6Bone
  - thus, 6Bone addresses start with 3FFE:
  - (binary 001 + 1 1111 1111 1110)
- Next 12 bits hold a “pseudo-TLA” (pTLA)
  - thus, each 6Bone pseudo-ISP gets a /28 prefix
- Not to be used for production IPv6 service

Global Unicast Addresses for Production Service



- LIRs receive by default /32
  - Production addresses today are from prefixes 2001, 2003, 2400, 2800, etc.
  - Can request for more if justified
- /48 used only within the LIR network, with some exceptions for critical infrastructures
- /48 to /128 is delegated to end users
  - Recommendations following RFC3177 and current policies
    - /48 general case, /47 if justified for bigger networks
    - /64 if only and only one network is required
    - /128 if it is sure that only and only one device is going to be connected

Link-Local & Site-Local Unicast Addresses

Link-local addresses for use during auto-configuration and when no routers are present:

1111111010	0	interface ID

Site-local addresses for independence from changes of TLA / NLA*: (deprecated)

1111111011	0	SLA*	interface ID

Unique Local IPv6 Unicast Addresses

IPv6 ULA (RFC4193)

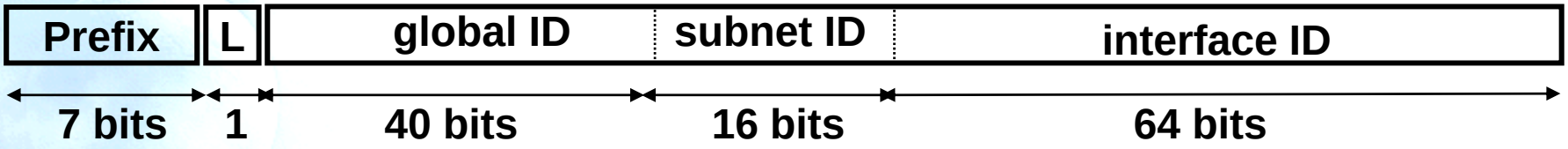
- Globally unique prefix with high probability of uniqueness
- Intended for local communications, usually inside a site
- They are not expected to be routable on the Global Internet
- They are routable inside of a more limited area such as a site
- They may also be routed between a limited set of sites
- Locally-Assigned Local addresses
  - vs Centrally-Assigned Local addresses

IPv6 ULA Characteristics

- Well-known prefix to allow for easy filtering at site boundaries
- ISP independent and can be used for communications inside of a site without having any permanent or intermittent Internet connectivity
- If accidentally leaked outside of a site via routing or DNS, there is no conflict with any other addresses
- In practice, applications may treat these addresses like global scoped addresses

IPv6 ULA Format

- Format:



- FC00::/7 Prefix identifies the Local IPv6 unicast addresses
- L = 1 if the prefix is locally assigned
- L = 0 may be defined in the future
- ULA are created using a pseudo-randomly allocated global ID
  - This ensures that there is not any relationship between allocations and clarifies that these prefixes are not intended to be routed globally

Centrally Assigned Unique Local IPv6 Unicast Addresses (1)

- Centrally-Assigned Local addresses
  - vs Locally-Assigned Local addresses
- Latest Draft:
  - draft-ietf-ipv6-ula-central-02.txt
  - June 2007
  - It defines the characteristics and requirements for Centrally assigned Local IPv6 addresses in the framework defined in IPv6 ULA – RFC4193

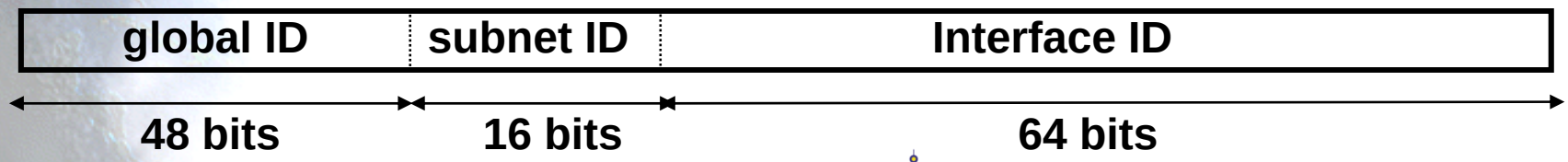
Centrally Assigned Unique Local IPv6 Unicast Addresses (2)

- The major difference between both assignments:
  - the Centrally-Assigned is uniquely assigned and the assignments are registered in a public database.
- It is recommended that sites planning to use Local IPv6 addresses use a centrally assigned prefix as there is no possibility of assignment conflicts. Sites are free to choose either approach.
- The allocation procedure for creating global-IDs for centrally assigned local IPv6 addresses is setting L=0. Remember that the allocation procedure for locally assigned local IPv6 addresses is thru L=1, as is defined in RFC4193.

Interface IDs

The lowest-order 64-bit field of unicast addresses may be assigned in several different ways:

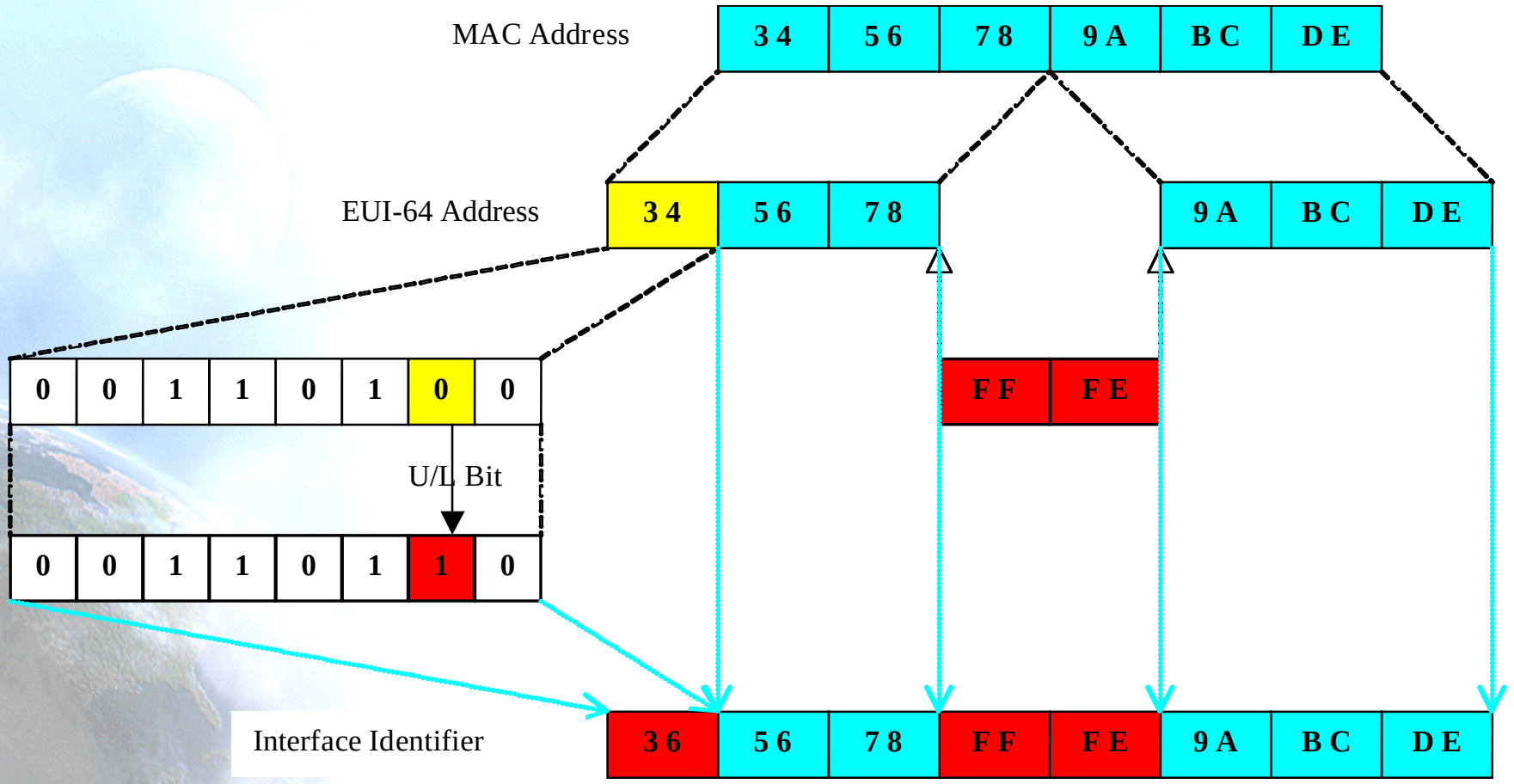
- auto-configured from a 48-bit MAC address (e.g., Ethernet address), expanded into a 64-bit EUI-64
- assigned via DHCP
- manually configured
- auto-generated pseudo-random number (to counter some privacy concerns)
- possibly other methods in the future



IPv6 in Ethernet

48 bits	48 bits	16 bits	
Ethernet Destination Address	Ethernet Source Address	1000011011011101 (86DD)	IPv6 Header and Data

EUI-64


MAC Address
3 4
5 6
7 8
9 A
B C
D E
EUI-64 Address
3 4
5 6
7 8
F F
F E
9 A
B C
D E
0 0 1 1 0 1 0 0
U/L Bit
0 0 1 1 0 1 1 0
Interface Identifier
3 6
5 6
7 8
F F
F E
9 A
B C
D E

Some Special-Purpose Unicast Addresses

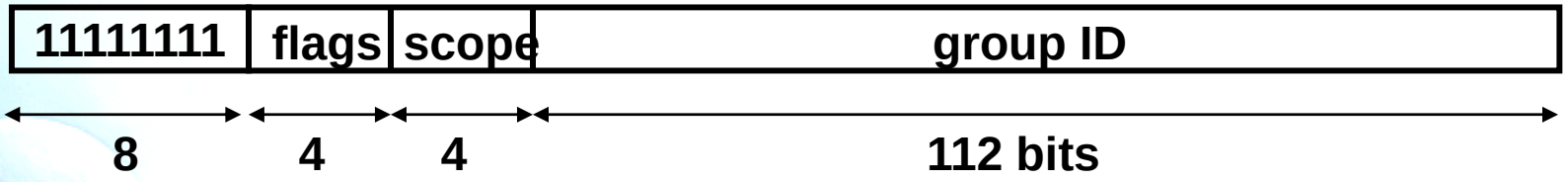
- The unspecified address, used as a placeholder when no address is available:

  0:0:0:0:0:0:0:0 (::/128)

- The loopback address, for sending packets to self:

  0:0:0:0:0:0:0:1 (::1/128)

Multicast Addresses



- Flags: **0RPT**: The high-order flag is reserved, and must be initialized to 0.
  - T: Transient, or not, assignment
  - P: Assigned, or not, based on network prefix
  - R: Rendezvous Point Addr. embedded, or not
- Scope field: 1 - Interface-local
  2 - link-local
  4 - admin-local
  5 - site-local
  8 - organization-local
  E - global

(3,F reserved)(6,7,9,A,B,C,D unassigned)

IPv6 Tutorial

4. Autoconfiguration

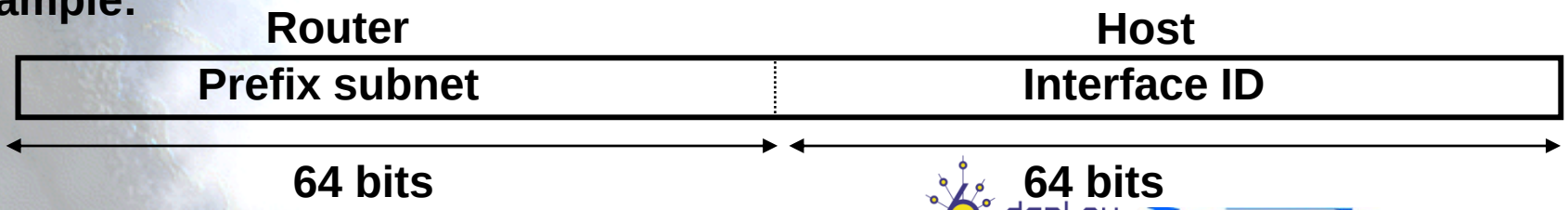
RFC2462

- The document specifies the steps a host takes in deciding how to autoconfigure its interfaces in IPv6.
- The autoconfiguration process includes creating a link-local address and verifying its uniqueness on a link, determining what information should be autoconfigured (addresses, other information, or both), and in the case of addresses, whether they should be obtained through the stateless mechanism, the stateful mechanism, or both.
- IPv6 defines both a stateful and stateless address autoconfiguration mechanism.
- Stateless autoconfiguration requires no manual configuration of hosts, minimal (if any) configuration of routers, and no additional servers.

Stateless or Serverless Autoconfiguration

- Stateless mechanism allows a host to generate its own addresses using a combination of locally available information and information advertised by routers.
- Routers advertise prefixes that identify the subnet(s) associated with a link.
- Hosts generate an "interface identifier" that uniquely identifies an interface on a subnet, locally generated, e.g., using MAC address.
- An address is formed by combining the both.
- In the absence of routers, a host can only generate link-local addresses.
- Link-local addresses are sufficient for allowing communication among nodes attached to the same link.

Example:



Stateful Autoconfiguration

- Hosts obtain interface addresses and/or configuration information and parameters from a server.
- Servers maintain a database that keeps track of which addresses have been assigned to which hosts.
- Stateless and stateful autoconfiguration complement each other.
- Both stateful and stateless address autoconfiguration may be used simultaneously.
- The site administrator specifies which type of autoconfiguration to use through the setting of appropriate fields in Router Advertisement messages.
- Example: DHCPv6

IPv6 Tutorial

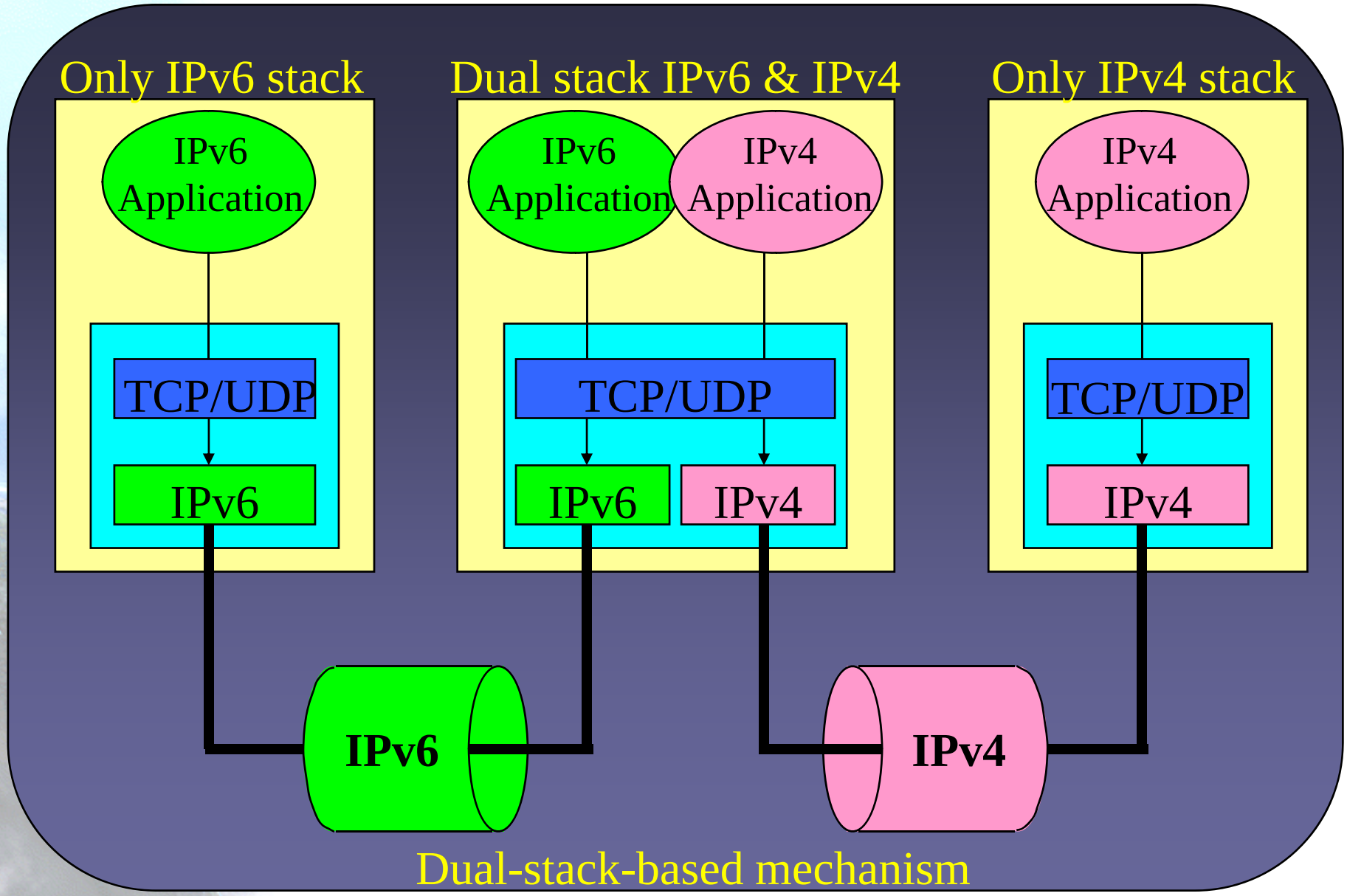
5. Transition and Coexistence

Transition / Co-Existence Techniques

- IPv6 has been designed for easing the transition and coexistence with IPv4
- Several strategies have been designed for coexisting with IPv4 hosts
  - Dual stack: Simultaneous support for both IPv4 and IPv6 stacks
  - Tunnels: IPv6 packets encapsulated in IPv4 ones
    - This is the commonest choice
  - Translation: This should be the last choice because it isn't perfect

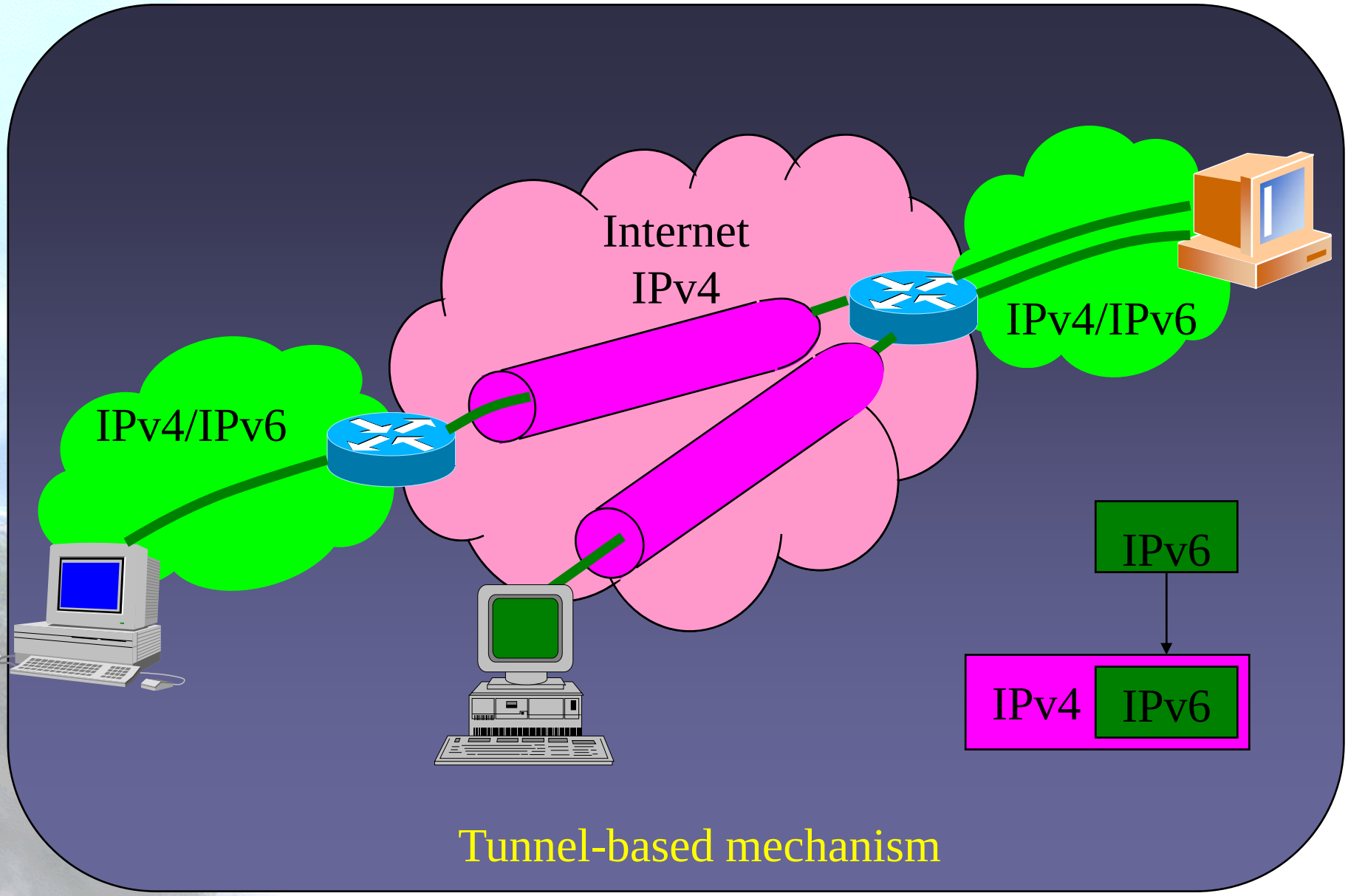
Dual-Stack Approach

- When adding IPv6 to a system, do not delete IPv4
  - this multi-protocol approach is familiar and well-understood (e.g., for AppleTalk, IPX, etc.)
  - note: in most cases, IPv6 will be bundled with new OS releases, not an extra-cost add-on
- Applications (or libraries) choose IP version to use
  - when initiating, based on DNS response:
    - if (dest has AAAA record) use IPv6, else use IPv4
  - when responding, based on version of initiating packet
- This allows indefinite co-existence of IPv4 and IPv6, and gradual app-by-app upgrades to IPv6 usage
- A6 record is experimental


Only IPv6 stack
Dual stack IPv6 & IPv4
Only IPv4 stack
IPv6 Application
IPv6 Application
IPv4 Application
IPv4 Application
TCP/UDP
TCP/UDP
TCP/UDP
IPv6
IPv6
IPv4
IPv4
IPv6
IPv4
Dual-stack-based mechanism

Tunnels to Get Through IPv6-Ignorant Routers

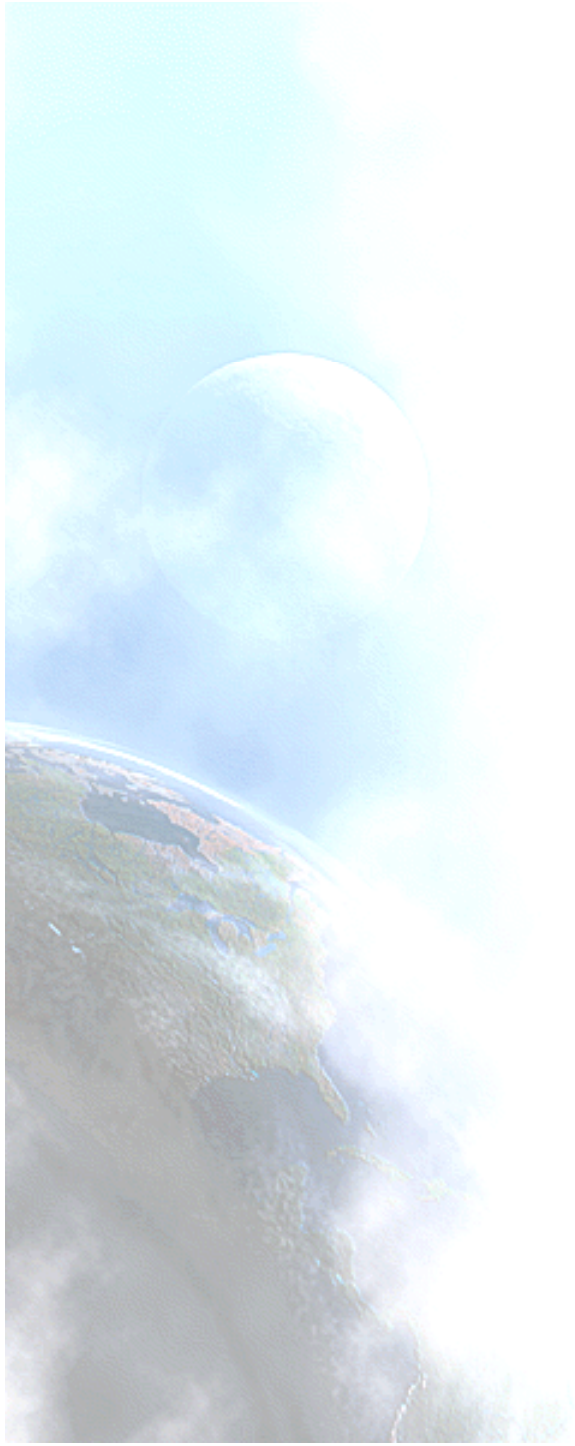
- Encapsulate IPv6 packets inside IPv4 packets (or MPLS frames) in order to provide IPv6 connectivity through IPv4-only networks
- Many methods exist for establishing tunnels:
  - manual configuration
  - "tunnel brokers" (using web-based service to create a tunnel)
  - "6over4" (intra-domain, using IPv4 multicast as virtual LAN)
  - "6to4" (inter-domain, using IPv4 addr as IPv6 site prefix)
- Can view this as:
  - IPv6 using IPv4 as a virtual link-layer, or
  - an IPv6 VPN (virtual public network), over the IPv4 Internet (becoming "less virtual" over time, we hope)



Tunnel-based mechanism

Translation IPv4/IPv6

- May prefer to use IPv6-IPv4 protocol translation for:
  - new kinds of Internet devices (e.g., cell phones, cars, appliances)
  - benefits of shedding IPv4 stack (e.g., serverless autoconfig)
- This is a simple extension to NAT techniques, to translate header format as well as addresses
  - IPv6 nodes behind a translator get full IPv6 functionality when talking to other IPv6 nodes located anywhere
  - they get the normal (i.e., degraded) NAT functionality when talking to IPv4 devices
  - methods used to improve NAT functionality (e.g, RSIP) can be used equally to improve IPv6-IPv4 functionality



IPv6 Tutorial

6. Routing

IPv6 Routing

- IPv6 uses same “longest-prefix match” routing as IPv4 CIDR
- Straightforward changes to existing IPv4 routing protocols to handle bigger addresses
  - –unicast: RIPv2, IS-IS, OSPFv3, BGP4+, …
  - –multicast: MOSPF, PIM, …
- Can use Routing header with anycast addresses to route packets through particular regions
  - –e.g., for provider selection, policy, performance, etc.

IPv6 Unicast Routing Protocols

- IPv6 supports widely deployed routing protocols such as
  - –RIPv2
  - –IS-IS
  - –**OSPFv3**
  - –**BGP4+**

OSPF for IPv6 (OSPFv3)

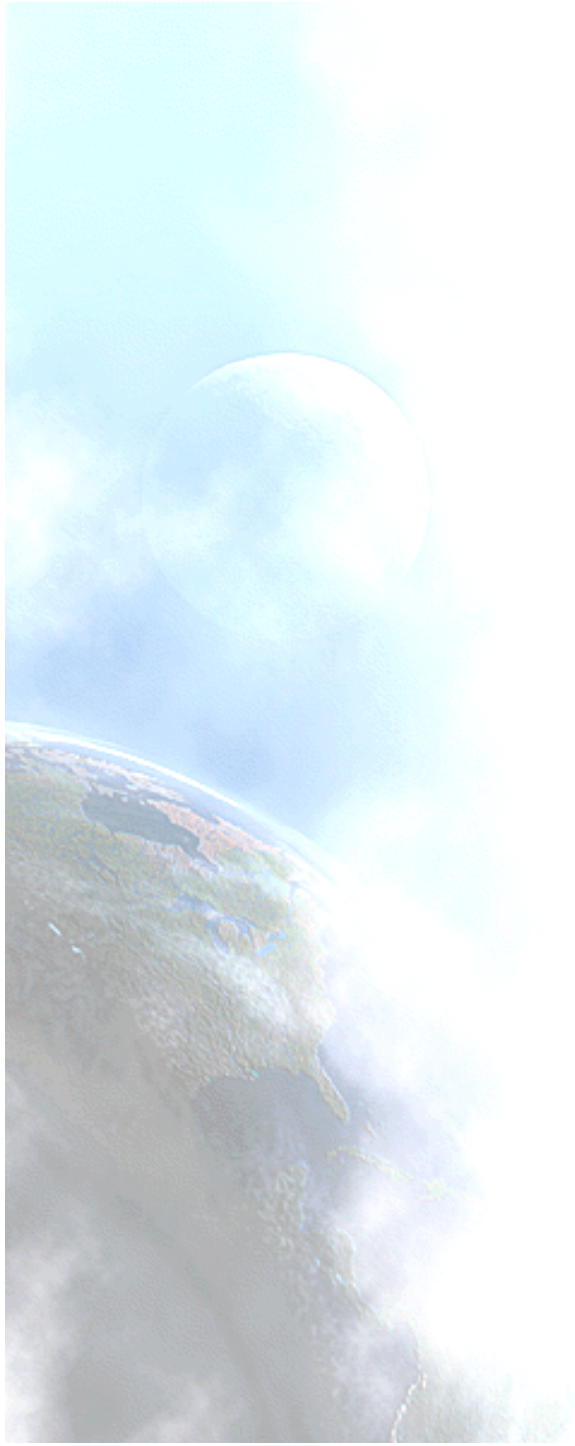
- Apart from the obsolete version 1, there are currently two versions of OSPF: OSPFv2 (for IPv4) and OSPFv3 (for IPv6). They are completely separate protocols that don't interact when both are enabled
  - –OSPF for IPv4 is described in RFC 2328
  - –OSPF for IPv6 is described in RFC 2740
- OSPF for IPv6 expands on OSPF for IPv4 to provide support for IPv6 routing prefixes
  - –Most of the algorithms from OSPF for IPv4 have preserved in OSPF for IPv6
  - –But some changes have been necessary
    - •either due to changes in protocol semantics between IPv4 and IPv6
    - •or simply to handle the IPv6 routing prefixes and the larger size of IPv6 addresses.

BGP for IPv6 (BGP4+) (1)

- The current version of BGP is version 4, i.e. BGP4
  - –BGP4 (BGP for IPv4) is described in RFC 4271
- The Multiprotocol BGP extensions, i.e. BGP4+, allow BGP4 to be used for different address families, such as IPv6 and Multicast
  - –Multiprotocol Extensions for BGP4 (BGP for IPv6) are described in RFC 4760
    - •This document defines the extensions to BGP4 to enable it to carry routing information for multiple Network Layer protocols (e.g., IPv6, IPX, L3VPN, etc.)

BGP for IPv6 (BGP4+) (2)

- Multiprotocol BGP extensions for IPv6 supports the same features and functionality as IPv4 BGP
  - –The IPv6 enhancements to multiprotocol BGP include support for
    - •The IPv6 address family and network layer reachability information (NLRI)
    - •The next hop (the next router in the path to the destination) attributes that use IPv6 addresses
- Multiprotocol BGP for the IPv6 Multicast Address family feature provides multicast BGP extensions for IPv6 and supports the same features and functionality as IPv4 BGP
  - –The IPv6 enhancements to multicast BGP include support for
    - •The IPv6 multicast address family and network layer reachability information (NLRI)
    - •The next hop (the next router in the path to the destination) attributes that use IPv6 addresses



IPv6 Tutorial

7. Mobility

Mobility in IP layer

- Implications
  - Communication = f(IP_source, Prt_source, IP_dest., Prt_dest)
  - If IP address changes, communication is not longer feasible
- Requirements
  - Compatibility with current applications and systems
  - No changes in routers
  - Transparency for applications
  - ……

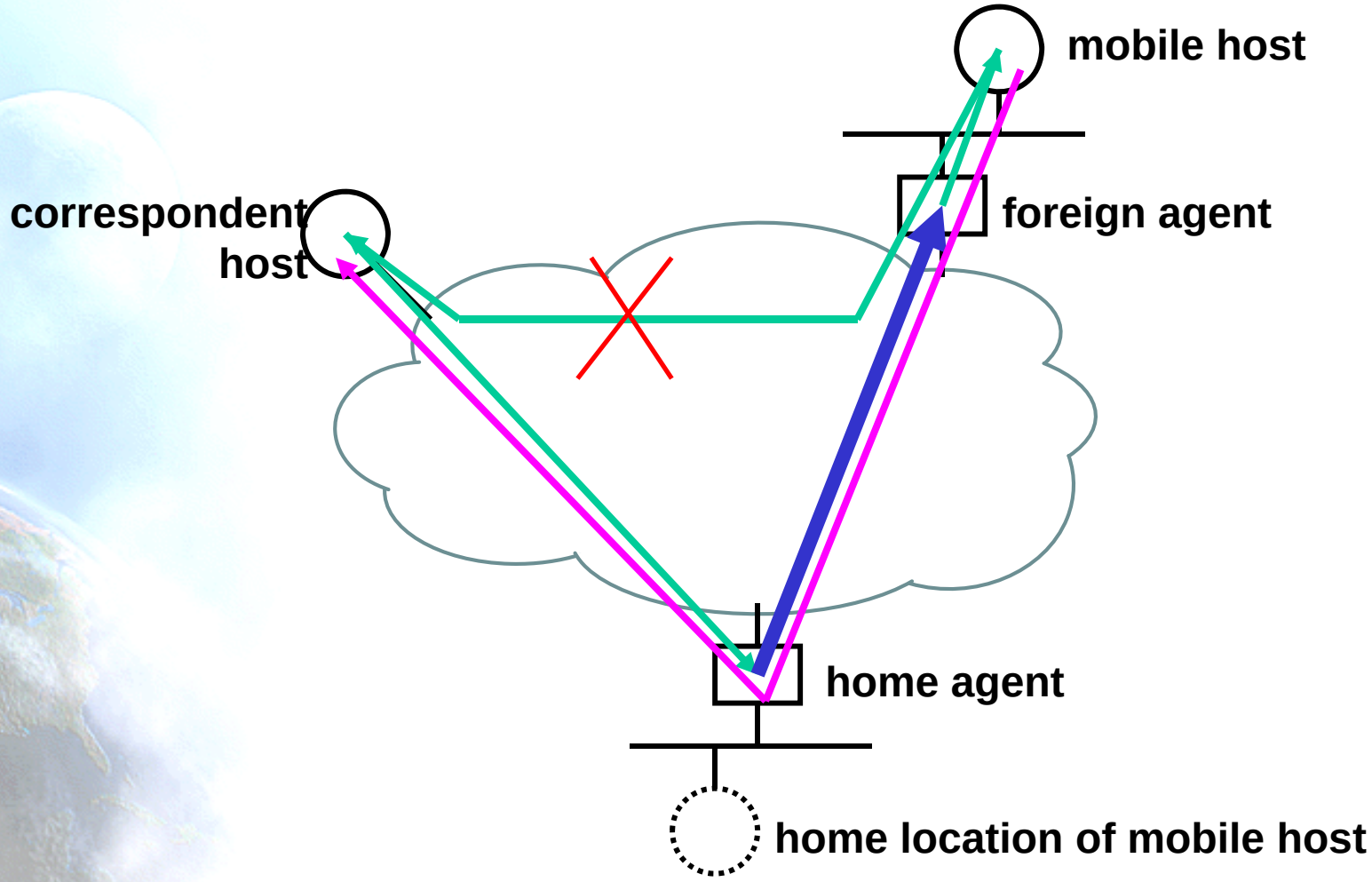
Mobility in IPv4 (1)

- Concepts
  - Home Agent: Server in “Home Network” (HN)
  - Foreign Agent: Server in foreign network
  - Mobile Node: Node which is away from HN
  - Correspondent Node: Node communicating to MN
  - Home Address: Address from the HN (HoA)
  - Care of Address: MN’s address obtained in the foreign network. It’s an address located into the FA, in a virtual interface

Mobility in IPv4 (2)

- A MN has one or more HoA
  - they are stable and can be associated to the host name through the DNS
- When MN is in a foreign network, it acquires another IP address
- It registers its new CoA with its HA
- Packets sent to the MN's HoA are intercepted by the HA y then forwarded to the FA by using tunneling
- Packets sent by the MN are delivered in two ways:
  - They are sent to the FA and this forwards them by using the HoA
    - This is an issue if ingress-filtering is implemented into the ISP
  - A tunnel with the HA is created and packets are sent through it

Mobility in IPv4 (3)



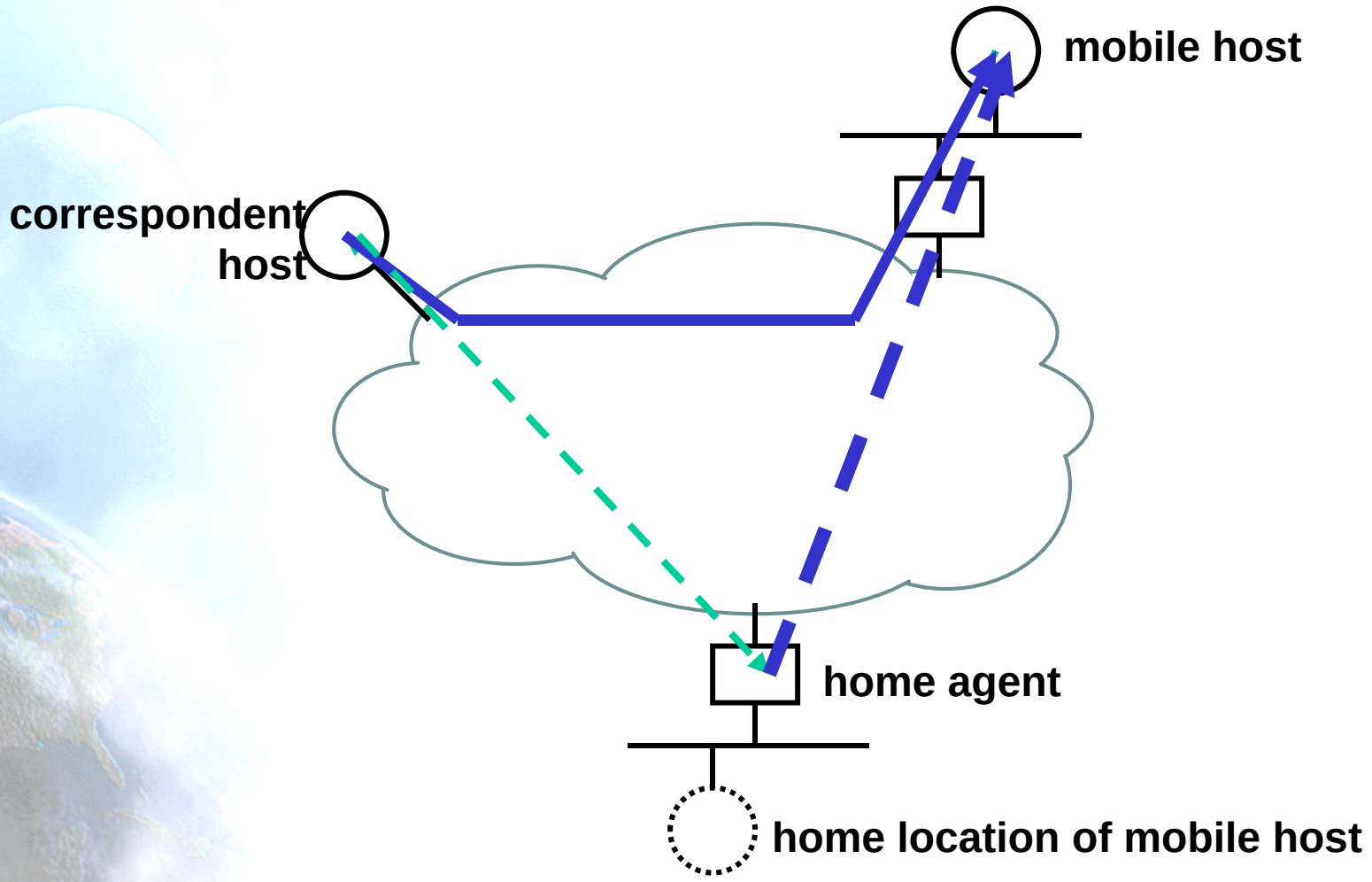
Mobility in IPv4 (4)

- Security
  - Authentication required
    - FA → HA
    - MN → FA
  - AAA infrastructure is usually used
- Issues with IPv4
  - Scarcity of public IPv4 addresses
    - FAs uses to be located behind routers implementing NAT, so IPv4 packets are modified in transit
  - Complexity and not broadly deployed AAA infrastructures
- Consequence
  - MIPv4 not operating

Mobility in IPv6 (1)

- IPv6 has two main features that much help to design a mobility solution
  - Neighbor discovery
  - Autoconfiguration
  - Both of them are used for:
    - Mobile Prefix Discovery: Similar to RS and RA
    - Dynamic HA Address Discovery. more than one HA are possible
- There are many differences to MIPv4, the most remarkable:
  - CoA is setup in the MN rather than the FA
  - There exists no FA
  - Authentication relations are different
    - MN → HA
    - MN → CN
  - ESP is used, so there is no need for AAA
  - Route optimization

Mobility in IPv6 (2)



Mobility in IPv6 (3)

- Route optimization is one of the most remarkable features:
  - At the beginning: CN → HA → MN
  - MN → CN (including the Header Option with its HoA)
    - Alternatively MN → HA → CN by using a tunnel
  - Once communication between CN and MN is setup: CN → MN
- This prevents the fact that HA is one single point-of-failure
- Also, unnecessary delays are prevented when the distance between CN → MN is lower than CN → HA → MN
- Authentication between CN → MN is required

Deploying IPv6 Mobility

- MIPv6 has been standardized 2004
  - It works for manual configurations ➔ Not scalable
- Deploying MIPv6 as network service has several implications
  - To define a scalable mechanism to provide the required parameters for MIPv6 to work without user's manual intervention
    - Bootstrapping: provide HoA, user's cryptographic credentials and HA address
  - To solve network issues to let the MIPv6 service work anywhere
    - HA load balancing
    - IPv4 MIPv6 interworking
    - Firewall traversal
- Most of these issues are being evaluated in the IETF MIPv6 WG
  - http://www.ietf.org/html.charters/mip6-charter.html
- Also other R&D projects deal with them
  - http://www.ist-enable.eu
  - http://www.nautilus6.org

Thanks !

Contact:

- César Olvera Morales (Consulintel): cesar.olvera@consulintel.es
- Jordi Palet Martínez (Consulintel): jordi.palet@consulintel.es

The IPv6 Portal:

http://www.ipv6tf.org